EnPress

Utilizing Machine Learning Algorithms for Predictive Analysis of Student Performance: A Database-Integrated Approach

Yizhou Zhou, Zhijia Li

National University of Singapore, Singapore.

Abstract: This study embarks on an innovative project aimed at leveraging machine learning algorithms to analyze and predict students' academic performance. By extracting meaningful data from existing datasets and arranging it according to specific test sets, the project seeks to develop a robust framework that facilitates a more personalized learning experience. Utilizing Python functions for database connectivity and MySQL queries for data retrieval, the initiative efficiently structures and sorts data, paving the way for detailed comparative analyses to identify the most precise prediction methods. Subsequent efforts will focus on recommending suitable exercises to students based on predicted scores and study times, enhancing the accuracy and effectiveness of learning strategies.

Keywords: Artificial intelligence (AI); Machine learning (ML); Education; Personalized learning; Student learning; Predictive Analysis; Database Integration

1. Introduction

1.1 Background and Context

In the contemporary educational landscape, it is widely acknowledged that each individual is a unique entity characterized by distinct personality traits, learning philosophies, and adaptable to varied instructional methods. The conventional teaching approach, which tends to perceive all students as a homogeneous group, evidently falls short of maximizing each student's potential learning capacity. This traditional paradigm often entails the allocation of identical homework assignments to all students, a strategy that often fails to achieve the intended outcomes. The limitations stem from the potential mismatch in difficulty levels for diverse students, resulting in limited impact and possibly undermining the effectiveness of the educational process.

Moreover, the burgeoning development in the domains of neural networks and deep learning present promising avenues to cater to the pressing demand for personalized student training. As we stand on the cusp of a revolution in education facilitated by technological advancements, the prospect of offering personalized education tailored to each student's nuances seems well within reach in the foreseeable future.

1.2 Objective of the Study

The study aims to develop a machine-learning-based intelligent system for personalized learning. It focuses on using predictive algorithms to understand individual learning patterns, thereby guiding personalized educational pathways. This enables educators to identify struggling students more efficiently, relieving instructional burden and enhancing teaching strategies. Ultimately, the study seeks to establish a more adaptive educational framework, tailored to individual learning needs.

2. Literature Review

2.1 Previous Studies on Machine Learning in Education

Early educational AI focused on intelligent tutoring systems and basic machine learning algorithms (Kumar & Kim, circa 2014). Advances in data mining laid the foundation for in-depth analytics on student learning behavior (Neumann & Waight, 2019; Tiwari, 2023). Algorithms like C4.5 are now used for accurate sentiment analysis in educational settings (Pahuriray et al., 2022). This evolution led to the modern, multifaceted AI-driven educational tools that employ more sophisticated algorithms.

2.2 Importance of Predictive Analysis in Educational Strategies

Predictive analysis is key for shaping educational strategies, especially in mathematics and tertiary education settings (Salles et al., 2020; Gray, 2014). It uses data analysis for assessing traditional competencies and identifying at-risk students, thereby enhancing learning outcomes.

2.3 Design a personalized e-learning system based on IRT and ANN

Web-based education often lacks personalization and interactivity. A proposed intelligent system tailors tests and adapts to learners' needs, similar to human instructors (Xu & Wang, 2006). It uses Item Response Theory (IRT) for student evaluation and offers adaptive post-tests based on this ability.

3. Methodology

3.1 Dataset Description

In the burgeoning field of modern education, a deep comprehension and analysis of student learning patterns have emerged as pivotal subjects of discussion. This study seeks to cultivate a nuanced understanding by building a comprehensive database system. Utilizing AI technology, the system performs a deep analysis and prediction of student learning characteristics, thereby providing personalized learning guidance.

3.1.1 Database structure

The database system comprises three main components aimed at tracking and analyzing student information, exercise topics, and performance metrics.

