Nexus Edge: Student Performance Prediction using a Deterministic Linear Model

Ananya V[1], K Hrishikesh[2], Neha K Karatekar[3], Prathik SR[4], Dr. Arjun U[5]
[1,2,3,4]*Department of Computer Engineering, PES Institute of Technology and Management, Shimoga, India*
[5]*Head of Department of Computer Engineering, PES Institute of Technology and Management, Shimoga, India*

***Abstract*—Student performance prediction is a critical component of modern educational management systems. While machine learning-based approaches have demonstrated high predictive accuracy, they introduce significant complexity in terms of data requirements, computational overhead, and model interpretability. This paper presents Nexus Edge; a lightweight student performance prediction system grounded in a deterministic linear model. The model computes a performance score through a heuristic linear scoring function: $y = 0.55x_1 + 2.5x_2 + 2.0x_3$, where x_1 denotes previous examination marks, x_2 denotes weekly study hours, and x_3 denotes the Cumulative Grade Point Average (CGPA). The computed score is clipped to the interval [0, 100] and classified into four discrete performance bands. Unlike machine learning methods, this model requires no training data, has no convergence dependencies, and produces fully interpretable, auditable outputs. The system is specifically designed for data-scarce academic environments and real-time prediction contexts where simplicity, transparency, and low computational overhead are paramount. Conceptual evaluation confirms the model's consistency, determinism, and practical utility as a principled baseline performance predictor.**

***Index Terms*—student performance prediction; deterministic linear model; heuristic linear scoring function; educational data mining; interpretable model; academic analytics**

I. INTRODUCTION

The accurate prediction of student academic performance has emerged as a central research problem within the domain of Educational Data Mining (EDM) and Learning Analytics (LA). Institutions that can identify at-risk students at an early stage are better positioned to deploy targeted interventions, optimize resource allocation, and improve overall academic outcomes. The demand for such predictive capabilities has accelerated in recent years, driven by the proliferation of digital learning platforms and the increasing availability of student activity data [1].

Contemporary approaches to student performance prediction predominantly rely on supervised machine learning algorithms, including Random Forests, Support Vector Machines (SVMs), artificial neural networks, and ensemble methods. These approaches, while demonstrably powerful in large-scale data settings, carry substantial prerequisites: they require labeled training datasets of sufficient size, computationally intensive training pipelines, and extensive hyperparameter tuning. Furthermore, the resulting models often sacrifice interpretability in pursuit of marginal accuracy gains, producing "black-box" predictors whose outputs are difficult for educators and administrators to audit or explain [2].

In contrast, a wide class of practical deployment scenarios in academic institutions—particularly those in resource-constrained environments demands a different set of trade-offs. These scenarios are characterized by: (i) the absence of large historical datasets, (ii) the need for real-time or near-real-time predictions without batch training latency, and (iii) the requirement for explainable outputs that can be directly communicated to students and faculty.

This paper introduces Nexus Edge, a student performance prediction system based on a deterministic linear model. The core of the system is a heuristic linear scoring function that combines three primary academic indicators previous marks, study hours, and CGPA using domain-informed weights.

The model does not require training, eliminates uncertainty from stochastic optimization, and produces identical outputs for identical inputs under all execution conditions. The system classifies predicted scores into four actionable performance bands: Critical, Needs Improvement, Good but Can Improve, and Excellent.

The principal contribution of this work is the design and evaluation of a principled, interpretable baseline model that is explicitly positioned as an alternative to data-intensive machine learning pipelines for the specific context of data-scarce, real-time academic prediction. The system is framed not as a competitor to state-of-the-art machine learning methods, but as a complement providing an immediately deployable, fully transparent, and computationally trivial predictor for environments where machine learning is impractical.

The remainder of this paper is structured as follows. Section II surveys relevant literature in student performance prediction. Section III details the mathematical formulation and design rationale of the deterministic model. Section IV describes the system architecture. Section V presents conceptual evaluation and discussion. Section VI and VII provide conclusions and future work directions. Section VIII lists all references.

