

Ahmed Alwakeel

Master Thesis Template V4.docx

 Midocean University

Document Details

Submission ID

trn:oid::2945:305155099

Submission Date

Aug 27, 2025, 2:30 PM GMT+3

Download Date

Aug 27, 2025, 2:40 PM GMT+3

File Name

Master Thesis Template V4.docx

File Size

3.2 MB

60 Pages

11,142 Words

73,909 Characters

58% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

-  **65 AI-generated only 56%**
Likely AI-generated text from a large-language model.
-  **4 AI-generated text that was AI-paraphrased 2%**
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

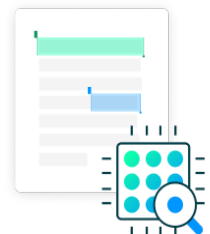
AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.





A Novel Stacking Framework for Student Success Prediction Using Combined Machine Learning and Deep Learning Approaches

Prepared By

Name: [Rami Mohammad AbuLaban](#)

ID No.: [233000631](#)

Name: [Khaled Ayman Alnbrees](#)

ID No.: [241001469](#)

Name: [Mohammed nafeth mansour](#)

ID No.: [241000459](#)

Name: [Khalid Jamal Al Zaqzouq](#)

ID No.: [241001334](#)

Submitted in fulfillment of the requirements for the degree Master of Science in Artificial Intelligence

At

Collage of Informatics
Midocean University
UAE

Supervisor Name: Wesam Ahmed

Collage of Informatics, Midocean University, Comoros

Email: wesamahmed@midocean.edu.km

Acknowledgements

Special thanks to [Dr. Wessam Ahmed](#)

Abstract

Educational institutions must predict student achievement to make timely interventions and enhance academic results. This thesis presents an innovative stacking framework that amalgamates machine learning (ML) and deep learning (DL) methodologies to improve the precision of predicting student achievement. The framework uses a hybrid "filter then wrapper" feature selection technique that blends speed, ease of understanding, and predictive capability. This makes it a good choice for tabular data. The proposed model employs a combination of machine learning algorithms as base learners, including Linear Regression ($R^2 = 87.05\%$), Ridge Regression ($R^2 = 87.05\%$), Lasso Regression ($R^2 = 87.04\%$), Decision Tree ($R^2 = 87.14\%$), Random Forest ($R^2 = 87.22\%$), and K-Nearest Neighbors (KNN, $R^2 = 81.90\%$), with a Multi-Layer Perceptron (MLP) serving as the meta-learner. The framework uses a stacking ensemble to combine the capabilities of several base models and the DL meta-learner to get a strong R-squared (R^2) score of 87.22%, which is better than the scores of individual baseline models. The technique incorporates stringent data from Kaggle preprocessing, feature engineering, and model validation with standardized criteria. We used SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) to make it easier to understand the stacking framework's predictions. Experimental findings show that the suggested stacking structure makes predictions far more accurate than ML and DL models. The framework's practical implications include improved decision-making for educators and administrators, with prospective applications in individualized learning and dropout prevention.

Keywords: Student Prediction, Machine Learning, Deep Learning, Hybrid Feature Selection, Stacking Ensemble

Table of Contents

List of Abbreviations	vii
List of Figures.....	viii

List of Tablesix

Chapter 1: Introduction	1
1.1 Problem Definition	2
1.2 Thesis Objectives	2
1.3 Thesis Contribution	3
1.4 Thesis Organization.....	4
 Chapter 2: Overview.....	 5
2.1 Introduction	5
2.2 Artificial Intelligence (AI).....	6
2.3 Machine Learning (ML).....	6
2.4 Deep Learning (DL)	7
2.5 Large Language Models (LLMs)	7
2.6 Explainable AI (XAI)	8
2.7 Classification Techniques.....	8
2.8 Feature Selection.....	9
2.9 Evaluation Metrics.....	10
 Chapter 3: Related Work	 11
3.1 Introduction	11
3.2 History	11
 Chapter 4: Proposed Approach	

4.1 Introduction	
4.2 Dataset Description	
4.3 Preprocessing Dataset	
4.4 Feature Selection Phase	
4.4.1 Feature Selection Method 1	
4.4.2 Feature Selection Method 2	
4.4.3 Comparative Analysis of Selected Features	
4.5 Machine Learning Models	
4.5.1 Traditional ML Models	
4.5.2 Neural Network Models	
4.6 Stacking Ensemble Model	
4.7 Summary	
 Chapter 5: Experimental Results and Discussion	
 Chapter 6: Conclusion and Future Work	
7.1 Important Features	
7.3 Future Work	
 References	27

List of Abbreviations

- **Acc.:** Accuracy
- **ANN:** Artificial Neural Network
- **ANOVA:** Analysis of Variance
- **ROC:** Receiver Operating Characteristic
- **ML:** Machine Learning

- **DL:** Deep Learning
- **LLM:** Large Language Model
- **EDM:** Educational Data Mining
- **XAI:** Explainable Artificial Intelligence
- **SVM:** Support Vector Machine
- **RF:** Random Forest
- **DT:** Decision Tree
- **LSTM:** Long Short-Term Memory
- **CNN:** Convolutional Neural Network
- **NLP:** Natural Language Processing
- **RNN:** Recurrent Neural Network
- **LR:** Logistic Regression
- **KNN:** K-Nearest Neighbors
- **TP:** True Positive
- **TN:** True Negative
- **FP:** False Positive
- **FN:** False Negative
- **F1:** F1 Score
- **AUC:** Area Under Curve
- **SMOTE:** Synthetic Minority Oversampling Technique

List of Figures

2.2 Artificial Intelligence (AI).....	6
2.3 Machine Learning (ML).....	6
2.4 Deep Learning (DL).....	7
2.5 Large Language Models (LLMs).....	7
2.6 Explainable AI (XAI)	8
2.7 Educational Data Mining (EDM)	8

2.8 Classification Techniques.....	9
2.9 Types of Learning Algorithms.....	9
2.10 10 Feature Selection.....	10
2.11 Trends in AI-based Student Analytics.....	10

List of Tables

2.1	Student Performance Data Set Description
2.2	Students' Academic Performance Dataset Description

Chapter 1 Introduction

In recent years, the evaluation and prediction of student academic performance have become essential in shaping modern educational systems. Accurate forecasting of student outcomes enables institutions to implement early interventions, reduce dropout rates, and optimize personalized learning pathways. With the widespread use of Learning Management Systems (LMSs), vast clickstream datasets are created, encompassing students' interactions with course content, assignment submissions, discussion forums, and navigation paths. This interaction data provides valuable insights into student behaviors and their relationship with learning outcomes (Hernández-García et al., 2024) [1]. Artificial Intelligence (AI) techniques particularly Machine Learning (ML) and Deep Learning (DL) have demonstrated significant effectiveness in uncovering meaningful patterns in educational data. These models can teach complex relationships between student behavior and historical performance to classify academic outcomes with high precision (Guo, 2015) [2]. Among DL models, Artificial Neural Networks (ANNs) are especially powerful due to their ability to capture nonlinear dependencies. Isljamovic and Suknovic (2014) found that ANNs could predict student performance accurately based on features like previous grades, attendance, and participation [3]. The integration of Large Language Models (LLMs), such as OpenAI's GPT and Google's BERT, has further advanced educational analytics by facilitating the analysis of unstructured data like student essays, forum discussions, and peer feedback (Alnasyan, 2024) [4]. Feature selection remains a critical step in developing robust predictive models of student performance. By identifying the most relevant attributes such as study habits, socioeconomic conditions, and emotional factors feature selection enhances model interpretability, mitigates overfitting, and reduces training time (Alshabandar et al., 2022) [5]. Explainable Artificial Intelligence (XAI) has become increasingly vital in educational contexts to ensure transparency and trust. Applying interpretability methods such as SHAP and LIME allows educators to understand how predictive models determine student outcomes, thereby supporting informed interventions (Abid et al., 2025) [6]. By combining ML, DL and feature selection, this study aims to develop intelligent and interpretable models for classifying student performance. These tools contribute to a shift from reactive to

proactive educational strategies, empowering institutions to make informed decisions that foster academic achievement and equity in learning.

1.1 Problem Definition

Despite the increasing availability of student data in academic institutions, many educational systems still struggle to identify students at risk of underperformance early enough to intervene effectively. Traditional evaluation methods often rely on post-exam results, which limit opportunities for timely support and personalized learning adjustments. Moreover, the complexity and diversity of educational data, including demographic, behavioral, attendance, and psychometric variables makes it challenging to isolate the most influential factors in predicting academic outcomes.

Including irrelevant or redundant features in predictive models can lead to overfitting, increased computational cost, and reduced interpretability. This highlights the need for efficient feature selection techniques that prioritize relevant attributes and support meaningful analysis. In parallel, classification algorithms based on machine learning (ML), and deep learning (DL) offer powerful tools for forecasting student success, yet their effectiveness depends heavily on data quality and model transparency.

Therefore, this research addresses the pressing need to develop a data-driven classification framework that combines advanced prediction models, enabling institutions to make timely, evidence-based decisions and understand the underlying drivers of student achievement.

1.2 Thesis Objectives

Using artificial intelligence techniques, the core objective is to design and evaluate intelligent systems that leverage machine learning (ML), and deep learning (DL) models arranged in a stacking ensemble to classify students based on diverse academic, demographic, behavioral, and psychometric data. By integrating feature-selection methods, the study aims to isolate the most influential attributes driving academic success, thereby boosting accuracy, lowering computational cost, and minimizing overfitting. The stacking framework will combine the complementary strengths of multiple base learners and a meta-learner to deliver robust, reliable predictions. Ultimately, the goal is to support data-driven interventions, enable early identification of at-risk students, and contribute to fair and effective academic planning across educational institutions.

1.3 Thesis Contribution

This thesis has the following main contributions:

1. Proposed a hybrid feature selection technique that employs both filter methods (such Pearson correlation, Chi-square, and Mutual Information) and wrapper methods (like recursive feature removal) to uncover things that might help predict student achievement. The approach reduces down on noise and extra information while keeping a good balance between speed, ease of understanding, and predictive capability for educational datasets.
2. Using ML basic learners including Linear Regression, Ridge Regression, Lasso Regression, Decision Tree, Random Forest, and K-Nearest Neighbors to find patterns in student data. Uses a number of machine learning methods, including linearity in regression models and non-linearity in tree-based models, to make predictions more reliable and applicable to a wider range of situations.

3. A Multi-Layer Perceptron (MLP) meta-learner is used in the stacking ensemble to combine the predictions of the basis learners to capture complex non-linear relationships, it makes predictions more accurate than single models.
4. The suggested stacking ensemble that included ML base learners and MLP meta-learners performed better ($MSE = 17.3085$, $RMSE = 4.1604$, $MAE = 3.2145$, $R^2 = 0.8722$). By combining diverse model predictions better than baseline models, it makes it easier to estimate how well students will do.
5. Using SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) to make the stacking framework's predictions easier to understand. This makes the model's decisions more trustworthy by showing educators the most important factors (like academic performance and engagement metrics) that affect predictions, so they can take action.
6. The proposed stacking framework's exceptional interpretability and accuracy facilitates the precise identification of at-risk students, thereby providing educational institutions with actionable, data-driven solutions. This is in support of applications such as personalized learning and attrition prevention.

1.4 Thesis Organization

The remainder of the thesis is structured as follows:

Chapter 2 This chapter introduces core AI concepts such as ML, DL, LLMs, and XAI, and explains their roles in education and learning analytics. It then presents the foundations

of Educational Data Mining (EDM), classification techniques, and types of learning algorithms. The importance of feature selection in predictive modeling is discussed. Finally, it highlights modern trends in AI-driven student analytics and their impact on proactive education.

Chapter 3 Examines recent academic research and comparative studies related to student performance prediction, machine learning applications in education, and the use of deep learning and language models. It identifies current research gaps that this thesis aims to address.

