

Predicting Student Academic Performance Using GenAI-Enhanced Deep Learning Features: A Comparative Study

Khushvir Singh*, Anurag Tewari¹

*Research Scholar, School of Computing and Artificial Intelligence, Lovely Professional University (LPU), Punjab, India

¹Assistant Professor, School of Computing and Artificial Intelligence, Lovely Professional University (LPU), Punjab, India

*Corresponding author: khushvirsingh100@gmail.com



Abstract

Predicting student academic performance is a central task in educational data mining, with implications for early intervention and personalized learning. This paper evaluates whether Generative AI (GenAI)-derived behavioral and contextual features add predictive value beyond traditional educational attributes for five-class grade prediction. We benchmark five deep learning architectures (Residual MLP, Bidirectional LSTM, CNN with Global Average Pooling, Feature Tokenizer Transformer, and a Hybrid CNN-LSTM) and four tree-ensemble baselines (Random Forest, XGBoost, LightGBM, CatBoost) under two feature conditions: a Traditional set of 14 features and a Full set of 30 features that adds four scores generated by an open-weight instruction-tuned language model (Qwen2.5-1.5B-Instruct) and 12 engineered interaction and polynomial features. Experiments use a synthetic dataset of 14,003 student records. Under 10-fold stratified cross-validation, GenAI augmentation does not help in general. Only the FT-Transformer improves (+3.3 percentage points; paired t-test $p = 0.0002$; $p = 0.003$ after correcting for overlapping training sets), and this remains significant after Bonferroni adjustment across nine models. Random Forest declines by 0.65 points (uncorrected $p = 0.040$), but the decline is not significant after correction ($p = 0.13$), and the remaining seven models change by less than half a point (corrected $p \geq 0.29$). An ablation on the Hybrid CNN-LSTM shows no benefit from keeping the GenAI features (33.01% $\pm$ 1.32% without them, 32.76% $\pm$ 0.89% with them). The best result is CatBoost on Traditional features (34.64%), only 5.98 points above the 28.66% majority-class baseline, and neither ensemble configuration exceeds it (best 34.36%). Because the GenAI scores are generated from the same Traditional attributes, they cannot add information beyond those attributes, and the results are consistent with that: GenAI-based feature engineering should not be assumed to give predictive lift, and any benefit appears to depend on the model architecture.

Keywords

educational data mining; student performance prediction; generative AI; deep learning; FT-Transformer; feature augmentation; grade classification

Manuscript

Received:

September 2, 2026

Revised:

September 30, 2026

Accepted:

October 6, 2026

1. Introduction

The ability to accurately predict student academic outcomes before or during a learning cycle has far-reaching implications for educators, institutions, and policy makers [1]. Early identification of at-risk students allows for timely intervention, while performance forecasting can guide curriculum design and resource allocation [2]. Despite decades of research in Educational Data Mining (EDM), the field continues to

grapple with a fundamental challenge: the richness of the factors that determine academic success is rarely captured by simple attendance and grade records alone [3].

Traditional machine learning approaches to grade prediction have largely relied on structured demographic and behavioral features — study hours, attendance rates, assignment completion ratios, and similar metrics [4]. While these approaches have shown reasonable utility, they inevitably leave out the softer, more nuanced dimensions of the student experience: motivational state, stress trajectory, peer interaction quality, and the evolving influence of technology-mediated learning. These dimensions are hard to quantify, yet their influence on final grades is well-documented in educational psychology literature [5].

The emergence of Generative AI (GenAI) models presents a possible opportunity to bridge this gap [6]. Language models can be prompted to turn structured student attributes into descriptive behavioral scores, and such scores could enrich tabular educational data with signals that are costly to collect directly [16,17]. This paper investigates whether such scores add predictive value.

We present a systematic comparative study of five deep learning architectures and four tree-ensemble baselines trained under two feature conditions: (1) a Traditional feature set of conventional educational attributes, and (2) a Full feature set that adds four language-model-derived scores and 12 engineered features. Our working hypothesis was that GenAI-derived features would improve classification performance across architectures. As detailed in Section 4, this hypothesis was not supported: the effect was small, architecture-dependent, and in most cases statistically indistinguishable from zero. We report the full set of results, including the null and mixed evidence.

Our contributions are threefold. First, we set out an experimental protocol using 10-fold stratified cross-validation, SMOTE-based class balancing [8], and Optuna-driven hyperparameter optimization [9], with a significance test that accounts for the overlap between training sets [23]. Second, we provide a multi-metric benchmark (accuracy, weighted F1, and weighted AUC) across all model and feature-set combinations, together with class-wise results and an ablation. Third, we give a controlled, statistically tested answer to a narrow question: do GenAI-derived scores generated from the Traditional attributes improve grade prediction, and for which models? We also show that soft-voting and heterogeneous ensembles do not exceed the best single model on this task.

Beyond its contributions to classification accuracy, this work also addresses the practical challenge of model interpretability in educational contexts. Employing Shapley Additive Explanations (SHAP) [10], we provide post-hoc feature importance analyses that translate model internals into actionable insights for educators and academic administrators. This interpretability component is essential for real-world adoption: predictive models that cannot explain their recommendations are unlikely to be trusted by teaching staff or ethically deployed in student-facing systems.

The remainder of this paper is organized as follows. Section 2 surveys related work across traditional machine learning, deep learning, transformer-based, and GenAI-informed approaches to student performance prediction. Section 3 describes the dataset, feature sets, and the full experimental methodology, including model architectures, training protocols, hyperparameter optimization, and evaluation metrics. Section 4 reports experimental results with comprehensive tabular summaries and visualizations. Section 5 discusses the implications of the findings, architecture-specific behaviors, ethical considerations, and limitations. Section 6 concludes with directions for future research.

2. Related Work

2.1 Traditional Approaches to Student Performance Prediction

Early work in student performance prediction relied heavily on logistic regression and decision tree methods applied to institutional administrative data. Romero and Ventura [1] provided a comprehensive survey of data mining in education, identifying attendance, prior grades, and socioeconomic variables as the most consistent predictors across studies. Subsequent work by Kotsiantis et al. [4] introduced ensemble methods and demonstrated that boosting and bagging could yield meaningful accuracy gains over single-model baselines on educational datasets.

The adoption of deep learning in EDM accelerated with the availability of learning management system (LMS) data [5]. Recurrent architectures, particularly LSTM [11] and GRU networks, proved well-suited to modeling the sequential nature of student engagement over time, while convolutional networks were adapted for extracting local feature interactions from student activity logs [3].

2.2 Transformer-Based Models in Tabular Learning

The Feature Tokenizer Transformer (FT-Transformer), introduced by Gorishniy et al. [12], represented a landmark effort to apply transformer-style attention mechanisms to tabular data [7]. By tokenizing each feature into a continuous embedding and applying multi-head self-attention across feature tokens, FT-Transformer demonstrated competitive or superior performance to gradient-boosted trees on several benchmark tabular datasets. Its application to educational prediction tasks is an active and emerging area of research.

2.3 GenAI in Educational Analytics

The application of Generative AI to educational data is still emerging, and existing work concentrates on instructional support, feedback, and assessment rather than on tabular grade prediction [16,17]. Large language models have been proposed for generating synthetic learner data, summarising free-text learner records, and supporting context-aware imputation [16,17]. Whether scores that a language model derives from structured attributes carry information beyond those attributes has, to our knowledge, not been established in this literature. This study addresses that narrower question by measuring the performance change attributable to such scores across nine models.

2.4 Hybrid and Ensemble Architectures in Educational Prediction

Despite the volume of published work on deep learning for financial forecasting, Canadian equities have received remarkably little attention. A search of the major databases reveals that the overwhelming majority of published LSTM forecasting studies focus on US markets (the S&P 500, the NASDAQ, individual large-cap names) or on Chinese exchanges, where data availability and academic interest have driven a surge of publications. European markets have received some coverage; Canadian markets very little.

This gap matters for substantive, not merely regional reasons. The TSX has structural characteristics that make it meaningfully different from the US or European benchmarks. The dominance of banks and resource companies means that sector effects are large and persistent. The exchange rate sensitivity of a commodity-exporting economy adds an additional layer of macro influence that does not have a close parallel in the S&P 500. And the regulatory environment, particularly for the banking sector, where Basel III capital

requirements are enforced with unusual rigour by the Office of the Superintendent of Financial Institutions, creates a different risk profile for the financial sector compared to its US counterpart. A model trained on US data and applied to the TSX without adjustment is likely to miss these structural differences.