Database Component	Key Fields	Purpose
Student Information	id, name, gender, major	Facilitates identification and tracking of each student's basic information.
Exercises	exer_id,exer_category, exer_fullmark, etc.	Enables evaluation of exercise difficulty and popularity, assists in student performance analysis.
Time-Mark	tm_id, stu_id, exer_id, finish_time, mark	Tracks and analyzes students' learning progress, including completion time and scores on exercises.

3.1.1 Student Feature Design Principle

During the construction of this AI-assisted educational system, various students exhibiting notably diverse learning characteristics were selected to demonstrate the system's capability to accurately identify and analyze multifaceted learning patterns. Each student representation encapsulates different learning intensities, abilities, and habits, aiming to encompass a plethora of possible learning situations and challenges. If data designed around diverse student characteristics can be precisely analyzed and predicted by machine learning algorithms, it would substantiate the efficacy of the algorithms in fostering personalized learning environments.

3.2 Tools and Techniques used

This study leverages a multi-faceted toolkit to foster personalized education through data analysis. Central components include:

SQL Databases: Serve as the backbone for storing, retrieving, and managing rich datasets, including student profiles and performance metrics.

Jupyter Notebooks: Enable real-time data analysis and visualization, crucial during data preprocessing and exploratory phases.

Python Programming: Chosen for its ease and extensive libraries like Pandas, NumPy, and Scikit-learn, Python facilitates the coding and analytical phases. It also integrates with graphical libraries like Seaborn and Matplotlib for effective visualization.

These tools collectively create a streamlined data management and analytical ecosystem, enhancing the study's efficacy in delivering personalized educational experiences.

3.2.1 Linear Prediction

Utilizing linear prediction, we performed regression analyses to forecast student performance. The established linear relationships between factors allow for targeted, personalized learning strategies.

3.2.2 Support Vector Machine (SVM)

SVM classifies students based on learning patterns and preferences. Feature scaling and the kernel trick enhance precision and resolve complex classification problems.

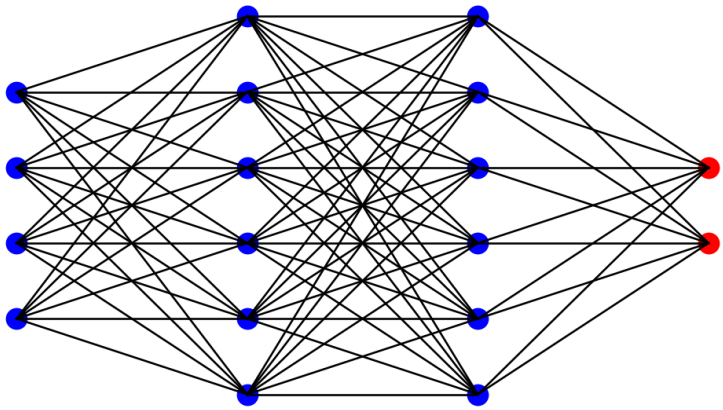
3.2.3 Random Forests (RF)

Random forests employ multiple decision trees for robust classification. Feature scaling and grid search with optimal parameters refine the model, providing detailed insights into student learning paths.

3.2.4 Index Analysis Prediction

Index Analysis Prediction captures complex, non-linear trends in student data. Variable weights and feature indices enable nuanced, question-specific predictions.

3.2.5 Artificial neural network (*ANN*)



We've incorporated Artificial Neural Networks (ANN) to enhance prediction accuracy in student analytics. Trained on datasets that include exercise indicators, test full scores, average grades, and question frequencies, the ANN model aims to predict student completion time and test scores. Post-training, this model serves as an analytical tool for gauging student learning states. It offers precise predictions on key metrics like completion time and test scores, contributing to more personalized and efficient educational strategies.

3.2.6 Implementation Strategy

The system adopts a phased approach, initially aggregating data in SQL databases as a foundation for analysis. Python scripts in Jupyter Notebooks handle data processing and kickstart predictive model development, integrating a range of machine learning and forecasting techniques for student analytics.