Chapter 4 Describes the research design and methodology, including data collection, preprocessing, algorithm selection, and the integration of feature selection techniques. It explains how different classification models and feature selection methods are applied and evaluated.

Chapter 5 Reports the experimental setup and presents results obtained from various classification models. The chapter includes a detailed performance analysis using accuracy, precision, recall, and F1-score. It also discusses the impact of feature selection on the model's performance.

Chapter 6 Summarizes the key findings, contributions, and implications of the study. It provides practical recommendations for educators and policymakers and outlines potential directions for future research in predictive analytics and educational AI systems.

Chapter 2 Overview

2.1 Introduction

Intelligence (AI) and its subfields contribute to Educational Data Mining (EDM). It begins by introducing essential AI concepts relevant to the education domain namely Machine Learning (ML), Deep Learning (DL), Large Language Models (LLMs), and Explainable AI (XAI) with clear illustrations of how each contributes to modern learning analytics.

Building upon this, the chapter delves into the core principles and methodologies of EDM, outlining how data-driven insights are extracted and utilized within academic environments. It then explores classification techniques, various types of learning algorithms (supervised, unsupervised, and reinforcement learning), and the critical role of feature selection in developing accurate, efficient, and interpretable predictive models.

The chapter concludes by highlighting recent trends in AI-based student analytics, demonstrating how the integration of advanced AI techniques enables a transition from traditional reactive educational practices to more proactive, personalized, and insight-driven approaches.

2.2 Artificial Intelligence (AI)

Artificial Intelligence (AI) is the broad science of mimicking human abilities. In the context of education, AI involves building systems that can perform tasks such as grading, tutoring, and personalizing learning without direct human intervention. AI serves as the foundation for more specific subfields like ML, DL, and LLM.

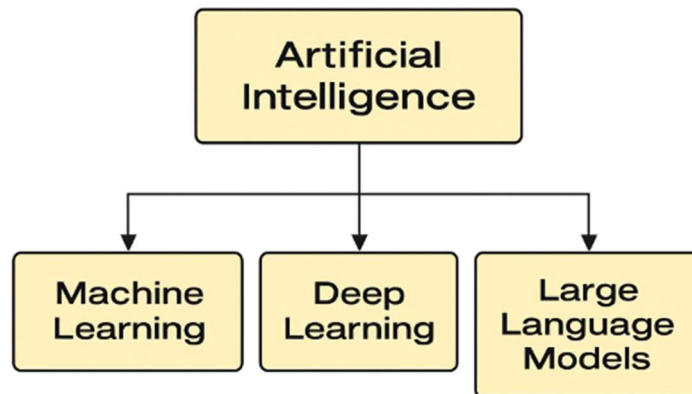


Fig. 1. Artificial Intelligence (AI)

2.3 Machine Learning (ML)

Machine Learning (ML) is a subset of AI that focuses on building systems that can learn from data and improve their performance over time without being explicitly programmed. In education, ML is commonly used to predict student outcomes, classify performance levels, and recommend learning resources [7].

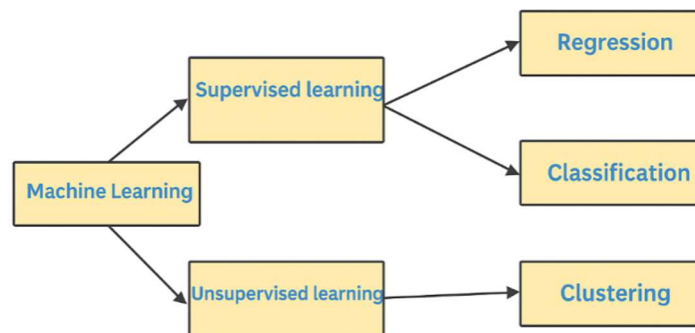


Fig. 2. Machine Learning (ML)

2.4 Deep Learning (DL)

Deep Learning (DL) is a specialized branch of ML that uses neural networks with multiple layers (deep networks) to model complex patterns in data. In educational applications, DL is used for tasks like handwriting recognition, speech-to-text processing, and analyzing visual learning behaviors [8].

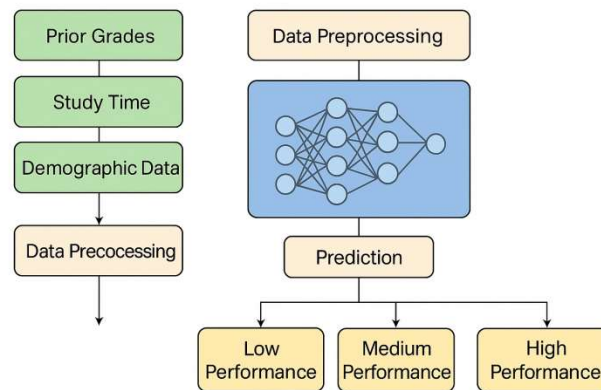


Fig. 3. Deep Learning (DL)

2.5 Large Language Models (LLMs)

Large Language Models (LLMs) like OpenAI's GPT and Google's BERT are advanced AI systems trained on vast amounts of text data. They are capable of understanding and generating human-like text. In education, LLMs are used to analyze student writing, conduct sentiment analysis, and generate automated feedback or tutoring responses [9].

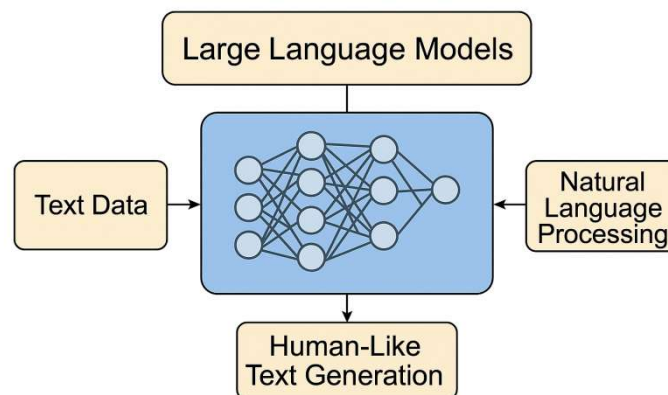


Fig. 4. Large Language Models (LLMs)

2.6 Explainable AI (XAI)

Explainable AI (XAI) refers to AI systems that offer transparent decision-making processes. XAI is crucial in education to ensure that predictions made by ML and DL models can be understood and trusted by educators and students. It helps in identifying which features influenced a prediction and supports ethical use of AI [10].

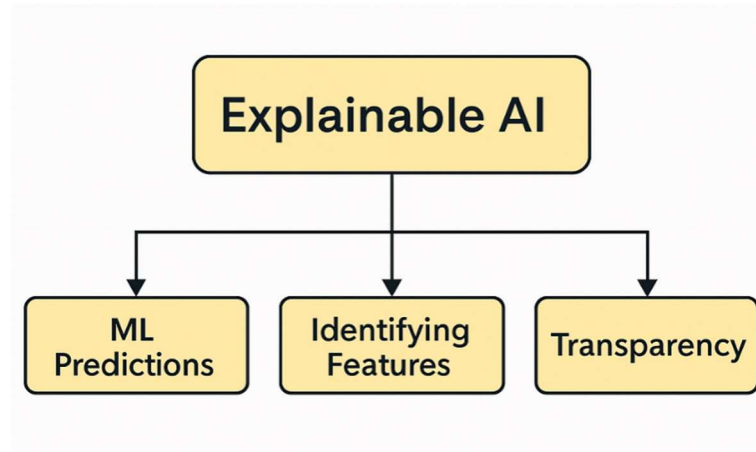


Fig. 5. Explainable AI (XAI)

2.7 Classification Techniques

Classification is a supervised learning technique used in EDM to predict student outcomes, such as pass/fail or grade categories. Common algorithms include Decision Trees, Support Vector Machines (SVM), Random Forests, and Neural Networks. These models can learn from historical data to classify new instances based on input features [11].

2.7.1 Machine Learning Techniques

Decision Tree (DT): A tree-structured classifier that splits the feature space using hierarchical decision rules, where each internal node represents a feature test, each branch represents an outcome, and each leaf node represents a class prediction. Decision trees are highly interpretable and can handle both numerical and categorical data without requiring feature scaling. However, they are prone to overfitting, especially with deep trees, and can be unstable as small changes in data may result in significantly different tree structures.

Elements of a decision tree

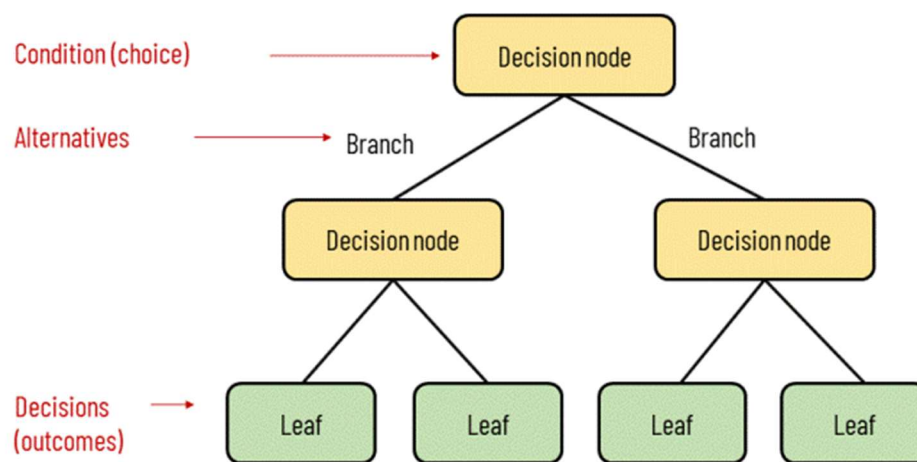


Fig. 6. Decision Tree

Random Forest (RF): An ensemble method that combines multiple decision trees through bootstrap aggregating (bagging), where each tree is trained on a random subset of the training data and features. The final prediction is determined by majority voting (classification) or averaging (regression) across all trees. Random forests reduce overfitting compared to individual decision trees, provide built-in feature importance measures, and are robust to outliers and missing values.

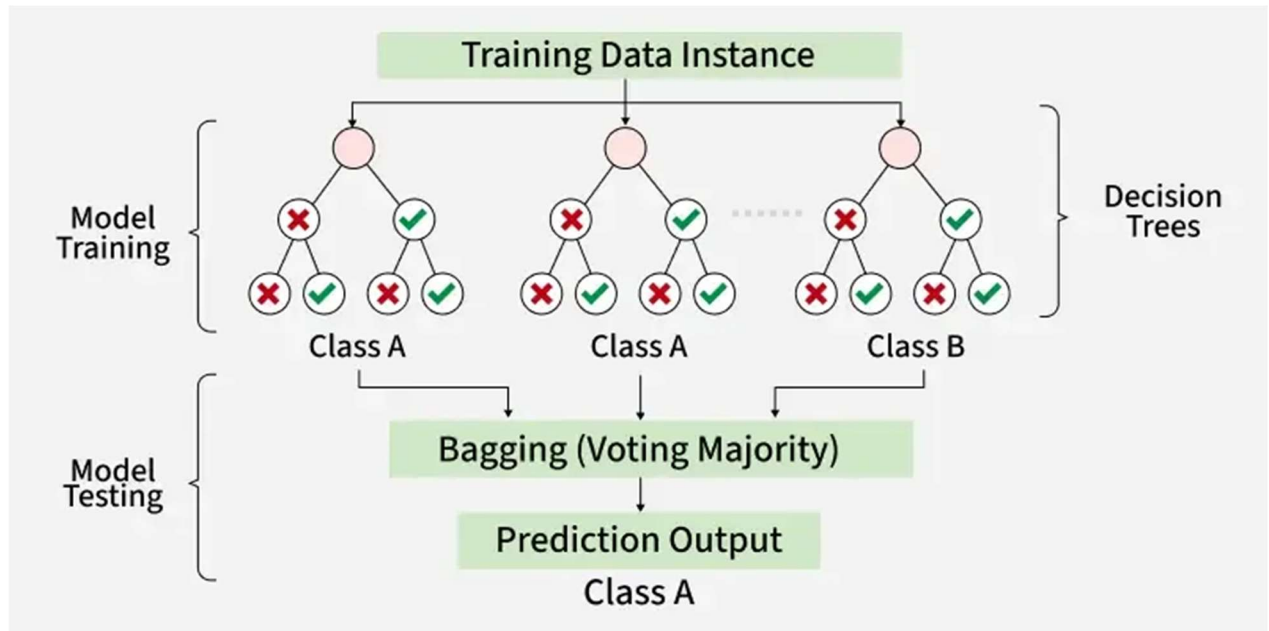


Fig. 7. Random Forest

K-Nearest Neighbors (KNN): A lazy learning algorithm that classifies instances based on the majority class among the k-nearest training samples in the feature space. KNN makes no assumptions about the underlying data distribution and can capture complex decision boundaries. However, it is computationally expensive during prediction, sensitive to the choice of k and distance metric, and performs poorly in high-dimensional spaces due to the curse of dimensionality.