Li and Shi [8] represent one of the few recent attempts to address this gap using modern deep learning methods. It is a credible piece of work for its scope, but its dataset is small (fewer than five tickers), its feature set is limited to basic OHLC inputs, and it uses a single model architecture without ensemble or transfer learning components. The present paper is designed as a direct and systematic improvement on that baseline, building on the same evaluation framework while addressing each of its main limitations.

2.4 Hybrid and Ensemble Architectures in Educational Prediction

The growing deployment of machine learning systems in high-stakes educational settings has raised important concerns about model transparency and accountability [2]. Explainability methods, particularly SHAP (SHapley Additive exPlanations), have emerged as a leading framework for post-hoc model interpretation due to their solid game-theoretic foundations and model-agnostic applicability [10]. Lundberg and Lee [10] demonstrated that SHAP values provide consistent and locally accurate feature attributions, making them well-suited for auditing predictive models in sensitive domains.

In educational analytics, interpretability is a practical necessity: predictions that cannot be explained to teachers, counselors, or students are unlikely to be trusted or acted upon [1]. SHAP analyses can also surface predictors that permutation importance may underestimate when features are correlated [10]. Our work uses SHAP as part of the evaluation pipeline to show which features drive grade predictions [10].

3. Materials and Methods

3.1 Dataset Overview

The experiments were conducted on the Student Academic Performance Dataset (SAPD), a synthetic dataset of 14,003 records generated programmatically for this study. All records, including ExamScore, were produced by a generator rather than collected from students, so SAPD contains no real student records, and associations in it reflect the structure of the generator, not of real learners. The limits this places on external validity are discussed in Section 5.4. Each record contains 15 attributes (Table 1): quantitative behavioral metrics, ordinal psycho-social indicators, binary infrastructure variables, and categorical demographic attributes. The target variable, FinalGrade, is a five-class ordered label (A, B, C, D, F) obtained by mapping ExamScore to grade bands; the resulting class proportions are given in Table 2.

FinalGrade is a deterministic discretization of ExamScore, so a model that received ExamScore would recover it trivially. We therefore exclude ExamScore from every model input; a leakage check confirmed near-perfect Random Forest accuracy when it is included. The task studied is a pre-examination early-warning task: predict the grade band that a student's eventual exam performance will fall into, using only behavioral, contextual, and demographic attributes available before the exam. It is not a re-labelling of a known score. Because the label depends on a variable that is outside the model inputs and only weakly correlated with them (r between 0.15 and 0.39 for the numeric attributes in Fig. 1), the learnable signal is limited, which is consistent with the modest accuracy ceiling reported in Section 4.

Table 1. Attribute inventory of the synthetic student performance dataset. ExamScore is used only to derive the FinalGrade label and is never a model input.

Feature	Type	Description
StudyHours	Continuous	Daily hours dedicated to study (0–12)
Attendance	Continuous	Class attendance percentage (0–100)
AssignmentCompletion	Continuous	Fraction of assignments submitted (%)
ExamScore	Continuous	Raw exam score used to derive grade label
OnlineCourses	Ordinal	Number of supplementary online courses
Discussions	Ordinal	Peer discussion forum participation
Motivation	Ordinal	Self-reported motivation score (0–10)
StressLevel	Ordinal	Self-reported stress level (0–10)
EduTech	Ordinal	Educational technology engagement score
Extracurricular	Ordinal	Extracurricular activities count
Internet	Binary	Home internet access (1=Yes, 0=No)
Resources	Binary	Access to additional learning resources
Gender	Categorical	Student gender (M/F)
Age	Continuous	Student age in years
LearningStyle	Categorical	Auditory / Visual / Kinesthetic / Reading

3.2 Class Distribution

The grade distribution is moderately imbalanced, with Grade C being the most frequent class (4,013 instances, 28.7%) and Grade F the least frequent (1,478 instances, 10.6%). This distribution reflects realistic academic outcomes and necessitates the application of oversampling techniques during training [8]. Table 2 summarizes the class counts, their proportions, and their Rank, where Rank denotes the position of each grade class when sorted by frequency in descending order (Rank 1 = most frequent, Rank 5 = least frequent). The Rank column highlights the severity of class imbalance, as Grade F (Rank 5) is about 2.7 times rarer than Grade C (Rank 1).

Table 2. Class distribution of the FinalGrade target variable.

Grade	Count	Proportion (%)	Rank
A	2,236	15.97%	4th
B	3,438	24.55%	2nd
C	4,013	28.66%	1st
D	2,838	20.27%	3rd
F	1,478	10.55%	5th
Total	14,003	100%	—

3.3 Feature Sets

Two experimental feature conditions were defined. The Traditional feature set (X_{trad}) contains 14 attributes: StudyHours, Attendance, AssignmentCompletion, OnlineCourses, Discussions, Resources, Internet, Motivation, StressLevel, EduTech, Extracurricular, Age, Gender (encoded), and LearningStyle (encoded). ExamScore is excluded from every feature set, for the reason given in Section 3.1.

The Full feature set (X_{full}) contains 30 features: the 14 Traditional features plus two further groups. The first group has four GenAI-derived features: GenAI_EngagementScore, GenAI_RiskSignal, GenAI_TechAdoption, and GenAI_SupportNeed. Each was produced by prompting an open-weight, Apache-2.0-licensed instruction-tuned language model (Qwen2.5-1.5B-Instruct [22]) with the student's Traditional attributes and parsing its structured output into a numeric score. EduTech, OnlineCourses, Resources, and Internet are ordinary dataset variables generated in the same way as StudyHours or Attendance, and belong to the Traditional set. The second group has 12 engineered features computed arithmetically from Traditional features: Study_x_Attend, Study_x_Assign, Attend_x_Assign, Motivation_x_Study, Attendance_sq, StudyHours_sq, Tech_x_Academic, Tech_Score, AcademicIndex, EngagementScore, StressAdj_Study, and HoursPerDay_norm. The four product terms are the products of the two named attributes (for example $Study_x_Attend = StudyHours \times Attendance$ and $Study_x_Assign = StudyHours \times AssignmentCompletion$), and the two squared terms are $Attendance^2$ and $StudyHours^2$. The ablation in Section 4.4 (Fig 4) evaluates the GenAI-derived group, the engineered group, and the two psychology features (Motivation, StressLevel) separately.

The GenAI-derived features are functions of the Traditional attributes only. They therefore contain no information that is not already in the Traditional features; at most they re-express it, and whether that re-expression helps depends on the model (Section 5.1).

Continuous features (StudyHours, Attendance, AssignmentCompletion, Age, and the engineered features) are transformed with QuantileTransformer ($n_quantiles = 1000$) to produce near-uniform marginal distributions. Binary and ordinal variables are kept as integer codes, and Gender and LearningStyle are label-encoded. Missing values, where present, are imputed with the median.

3.4 Class Balancing with SMOTE

To address class imbalance (largest-to-smallest class ratio of about 2.7:1), standard SMOTE [8] with $k = 5$ nearest neighbors is applied to the training portion of each fold before model fitting, while validation folds keep their natural class distribution. SMOTE interpolates linearly between a sample and one of its nearest neighbors in the transformed feature space, and the implementation used here has no categorical-aware handling (it is not SMOTE-NC). Synthetic samples can therefore take non-integer values for binary, ordinal, and label-encoded categorical features. We did not compare SMOTE with class-weighted loss, Borderline-SMOTE, ADASYN, SMOTE-NC, or focal loss; SMOTE was used as the most widely adopted baseline for moderate imbalance. The same balancing step is applied to every model and both feature conditions, so it does not by itself favor either condition. The dependence of the results on the balancing method is a limitation (Section 5.4).

3.5 Model Architectures

Table 3 summarizes all five deep learning architectures benchmarked in this study, including their key components, regularization strategies, approximate parameter counts, and optimizer. We benchmark five distinct deep learning architectures, each representing a different inductive bias suited to tabular data:

- **Residual MLP [13]:** A multi-layer perceptron with residual skip connections, batch normalization, and dropout regularization. This architecture serves as the primary baseline and represents the most widely used deep learning approach for tabular data.