II. LITERATURE SURVEY

A. Foundational Works in Educational Data Mining

The use of data mining and predictive modeling in educational contexts has a well-established research history. Cortez and Silva [1] presented one of the most influential early empirical studies in this domain, applying decision trees, random forests, neural networks, and SVMs to predict secondary school student performance using demographic, social, and academic features from Portuguese secondary schools. Their work systematically demonstrated the feasibility of academic prediction from relatively small, feature-rich datasets and established key benchmarks for comparative evaluation. They identified study time, past failures, parental education, and absences as the most discriminative predictors.

Kotsiantis, Pierrakeas, and Pintelas [2] investigated student performance prediction in distance learning environments using multiple machine learning classifiers, including naive Bayes, decision trees, neural networks, and SVMs. Their study contributed critical insights into the particular challenges of non-residential academic settings, where traditional attendance-based features are unavailable. The study emphasized the importance of feature selection and highlighted the relatively limited gains from complex model architectures when applied to small, noisy educational datasets.

Romero and Ventura conducted two landmark surveys [3][4] that mapped the trajectory of EDM research from its origins through the mid-2000s. Their 2007 survey [4] identified the principal task categories in EDM classification, clustering, association rule mining, and visualization and reviewed over 50 published studies. Their 2010 survey [3] extended this analysis within the IEEE framework, positioning EDM as a mature subdiscipline with increasing intersections with intelligent tutoring systems, adaptive learning platforms, and learning management system (LMS) data mining. These surveys remain canonical references for situating new contributions within the broader EDM landscape.

Baker and Yacef [5] provided a complementary review of the state of EDM in 2009, emphasizing the shift from prediction-only paradigms toward discovery-with-models and relationship mining. Their work foregrounded the growing importance of real-time data streams from digital learning environments and articulated the distinction between post-hoc analytics and formative, actionable prediction.

B. Modern Machine Learning Approaches

Since the foundational period, the field has adopted progressively more sophisticated algorithmic approaches. Ensemble methods, particularly the random forest algorithm introduced by Breiman [6], have been widely applied to educational prediction tasks, offering strong generalization and built-in feature importance estimation. SVMs, whose theoretical foundations were established by Cortes and Vapnik [7], have been applied to binary and multi-class academic performance classification, with studies reporting competitive accuracy on moderate-sized datasets.

Deep learning methods, reviewed comprehensively by LeCun, Bengio, and Hinton [8], have been explored for student outcome prediction, particularly in large-scale MOOC (Massive Open Online Course) environments where interaction log data is abundant.

Recurrent neural networks (RNNs) and long short-term memory (LSTM) networks have been applied to model temporal patterns in student engagement and predict dropout risk. However, the data requirements for these approaches are substantially higher than those of classical methods, typically demanding tens of thousands of labeled samples for meaningful generalization.

More recent research trends (2020–2025) in the EDM and learning analytics literature have explored: federated learning for privacy-preserving student modeling; transformer-based architectures adapted from natural language processing for sequential learning behavior analysis; graph neural networks for modeling social and collaborative learning structures; and multi-task learning frameworks that simultaneously optimize for performance prediction and dropout prevention. These developments reflect the field's increasing access to large-scale, fine-grained digital learning data.

C. Limitations of Machine Learning Approaches

Despite their impressive predictive capabilities, machine learning approaches present several systemic limitations when deployed in resource-constrained educational institutions. First, training-based models require labeled historical datasets of sufficient size; institutions with fewer than two to three years of structured digital records frequently lack the data density required for reliable model training. Second, model maintenance is non-trivial: as student cohorts, curricula, and assessment formats evolve, models degrade in predictive validity and must be periodically retrained. Third, the opacity of complex models particularly ensemble methods and neural networks pose significant practical barriers to adoption; instructors and academic counselors require predictions that can be explained, questioned, and contextualized, not merely accepted as numerical outputs from an opaque system. Fourth, the deployment infrastructure for machine learning models (serving environments, dependency management, versioning) introduces operational overhead that many institutions lack the technical capacity to sustain.