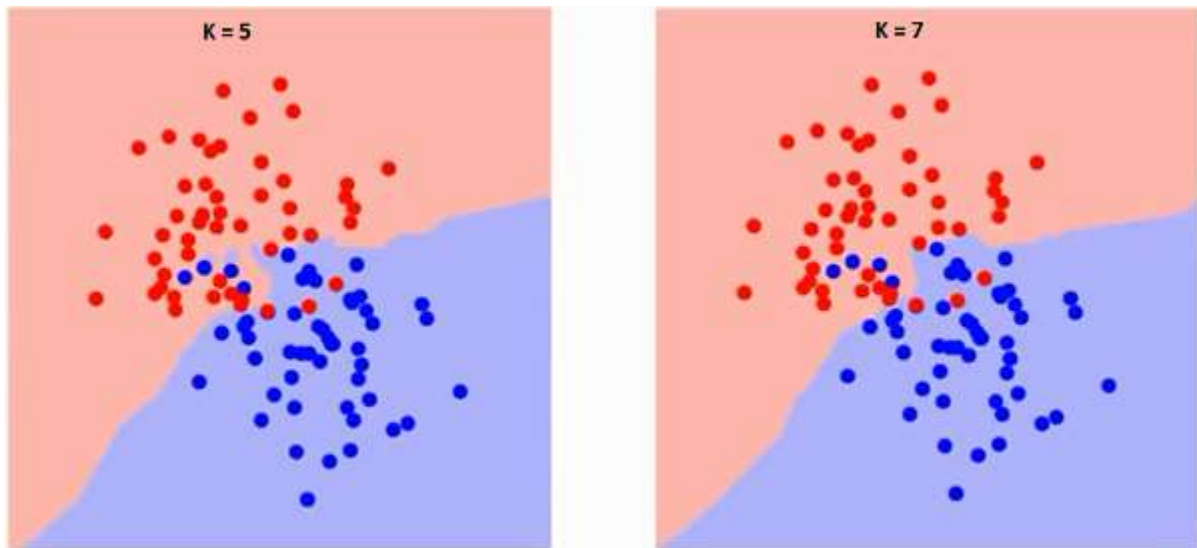


Fig. 8. KNN

Support Vector Classifier (SVC): A discriminative classifier that finds the optimal hyperplane to separate classes by maximizing the margin between different class boundaries. SVC can handle non-linearly separable data through kernel functions (polynomial, RBF, sigmoid) that map features into higher dimensional spaces. It is effective in high-dimensional spaces and memory efficiency but requires careful hyperparameter tuning and feature scaling.

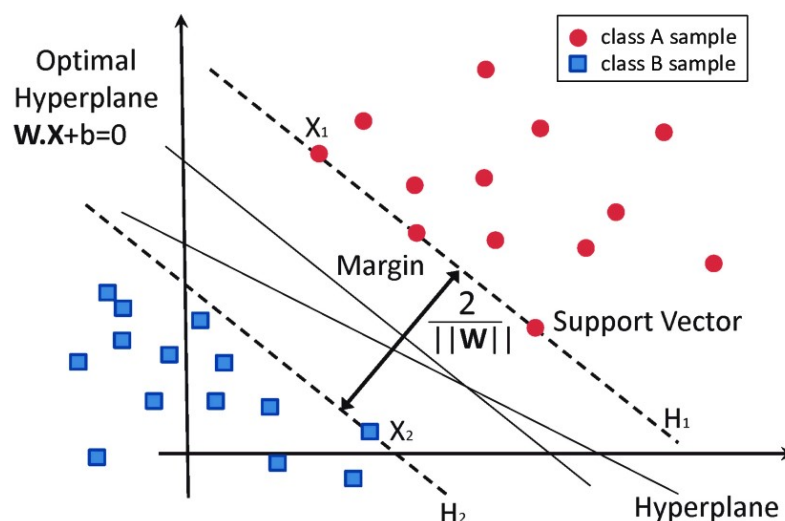


Fig. 9. SVC Algorithm

2.7.2 Deep Learning Techniques

Multi-Layer Perceptron (MLP): A feedforward artificial neural network consisting of multiple layers of interconnected neurons, including input, hidden, and output layers. Each neuron applies to a weighted linear combination of inputs followed by a non-linear activation function. MLPs can learn complex non-linear relationships through backpropagation training and are universal function approximators. They require careful architecture design, regularization techniques to prevent overfitting, and sufficient training data to achieve optimal performance.

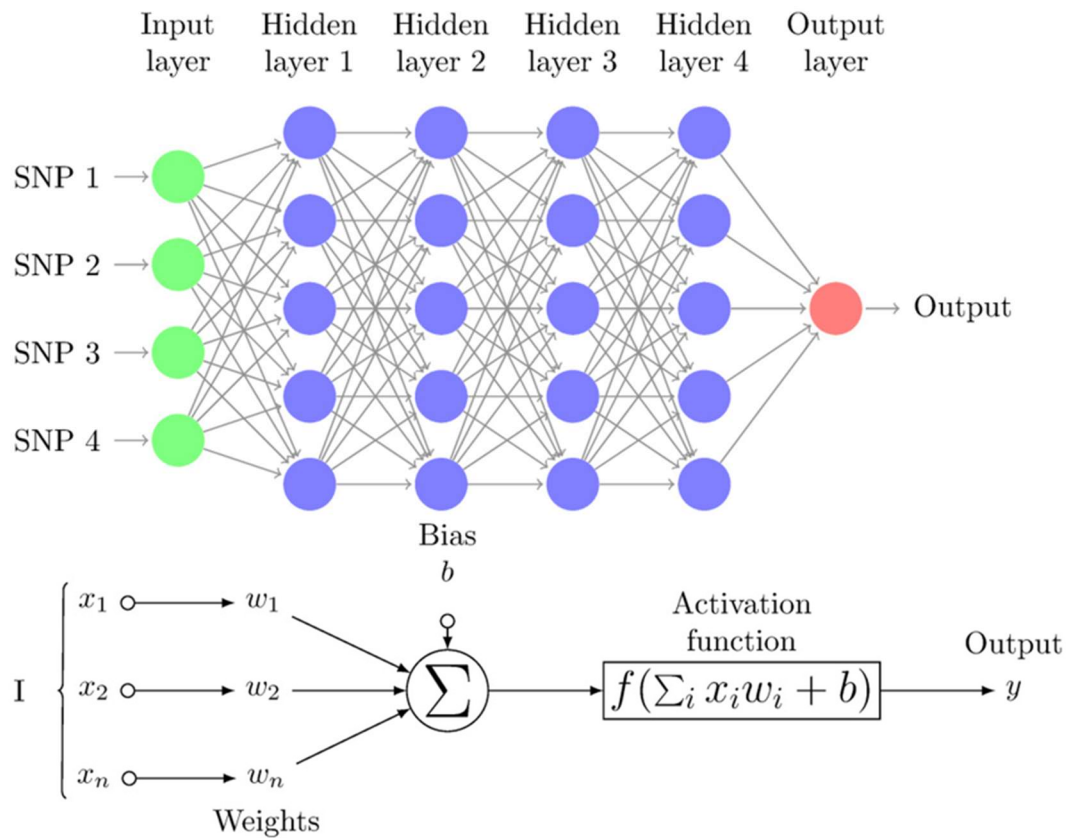


Fig. 10. MLP Diagram

Recurrent Neural Network (RNN): A neural network architecture designed for sequential data processing, where connections between nodes form directed cycles, allowing information to persist across time steps. RNNs maintain internal memory states that capture temporal dependencies in sequences, making them suitable for time-series analysis, natural language processing, and sequential decision-making tasks. However, traditional RNNs suffer from vanishing gradient problems in long sequences, often requiring advanced variants like LSTM or GRU for better performance.

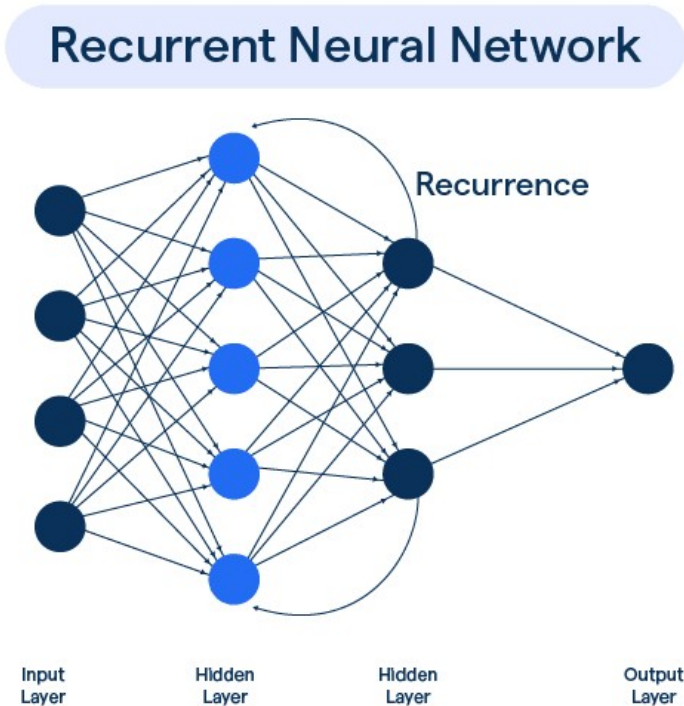


Fig. 11. RNN architecture Diagram.

2.8 Feature Selection

Feature selection is crucial in building efficient and interpretable models. By choosing the most relevant attributes (e.g., attendance, study time, prior grades), it enhances model performance and reduces complexity. Techniques include filter methods, wrapper methods, and embedded methods [12].

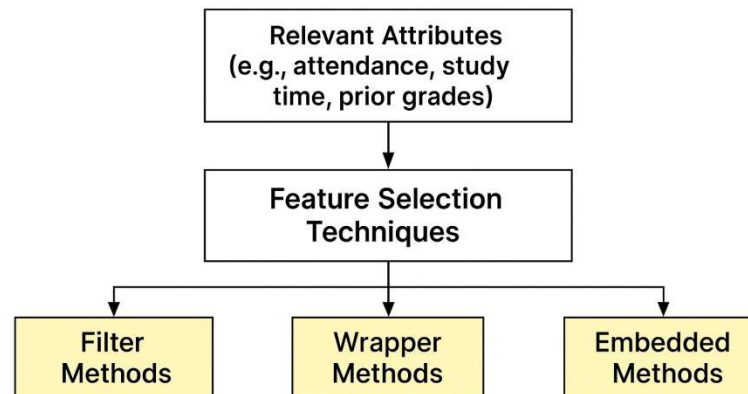


Fig. 12. Feature Selection

2.9 Evaluation Metrics

In machine learning classification tasks, evaluating model performance is essential to ensure that predictions are not only accurate but also meaningful in real-world contexts. The most widely used evaluation metrics including accuracy, recall, precision, and F1-Score are derived from the confusion matrix and are particularly valuable when dealing with imbalanced datasets (Opitz, 2024) [13].