3.3 Data Parsing and Retrieval

3.3.1 Python Functions

1. Central Executor: Serves as the hub. Takes student ID to trigger various sub-functions, including database calls and data transformation. Outputs detailed student performance metrics and graphs. Segments and analyzes data with predictive methods.

2. Index Tuner: Customizes special indices based on student profiles, converting them into SQL commands for future retrieval.

3. Student Simulator: Simulates student performance using random functions, storing the data in SQL format. Utilizes both normal and random distributions for realism.

4. Bulk Predictor: Extends the single-student predictive function of the main notebook to multiple student IDs. Exports results as structured CSV files.

5.Neural Network Trainer: Handles machine learning through neural networks, employing a three-layer model for predictive analytics.

6. Result Visualizer: Imports and visualizes results from different methodologies using Seaborn and Matplotlib for comparative analysis.

These notebooks integrate seamlessly, forming a cohesive and efficient data analysis pipeline.

3.3.2 MySQL Queries

The MySQL query in this study employs a nested sub-query approach focused on filtering records based on stu_id and exer_category. These filtered records are further sorted by exer_id. The sub-query links the time_mark table with the exercises table, incorporating details like exer_fullmark, exer_avgmark, exer_time, and exer_popularity. These fields serve to track exercises, evaluate student performance, analyze time management, and gauge exercise popularity.

3.3.3 Analytical Goals

The query is designed with multiple facets of in-depth analysis in mind, encompassing performance metrics, time analysis, comparative analysis, and trend prediction. Fields like mark, exer_fullmark, and exer_avgmark facilitate detailed analysis of individual and collective performance. Additionally, finish_time and avgtime are key for studying time management skills. Average scores and time metrics also allow for the relative assessment of an individual student's performance. The comprehensive data set paves the way for applying machine learning algorithms for predictive analytics.

4. Implementation

4.1 Development and Features

The code underwent 11 iterations, evolving from basic data retrieval in version 1.0 to intricate analytics and features by version 4.1. Error handling, data visualization, and student-specific analytics were sequentially introduced. Version 6.1 and onwards included advanced scoring indices and a recommendation engine.

4.2 Code Maturation and Analytics

Versions 7.1 to 9.1 focused on code maintenance, bug fixes, and documentation. A tag-assignment mechanism was introduced in version 8.1. By version 9.1, the system gained enhanced analytical capabilities through integration with ChatGPT API.

4.3 Predictive Analytics

In version 10.1, linear regression was initially used for score prediction. Later versions broadened the range of predictive algorithms, culminating in a multifaceted visualization in version 11.3 for easy comparison of model efficacy.

Overall, the iterative process led to a streamlined, robust analytics system capable of intelligent insights and predictive modeling.