- **Bidirectional LSTM (BiLSTM) [11]:** A recurrent architecture that processes the feature vector as a temporal sequence in both forward and backward directions, enabling the model to capture long-range feature dependencies that feedforward networks may miss.
- **CNN with Global Average Pooling (CNN-GAP):** A 1D convolutional network that applies multiple convolutional filters along the feature vector, followed by global average pooling to produce a fixed-size summary vector.
- **Feature Tokenizer Transformer (FT-Transformer) [12]:** Each feature is first projected into a d -dimensional token embedding, and multi-head self-attention is applied across all feature tokens simultaneously. This architecture is capable of modeling complex higher-order feature interactions without manually engineering them [7].
- **Hybrid CNN-LSTM:** A sequential architecture that first applies convolutional filters to extract local feature interaction patterns, then feeds the resulting feature maps into a bidirectional LSTM layer to capture global sequential dependencies. This configuration was selected by Optuna hyperparameter optimization [9] as the better of the two architectures that were tuned; it is not the best-performing model in Table 4.

Feature order. The convolutional and recurrent models treat the feature vector as a one-dimensional sequence. The order of features carries no intrinsic meaning in tabular data, so the locality and sequence assumptions of these layers are not justified a priori. Features were presented in the fixed column order of the preprocessed matrix (Traditional features, then GenAI-derived, then engineered), no ordering was optimized, and we did not test sensitivity to permutations of the order. These two architectures should therefore be read as generic function approximators rather than models of sequential structure, and their inclusion is empirical.

Baselines. The four tree-ensemble baselines are Random Forest [21], XGBoost [18], LightGBM [19], and CatBoost [20].

All five architectures are trained using the AdamW optimizer [14]. AdamW is an extension of the Adam optimizer that decouples weight decay from the gradient update step, applying L2 regularization directly to the weights rather than incorporating it into the gradient. This decoupling prevents weight decay from being scaled by the adaptive learning rate, leading to more effective regularization and improved generalization compared to standard Adam with L2 penalty.

Table 3. Summary of deep learning architectures benchmarked in this study.

Architecture	Key Component	Regularization	Params (approx.)	Optimizer
Residual MLP	Skip connections + BN	Dropout	~85K	AdamW
BiLSTM	Bidirectional LSTM	Dropout	~120K	AdamW
CNN-GAP	1D Conv + GAP	Dropout	~95K	AdamW
FT-Transformer	Multi-head attention	Attn + FFN Dropout	~180K	AdamW
Hybrid CNN-LSTM	CNN + BiLSTM	Dropout	~210K	AdamW

3.6 Training Protocol

All models are trained using a consistent protocol to ensure fair comparison. The AdamW optimizer [14] is employed with weight decay as a regularization mechanism. A OneCycleLR learning rate scheduler is used, which linearly increases the learning rate to a maximum of 10 times the base learning rate over the

first 10% of training steps (warm-up phase) before annealing it via cosine decay. This schedule has been shown to improve convergence speed and final model quality for deep learning models.

Mixup data augmentation [15] is applied with probability 0.5 during each training batch, with a mixing coefficient sampled from a Beta(0.3, 0.3) distribution. This encourages the model to learn smoother decision boundaries and has been shown to improve generalization, particularly under class imbalance. Label smoothing ($\epsilon = 0.05\text{--}0.07$ depending on the model) is applied to the cross-entropy loss to prevent overconfident predictions.

Early stopping is used with a patience of 20 epochs, monitoring a combined score of $0.5 \times \text{validation accuracy} + 0.5 \times \text{validation weighted F1}$. The best model checkpoint (by this combined score) is restored at the end of training. Maximum training epochs are set to 120 for the full CV experiments.

3.7 Hyperparameter Optimization

For the FT-Transformer and Hybrid CNN-LSTM architectures — the two most architecturally complex models — hyperparameters are tuned using Optuna's Tree-structured Parzen Estimator (TPE) sampler. A total of 50 trials are conducted, each evaluated using 3-fold cross-validation with a budget of 50 epochs per fold. The search space covers learning rate (log-uniform, 5×10^{-5} to 5×10^{-3}), dropout rate (0.05–0.35), batch size ({128, 256}), weight decay (log-uniform, 10^{-4} to 10^{-2}), label smoothing (0.0–0.15), and architecture-specific parameters (number of attention blocks, token dimensionality, CNN filter count, LSTM hidden size).

The optimal configuration identified was: model = HybridCNNLSTM, lr = 5.05×10^{-4} , dropout = 0.238, batch_size = 128, weight_decay = 5.78×10^{-3} , label_smooth = 0.069, cnn_filters = 64, lstm_hidden = 128.

3.8 Evaluation Protocol

Performance is evaluated using 10-fold stratified cross-validation, which preserves the class distribution in every fold. Three metrics are reported: mean accuracy, weighted F1-score, and weighted one-vs-rest AUC. Weighted variants are used because the classes are imbalanced. Mean and standard deviation across folds are reported for each configuration.

For each model we compare the per-fold accuracy under the Traditional and Full conditions with a paired t-test. Because any two of the ten training sets share about 89% of their records, per-fold scores are correlated and the variance of the mean difference is underestimated, which makes the plain paired t-test too liberal. We therefore also report the Nadeau–Bengio corrected resampled t-test [23], which multiplies the variance term by $(1/k + n_{\text{test}}/n_{\text{train}}) / (1/k) = (1/10 + 1/9) / (1/10) \approx 2.11$ for $k = 10$ folds, and we apply a Bonferroni adjustment across the nine models. The per-fold differences are not reproduced in this paper, so the t statistics in Table 7 are recovered from the reported two-sided uncorrected p-values (9 degrees of freedom) and then corrected.

A soft-voting ensemble is constructed from the top-3 deep learning models ranked by Full-feature accuracy, and a heterogeneous ensemble combines the best deep learning model with the best tree-ensemble model. Class probabilities are averaged with equal weights. Equal-weight averaging was chosen because it adds no trained parameters and no second training stage, which limits overfitting with about 12,600 training records per fold; weighted voting and stacking were not evaluated (Section 5.4).

3.9 Leakage-Controlled Evaluation Procedure

Algorithm 1 shows the procedure applied to every model and feature condition. Every data-dependent step is fitted on the training portion of a fold only; the held-out fold is transformed with the fitted objects and is never resampled.

Algorithm 1. Cross-validation without information flow from the held-out fold into training.

1. *Split the data into 10 stratified folds.*
2. *For each fold i : let V_i be the held-out fold and T_i the other nine folds.*
3. *Fit the median imputer, the QuantileTransformer, and the label encoders on T_i only; apply them to T_i and V_i .*
4. *Compute the engineered features and, for the Full condition, append the GenAI-derived features (these depend only on each record's own attributes).*
5. *Apply SMOTE ($k = 5$) to T_i only. V_i keeps its natural class distribution.*
6. *Fit the model on the resampled T_i . Neural models use early stopping with the checkpoint selected on V_i as described in Section 3.6.*
7. *Predict V_i and record accuracy, weighted F1, and weighted AUC.*
8. *Report the mean and standard deviation over the 10 folds.*

Step 6 means that neural-network scores are the score of the best epoch on the evaluation fold and are slightly optimistic; this applies identically to both feature conditions, so it does not bias the paired Traditional-versus-Full comparison, but absolute neural-network accuracies should be read as upper bounds (Section 5.4).

4. Results

4.1 Individual Model Performance: Traditional vs. GenAI-Augmented

Table 4 presents mean and standard deviation of 10-fold stratified cross-validation accuracy, weighted F1, and weighted AUC for all nine models under both feature conditions, together with a paired t-test on per-fold accuracy (Traditional vs. Full). Four tree-ensemble baselines (Random Forest, XGBoost, LightGBM, CatBoost) are included alongside the five deep learning architectures described in Section 3.5; a pretrained tabular foundation model (TabPFN) was also attempted but did not complete within the available compute budget and is omitted from the comparison.