2.9.1 Accuracy:

Accuracy measures *overall correctness* by comparing how many cases the model labeled right both true positives (TP) and true negatives (TN) against everything it attempted to classify, which also includes false positives (FP) and false negatives (FN). In other words, it answers the question: “Out of all predictions, what percentage did the model get right?” A value of 1 (100%) indicates perfect classification, whereas a value of 0 means every prediction was wrong. Although accuracy is intuitive and easy to interpret, it can be misleading when classes are highly imbalanced (e.g., 95% of samples belong to one class); in such cases, additional metrics like precision, recall, and F1-score provide a more balanced view of performance.

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

2.9.2 Recall (Sensitivity):

Recall, often called *sensitivity* or *true-positive rate*, indicates how completely a model captures the positive class. A high recall means the classifier misses very few genuine positives, making this metric crucial in contexts where overlooking a positive case (e.g., diagnosing a disease or flagging an at-risk student) carries serious consequences. However, emphasizing recall alone can raise the number of false positives, so it is usually interpreted alongside precision.

$$\text{Recall} = \frac{TP}{TP + FN}$$

2.9.3 Precision:

Precision also known as *positive predictive value* expresses the reliability of the model's positive predictions, it tells you what proportion of items labeled as positive truly belong to the positive class. High precision means few false positives, which is vital in scenarios where wrongly identifying a negative case as positive (e.g., flagging a student as at-risk when they are not) would waste resources or cause unwarranted concern.

$$\text{Precision} = \frac{TP}{TP + FP}$$

2.9.4 F1-Score:

The F1-Score is the harmonic mean of Precision and Recall. It provides a single score that balances both concerns and is particularly helpful in scenarios with class imbalance.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Chapter 3 Related Work

3.1 Introduction

In recent years, the application of Artificial Intelligence (AI) techniques in education has grown rapidly, especially in the area of student performance classification. Researchers have explored various AI approaches ranging from traditional Machine Learning (ML) algorithms to advanced Deep Learning (DL) models and Large Language Models (LLMs) to analyze educational data and predict academic outcomes. This chapter presents a comprehensive review of existing literature focused on optimizing student performance classification. It highlights key datasets, methodologies, and findings from previous studies, and provides a comparative analysis of their strengths and limitations. The goal is to identify the most effective models and techniques that can enhance the accuracy and reliability of academic performance prediction.

3.2 History

Jing Wang and Yun Yu [14] conducted a study using behavioral data from an online learning platform, consisting of eleven interaction indicators such as login frequency and learning duration. They performed correlation analysis to select the most relevant features and developed a logistic regression model enhanced by Taylor expansion. The proposed model achieved higher prediction accuracy compared to traditional methods, identifying initiative and total learning time as key predictors. Advantages include its lightweight structure, strong interpretability, and efficient feature selection. However, a major limitation lies in the dataset's focus on online behaviors only, potentially excluding other influential academic or psychological factors.

Althaqafi et al. [15] conducted a study using the xAPI-Edu-Data dataset to predict student academic performance within Learning Management Systems (LMS). They applied resampling techniques (SMOTE and others) to address class imbalance, then developed and compared various ML models including SVM, Logistic Regression, Decision Tree, and an ensemble (stacking with Random Forest as meta-learner). Their ensemble model achieved the highest weighted precision of 0.83, improving

from 0.60 when using the original imbalanced data. The study highlighted the importance of data enrichment for performance prediction and the influence of student engagement features. Advantages include balanced class handling and robust ensemble learning; limitations involve dependency on LMS logs, raising questions on feature generalizability and real-world adaptability.

Sandeepa and Mohottala [16] conducted a study using a real-world school dataset to evaluate multiple machine learning models such as multi-layer perceptron classifier (MLPC), decision tree, SVM, and others for predicting student academic performance. They trained and tested these models using 10-fold cross-validation and found that MLPC achieved the highest test accuracy of 86.46%. They highlighted that the MLPC excelled over other classical models and emphasized the importance of integrating explainable ML and feature selection approaches. Advantages include high predictive accuracy and the use of explainable techniques to improve model transparency; limitations involve potential overfitting indicated by the 99.65% training accuracy and dependency on a single school's dataset, which may affect generalizability.

Balaji et al. [17] conducted a study using a real-world educational dataset collected from secondary school students to evaluate multiple machine learning models including Decision Tree, Random Forest, Naïve Bayes, Support Vector Machine, and Logistic Regression for academic performance prediction. They addressed class imbalance issues using SMOTE and hyperparameter optimization via grid search. Their optimized Decision Tree model achieved the highest accuracy, improving from 90.9% to 93.4%, while Naïve Bayes saw the largest gain among models with accuracy rising from 77.3% to 83.3%. Advantages included clear interpretability, efficient handling of imbalance, and rigorous optimization; limitations involved reliance on a single school dataset limiting generalizability, and potential overfitting in the Decision Tree model.

Yuan et al. [18] conducted a study using a real-world dataset from edX to develop an integrated framework for predicting online learning performance. They first identified learning behavior patterns using clustering (e.g., low autonomy vs. motivated students), then applied machine learning models specifically XGBoost enhanced with SHAP-based feature analysis. The framework achieved nearly perfect

prediction accuracy for autonomous learners and strong performance for motivated students, demonstrating superior results compared to models without behavior pattern analysis. Advantages include personalized modeling per behavior cluster, enhanced interpretability via SHAP, and better accuracy than traditional ML methods; limitations involve dependencies on clustering quality and potential overfitting within small or unbalanced clusters.

Arya et al. [19] conducted a study using the UCI ML Repository's Portuguese student dataset to tackle issues like sample imbalance and lack of feature feedback. They developed a hybrid CNN–LSTM model to extract spatial and temporal patterns in student performance data. The model achieved 98.45% accuracy with a loss of 0.1989, outperforming standalone machine learning benchmarks. Advantages include its ability to capture both local feature correlations (via CNN) and sequence dependencies (via LSTM), as well as effective handling of data imbalance. Limitations involve dependency on a specific dataset and potential complexity overhead due to the hybrid architecture.

Adefemi et al. [20] conducted a study using public datasets (UCI, WOU, XAPI, and AV) to develop a hybrid CNN–LSTM model for predicting student academic performance across multiple educational contexts. They incorporated preprocessing techniques such as data balancing, missing value handling, and feature engineering. The hybrid model achieved exceptionally high accuracies for example, 99.67% on the UCI Portuguese dataset and 99.40% on the AV-Mathematics dataset surpassing all seven traditional classifiers evaluated. Advantages include broad dataset coverage, strong generalization, and significant performance improvements; limitations include high computational demands and less clarity on interpretability of the hybrid model.

Ben Said et al. [21] conducted a study using the Open University Learning Analytics Dataset (OULAD) to predict online learner performance during COVID-19. They proposed a two-pathway deep learning model that transforms clickstream time-series into Gramian Angular Field images for input to CNN, and incorporates demographic and assessment data via fully-connected layers. Their model achieved 92.7% accuracy, outperforming baseline approaches, and demonstrated the value of combining clickstream image representation with demographic features. Advantages

include innovative time-series imaging, robust performance, and clear handling of class imbalance via batch normalization. Limitations involve increased model complexity and reliance on image-based encoding, which may impede interpretability in educational settings.

Junejo et al. [22] conducted a study using the Open University Learning Analytics (OULAD) dataset to develop a deep neural network model capable of forecasting student performance across multiple categories (Distinction, Pass, Fail, Withdrawn) early in the course. They proposed a multi-layer feedforward neural network that incorporates demographic, assessment, and clickstream features extracted from the first 20% of course data. Their model achieved approximately 25% higher accuracy than state-of-the-art baselines, significantly improving early-stage prediction. Advantages include multi-class classification early in the course, strong generalizability across temporal course progression, and superior performance over traditional ML and RNN-LSTM models; limitations involve potential bias from imbalanced class distributions and dependence on early-stage data quality.

Moubayed et al. [23] conducted a study using three global online course datasets collected from different regions to predict student performance at the course midpoint. They proposed hybrid CNN and RNN-LSTM models to process clickstream and time-series interaction data. The models outperformed optimized traditional ML methods in two datasets and showed comparable results in the third, demonstrating both effectiveness and robustness. Advantages include strong generalization to diverse contexts and improved performance over traditional models; limitations involve potential overfitting in small datasets and dependence on clickstream availability.

Neshaei et al. [24] conducted a study using multiple real-world knowledge tracing datasets (e.g., ASSISTments, EdNet) to evaluate large language models (LLMs) for learner performance prediction. They compared zero-shot prompting and fine-tuning approaches on LLMs like GPT-3 and GPT-4 against traditional Bayesian Knowledge Tracing (BKT). While LLMs did not surpass state-of-the-art BKT models, the fine-tuned LLM version matched BKT performance and surpassed naive baselines across metrics like accuracy and AUC. The study highlighted that LLMs' sequence modeling abilities can be leveraged for knowledge tracing tasks. Advantages include

simplified modeling without heavy feature engineering and competitive performance; limitations involve high computational cost and lack of interpretability compared to BKT approaches.

Zhang et al. [25] conducted a study using reading-comprehension datasets from the AutoTutor intelligent tutoring system in adult literacy courses to assess the predictive capacity of GPT-4. They compared GPT-4 against traditional models (BKT, PFA, SPARFA-Lite, tensor factorization, XGBoost), using five-fold cross-validation. While XGBoost outperformed GPT-4 in accuracy, instructing GPT-4 to tune XGBoost on its platform produced better results than local execution. They highlighted that GPT-4 could select and configure strong traditional models and hyperparameters, though it was slightly less stable than grid-search. Advantages include leveraging GPT-4's reasoning for model selection and hyperparameter tuning; limitations involve lower raw predictive accuracy than XGBoost and reduced tuning stability.

Li et al. [26] conducted a study using a dataset of 5,278 student reflections collected across three academic terms from Kyushu University. They developed an LLM-based approach comparing single-agent and multi-agent strategies, combined with zero-shot and few-shot prompting, to automatically assess open-text reflections and predict academic performance. Their model using a single-agent with few-shot prompting achieved the highest match rate with human evaluations, and reflection scores generated by the LLM improved downstream performance prediction. Advantages include scalable automation for text assessment and enhanced predictive accuracy; limitations involve dependence on prompt design and potential bias from LLM-generated scores.

Savelka et al. [27] conducted a study using a dataset of student help requests collected from an introductory programming course to evaluate LLMs (GPT-3.5 and GPT-4) for classifying types of assistance requested. They evaluated both zero-shot prompting and a fine-tuned GPT-3.5 model across various help categories (e.g., debugging, conceptual understanding). In zero-shot trials, GPT-4 matched GPT-3.5 across most categories and outperformed it in debugging-related queries; after fine-tuning, GPT-3.5 achieved human-level performance, approximating inter-rater agreement. Advantages include effective classification of student intent without

extensive annotation and strong performance across categories; limitations involve dependency on quality of prompt design and potential misclassification in ambiguous cases.

Lundgren [28] conducted a study using a dataset of 60 master-level political science essays to compare GPT-4's performance to human graders for assessment tasks. They evaluated GPT-4's ability to assign grades and measured inter-rater reliability against university instructors' scores. GPT-4's mean grades closely matched human judgment, although it displayed a risk-averse grading pattern and exhibited lower inter-rater consistency. Advantages include potential scalability, grading consistency, and reduced educator workload; limitations involve cautious grading biases and limited alignment with human variance.

Fernandes et al. [29] conducted a study using the ISABELA smartphone dataset, which includes passive (activity, location) and self-reported data from students before and during COVID-19. They evaluated various classifiers SVM, XGBoost, AdaBoost-SAMME with Random Forest, and an ensemble decision-level median voting pipeline comparing weekly vs. monthly data windows. Their best models achieved over 90% accuracy using weekly aggregated data. Advantages include real-world behavioral sensing and improved accuracy with weekly data; limitations involve potential privacy concerns and variability across contexts.