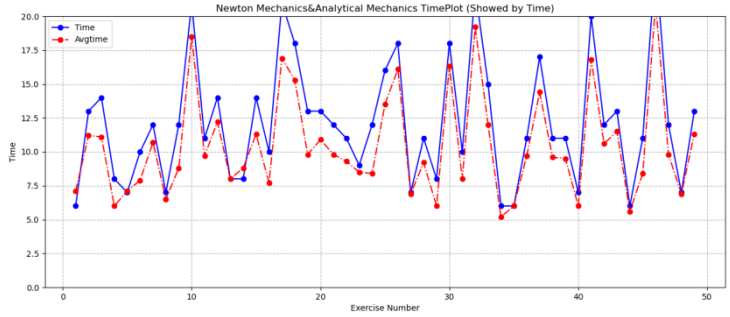
5. Results and Analysis

5.1 Case Study: Student ID 4 Analysis and Forecasting

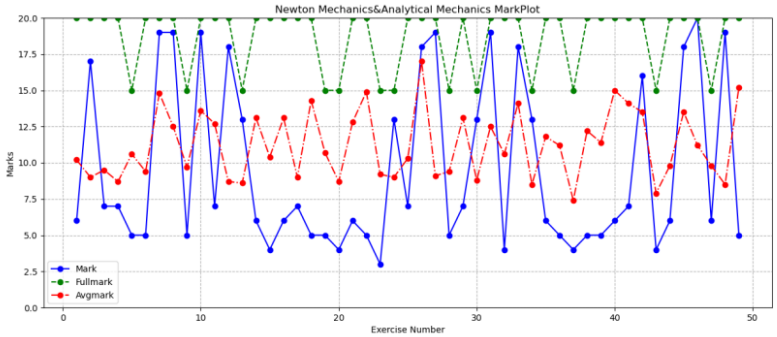
In the construction of the grade database for our analysis, specific characteristics were pre-defined for each student to simulate real-world variations. For Student ID 4, named Abel, his scoring attributes were intentionally designed as follows: On exercises with high popularity, his scores were set to be around 90%. Conversely, for less popular exercises, he was designed to score as low as 30%. These settings were implemented under the assumption that his overall average score would be 65%.

In the generated analytical report, the first section provides basic information about the student, including details such as student ID and gender.

In the report, the score analysis is presented first. A graph featuring a blue line displays Abel's scores across different exercises. The blue line illustrates significant volatility; some exercises are scored highly, while others are not.



Next, the analysis moves to the speed at which Abel completes exercises. In the corresponding graph, a blue line represents the time Abel takes for each exercise, and a red line indicates the average time across all students. Observations from the graph suggest that Abel's speed is marginally slower than average.



Key performance indices for Abel are as follows:

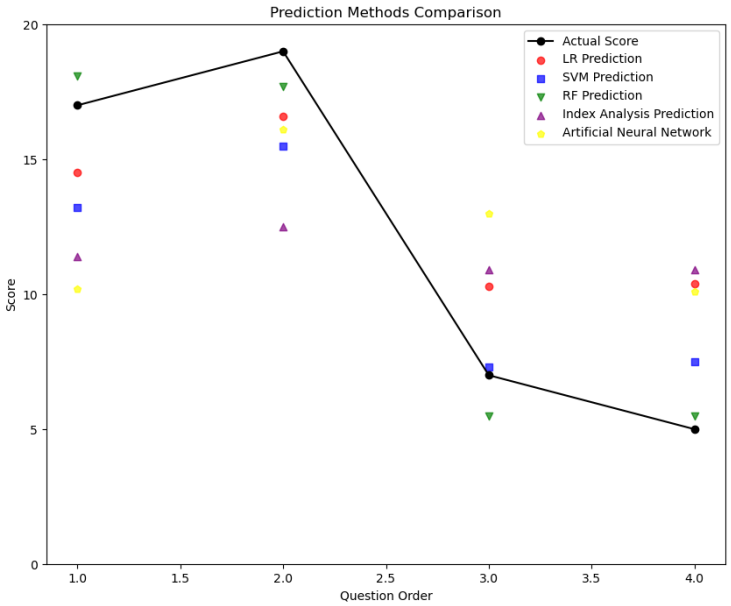
Index Category	Performance Indices for Abel
score_index	50.7
speed_index	49.2
strange_index	31.8
difficult_index	50.5
easy_index	50.9
popular_index	84.4

Abel's average score, converted to percentage, is 50.7, suggesting a weak grasp of Newton Mechanics&Analytical Mechanics. His capabilities are especially lacking in niche exercises while being better in commonly encountered easy exercises. The speed index of 49.2 indicates a slower solving pace.