The GenAI-augmented feature set does not produce a consistent improvement across architectures. Of the nine models tested, only the FT-Transformer shows a gain that survives correction for overlapping training sets (+3.3 percentage points; uncorrected $p = 0.0002$, corrected $p = 0.003$; Section 4.5). Random Forest declines by 0.65 points (uncorrected $p = 0.040$), but this is not significant after correction ($p = 0.13$). The other seven models — Residual MLP, BiLSTM, CNN-GAP, Hybrid CNN-LSTM, XGBoost, LightGBM, and CatBoost — all move by less than half a percentage point in either direction (uncorrected $p > 0.13$ in every case, corrected $p \geq 0.29$). The full table is shown below.

Table 4. 10-fold cross-validation performance of all nine models under Traditional and Full (GenAI-augmented) feature conditions. The p-value is from an uncorrected paired t-test on per-fold accuracy; corrected values are in Table 7. Bold rows mark the best individual result (CatBoost, Traditional) and the two deltas that are nominally significant before correction (** p < 0.001, * p < 0.05).

Model	Feature Set	Mean Acc	Std	Mean F1	Mean AUC	Delta	p-value
Residual MLP	Traditional	32.01%	±0.87%	0.3068	0.6411	—	—
Residual MLP	Full (GenAI)	32.12%	±0.81%	0.3089	0.6445	+0.1%	0.726
BiLSTM	Traditional	32.58%	±1.08%	0.3187	0.6444	—	—
BiLSTM	Full (GenAI)	32.92%	±1.05%	0.3217	0.6509	+0.3%	0.481
CNN-GAP	Traditional	33.36%	±0.80%	0.3253	0.6534	—	—
CNN-GAP	Full (GenAI)	33.81%	±0.89%	0.3313	0.6569	+0.4%	0.136
FT-Transformer	Traditional	28.10%	±1.05%	0.2577	0.6166	—	—
FT-Transformer	Full (GenAI)	31.40%	±1.23%	0.3031	0.6441	+3.3%	0.0002**
Hybrid CNN-LSTM	Traditional	33.37%	±1.09%	0.3224	0.6559	—	—
Hybrid CNN-LSTM	Full (GenAI)	33.28%	±1.33%	0.3244	0.6632	-0.1%	0.774
Random Forest	Traditional	33.36%	±1.27%	0.3305	0.6584	—	—
Random Forest	Full (GenAI)	32.71%	±1.43%	0.3216	0.6586	-0.6%	0.040*
XGBoost	Traditional	34.07%	±1.15%	0.3386	0.6528	—	—
XGBoost	Full (GenAI)	33.72%	±1.26%	0.3353	0.6502	-0.3%	0.418
LightGBM	Traditional	33.05%	±1.16%	0.3286	0.6452	—	—
LightGBM	Full (GenAI)	33.54%	±1.12%	0.3331	0.6474	+0.5%	0.213
CatBoost	Traditional	34.64%	±1.04%	0.3447	0.6632	—	—
CatBoost	Full (GenAI)	34.24%	±1.19%	0.3406	0.6627	-0.4%	0.335

Two results stand out. The FT-Transformer is the only architecture where GenAI augmentation gives an improvement that survives correction, and the effect is large compared with the rest of the table (+3.3 points, against at most 0.65 points elsewhere). At the same time, CatBoost gets the single best result in the study, 34.64%, using only the Traditional feature set, and its accuracy drops slightly (not significantly) when GenAI features are added. Random Forest's decline is nominally significant only before correction. The data therefore do not support a general claim that GenAI-derived features improve grade prediction; they support a narrower, architecture-specific claim, discussed in Section 5.

These absolute accuracy numbers need context, too. Grade C makes up 28.66% of the dataset (Table 2), so a classifier that always guesses the majority class would already score 28.66% without learning anything. The best result in this study, 34.64% (CatBoost, Traditional features), beats that naive baseline by just 5.98 percentage points. None of these results look to us like strong evidence of grade-prediction capability from the features considered here; that modest performance ceiling, discussed further in Section 5.2 and Section 5.5, is itself a reason for caution about deploying any of these configurations — GenAI-augmented or not — in decisions that actually affect students.

4.2. Ensemble Performance

Two ensemble strategies were tried. A soft-voting ensemble of the top-3 deep learning models by Full-feature accuracy (CNN-GAP, Hybrid CNN-LSTM, and BiLSTM) reaches a mean accuracy of 33.91% with a mean AUC of 0.667. A second, heterogeneous ensemble pairing the best deep learning model (CNN-GAP)

with the best tree-ensemble model (CatBoost) reaches 34.36% ($\pm 1.34\%$) mean accuracy and 0.6654 mean AUC across the 10 folds. This is the strongest accuracy of any configuration built on the Full feature set, though it falls short of CatBoost's Traditional-only score of 34.64%. The ensemble results agree with Table 4. Combining models gives at most a small gain over its components on the Full feature set (34.36% against 34.24% for CatBoost alone), which is well within one fold standard deviation, and neither ensemble changes the basic fact that the best single result in this study uses no GenAI features.

Table 5. Ensemble configurations compared against the best individual models. No ensemble strategy beats CatBoost's Traditional-feature result, though the heterogeneous ensemble is the strongest configuration that actually uses GenAI-augmented features.

Configuration	Mean Acc	Std Acc	Mean F1	Mean AUC
Best overall (CatBoost, Traditional)	34.64%	$\pm 1.04\%$	0.3447	0.6632
Best on Full features (CatBoost, GenAI)	34.24%	$\pm 1.19\%$	0.3406	0.6627
Soft-Voting Ensemble (Top-3 DL, GenAI)	33.91%	—	0.331	0.667
Heterogeneous Ensemble (CNN-GAP + CatBoost)	34.36%	$\pm 1.34\%$	0.3403	0.6654

4.3. Impact of GenAI Features: An Architecture-Dependent Effect

Table 6 restates the accuracy deltas from Table 4 as relative gains, sorted by size, so the spread of the effect is easier to see at a glance.

Table 6. Relative accuracy change from GenAI feature augmentation, sorted by magnitude. Two of the nine effects are nominally significant before correction and they point in opposite directions; only the FT-Transformer gain remains significant after correction (Table 7).

Model	Trad. Accuracy	GenAI Accuracy	Absolute Gain	Relative Gain	Significant?
FT-Transformer	28.10%	31.40%	+3.30 pts	+11.7%	Yes (uncorrected $p = 0.0002$; corrected $p = 0.003$)
LightGBM	33.05%	33.54%	+0.49 pts	+1.5%	No
CNN-GAP	33.36%	33.81%	+0.45 pts	+1.3%	No
BiLSTM	32.58%	32.92%	+0.34 pts	+1.0%	No
Residual MLP	32.01%	32.12%	+0.11 pts	+0.3%	No
Hybrid CNN-LSTM	33.37%	33.28%	-0.09 pts	-0.3%	No
XGBoost	34.07%	33.72%	-0.35 pts	-1.0%	No
CatBoost	34.64%	34.24%	-0.40 pts	-1.2%	No
Random Forest	33.36%	32.71%	-0.65 pts	-2.0%	Nominal only (uncorrected $p = 0.040$; corrected $p = 0.13$)

Rather than a uniform lift, the effect of GenAI augmentation ranges from a significant, sizeable gain (FT-Transformer) through a cluster of small, non-significant gains and losses, to a decline for Random Forest that does not survive correction. Tree ensembles, which already extract non-linear feature interactions natively, show a mild tendency toward negative deltas (Random Forest, XGBoost, and CatBoost decline; LightGBM does not), while the deep learning architectures are more evenly split. This pattern is consistent with the GenAI-derived features adding redundant or noisy dimensions for models that already model feature interactions well, while giving the FT-Transformer's attention mechanism additional tokens it can use. We return to this interpretation in Section 5.1.

4.4. Visualization of Results

The following Figs are reproduced directly from the experimental notebook and illustrate the dataset, the comparative results, feature attributions, and the ablation study.