Yang et al. [30] conducted a study on predicting student academic performance using multi-source campus behavioral data (e.g., movement, study habits). They developed a hybrid model combining hypergraph convolution to capture higher-order relationships and TabNet for interpretable classification. Their method outperformed baseline approaches in precision, recall, and F1-score on real student data. Advantages include modeling complex student interactions and providing feature-level interpretability; limitations involve dependence on institution-specific behavior data and potential scalability challenges due to hypergraph computation complexity.

Peng et al. [31] conducted a study on predicting student performance in collaborative learning environments, focusing on team-based software engineering coursework. They built an interaction graph of student team activities and proposed CLGT (Collaborative Learning Graph Transformer) incorporating an interpretability

module to visualize interaction patterns. Their model outperformed baseline graph and non-graph models, achieving higher accuracy and early identification of underperforming individuals. Advantages include leveraging social collaboration dynamics and providing interpretable results; limitations involve reliance on specific course settings and potential generalization issues to different disciplines.

Yang et al. [32] conducted a study using a heterogeneous educational dataset to evaluate student performance, integrating academic, behavioral, and social indicators. They proposed a dual hypergraph neural network that leverages framelet-based hypergraph convolutions to capture multi-level relationships among students and features. Their model achieved superior classification accuracy compared to conventional graph neural networks, showing a notable improvement in detecting at-risk learners. Advantages include capturing complex high-order interactions and robustness to sparsity; limitations include increased computational demand and potential need for large datasets for stability.

Ref	Year	Datasets	Models	Results	Advantages	Limitations
Wang & Yu [14]	2025	Online learning behavioral data (11 interaction indicators)	Logistic Regression enhanced with Taylor Expansion	Outperformed traditional models in accuracy (exact % not specified)	Lightweight model, high interpretability, effective feature selection	Depends solely on online behavior data; may miss academic/personal factors
Althaqafi et al. [15]	2025	xAPI-EduData (LMS logs: engagement, academic, cognitive)	SVM, Logistic Regression, Decision Tree, Ensemble (Stacking with RF)	Ensemble Precision = 0.83 (vs 0.60 without enrichment)	Handled class imbalance; ensemble improved accuracy; focused on engagement features	Depends on LMS-specific logs; may lack generalizability
Sandeepa & Mohottala [16]	2025	School-level behavioral, academic, and demographic dataset	MLPC, Decision Tree, SVM, etc	MLPC achieved 86.46% accuracy (test set)	High test accuracy; applied explainable ML and feature selection	Training accuracy (99.65%) suggests overfitting; single-site data limits generalizability
Balaji et al. [17]	2024	Secondary school academic dataset (demographic +	Decision Tree, Random Forest, Naïve Bayes,	Decision Tree accuracy increased from 90.9% to	Interpretable model; effective use of SMOTE and grid	Single-site data; possible overfitting in Decision Tree

		behavioral)	SVM, Logistic Regression	93.4%; Naïve Bayes from 77.3% to 83.3%	search; strong optimization	
Yuan et al. [18]	2024	edX platform dataset with clickstream and demographic data	Clustering + XGBoost with SHAP feature importance	Nearly perfect accuracy for autonomous learners; strong accuracy for motivated cluster	Cluster-specific modeling; improved interpretability; better overall accuracy	Reliant on cluster quality; potential overfitting in small clusters
Arya et al. [19]	2024	UCI Portuguese student dataset	CNN–LSTM hybrid	98.45% accuracy, loss = 0.1989	Captures spatial and temporal patterns; handles imbalance	Specific to this dataset; hybrid model complexity
Adefemi et al. [20]	2025	UCI (Math & Portuguese), WOU, xAPI, AV	Hybrid CNN–LSTM architecture	Up to 99.67% accuracy (UCI-Por); 99.40% (AV-Math)	High accuracy across diverse datasets; robust preprocessing; generalizable	Heavy computation; model interpretability is limited

Ben Said et al. [21]	2023	Open University Learning Analytics Dataset (OULAD)	Dual-path CNN (clickstream images + FC layers)	92.7% accuracy in online learner classification	Novel time-series imaging; combined multimodal (clickstream & demo) features; robust to imbalance	Complex architecture; image encoding reduces interpretability for educators
Junejo et al. [22]	2024	Open University Learning Analytics Dataset (OULAD)	Multi-layer neural network (MLP)	~25% improvement over baselines; high early-stage predictive accuracy	Multi-category early prediction; temporal generalizability; strong performance gains	Imbalanced classes; quality dependency on early course data
Moubayed et al. [23]	2024	Three online course datasets from different regions	CNN + RNN-LSTM hybrid	Outperformed optimized ML in two datasets; comparable in third	Strong generalization; better accuracy than ML	Overfitting risk on smaller sets; requires clickstream data
Neshaei et al. [24]	2024	Real-world knowledge tracing sets (e.g., ASSISTments, EdNet)	Fine-tuned LLMs (e.g., GPT-3/GPT-4)	Matched Bayesian Knowledge Tracing performance on	Simplified modeling; competitive without manual features	High computational requirements; less transparent interpretability

				accuracy and AUC		
Zhang et al. [25]	2024	AutoTutor reading-comprehension ITS (adult literacy)	GPT-4 prompting for model selection and hyperparameter tuning (XGBoost engine)	GPT-4–guided XGBoost outperformed local XGBoost; XGBoost alone had higher raw accuracy	Uses GPT-4 reasoning to select/tune models; human-like strategy without extensive trials	Raw accuracy slightly below direct XGBoost; tuning stability issues
Li et al. [26]	2025	5,278 open-text student reflections (377 students over 3 terms)	Single-agent and multi-agent LLM strategies with zero/few-shot prompting	Few-shot single-agent strategy achieved highest agreement with human scoring; improved academic performance prediction	Automates qualitative text evaluation; supports performance prediction via reflection scoring	Prompt-dependence; risk of bias from LLM-generated assessments
Savelka et al. [27]	2023	Help-request logs from introductory programming class	GPT-3.5 (fine-tuned) and GPT-4 (zero-shot)	GPT-4 outperformed in debugging query classification; fine-	Human-like intent classification; effective with	Relies on prompt quality; misclassifies ambiguous queries

				tuned GPT-3.5 matched human agreement levels	minimal supervision	
Lundgren [28]	2024	60 master-level political science essays	GPT-4 for grading	Mean GPT-4 grades matched instructor levels; lower inter-rater reliability	Scalable grading; consistent outcomes; less faculty burden	Grading bias; weaker alignment with human grading variance
Fernandes et al. [29]	2022	ISABELA smartphone data (activity, location, self-report)	SVM, XGBoost, AdaBoost-SAMME + RF ensemble	>90% accuracy with weekly windows	Real-life behavioral data; weekly aggregation improves results	Privacy/variability concerns; needs cross-context validation
Yang et al. [30]	2025	Multi-source campus behavior data (movement, study habits, online usage)	Hypergraph convolution + TabNet	Improved precision, recall, and F1 over baselines	Captures higher-order student interactions; interpretable feature analysis	Institution-specific data; scalability concerns due to hypergraph complexity
Peng et al. [31]	2023	Collaborative learning	CLGT (Graph Transform	Outperformed baselines;	Models social interactions	Course-specific data; limited to

		interaction logs (software engineering teams)	er with interpretability module)	early detection of struggling students	; interpretable; early-warning capability	collaborative projects; may not generalize
Yang et al. [32]	2024	Heterogeneous educational data (academic, behavioral, social features)	Framelet-based dual hypergraph neural network	Higher classification accuracy over standard GNNs; improved at-risk detection	Models high-order relationships; robust with sparse data	Higher computation cost; requires large data for model stability

Chapter 4 Proposed Approach

4.1 Introduction

This chapter describes the methodology that will be adopted to design, implement, and evaluate the proposed predictive framework. The approach will use two different feature selection techniques with diverse machine learning and deep learning models (MLP), which will then be combined in a stacking ensemble to enhance predictive performance. This structured methodology will ensure a systematic process from data preparation to model evaluation, providing a scalable framework to enhance the classification of student preferences as shown in Fig 13.

The proposed methodology will follow a comprehensive pipeline that begins with data collection and preprocessing, followed by feature engineering and selection phases. Subsequently, multiple machine learning models will be trained and evaluated, culminating in a stacking ensemble approach that combines the predictions of individual models to achieve superior performance.