In the predictive segment, questions 67, 44, 17, and 162 were selected for analysis. Five different forecasting techniques were used: Linear Regression (LR), Support Vector Machine (SVM), Random Forest (RF), Index Analysis, and Neural Network.

stu_id	exer_id	mark	LR Prediction	SVM Prediction	RF Prediction	Index Analysis Prediction	Netural Network
4	67	7	10.3	7.3	5.5	10.9	9.8
4	44	19	16.6	15.5	17.7	12.5	13.4
4	17	17	14.5	13.2	18.1	11.4	10.5
4	162	5	10.4	7.5	5.5	10.9	9.6

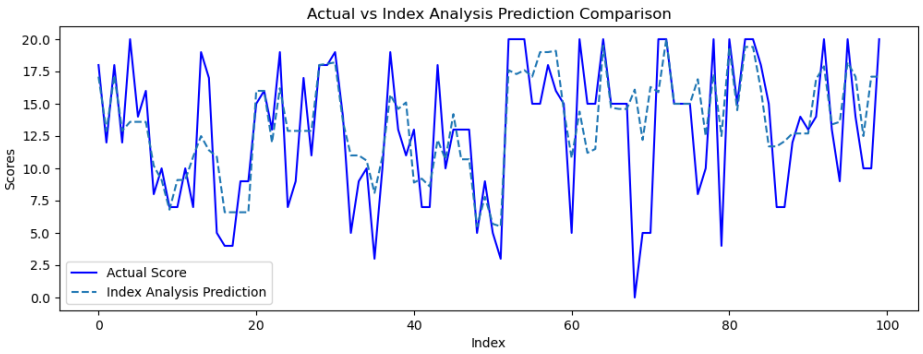
Following the predictions, a comparative visual graph was generated based on the results from the five forecasting methods.

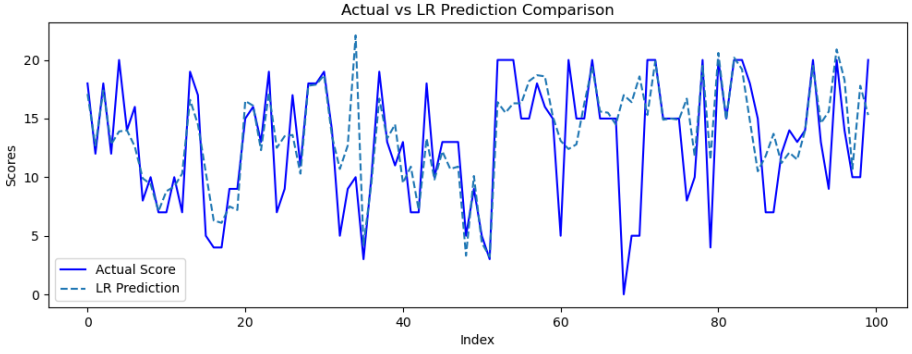


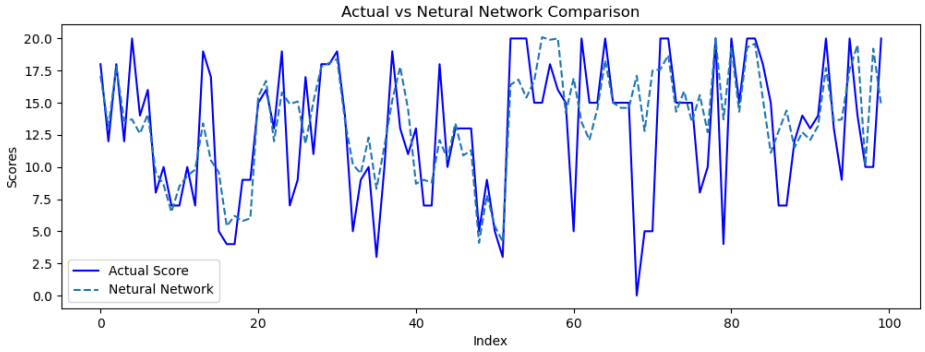
In summary, the analysis successfully pinpoints Abel's academic characteristics. The forecasting algorithms accurately identify his learning traits and predict scores for the selected questions based on these traits. The Random Forest method demonstrated remarkable precision, with an average prediction error of only 1.1 points per question, validating the scientific and effective nature of the algorithms.