D. Research Gap and Positioning

The surveyed literature reveals a consistent pattern: the dominant research focus has been on maximizing predictive accuracy using increasingly complex models, with comparatively limited attention to interpretability, deployability, and suitability for data-scarce contexts. Simple, deterministic, and interpretable baseline systems have received limited formal treatment in the academic literature, despite their practical utility.

This paper addresses the identified gap by presenting a deterministic linear model as a formally characterized, interpretable baseline system. The model is explicitly designed for environments where machine learning is impractical due to data scarcity, interpretability requirements, or infrastructure limitations. Rather than competing with state-of-the-art methods on accuracy benchmarks, Nexus Edge occupies a distinct niche: a principled, explainable, zero-training-cost predictor that provides immediate, consistent value in real-world academic deployments.

III. METHODOLOGY

A. Model Formulation

The Nexus Edge prediction engine is implemented as a deterministic linear model. The model computes a predicted performance score y for a given student based on three input variables: previous examination marks (x_1), weekly study hours (x_2), and Cumulative Grade Point Average (x_3). The scoring function is defined as:

$$y = 0.55 \cdot x_1 + 2.5 \cdot x_2 + 2.0 \cdot x_3$$

The computed value y is subsequently clipped to the closed interval [0, 100] to enforce the semantic validity of the score within the conventional academic grading range:

$$y_{au} = \min(\max(y, 0), 100)$$

This formulation constitutes a heuristic linear scoring function. It does not involve iterative optimization, gradient descent, cost minimization, or any form of statistical learning from data. The weights (0.55, 2.5, 2.0) are domain-informed constants that encode expert judgment about the relative predictive contribution of each input feature to final academic performance.

B. Variable Definitions and Input Ranges

The three input variables are defined as follows. The variable x_1 (previous marks) represents a student's most recent cumulative examination score, expressed on a 0–100 scale. It serves as a direct empirical proxy for prior academic achievement and is the most objective of the three predictors. The variable x_2 (study hours) represents the average number of hours per week that the student dedicates to self-directed academic study, exclusive of formal instruction time. Study hours are a behaviorally grounded predictor that captures effort and academic engagement independent of prior achievement. The variable x_3 (CGPA) represents the student's Cumulative Grade Point Average on a standard 0–10 scale. CGPA encodes longitudinal academic performance across multiple semesters and subjects, providing a temporally integrated measure that complements the recency-weighted signal of x_1.

C. Weight Rationale and Design Reasoning

The weight assigned to x_1 (0.55) is intentionally moderate. While previous marks are informative, they may reflect performance under specific, non-generalizable conditions such as a single high-stakes examination or a period of anomalous performance. A lower coefficient attenuates the risk of over-anchoring predictions to a single data point.

The weight assigned to x_2 (2.5) is the largest in the function, reflecting the widely documented positive relationship between time-on-task and academic outcomes in the educational psychology literature. Study hours represent a volitional, controllable factor, and their elevated weight is also intended to create a scoring function that rewards and thereby implicitly motivates sustained academic effort.

The weight assigned to x_3 (2.0) is high, reflecting the strong predictive validity of cumulative academic record. CGPA integrates performance signals across the full academic history of the student and is thus a robust, low-noise predictor. Its slightly lower weight relative to x_2 reflects the design principle that current effort should carry at least equal prognostic weight to historical outcome.

These weights represent domain-informed heuristics. They are not derived from statistical optimization over a training set, and their selection is accordingly transparent, auditable, and subject to straightforward institutional calibration. Any stakeholder can examine, question, and modify these weights without requiring expertise in statistical modeling or machine learning.

D. Performance Classification

The output score y_{aut} is mapped to one of four discrete performance classifications via a deterministic threshold function. Table I summarizes the classification scheme.