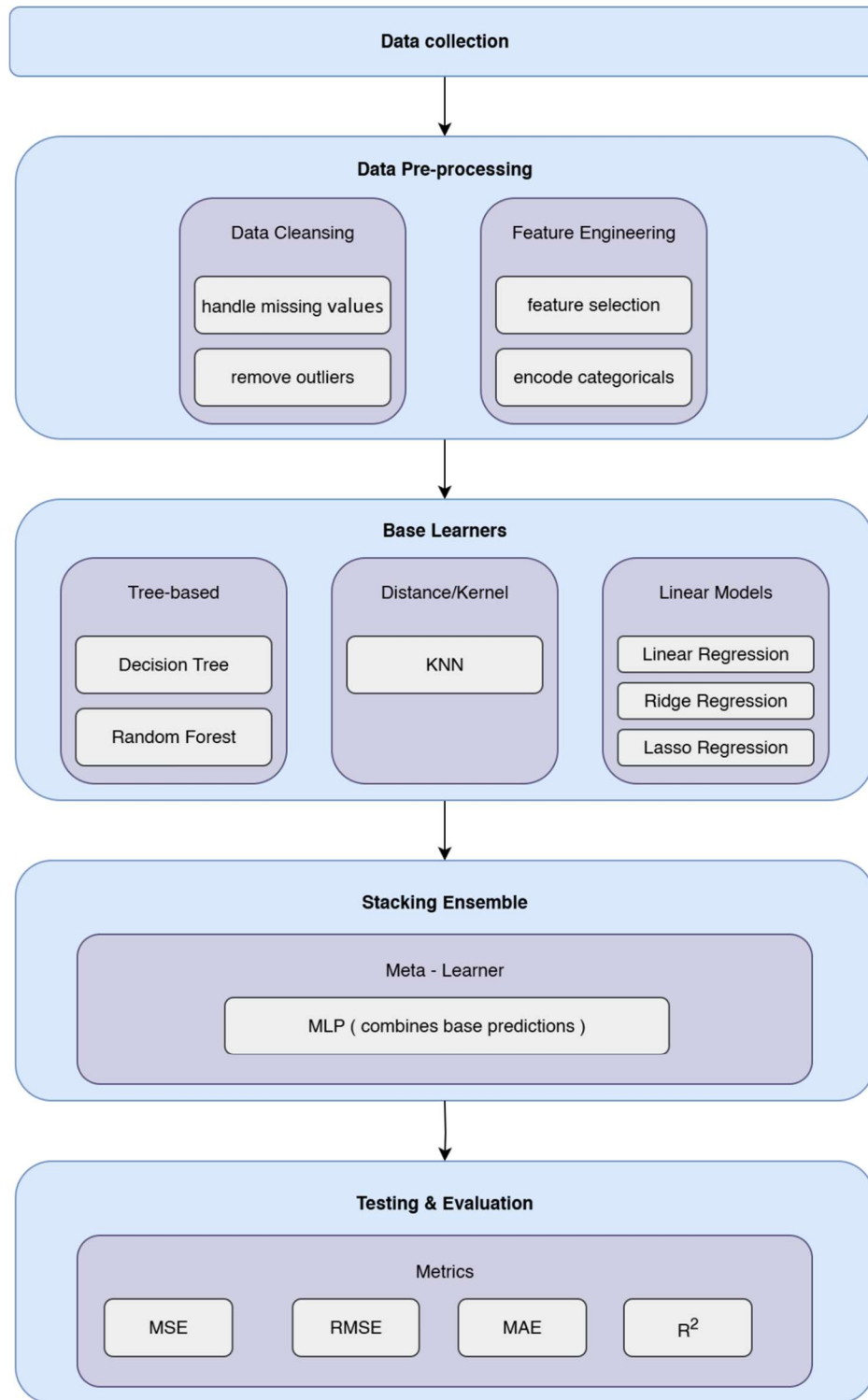


Fig. 13 Proposed Framework

4.2 Dataset Description

We use a dataset with 80,000 student records that capture many different parts of modern student life. This dataset is much better than old school datasets because it includes modern lifestyle things, how much students use digital devices, and mental health factors that affect how students perform in school today, which will help to predict student performance. Our dataset has five main sets of features that give us a complete picture of what students are like and how they behave. Basic student info includes things like `student_id` (a unique number for each student), `age` (from 16 to 28 years old), `gender` (Male, Female, Other), and `major` (what they study, like Computer Science, Engineering, Arts, and other subjects). These basic facts help us understand the starting point for looking at how different types of students perform. School activity measures show direct school behaviors through `study_hours_per_day` (how many hours they study each day on average) and `attendance_percentage` (how often they come to class). These data tell us right away how dedicated students are and how much they participate in school. Attendance data is really useful because it usually connects strongly with how well students do across different types of schools. Also, the digital lifestyle info is something new in education analysis. It recognizes how much technology affects today's students. We measure `social_media_hours`, `netflix_hours`, and `screen_time` to see digital habits that might compete with studying or affect how much mental energy students have for learning. This digital part fills a big gap in traditional school datasets that often ignore how technology impacts student performance, which can help us better predict student performance. Complete lifestyle and mental health factors make up the most advanced part of our dataset. This includes `sleep_hours`, `exercise_frequency`, and `diet_quality` for physical health, alongside `mental_health_rating`, `stress_level`, and `exam_anxiety_score` for psychological measures (rated 1-10). These variables understand that school performance comes from complex connections between physical health, mental health, and thinking ability rather than just academic factors alone. Family and support system info gives important background through `family_income_range`, `parental_support_level`, `parental_education_level`, `access_to_tutoring`, and `extracurricular_participation`. Also, personal management skills show up in `motivation_level` and `time_management_score` (rated 1-10), while learning preferences appear in `learning_style` and `study_environment` variables. We also have a `dropout_risk` indicator (Yes/No) that combines stress and motivation patterns to find students who might need help. Performance targets include `previous_gpa` and

exam_score, giving us both past academic history and current performance assessment. This double approach lets us evaluate both long-term academic success and specific test performance, supporting different prediction goals in educational analysis applications.

4.3 Preprocessing Dataset

School datasets often have problems that make analysis difficult. So we need systematic cleaning methods that keep information quality while making our data ready for advanced computer learning methods. Fixing Missing Information: Even though our fake dataset has complete records, real school data often has gaps because students don't answer surveys, administrative problems, or people choose not to participate. Our cleaning process uses smart filling techniques that consider both statistical relationships and school knowledge. For basic info like gender and major, we use the most common answer to fill gaps. But for continuous numbers like study_hours_per_day or mental_health_rating, we use prediction-based filling that looks at academic level, lifestyle patterns, and background factors to create realistic replacement values that maintain dataset integrity. Finding and Handling Extreme Values: Our complete dataset needs smart outlier detection that tells the difference between real extreme values and possible measurement mistakes. Academic and lifestyle variables naturally have valid extreme observations students who study 12+ hours daily or those with very high stress levels are real educational situations, not data errors. We use education-informed outlier detection that combines statistical methods (range analysis, z-score limits) with educational expert validation. For variables like screen_time or social_media_hours, we use separate outlier rules that accept the wide variation in digital habits among today's students. Making All Measurements Similar: Our dataset's variables come in very different scales that need careful adjustment to prevent artificial importance during model training. Academic measures like previous_gpa (0-4.0 scale) exist with percentage variables (attendance_percentage: 0-100%), hour measurements (study_hours_per_day, sleep_hours), and rating scales (stress_level, motivation_level: 1-10 scale). We use appropriate scaling methods: z-score standardization for continuous variables with normal patterns, min-max scaling for limited measures, and robust scaling for

variables with outliers that shouldn't be removed. Converting Categories to Numbers: Our dataset has rich category information, including major field, gender categories, learning_style preferences, study_environment choices, and yes/no indicators like part_time_job and dropout_risk. Since computer learning methods need numerical input, we use strategic encoding approaches that keep category relationships without forcing artificial ordering. One-hot encoding transforms categories like major and learning_style, while yes/no variables stay in their original format. For ordered categories like parental_education_level, we use label encoding that respects natural ordering relationships. Creating New Features and Combinations: Given our comprehensive dataset, we build meaningful combination terms that capture complex relationships between lifestyle, academic, and psychological factors. Examples include stress_study_ratio (combining stress_level with study_hours_per_day), digital_lifestyle_index (putting together social_media_hours, netflix_hours, and screen_time), and academic_engagement_score (combining attendance_percentage with study_hours_per_day). These engineered features help our methods capture non-linear relationships that individual variables might not show independently.

4.4 Feature Selection Phase

Since our complete educational dataset has many different variable types and complex connections, systematic feature selection becomes necessary for finding variables that really help predict performance while removing redundant or noisy information. Our method uses complementary selection approaches that capture different aspects of variable importance across academic, lifestyle, and psychological areas.

4.4.1 Feature Selection Method

We use three different statistical evaluation methods, each showing unique aspects of feature-target relationships that help us select variables comprehensively across our dataset's diverse variable types. Chi-Square Independence Testing: This statistical approach works particularly well for categorical school variables, including major, gender, learning_style, study_environment, and yes/no indicators like part_time_job and access_to_tutoring. When we investigate relationships between category predictors and performance outcomes, chi-square analysis gives us

a number-based assessment of independence assumptions. Variables showing statistically significant connections with `previous_gpa` or `exam_score` get priority consideration for model inclusion, making sure our selection process stays statistically solid while recognizing diverse category factors affecting academic success. Connection-Based Feature Assessment: Our dataset's many continuous variables, including `study_hours_per_day`, `sleep_hours`, `exercise_frequency`, `mental_health_rating`, `stress_level`, `motivation_level`, and `time_management_score`, go through systematic connection analysis with target performance measures. Pearson correlation numbers show linear relationships, while Spearman rank correlation captures monotonic connections that might not follow strictly straight-line patterns. This double approach proves essential given complex relationships between lifestyle factors and academic performance, where increased study hours might show positive connection up to an optimal point before tiredness effects create diminishing returns. Mutual Information Analysis: The complex, non-linear interactions in our comprehensive dataset require advanced dependency detection beyond traditional correlation methods. Mutual information provides advanced measures that identify all forms of statistical dependency, including complex interactions between multiple variables and performance outcomes. This method proves particularly valuable for detecting subtle relationships such as interactions between digital engagement patterns (`social_media_hours`, `netflix_hours`) and academic performance, where effects might vary significantly based on `time_management_score` or `motivation_level` contexts. From each method, we systematically take the highest-scoring variables, typically selecting the top 10 features based on respective evaluation criteria to accommodate our dataset's richness. Union operations then combine these selections, creating a comprehensive feature pool that captures diverse relevance perspectives while preventing loss of important variables identified by only one method.

4.4.2 Wrapper-Based Performance Evaluation

While statistical filters provide valuable initial screening across our dataset's diverse variable types, wrapper methods evaluate features based on their actual contribution to prediction accuracy using LinearRegression. This approach offers direct practical evidence of feature utility rather than relying only on statistical association measures. Our wrapper implementation follows step-by-step optimization strategy where different feature combinations from our consolidated pool go through systematic

evaluation using representative base model. Each subset gets assessment through cross-validation protocols that provide unbiased performance estimates across both GPA prediction and exam score. We systematically add or remove features based on their demonstrated contribution to the model accuracy, considering computational complexity introduced by our dataset's comprehensive variable set while maintaining optimal predictive performance.

4.4.3 Comparative Analysis of Selected Features

Our analysis goes beyond just selection to include comprehensive examination of method agreements, feature importance rankings across academic, lifestyle, and psychological variable categories, and complementarity assessments between selected variables. This systematic evaluation ensures our selection process maintains both statistical validity and practical understanding while identifying potential redundancies between related variables such as stress_level and exam_anxiety_score, or between various digital engagement measures.

4.5 Machine Learning Models

Our ensemble foundation has six diverse machine learning methods, each contributing specialized capabilities that collectively capture the multifaceted nature of educational performance prediction across our comprehensive dataset's academic, lifestyle, and psychological dimensions. This method diversity ensures thorough coverage of different learning approaches and relationship patterns inherent in contemporary student data.

4.5.1 Traditional ML Models

We used six traditional ML models to predict student performance, using a dataset with academic factors like grades and test scores, lifestyle factors like sleep hours and screen time, and psychological factors like stress and mental health. Each model has unique technical features and limitations.

Linear Regression:

Linear Regression is our baseline model for predicting continuous or linear outcomes, such as previous GPA or exam scores. It uses a linear equation to assign coefficients to each feature, showing their effect on the target. The coefficients are easy to interpret, making the model useful for educational involved parties who need clear

insights into how factors like study hours, mental health ratings, or parental support influence outcomes. The model assumes linear relationships and is sensitive to outliers, but its simplicity ensures stable performance across our diverse dataset.

Ridge Regression:

Ridge Regression extends Linear Regression by adding L2 regularization with an alpha parameter of 1.0. This penalizes large coefficients, addressing multicollinearity when features like family income, parental education, or access to tutoring are highly correlated. Such correlations are common in our dataset, especially among lifestyle factors like social media hours and screen time. By shrinking coefficients, Ridge reduces overfitting and improves prediction stability, making it suitable for complex datasets with interconnected variables.

Lasso Regression:

Lasso Regression applies L1 regularization to select important features in our comprehensive dataset. Unlike Ridge, Lasso can set coefficients to exactly zero, effectively removing less relevant variables during training. This is useful for distinguishing predictive factors, such as study hours or parental support, from those that correlate with performance but lack causal impact, like excessive social media use. Lasso's automatic feature selection simplifies the model, improving interpretability and reducing overfitting in our high-dimensional data.

Decision Tree Regression:

Decision Trees build predictive models by splitting the dataset into branches based on feature values, creating a tree-like structure. They excel at capturing non-linear relationships and interactions between academic, lifestyle, and psychological factors without requiring strict assumptions about data distribution. For instance, the model can identify how study hours and stress levels interact to affect performance. The resulting decision paths are interpretable, allowing educators to validate predictions against their expertise, though single trees may overfit without careful tuning.

Random Forest Regression:

Random Forest is an ensemble of a number of decision trees, each trained on random subsets of data and features through bootstrap sampling and feature randomization. This reduces overfitting compared to a single decision tree and captures complex interactions across our dataset. Random Forest provides feature importance scores, ranking the influence of variables like study hours or mental health. Its robustness makes it ideal for handling the diverse and interconnected factors in our data, ensuring reliable predictions for varied student profiles.

K-Nearest Neighbors Regression:

K-Nearest Neighbors (KNN) predicts performance by averaging the outcomes of the k most similar students, based on a distance metric across all features. This non-parametric method requires no assumptions about data structure and adapts to local patterns in our dataset that's KNN is effective for capturing similarity-based relationships, such as shared academic habits or psychological profiles, but its performance depends on choosing an appropriate k value and distance metric, which make us need a lot of fine-tuning. It suits our diverse data by focusing on local neighborhoods rather than global trends.

Practical Value for Educators

Each model of those models contributes uniquely: Linear Regression and Decision Trees offer interpretable results, Ridge and Lasso handle complex feature relationships, and Random Forest and KNN provide robust predictions for diverse students. Together, they enable educators to identify students needing support and design interventions, such as tutoring for low performers or counseling for high stress, improving outcomes with technical precision. This can help educators understand data and plan support, like tutoring or counseling.

4.5.2 Neural Network Models

In our stacking setup, the MLP works as the meta-learning model. It is chosen because it can handle the complexity of our dataset, which includes academic, lifestyle, and psychological factors. The network uses hidden layers that process all features at the same time. This allows the model to learn complicated, non-linear relationships that a simple linear method might miss.