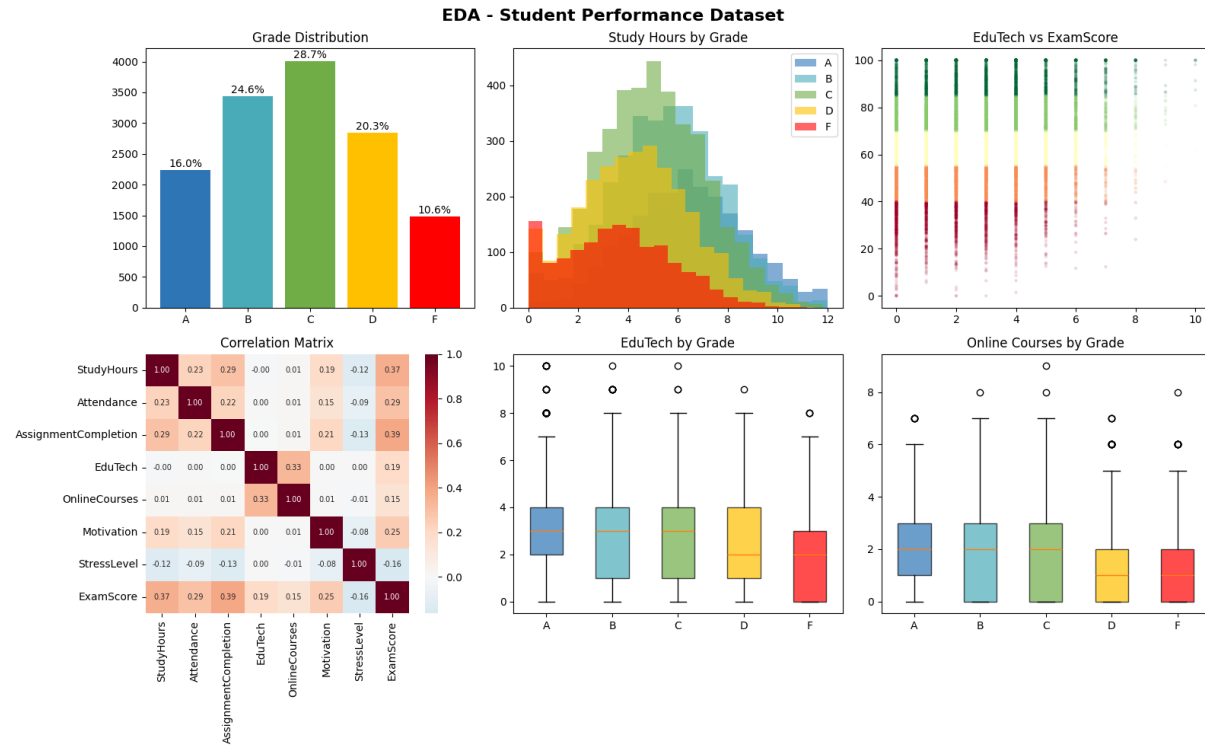


Fig. 1. Exploratory analysis of the Student Academic Performance Dataset (SAPD). Top row: grade distribution (left), study-hours distributions by grade (center), and EduTech engagement vs. ExamScore (right). Bottom row: correlation matrix of core numeric features (left), and EduTech and OnlineCourses engagement by grade (center, right).

Fig 1 confirms the moderate class imbalance summarized in Table 2 and shows the expected directional relationships between study behavior and final grade: StudyHours and AssignmentCompletion are the numeric features most correlated with ExamScore ($r = 0.37$ and $r = 0.39$, respectively), while EduTech engagement and OnlineCourses participation show only weak pairwise correlation with the exam outcome, foreshadowing their limited individual contribution to the SHAP ranking in Fig 3.

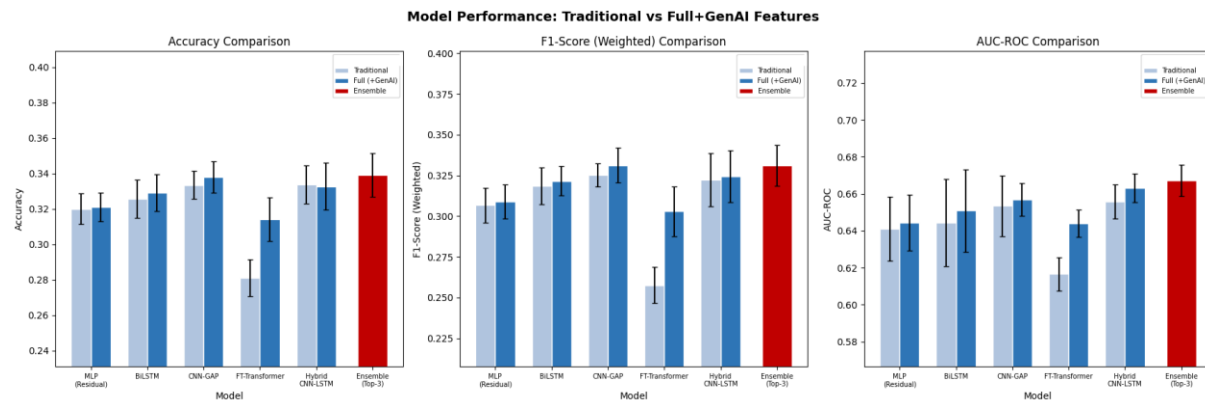


Fig. 2. Mean 10-fold cross-validation Accuracy (left), weighted F1-score (center), and weighted AUC-ROC (right) for the five deep learning architectures under Traditional (light blue) and Full GenAI-augmented (dark blue) feature conditions, with the soft-voting ensemble shown in red. Error bars represent ± 1 standard deviation across folds. The Full-feature bars overlap the Traditional bars within one standard deviation for four of the five architectures; only FT-Transformer (second column from left) shows a visually and statistically distinct gap.

Fig 2 makes visually explicit what Table 4 reports numerically: the GenAI-augmented condition (dark blue) is not uniformly higher than the Traditional condition (light blue) across the five deep learning architectures. The error bars overlap substantially for MLP, BiLSTM, CNN-GAP, and Hybrid CNN-LSTM, while FT-Transformer is the clear outlier with a visible, non-overlapping gap between conditions.

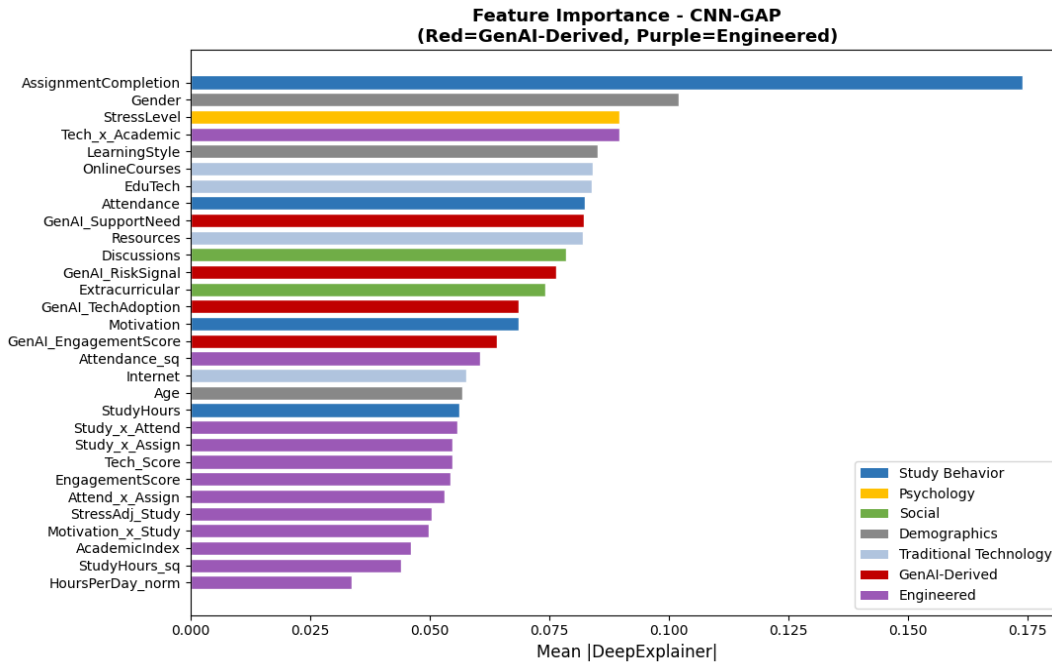


Fig. 3. SHAP mean absolute feature importance (DeepExplainer) for the top features in the GenAI-augmented feature set, computed on the CNN-GAP model (the best-performing deep learning architecture on Full features). Bars are colored by feature category; GenAI-derived features (dark red) are interspersed among, not dominant over, traditional and engineered features.