Table I Performance Classification Scheme for Nexus Edge

Score Range	Classification	Recommended Action
≤ 40	Critical	Immediate academic intervention required
41 – 60	Needs Improvement	Targeted support and remedial action advised
61 – 80	Good — Can Improve	Satisfactory performance with room for growth
≥ 81	Excellent	High academic performance; maintain trajectory

The thresholds at 40, 60, and 80 are consistent with conventional academic grading conventions widely adopted across Indian technical universities, and directly mirror the grade boundaries defined under the CGPA systems prescribed by AICTE-affiliated institutions. This alignment ensures immediate institutional familiarity and reduces the semantic overhead of communicating results to students and faculty.

E. Determinism and Reproducibility

A defining property of the Nexus Edge model is strict determinism: for any fixed input vector (x_1, x_2, x_3), the model produces a unique, invariant output y_{aut} under all execution conditions, platforms, and system states. This property distinguishes the model fundamentally from stochastic machine learning approaches, including those with random initialization, dropout regularization, or probabilistic inference. Determinism is an operational advantage in educational contexts, where the reproducibility of predictions is essential for procedural fairness, appeals processes, and regulatory compliance.

IV. SYSTEM ARCHITECTURE

A. Architectural Overview

Nexus Edge is implemented as a three-tier web application. The system comprises a client-side presentation layer (frontend), a server-side application layer (backend), and an embedded model execution module. The architecture is intentionally lightweight, eschewing external machine learning frameworks, model-serving infrastructure, and database dependencies that would increase operational complexity.

B. Frontend Layer

The frontend is implemented as a responsive web interface accessible via standard web browsers on both desktop and mobile devices. The user interface exposes three input fields corresponding to the model variables x_1 (previous marks), x_2 (study hours), and x_3 (CGPA). Input validation is applied client-side to ensure that submitted values are within semantically valid ranges (0–100 for marks, 0–24 for daily hours, 0–10 for CGPA). Upon submission, the frontend dispatches an asynchronous HTTP request to the backend prediction endpoint and renders the returned score and classification band with appropriate visual feedback. Color-coded result indicators (red for Critical, amber for Needs Improvement, green for Good or Excellent) provide immediate, perceptible performance feedback without requiring the user to interpret numerical scores.

C. Backend Layer

The backend application server receives the validated input vector, invokes the deterministic linear scoring function, applies the output clipping operation, and maps the result to the appropriate classification label. The computation requires no external library invocations beyond standard arithmetic operations and is completed in constant time O(1) with negligible latency. The server returns a structured JSON response containing the numeric score, the classification label, and an advisory message corresponding to the classification band. The simplicity of the backend computation eliminates the need for GPU resources, model file loading, or inference session management—resources that machine learning–based alternatives require.

D. Model Execution Flow

The end-to-end data flow proceeds as follows: (1) The student submits academic input data via the frontend form. (2) Client-side validation confirms input range compliance. (3) The frontend transmits the input tuple (x_1, x_2, x_3) to the backend via HTTP POST. (4) The backend computes $y = 0.55 \cdot x_1 + 2.5 \cdot x_2 + 2.0 \cdot x_3$. (5) The result is clipped: $y_{aut} = \min(\max(y, 0), 100)$. (6) The classification threshold function maps y_{aut} to the appropriate band. (7) The structured response is returned to the frontend and displayed. The entire pipeline from submission to result display completes in under 50 milliseconds in typical deployment conditions, making the system suitable for real-time, high-throughput use.

V. RESULTS AND DISCUSSION

A. Conceptual Model Evaluation

The evaluation of the Nexus Edge deterministic linear model is necessarily conceptual in character, given that the model is not trained on a dataset and does not optimize a statistical objective function. Accordingly, conventional machine learning evaluation metrics such as classification accuracy, area under the ROC curve, precision-recall trade-offs, and cross-validation scores are not applicable in their standard form. Instead, the model is evaluated against a set of functional criteria relevant to the requirements of the target deployment context: determinism, monotonicity, input sensitivity, classification coherence, and boundary behavior.