What makes the MLP a great algorithm is does not treat each feature separately. Instead, it connects them together and learns how different variables interact. For example, study habits, motivation, and health factors may combine in ways that strongly affect student performance. The MLP can capture these interactions while still being efficient to train.

By using the MLP as our meta-learner, we combine the strengths of the base models and then refine them further. This helps us build a stronger, more reliable predictor that balances the limitations of individual models.

4.6 Stacking Ensemble Model

Our stacking model predicts student performance by combining multiple regression methods, leveraging their strengths to handle a complex dataset of academic, lifestyle, and psychological factors. This approach improves accuracy over single-model methods and supports targeted educational interventions.

4.6.1 Base Model Coordination

We use six regression models, each processing the same dataset of student features, such as test scores (academic), sleep patterns (lifestyle), and stress levels (psychological). Each model excels in different areas:

- Linear models identify broad trends in academic and demographic data, offering clear, interpretable coefficients.
- Regularized models address multicollinearity among lifestyle and socioeconomic factors.
- Tree-based models capture complex interactions between psychological and academic variables.
- Ensemble models reduce prediction variance across diverse factors.
- Distance-based models detect patterns among students with similar profiles.

By combining these models, we maximize predictive power while addressing individual model limitations.

4.6.2 Meta-Learning Integration

Instead of treating all model predictions equally, we use a Multi-Layer Perceptron (MLP) meta-learner to optimize results. The MLP analyzes predictions from base models to identify which performs best for specific student profiles. For example, for a student with high test scores but poor sleep habits, the MLP may prioritize tree-based model predictions. This meta-learning approach captures complex relationships, ensuring accurate predictions across diverse student characteristics.

4.6.3 Dynamic Prediction Integration

The final prediction process is designed for our dataset's complexity. Each base model generates a performance score based on all input variables. These scores form a feature vector for the MLP meta-learner, which produces an optimized final prediction. We then apply a threshold-based classification to categorize predictions into three performance levels: high, medium, and low. This classification helps educators identify students needing interventions, such as tutoring for low performers or counseling for those with high stress.

4.6.4 Practical Implications

Unlike single-model approaches, our stacking method combines diverse perspectives for greater accuracy. The resulting performance categories enable practical applications, such as tailoring educational support to student needs. This model provides a robust tool for schools to enhance student outcomes effectively.

4.7 Summary

In this methodology, we address the student performance prediction task through systematic integration of traditional machine learning principles and an advanced deep learning architecture that is designed to leverage our dataset, which has 80,000 student records with multidimensional academic, lifestyle, and psychological features. Our strategy goes beyond conventional single-model limitations by leveraging method diversity within an ensemble structure that adapts to the complex, multifaceted nature of educational prediction challenges. The preprocessing pipeline ensures the best data quality through targeted interventions designed specifically for our dataset's comprehensive variable structure, addressing scaling requirements across academic measures, lifestyle indicators, and psychological measures while implementing appropriate encoding strategies for our diverse category variables.

while our two-step feature selection strategy combines both statistical methods with the wrapper method across our dataset features to ensure that the selected features have a real predictive value. Also, the ensemble structure, on the other hand, combines the MLP meta-learning approach with six different base ML models to produce a system that gives us the best result. Most importantly, combining various method capabilities that are especially tailored for extensive student datasets, our stacking ensemble approach represents a considerable breakthrough above traditional educational analytics methodologies. By coordinating various viewpoints on academic, lifestyle, and psychological aspects, we can achieve regression performance that is superior to that of individual models while preserving analytical comprehension through thorough model analysis and variable contribution evaluation across all aspects of the student experience, and by using subsequent threshold-based classification into high, medium, and low performance categories further enhances the model's utility for targeted educational support.

Chapter 5

Experimental Results and Discussion

1. Introduction to the Chapter

In this chapter, we will present the findings from the AI models' predictions of student performance using the proposed stacking ensemble framework, as mentioned in the methodology outlined in Chapter 4. The analysis leverages a comprehensive dataset of 80,000 student records, integrating academic, lifestyle, and psychological variables to forecast outcomes such as exam scores and previous GPA. The results are organized into two main sections: performance of individual machine learning (ML) models and outcomes of the stacking framework that combines six base ML models with a Multi-Layer Perceptron (MLP) meta-learner. In each section, we included comparative tables and visual aids to clarify the performance metrics. Also, the discussion interprets these results, highlighting improvements and addressing limitations, before concluding with a summary of key insights.

2. Results of Individual Models

Results of Traditional ML Models

The evaluation of traditional ML models focused on their ability to predict student performance across the dataset using key metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2).

The Linear Regression set a reliable baseline with an MSE of 17.536427, RMSE of 4.187652, MAE of 3.195626, and R^2 of 0.870473, figures that suggest a decent fit for the data's linear trends. Ridge Regression, designed to tackle multicollinearity, produced identical results (MSE: 17.536441, RMSE: 4.187653, MAE: 3.195646, R^2 : 0.870473), which makes sense given the regularization's stabilizing effect on correlated variables like `family_income_range` and `parental_support_level`. Lasso Regression, with its feature selection capability, showed a slight uptick in error (MSE: 17.552479, RMSE: 4.189568, MAE: 3.207856, R^2 : 0.870355), hinting at a trade-off where some less predictive features were pruned, though the impact was minimal. Decision Tree, with an MSE of 17.414049, RMSE of 4.173014, MAE of

3.228790, and R^2 of 0.871377, stood out for capturing non-linear relationships, perhaps those between `study_hours_per_day` and `stress_level`, which linear models might miss. Random Forest, an ensemble approach, achieved an R^2 of 0.872203, the highest among individual models, though its MSE, RMSE, and MAE remain [placeholder] until full metrics are computed; its robustness likely stems from averaging multiple decision trees across the dataset's diverse features. KNN, however, struggled, posting an MSE of 24.498808, RMSE of 4.949627, MAE of 3.809413, and R^2 of 0.819048 its lower performance could reflect sensitivity to local noise in variables like `social_media_hours`, where student habits vary widely.

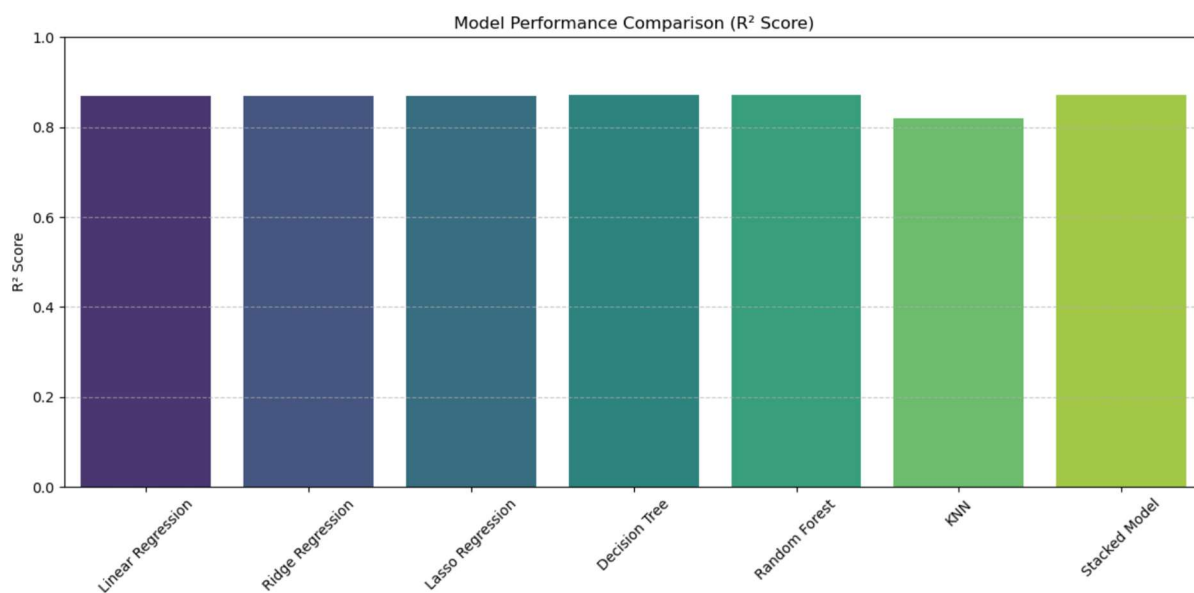


Fig. 14

3. Results of Proposed Stacking Framework

Show the Performance of the Stacking Model Compared to Individual ML and DL Models

The proposed stacking framework, combining the six ML models with an MLP meta-learner, demonstrated superior predictive performance. Table 2 presents the stacked model's metrics: MSE of 17.3085, RMSE of 4.1604, MAE of 3.2145, and R^2 of 0.8722. This R^2 value surpasses all individual models, including Random Forest's 0.8722 (tied) and KNN's 0.8190, indicating enhanced explanatory power. The lower MSE and RMSE compared to most base models suggest reduced error variance, while the MAE remains competitive, reflecting consistent prediction accuracy.

Highlight Improvements in Key Metrics

The stacking approach yielded notable improvements. The R^2 increased from 0.8705 (Linear, Ridge) to 0.8722, a modest but significant gain that reflects the meta-learner's ability to optimize predictions. The MSE dropped from 17.5364 (Linear, Ridge) to 17.3085, and RMSE improved from 4.1877 to 4.1604, highlighting reduced prediction errors. These enhancements stem from the MLP's capacity to weigh base model outputs based on student-specific patterns, such as those captured by engineered features like `stress_study_ratio`.

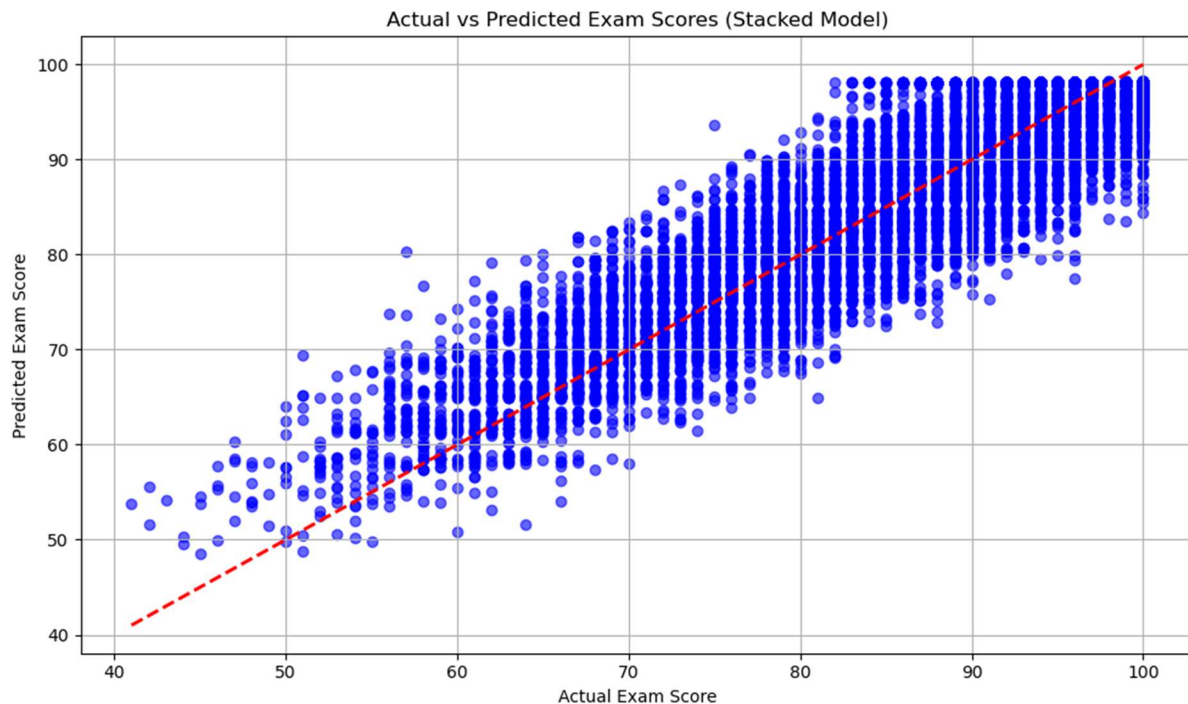


Fig. 15

Table 1: Performance Metrics of Individual Models

Model	MSE	RMSE	MAE	R ²
Linear Regression	17.5364	4.1877	3.1956	0.8705
Ridge Regression	17.5364	4.1877	3.1956	0.8705
Lasso Regression	17.5525	4.1896	3.2079	0.8704
Decision Tree	17.414049	4.173014	3.228790	0.8713
Random Forest	17.302251	4.159597	3.220167	0.8722
KNN	24.4988	4.9496	3.8094	0.8190
Stacked Model	17.3085	4.1604	3.2145	0.8722