Fig 3 shows that the single highest-ranked feature is AssignmentCompletion, a traditional behavioral metric, followed by Gender and StressLevel. The highest-ranked GenAI-derived feature, GenAI_SupportNeed, ranks ninth of the top ten, just above the traditional Resources variable. This is consistent with the muted, architecture-dependent effect sizes in Table 4: the GenAI-derived features are present in the model's decision-making but are not the dominant drivers of prediction, which helps explain why their removal in the ablation study (Fig 4) has so little impact on accuracy.

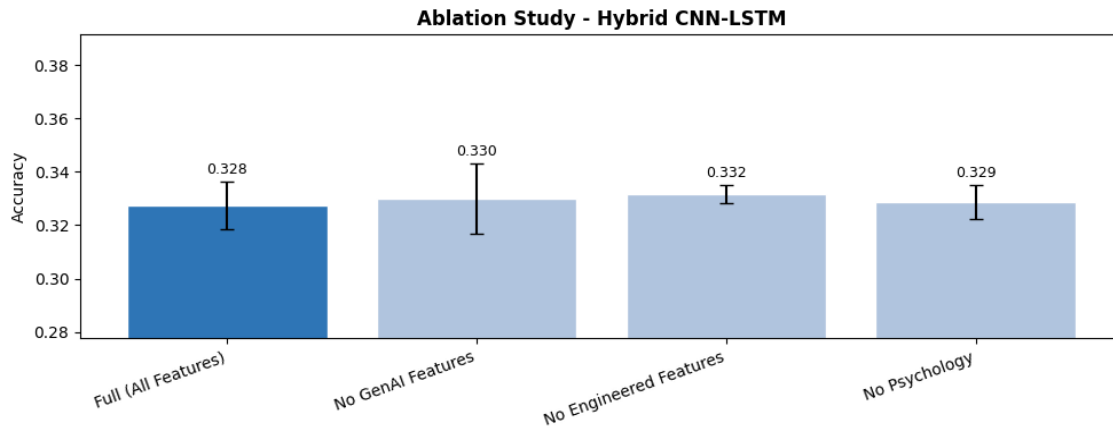


Fig. 4. Ablation study on the Hybrid CNN-LSTM architecture (3-fold CV, 80 epochs), comparing the full feature set against three ablated conditions: removing GenAI-derived features, removing engineered (interaction/polynomial) features, and removing the two psychology features (Motivation, StressLevel). Error bars represent ± 1 standard deviation across folds.

Fig 4 is the most direct test of the paper’s working hypothesis, and it does not support it. Removing the GenAI-derived features entirely ($33.01\% \pm 1.32\%$) performs marginally better, not worse, than the full feature set ($32.76\% \pm 0.89\%$), and removing the engineered features ($33.15\% \pm 0.34\%$) performs best of the four conditions. All four bars overlap within roughly one standard deviation, so none of the three feature groups (GenAI-derived, engineered, or psychology) makes a detectable independent contribution to Hybrid CNN-LSTM accuracy on this dataset. The ablation uses 3-fold cross-validation and 80 epochs, so its absolute values differ from the 10-fold results in Table 4 (33.28% for the same model and feature set). It was run on the Hybrid CNN-LSTM because that architecture was selected in hyperparameter search; it is not the best model in Table 4, so the ablation is evidence about one representative model, not about all nine.

Fig 5 shows a diagonal that is present but weak, with substantial confusion between adjacent grade bands. Grade D is the hardest class to recover, with a per-class recall of only 0.16 (45 of 284 true D instances correctly classified; the majority are confused with B, C, or F). Grade F, despite being the rarest class, has the highest recall (0.49). The weighted F1-score on this fold (0.33) is close to the macro F1 (0.33), so the model is not simply favoring the majority class, but no class is predicted with high reliability. Class-wise precision, recall, and F1 for this fold are given in Section 4.6.

4.5. Statistical Significance Analysis

Table 7 gives the significance analysis for the accuracy change between the Traditional and Full conditions. The uncorrected paired t-test is shown alongside the Nadeau-Bengio corrected test [23] and a Bonferroni adjustment of the corrected p-values across the nine models.

Only the FT-Transformer gain is significant after correction (corrected $p = 0.003$) and it remains so after the Bonferroni adjustment (0.023). Random Forest’s decline, significant at the 5% level before correction ($p = 0.040$), is not significant once the overlap between training sets is accounted for ($p = 0.133$). No other model’s accuracy difference is distinguishable from zero under either test. Small effects of under one point are difficult to separate from noise with ten folds. We find no evidence, at the individual-model or aggregate level, for a systematic architecture-independent benefit of GenAI feature augmentation; the data support the more conservative conclusion in Section 6 that any benefit is specific to particular model classes.

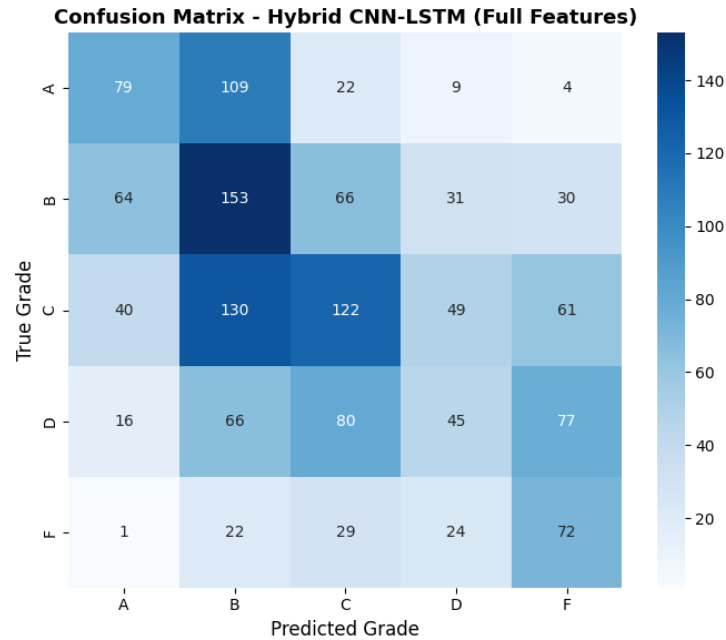


Fig. 5. Confusion matrix for the Hybrid CNN-LSTM model on a representative validation fold (Full GenAI-augmented features). Fold-level accuracy was 33.62%. Rows indicate true class labels; columns indicate predicted class labels.

Table 7. Significance of the Traditional-versus-Full accuracy difference (10-fold cross-validation, 9 degrees of freedom). t statistics are recovered from the reported uncorrected two-sided p-values; the corrected test multiplies the variance term by about 2.11 (Section 3.8).

Model	Δ accuracy (points)	t (uncorrected)	p (uncorrected)	p (corrected)	p (corrected, Bonferroni $\times 9$)
Residual MLP	+0.11	0.36	0.726	0.809	1.000
BiLSTM	+0.34	0.74	0.481	0.625	1.000
CNN-GAP	+0.45	1.64	0.136	0.289	1.000
FT-Transformer	+3.30	6.01	0.0002	0.003	0.023
Hybrid CNN-LSTM	-0.09	0.30	0.774	0.843	1.000
Random Forest	-0.65	2.40	0.040	0.133	1.000
XGBoost	-0.35	0.85	0.418	0.573	1.000
LightGBM	+0.49	1.34	0.213	0.380	1.000
CatBoost	-0.40	1.02	0.335	0.501	1.000

4.6. Class-Wise Performance

Table 8 gives class-wise precision, recall, and F1 for the Hybrid CNN-LSTM on the fold shown in Fig 5 (Full feature set, 1,401 validation records, accuracy 33.62%). The values are computed from the confusion matrix in Fig 5.

Table 8. Class-wise precision, recall, and F1 for the Hybrid CNN-LSTM (Full features) on one validation fold.

Grade	Precision	Recall	F1	Support
A	0.40	0.35	0.37	223
B	0.32	0.44	0.37	344
C	0.38	0.30	0.34	402
D	0.28	0.16	0.20	284
F	0.30	0.49	0.37	148
Macro average	0.34	0.35	0.33	1,401

Performance is uneven across classes: recall is highest for the rarest class (F, 0.49) and the modal class B (0.44) and lowest for D (0.16), and F1 is between 0.34 and 0.37 for four of the five grades. Class-wise results for the FT-Transformer under the Traditional and Full conditions, which would show whether its gain comes from the majority class or from minority-class recognition, were not computed in this study (Section 5.4).