B. Determinism and Consistency

The deterministic linear model guarantees output consistency: identical input vectors always produce identical outputs. This property is trivially verifiable by inspection of the scoring function and requires no empirical testing. In contrast, stochastic machine learning models may produce varying predictions under random seed changes or when operating near decision boundaries with probabilistic outputs. The strict determinism of Nexus Edge constitutes a fundamental advantage in institutional settings where prediction reproducibility is a legal or procedural requirement.

C. Monotonicity Analysis

A well-designed performance prediction model should exhibit monotonicity with respect to each input: holding other variables constant, an increase in any input variable should produce a non-decreasing output. The Nexus Edge scoring function satisfies this criterion by construction. Since all three weights (0.55, 2.5, 2.0) are strictly positive, the partial derivative of y with respect to each input variable is strictly positive. Specifically: $\partial y/\partial x_1 = 0.55 > 0$; $\partial y/\partial x_2 = 2.5 > 0$; $\partial y/\partial x_3 = 2.0 > 0$. Monotonicity is a desirable and practically important property: a predictor that assigns lower performance scores to students with higher CGPA or greater study hours would lack face validity and would not be trusted or adopted by educators.

D. Input Sensitivity and Weight Interpretation

The relative magnitudes of the weights provide an interpretable characterization of the model's sensitivity to each input. Study hours (weight 2.5) exerts the strongest marginal influence: a one-unit increase in weekly study hours increases the predicted score by 2.5 points. CGPA (weight 2.0) exerts the second-strongest influence: a one-unit increase in CGPA increases the predicted score by 2.0 points. Previous marks (weight 0.55) exert the weakest marginal influence per unit increment, consistent with the design rationale of moderating the effect of a single-point-in-time measurement.

These sensitivity characteristics are practically meaningful: they imply that a student with moderate previous marks ($x_1 = 50$) but high study engagement ($x_2 = 12$ hours/week) and a strong CGPA ($x_3 = 8.5$) would receive a predicted score of $0.55 \times 50 + 2.5 \times 12 + 2.0 \times 8.5 = 27.5 + 30.0 + 17.0 = 74.5$, classified as "Good but Can Improve." This result is intuitively plausible and communicates a clear, actionable message to the student and their academic advisor.

E. Boundary and Clipping Behavior

The output clipping operation ensures that the model never produces physically impossible scores outside [0, 100], even for extreme input combinations. For example, a student with $x_1 = 100$, $x_2 = 20$, and $x_3 = 10$ would yield a raw score of $0.55 \times 100 + 2.5 \times 20 + 2.0 \times 10 = 55 + 50 + 20 = 125$, which is clipped to 100. Conversely, a student with $x_1 = 0$, $x_2 = 0$, $x_3 = 0$ yields a raw score of 0, which remains at 0 post-clipping. The clipping operation does not introduce discontinuities that would distort the relative ordering of predictions in the interior of the input space, and its behavior at the boundaries is fully transparent.

F. Advantages and Limitations

The primary advantages of the Nexus Edge deterministic linear model relative to machine learning alternatives are: (i) zero training data requirement, enabling immediate deployment without historical records; (ii) complete interpretability, since the contribution of each input variable to any prediction is directly computable and communicable; (iii) computational triviality, requiring only three multiplications and two additions per prediction; (iv) strict determinism, supporting reproducibility and procedural fairness; and (v) full institutional controllability, since weights can be adjusted by domain experts without algorithmic expertise.

The principal limitations of the model are: (i) fixed weights that cannot adapt to the statistical structure of a particular institutional population; (ii) absence of learning, meaning the model cannot improve its predictions over time as more data accumulates; (iii) restriction to three input variables, potentially omitting relevant predictors such as attendance, extracurricular engagement, or socioeconomic factors; (iv) linear structure that cannot capture non-linear interactions between predictors; and (v) lower predictive ceiling relative to well-trained machine learning models applied to sufficiently large datasets.