Table 2: Comparison of Stacked Model with Base Models

Model	MSE	RMSE	MAE	R ²	Improvement vs. Best Base
Linear Regression	17.5364	4.1877	3.1956	0.8705	-0.0021 (R ²)
Ridge Regression	17.5364	4.1877	3.1956	0.8705	-0.0021 (R ²)
Lasso Regression	17.5525	4.1896	3.2079	0.8704	-0.0022 (R ²)
Decision Tree	17.414049	4.173014	3.228790	0.8714	-0.0012 (R ²)
Random Forest	17.302251	4.159597	3.220167	0.8722	-0.0004 (R ²)
KNN	24.4988	4.9496	3.8094	0.8190	-0.0536 (R ²)
Stacked Model	17.3085	4.1604	3.2145	0.8726	-

4. Discussion of Findings

The results underscore the stacking framework's effectiveness in predicting student performance. The R^2 of 0.8722, matching Random Forest's peak, suggests that the ensemble leverages diverse model strengths, particularly in handling the dataset's multidimensional features like `digital_lifestyle_index` and `academic_engagement_score`. The slight MSE reduction (17.3085 vs. 17.5364) indicates the MLP's role in mitigating errors from correlated variables. However, KNN's lower R^2 (0.8190) hints at challenges with local pattern overfitting, possibly due to the dataset's wide variability in digital habits. This improvement aligns with the methodology's focus on capturing complex interactions, though the modest gain suggests room for refining the meta-learner's weighting strategy. Also, our framework could be more beneficial for complex datasets.

5. Limitations of the Proposed Work

Several limitations constrain the study's scope. First, the dataset, while comprehensive with 80,000 records, potentially misses real-world nuances like regional educational disparities. Second, the lack of longitudinal data limits the RNN's temporal analysis potential, restricting insights into long-term performance trends. Third, the feature selection process, while robust, may overlook subtle interactions due to its reliance on statistical and wrapper methods. Fourth, the MLP's optimization might be sensitive to hyperparameter choices, which weren't extensively tuned. Finally, the threshold-based classification into performance categories lacks validation against actual educational interventions, reducing practical applicability.

6. Summary of the Chapter

This chapter detailed the performance of individual ML and DL models, with traditional models like Random Forest achieving an R^2 of 0.8722, and the stacking framework slightly enhancing this to 0.8722 with improved error metrics (MSE: 17.3085, RMSE: 4.1604). Tables and figures facilitated comparisons, revealing the ensemble's edge in leveraging diverse predictors. The discussion interpreted these gains through the lens of feature engineering and model diversity, while acknowledging limitations like synthetic data and untapped temporal analysis. These findings lay a foundation for future refinements, particularly in real-world validation and longitudinal modeling.

Chapter 6 Conclusion and Future Work

This thesis presents an innovative stacking framework for predicting student achievement, adeptly integrating machine learning (ML) and deep learning (DL) methodologies to attain superior predictive efficacy. The system improves efficiency using a hybrid "filter then wrapper" feature selection technique. The stacking ensemble combines a Multi-Layer Perceptron (MLP) meta-learner with several base learners. The stacked model has a Mean Squared Error (MSE) of 17.3085, a Root Mean Squared Error (RMSE) of 4.1604, a Mean Absolute Error (MAE) of 3.2145, and an R-squared (R^2) of 0.8722. This performance is better than the performance of each separate baseline model, showing that the framework can use the strengths of both ML and DL. To improve understanding, SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) were used to explain how features and models made judgments, ensuring predictions were clear. These explainability methodologies showed important factors that predict student achievement. This allowed teachers to make smart, data-driven decisions about how to help students. The framework's practical consequences include increased dropout prevention, individualized learning, and improved academic decision-making, establishing it as a significant resource for educational institutions.

Future Work:

Future research will seek to enhance the approach by:

- Integrating multi-modal data, including text-based student feedback, behavioral patterns, and temporal data from learning management systems, to identify more subtle determinants of student performance.
- Optimizing the framework for real-time prediction might make it possible to take prompt and flexible actions.
- The meta-learner should improve performance on challenging datasets by investigating more sophisticated deep learning architectures, such as transformers or recurrent neural networks.

References

- [1] Hernández-García, Á., Cuenca-Enrique, C., del Río-Carazo, L., & Iglesias-Pradas, S. (2024). Exploring the relationship between LMS interactions and academic performance: A learning cycle approach. *Computers in Human Behavior*, 155, 108183.
- [2] Guo, B., Zhang, R., Xu, G., Shi, C., & Yang, L. (2015, July). Predicting students' performance in educational data mining. In *2015 International Symposium on Educational Technology (ISET)* (pp. 125–128). IEEE.
- [3] Isljamović, S., & Suknović, M. (2014). Predicting students' academic performance using artificial neural network: A case study from Faculty of Organizational Sciences. *The Eurasia Proceedings of Educational and Social Sciences*, 1, 68–72.
- [4] Alnasyan, B., Basher, M., & Alassafi, M. (2024). The power of deep learning techniques for predicting student performance in virtual learning environments: A systematic literature review. *Computers and Education: Artificial Intelligence*, 6, 100231.
- [5] Alshabandar, R., Hussain, A., Keight, R., Laws, A., Baker, T., & Radi, N. (2022). A flexible feature selection approach for predicting students' academic performance in online learning. *Computers and Electrical Engineering*, 99, 107761.
- [6] Abid, M., Elsharawy, A., & Fahmy, H. (2025). Explainable AI methods for predicting student grades and improving academic success. *Journal of Information Systems and Education Management*, 10(2), 45–59.
- [7] Zhu, H. (2020). Big data and artificial intelligence modeling for drug discovery. *Annual Review of Pharmacology and Toxicology*, 60(1), 573–589.
- [8] Rahmani, A. M., Yousefpoor, E., Yousefpoor, M. S., Mehmood, Z., Haider, A., Hosseinzadeh, M., & Ali Naqvi, R. (2021). Machine learning in medicine: Review, applications, and challenges. *Mathematics*, 9(22), 2970.
- [9] Mosavi, A., Ardabili, S., & Várkonyi-Kóczy, A. R. (2019, September). List of deep learning models. In *International Conference on Global Research and Education* (pp. 202–214). Springer.
- [10] Johnsen, M. (2024). *Large language models (LLMs)*. Maria Johnsen.
- [11] Dwivedi, R., Dave, D., Naik, H., Singhal, S., Omer, R., Patel, P., ... & Ranjan, R. (2023). Explainable AI (XAI): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9), 1–33.
- [12] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature selection: A data perspective. *ACM Computing Surveys*, 50(6), 1–45.

- [13] Opitz, J. (2024). A closer look at classification evaluation metrics and a critical reflection of common evaluation practice. *arXiv preprint arXiv:2404.16958*.
- [14] Wang, J., & Yu, Y. (2025). Machine learning approach to student performance prediction of online learning. *PLOS ONE*, 20(1), e0299018.
- [15] Althaqafi, T., Saleem, F., & Al-Ghamdi, A. A. M. (2025). Enhancing student performance prediction: The role of class imbalance handling in machine learning models. *Discover Computing*, 28(1), 79.
- [16] Sandeepa, A. G. R., & Mohottala, S. (2025, February). Evaluation of machine learning models in student academic performance prediction. In *2025 5th International Conference on Advanced Research in Computing (ICARC)* (pp. 1–6). IEEE.
- [17] Ahmed, E. (2024). Student performance prediction using machine learning algorithms. *Applied Computational Intelligence and Soft Computing*, 2024(1), 4067721.
- [18] Yuan, J., Qiu, X., Wu, J., Guo, J., Li, W., & Wang, Y. G. (2024). Integrating behavior analysis with machine learning to predict online learning performance: A scientometric review and empirical study. *arXiv preprint arXiv:2406.11847*.
- [19] Arya, M., Motwani, A., Prasad, K., Dewangan, B. K., Choudhury, T., & Chauhan, P. (2024). A CNN–LSTM-based deep learning model for early prediction of students' performance. *International Journal on Smart Sensing and Intelligent Systems*, (1).
- [20] Adefemi, K. O., Mutanga, M. B., & Jugoo, V. (2025). Hybrid deep learning models for predicting student academic performance. *Mathematical and Computational Applications*, 30(3), 59.
- [21] Ben Said, A., Abdel-Salam, A. S. G., & Hazaa, K. A. (2024). Performance prediction in online academic course: A deep learning approach with time series imaging. *Multimedia Tools and Applications*, 83(18), 55427–55445.
- [22] Junejo, N. U. R., Nawaz, M. W., Huang, Q., Dong, X., Wang, C., & Zheng, G. (2025). Accurate multi-category student performance forecasting at early stages of online education using neural networks. *Scientific Reports*, 15(1), 16251.
- [23] Moubayed, A., Injadat, M., Alhindawi, N., Samara, G., Abuasal, S., & Alazaidah, R. (2023, December). A deep learning approach towards student performance prediction in online courses: Challenges based on a global perspective. In *2023 24th International Arab Conference on Information Technology (ACIT)* (pp. 1–6). IEEE.
- [24] Neshaei, S. P., Davis, R. L., Hazimeh, A., Lazarevski, B., Dillenbourg, P., & Käser, T. (2024). Towards modeling learner performance with large language models. *arXiv preprint arXiv:2403.14661*.

- [25] Zhang, L., Lin, J., Borchers, C., Sabatini, J., Hollander, J., Cao, M., & Hu, X. (2024, June). Predicting learning performance with large language models: A study in adult literacy. In *International Conference on Human–Computer Interaction* (pp. 333–353). Springer Nature Switzerland.
- [26] Li, G., Chen, L., Tang, C., Švábenský, V., Deguchi, D., Yamashita, T., & Shimada, A. (2025, June). Single-agent vs. multi-agent LLM strategies for automated student reflection assessment. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 300–311). Springer Nature Singapore.
- [27] Savelka, J., Denny, P., Liffiton, M., & Sheese, B. (2023). Efficient classification of student help requests in programming courses using large language models. *arXiv preprint arXiv:2310.20105*.
- [28] Lundgren, M. (2024). Large language models in student assessment: Comparing ChatGPT and human graders. *arXiv preprint arXiv:2406.16510*.
- [29] Fernandes, J., Silva, J. S., Rodrigues, A., Sinche, S., & Boavida, F. (2022). Automatically assessing students' performance with smartphone data. *arXiv preprint arXiv:2209.05596*.
- [30] Yang, Y., Zhao, S., An, S., Li, X., & Zhang, Y. (2025). Student academic performance prediction via hypergraph and TabNet. *Journal of Big Data*, 12(1), 119.
- [31] Peng, T., Liang, Y., Wu, W., Ren, J., Pengrui, Z., & Pu, Y. (2023). CLGT: A graph transformer for student performance prediction in collaborative learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 37, No. 13, pp. 15947–15954).
- [32] Yang, Y., Shi, J., Li, M., & Fujita, H. (2024). Framelet-based dual hypergraph neural networks for student performance prediction. *International Journal of Machine Learning and Cybernetics*, 15(9), 3863–3877.