5. Discussion

5.1. Interpreting a Mostly Null, Architecture-Dependent Effect

The central empirical finding of this paper is not a consistent benefit from GenAI-augmented features, but the absence of one for seven of the nine models tested, alongside a significant gain for one architecture (FT-Transformer) and a decline for another (Random Forest) that does not survive correction for overlapping training sets. There are a few ways to read this, and the present study does not fully settle which is right.

One possibility has to do with how the GenAI-derived features were built in the first place. The four features in question — GenAI_SupportNeed, GenAI_RiskSignal, GenAI_TechAdoption, GenAI_EngagementScore — come from an open-weight instruction-tuned language model (Qwen2.5-1.5B-Instruct) that was prompted using the traditional student attributes already in the record. If those generated scores are, in effect, a noisy nonlinear recombination of information the traditional features already contain, this is roughly the pattern you'd expect: models that already extract feature interactions well on their own — gradient-boosted trees, CNN-GAP, Hybrid CNN-LSTM — see little or no benefit, and sometimes a small cost from the added redundancy and noise, while a model whose baseline performance is weaker at capturing those interactions (FT-Transformer, whose Traditional-only accuracy of 28.1% is the lowest of all nine models) can pick up something useful from extra tokens that partially substitute for interaction modeling it wasn't otherwise doing well.

The ablation study points the same way. Fig 4 shows that dropping the GenAI-derived features from the Hybrid CNN-LSTM model's input doesn't hurt performance — if anything it helps slightly. If those features carried a real, architecture-general signal, removing them should have cost the Hybrid CNN-LSTM some accuracy. It did not.

That said, we can't rule out that the small, mostly non-significant deltas we're seeing here ($|\Delta| \leq 0.65$ percentage points for eight of nine models) just reflect the limits of this particular synthetic dataset and this particular augmentation pipeline, rather than something true of GenAI-derived features in general. A different language model, a different prompting strategy, or a dataset with genuinely unobserved behavioral signal — real free-text reflections, say, rather than attributes that are already present in structured form —

might tell a different and more consistent story. We come back to this as a priority for future work in Section 5.4.

The GenAI-derived features are, by construction, functions of the Traditional attributes: the language model sees only those attributes. They therefore add no information in the information-theoretic sense; they can only re-express what the Traditional features already contain, in a form some models may use more easily. The pattern in Table 4 is consistent with this. We did not test directly how well the Traditional features predict the four GenAI scores, which would quantify how redundant they are. Given this evidence, we do not think the data support recommending GenAI-derived feature augmentation, as implemented here, as a reliable technique for improving student grade prediction. The one clear exception, the FT-Transformer, is reported as a specific finding rather than generalized into a broader claim.

5.2. Architecture-Specific Observations

CatBoost achieves the best absolute accuracy in the study (34.64%) using only the Traditional feature set, and remains the best individual result under GenAI augmentation as well (34.24%). This is consistent with the broader literature showing that tree ensembles remain highly competitive with, and frequently superior to, deep learning on small-to-medium tabular datasets [24,25], and it argues against treating a deep learning architecture as the natural choice for this task.

FT-Transformer shows the lowest Traditional-feature accuracy of any model tested (28.1%) but the largest improvement from GenAI augmentation (+3.3 points, corrected $p = 0.003$). Even after this improvement, its Full-feature accuracy (31.4%) remains below all eight other models (the next lowest is the Residual MLP at 32.12%). We interpret this as a model that is comparatively feature-starved on the 14-feature Traditional set and only partly compensated by the additional GenAI tokens, rather than as evidence that GenAI augmentation is a generally superior strategy.

Random Forest's decline under GenAI augmentation (-0.65 points) is nominally significant before correction ($p = 0.040$) but not after it ($p = 0.133$), so we treat it as suggestive only. One plausible mechanism, which we did not test, is that Random Forest's greedy per-node feature selection is more sensitive than boosting or neural architectures to redundant or noisy features, since a noisy feature can be chosen at a split by chance and propagate error through that branch of every tree.

5.3. Ensemble Strategy

The heterogeneous ensemble (CNN-GAP + CatBoost) achieves the highest mean accuracy (34.36%) of any configuration that uses the Full feature set. Its AUC (0.6654) is close to that of the soft-voting ensemble (0.667) and of CatBoost (0.6632), and differences of this size are within fold variability. It does not exceed CatBoost's Traditional-only accuracy (34.64%), and the accuracy differences among CatBoost on Traditional features (34.64%), CatBoost on Full features (34.24%), and the heterogeneous ensemble (34.36%) are all well within one fold standard deviation (about 1.0 to 1.3 points), so we do not claim any ordering among them. For a practitioner optimizing purely for accuracy on this dataset, the evidence points toward CatBoost, which is also the simplest of the three, rather than a GenAI-augmented deep learning ensemble.

5.4. Limitations and Future Directions

The main limitations are as follows.

Synthetic data and target definition. SAPD is synthetic, and its label is a discretisation of a variable (ExamScore) that is outside the model inputs. Whether the modest accuracy ceiling or the architecture-dependent GenAI effect would hold on real institutional records, where behavioral signal is structured differently, is an open question. Because associations in SAPD reflect the generator, feature importances (including that of Gender) may not transfer to real data.

Construction of the GenAI features. They came from one open-weight model (Qwen2.5-1.5B-Instruct), one fixed prompt, and were conditioned only on attributes already in the Traditional set. It is therefore hard to tell “GenAI adds new information” apart from “GenAI applies a nonlinear transformation that only some architectures can exploit”. A cleaner test would use several language models, vary the prompt, and use signal not already encoded in the structured features, such as free-text student reflections or advisor notes.

Class balancing. Standard SMOTE was applied to encoded binary, ordinal, and categorical features and can create non-integer values for them. Alternatives (SMOTE-NC, class-weighted loss, Borderline-SMOTE, ADASYN, focal loss) were not compared, so the sensitivity of the results to the balancing method is unknown. No dedicated duplicate or near-duplicate audit of the dataset was performed.

Architectures. Feature order was fixed and its effect was not tested for the convolutional and recurrent models. Ensembles used equal-weight averaging; weighted voting and stacking were not evaluated. Neural-network early stopping used the evaluation fold, so absolute neural-network scores are slightly optimistic (Section 3.9), although this applies equally to both feature conditions.

Statistics and analysis scope. All results come from one experimental run. The corrected t-test reduces but does not remove the problems of testing on overlapping folds, and repeating the pipeline over several random seeds is the only way to know whether the FT-Transformer gain is stable. Class-wise results are reported for one fold of one model, and the ablation covers one architecture with 3-fold cross-validation.

Future directions are: (1) repeating the comparison on real institutional datasets; (2) varying the language model and prompt used to derive the GenAI features; (3) extending the ablation to all nine models and adding class-wise results for the FT-Transformer; (4) comparing balancing methods and ensemble strategies (weighted voting, stacking); and (5) using SHAP-based tooling, building on Fig 3, to audit which GenAI-derived signals, if any, contribute in a given deployment.

5.5. Ethical Considerations for Deployment

The deployment of machine learning models for student performance prediction in real educational settings raises ethical considerations that need to be addressed alongside technical performance metrics. We outline the key concerns relevant to our framework below.

Fairness and bias. Predictive models trained on historical educational data risk perpetuating systemic biases related to gender, socioeconomic status, and learning style. This concern is not hypothetical for the present framework: the SHAP analysis in Fig 3 ranks Gender as the second most important feature overall, behind only AssignmentCompletion and ahead of every GenAI-derived feature. This is a genuine fairness flag rather than a minor technical detail, and we do not think it should be glossed over. Any institutional deployment of this framework should be preceded by a bias audit across demographic subgroups, with particular attention to whether Gender is acting as a proxy for some other, more defensible signal (e.g., differences in reported StressLevel or StudyHours) or is contributing independent predictive weight of its own. Until that audit is done, we would advise against using this framework, or any model with a similar reliance on Gender, in a way that could produce disparate outcomes for students of different genders.