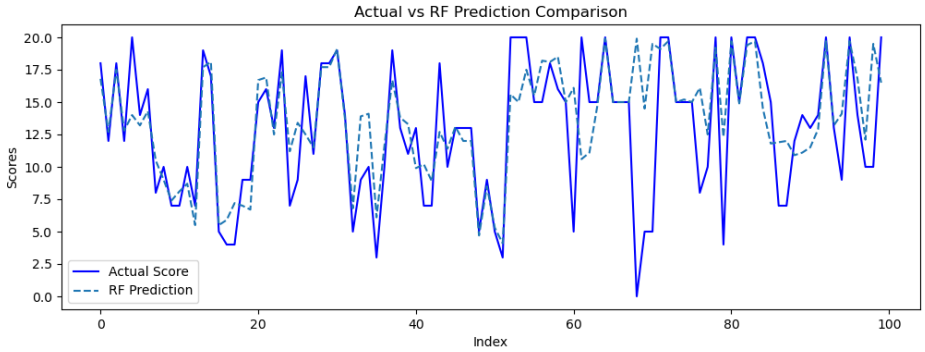
5.2 Accuracy Assessment: Random Sampling of 100 Exercises

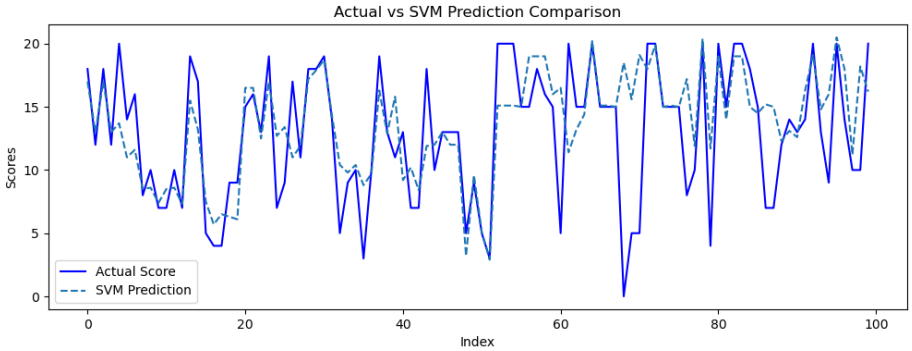
To assess the system's predictive accuracy, a random sample of 100 exercises was chosen. Various forecasting models were applied, including linear regression, SVR, and Random Forest Regression, Index Analysis Prediction and Artificial Neural Network. The outcomes were then compared with the real scores. The results demonstrated an impressive degree of accuracy, confirming that the system is robust enough for reliable educational analytics.







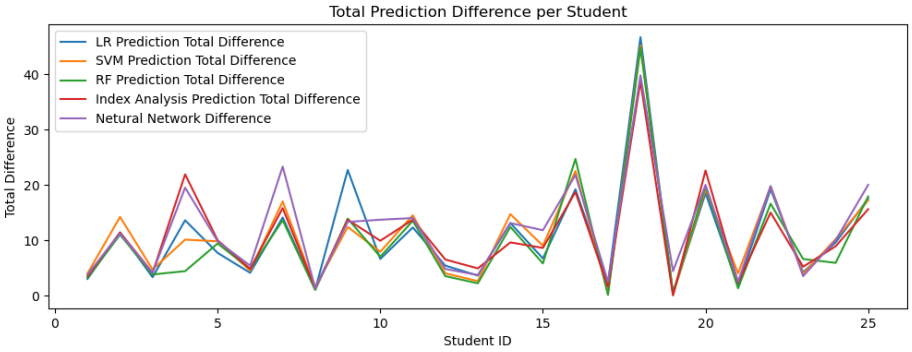




5.3 Comparative Analysis Across Students

The predictive models were further assessed by applying them to a diverse set of students. The results section, illustrated in the following chart, shows varying predictability among students. Some students' learning traits make them easier to forecast accurately, while others are more challenging to predict. In this assessment, the total predictive bias for four randomly selected exercises was calculated for each of four sample students. The results could include anomalous deviations, where a student may unexpectedly fail or succeed in an exercise, thereby skewing the aggregate predictive bias.