These limitations are acknowledged as inherent and accepted trade-offs in the design philosophy of the system. Nexus Edge is positioned not as the universally optimal predictor, but as the appropriate predictor for its target deployment context.

VI. CONCLUSION

This paper has presented Nexus Edge, a student performance prediction system based on a deterministic linear model and heuristic linear scoring function. The system computes predicted academic performance scores from three domain-validated input variables previous marks, study hours, and CGPA—using a fixed, interpretable weight vector, and classifies the resulting scores into four actionable performance bands.

The model occupies a distinct and underserved niche in the educational prediction literature: a principled, zero-training-cost, fully interpretable baseline predictor for data-scarce and real-time academic environments. It does not seek to match the predictive accuracy of complex machine learning models trained on large datasets; rather, it provides immediate, consistent, and explainable predictions in contexts where such models are impractical.

Conceptual evaluation has confirmed that the model satisfies key functional desiderata: strict determinism, input monotonicity, boundary validity, and interpretable input sensitivity. The system architecture is lightweight, platform-agnostic, and requires no specialized computational infrastructure. These characteristics collectively support the adoption of Nexus Edge as a practical, institutionally deployable tool for formative academic monitoring.

VII. FUTURE WORK

Several directions for future extension are identified. First, the fixed weights in the current model could be replaced by institutionally calibrated weights derived from retrospective analysis of historical grade data. If a sufficient dataset becomes available, a lightweight linear regression could be applied to estimate optimal weights for a specific institutional population, while retaining the interpretable linear structure.

Second, the model could be extended to incorporate additional predictor variables such as attendance percentage, assignment submission rates, and in-class participation scores. Each additional variable would require a new weight term in the scoring function and corresponding interface support, but the overall model structure would remain deterministic and interpretable.

Third, a longitudinal evaluation study could be conducted by deploying the system within an academic institution over one or more semesters, collecting ground-truth outcome data, and computing retrospective accuracy metrics. This would enable principled comparison with machine learning baselines on the same institutional data and provide an empirical characterization of the model's predictive validity.

Fourth, an adaptive variant of the model could be designed wherein the weights are periodically updated using a lightweight online learning rule such as a perceptron update or a stochastic gradient step applied to a small buffer of recent prediction-outcome pairs. This would introduce a limited form of adaptability while preserving the linear, interpretable structure of the model.

Fifth, the classification threshold boundaries could be made configurable at the institutional level, allowing departments to define their own performance bands in alignment with local grading policies and pedagogical priorities. This would increase the practical generalizability of the Nexus Edge system across diverse institutional contexts.

REFERENCES

[1] Cortez, P., and Silva, A., "Using data mining to predict secondary school student performance," in *Proc. 5th Future Business Technology Conf. (FUBUTEC 2008)*, Porto, Portugal, Apr. 2008, pp. 5–12.

[2] Kotsiantis, S. B., Pierrakeas, C., and Pintelas, P., "Predicting students' performance in distance learning using machine learning techniques," *Applied Artificial Intelligence*, vol. 18, no. 5, pp. 411–433, 2004.

[3] Romero, C., and Ventura, S., "Educational data mining: A review of the state of the art," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 6, pp. 601–618, Nov. 2010.

[4] Romero, C., and Ventura, S., "Educational data mining: A survey from 1995 to 2005," *Expert Syst. Appl.*, vol. 33, no. 1, pp. 135–146, Jul. 2007.

[5] Baker, R. S., and Yacef, K., "The state of educational data mining in 2009: A review and future visions," *J. Educ. Data Mining*, vol. 1, no. 1, pp. 3–17, 2009.

[6] Breiman, L., "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[7] Cortes, C., and Vapnik, V., "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[8] LeCun, Y., Bengio, Y., and Hinton, G., "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.