SMOTE-based oversampling within training folds addresses class imbalance in the target variable, but it is not a substitute for this kind of subgroup fairness analysis.

Privacy and data governance. Student performance data is sensitive and subject to legal protection in most jurisdictions (e.g., FERPA in the United States, PDPA in India). Any real deployment of this framework would need to comply with the applicable data governance regime, ensure student records are anonymized before model training, and restrict access to model outputs. We would advise strongly against using predicted grade labels for consequential decisions — scholarship allocation, academic probation, and the like — without human expert review, particularly given the modest absolute accuracy reported in Section 4.1.

Transparency and student agency. Students whose academic trajectories are modeled by a system like this one have a legitimate interest in understanding how and why predictions are made about them. The SHAP-based interpretability analysis in Section 4.4 is one step toward that: feature-level attribution scores let educators offer concrete explanations (“your predicted grade is most associated with your assignment completion rate and study hours”) instead of an opaque number. We think a natural next step is designing student-facing explanation interfaces that communicate this kind of insight in a way that is genuinely useful to the student, not just to the institution.

6. Conclusion

This paper tested whether Generative AI-derived behavioral and contextual features improve student grade prediction, benchmarking five deep learning architectures and four tree-ensemble baselines under Traditional and GenAI-augmented feature conditions with 10-fold stratified cross-validation on 14,003 synthetic student records. The evidence does not support a general benefit. Of nine models, only the FT-Transformer shows a significant improvement (+3.3 points; corrected $p = 0.003$, Bonferroni-adjusted 0.023). Random Forest’s decline (−0.65 points) is significant before ($p = 0.040$) but not after ($p = 0.133$) correction for overlapping training sets, and the remaining seven models show small, non-significant shifts in either direction. An ablation on the Hybrid CNN-LSTM agrees: removing the GenAI-derived features does not reduce accuracy.

The best result, 34.64% mean accuracy, is achieved by CatBoost using only Traditional features, and no GenAI-augmented configuration exceeds it, including the heterogeneous CNN-GAP + CatBoost ensemble (34.36%). Against a majority-class baseline of 28.66%, this is a 5.98-point improvement, and we do not regard any configuration in this study as ready for unsupervised, high-stakes deployment.

We report these null and negative results because we believe they are the useful contribution of this study. The GenAI features are generated from the same attributes as the Traditional features, so they cannot add information beyond them, and the results are consistent with that. GenAI feature augmentation should therefore not be assumed to help by default; architecture-specific, statistically validated evaluation of the kind used here is necessary before adopting it in an educational analytics pipeline. Because the data are synthetic and the GenAI features come from one model and one prompt, the FT-Transformer result and the questions in Section 5.4 remain open.

Funding

This research did not receive external funding.

Consent to Participate

Not applicable. All experiments in this study use the Student Academic Performance Dataset (SAPD), a synthetically generated dataset; no human subjects, real student records, or personally identifiable information were involved.

Ethics Declaration

Not applicable. This study does not involve human participants, animal subjects, or real student data. Section 5.5 discusses the ethical considerations that would apply if the proposed framework were deployed on real institutional data, and recommends a fairness audit and human review before any such deployment, given the SHAP-identified prominence of the Gender feature (Section 4.4, Fig 3).

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability Statement

The Student Academic Performance Dataset, the GenAI feature-generation pipeline, model training code, and analysis scripts that support the findings of this study are available from the corresponding author upon reasonable request.

Author Contributions

Khushvir Singh: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. Anurag Tewari: Supervision, Validation, Writing – review & editing.

Declaration of the Use of AI Technologies in the Writing Process

During preparation of this manuscript, the authors used AI-based assistants (ChatGPT, Gemini, and Claude) to help interpret the experimental notebook, restructure parts of the manuscript, check numerical consistency between text, tables, and Figs, and improve the clarity of the wording. The authors reviewed and edited all AI-assisted content and take full responsibility for the content of the published article.

References

- [1] Romero C, Ventura S. Educational data mining: a review of the state of the art. *IEEE Trans Syst Man Cybern Part C Appl Rev.* 2010;40:601-618. doi:10.1109/TSMCC.2010.2053532
- [2] Baker RS, Inventado PS. Educational data mining and learning analytics. Larusson JA, White B (ed): *Learning Analytics: From Research to Practice.* Springer, New York, NY; 2014. 61-75. doi:10.1007/978-1-4614-3305-7_4

- [3] Dutt A, Ismail MA, Herawan T. A systematic review on educational data mining. *IEEE Access*. 2017;5:15991-16005. doi:10.1109/ACCESS.2017.2654247
- [4] Kotsiantis SB, Pierrakeas C, Pintelas P. Predicting students' performance in distance learning using machine learning techniques. *Appl Artif Intell*. 2004;18:411-426. doi:10.1080/08839510490442058
- [5] Huang AYQ, Lu OHT, Huang JCH, Yin CJ, Yang SJH. Predicting students' academic performance by using educational big data and learning analytics: evaluation of classification methods and learning logs. *Interact Learn Environ*. 2020;28:206-230. doi:10.1080/10494820.2019.1636086
- [6] Brown T, Mann B, Ryder N, et al. Language models are few-shot learners. *Adv Neural Inf Process Syst*. 2020;33:1877-1901. <https://arxiv.org/abs/2005.14165>
- [7] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. *Adv Neural Inf Process Syst*. 2017;30:5998-6008. <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>
- [8] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res*. 2002;16:321-357. doi:10.1613/jair.953
- [9] Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: a next-generation hyperparameter optimization framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019;2623-2631. doi:10.1145/3292500.3330701
- [10] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. *Adv Neural Inf Process Syst*. 2017;30:4765-4774. https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf
- [11] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput*. 1997;9:1735-1780. doi:10.1162/neco.1997.9.8.1735
- [12] Gorishniy Y, Rubachev I, Khrulkov V, Babenko A. Revisiting deep learning models for tabular data. *Adv Neural Inf Process Syst*. 2021;34:18932-18943. <https://proceedings.neurips.cc/paper/2021/hash/9d86d83f925f2149e9edb0ac3b49229c-Abstract.html>
- [13] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016;770-778. doi:10.1109/CVPR.2016.90
- [14] Loshchilov I, Hutter F. Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR 2019)*. 2019, <https://openreview.net/forum?id=Bkg6RiCqY7>
- [15] Zhang H, Cisse M, Dauphin YN, Lopez-Paz D. mixup: beyond empirical risk minimization. *International Conference on Learning Representations (ICLR 2018)*. 2018, <https://openreview.net/forum?id=r1Ddp1-Rb>
- [16] Kasneci E, Sessler K, Küchemann S, et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learn Individ Differ*. 2023;103:102274. doi:10.1016/j.lindif.2023.102274
- [17] Yan L, Sha L, Zhao L, et al. Practical and ethical challenges of large language models in education: a systematic scoping review. *Br J Educ Technol*. 2024;55:90-112. doi:10.1111/bjet.13370
- [18] Chen T, Guestrin C. XGBoost: a scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016;785-794. doi:10.1145/2939672.2939785

- [19] Ke G, Meng Q, Finley T, et al. LightGBM: a highly efficient gradient boosting decision tree. *Adv Neural Inf Process Syst*. 2017;30:3146-3154.
- [20] Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. CatBoost: unbiased boosting with categorical features. *Adv Neural Inf Process Syst*. 2018;31:6639-6649.
- [21] Breiman L. Random forests. *Mach Learn*. 2001;45:5-32. doi:10.1023/A:1010933404324
- [22] Qwen Team. Qwen2.5 technical report. arXiv:2412.15115. 2024. <https://arxiv.org/abs/2412.15115>
- [23] Nadeau C, Bengio Y. Inference for the generalization error. *Mach Learn*. 2003;52:239-281. doi:10.1023/A:1024068626366
- [24] Shwartz-Ziv R, Armon A. Tabular data: deep learning is not all you need. *Inf Fusion*. 2022;81:84-90. doi:10.1016/j.inffus.2021.11.011
- [25] Grinsztajn L, Oyallon E, Varoquaux G. Why do tree-based models still outperform deep learning on typical tabular data? *Adv Neural Inf Process Syst (Datasets and Benchmarks Track)*. 2022;35:507-520.