Overall, for the majority of students, the cumulative predictive bias across four exercises did not exceed a 10-point margin, indicating a high level of reliability in the predictive models.



6. Discussion

6.1 Analysis of Predictive Results

The predictive models exhibited varied performance, corroborating our earlier observations of the differential predictability of student performance. Specifically, the ANN (Artificial Neural Networks) model displayed suboptimal accuracy. This might be due to its tendency to overfit the data or the complex nature of educational analytics that makes it challenging for ANN to make precise predictions.

Comparatively, SVM (Support Vector Machines) and RF (Random Forest) models were more accurate than Linear Regression. The former two are capable of capturing more complex relationships in the data, thus outperforming the latter in this context. Index Analysis Prediction, although the simplest, provided quick and reasonably accurate forecasts, making it a viable choice for large-scale student score predictions.

6.2 Future Implications of this study

The predictive power of these models paves the way for personalized education. By forecasting scores for unattempted exercises, we can identify the likely low-scoring areas for each student. Educators can then design targeted training on these problem areas, enabling students to improve their weaknesses efficiently. This not only helps in individual academic growth but also offers a scalable solution for personalized learning on a broader scale.

7. Conclusion

7.1 Summary of Findings

The study developed a multi-faceted educational analytics system that went through various stages of refinement. While ANN (Artificial Neural Networks) proved to be the most complex and time-consuming, it did not deliver as effectively in terms of predictive accuracy. On the other hand, RF (Random Forest) emerged as the most accurate, albeit with higher computational demands. SVM (Support Vector Machines) and Index Analysis Prediction offered a balanced approach. Though slightly less precise, they excelled in speed, making them ideal for large-scale predictive tasks.

7.2 Conclusion

The project's multiple predictive models underscore the vast potential of predictive analytics in the educational sector. From ANN's complexity to RF's accuracy and the rapid capabilities of SVM and Index Analysis Prediction, our system offers a range of options suited for various educational needs and scales. This diversity in analytical tools not only paves the way for more personalized and effective learning but also extends the scope for wide-scale implementation across different educational settings. As a culmination of prior discussions and analyses, this research lays a solid foundation for future endeavors to refine and diversify predictive models, thereby enriching the overall quality of education. Future work should target the existing limitations and investigate the system's adaptability to various educational landscapes.

References

[1] Kumar, R., Kim, J. Special Issue on Intelligent Support for Learning in Groups. Int J Artif Intell Educ 24, 1–7 (2014).

[2] Neumann, K., & Waight, N. (2019). Call for Papers: Science teaching, learning, and assessment with 21st century, cutting-edge digital ecologies. Journal of Research in Science Teaching. 56. 115-117.

[3] Tiwari, R. (2023). The Integration of AI and Machine Learning in Education and its Potential to Personalize and Improve Student Learning Experiences. International Journal of Scientific Research in Engineering and Management. 7.

[4] Pahuriray, Archolito V. et al. "Flexible Learning Experience Analyzer (FLExA): Sentiment Analysis of College Students through Machine Learning Algorithms with Comparative Analysis using WEKA." International Journal of Emerging Technology and Advanced Engineering (2022): n. pag.

[5] Salles, F., Dos Santos, R. & Keskpaik, S. When didactics meet data science: process data analysis in large-scale mathematics assessment in France. Large-scale Assess Educ 8, 7 (2020).

[6] Gray, G., McGuinness, C., Owende, P., & Carthy, A. (2014). A Review of Psychometric Data Analysis and Applications in Modelling of Academic Achievement in Tertiary Education. Journal of Learning Analytics, 1, 75-106.