A ROBUST PSO-LSTM FRAMEWORK FOR PREDICTING ACADEMIC PERFORMANCE WITH EXPERIMENTAL VALIDATION

Wenting Cui, Adisak Sangsongfa, and Noppadol Amdee*

Received: November 07, 2025; Revised: April 25, 2026; Accepted: April 30, 2026

Abstract

Accurately predicting students' academic performance is crucial for identifying students with learning difficulties and tailoring educational interventions effectively. This study proposed a PSO-LSTM framework that combines Particle Swarm Optimization (PSO) with Long Short-Term Memory (LSTM) to improve prediction accuracy by optimizing LSTM hyperparameters. The model was evaluated on a dataset of 722 students from the Computer Science program at Wenzhou Business College, covering six academic years (2018–2024). Two experimental scenarios were designed: semester-based GPA prediction and year-based CGPA prediction. The results indicated that PSO-LSTM outperformed traditional machine learning models (Decision Tree, Random Forest, SVM, KNN) and standard deep learning models (ANN, LSTM) in both experimental settings. In predicting the cumulative grade point average (CGPA), the model's accuracy increased from 87.59% in DSTU5 to 95.17% in DSTU7, and further to 96.55% after applying SMOTE. This method effectively alleviates bias caused by class imbalance, particularly by improving recall for identifying the minority (at-risk) class, thereby significantly enhancing fairness and robustness (e.g., DSTU5 increased by 8.0%). These findings highlight PSO-LSTM as a highly effective framework for predicting academic performance, with strong potential for early intervention strategies and personalized learning recommendations.

Keywords: **Academic performance prediction; Imbalanced data; PSO-LSTM; SMOTE**

Introduction

Academic performance prediction is of great significance in higher education, as it enables the identification of students at academic risk early, allowing for timely and accurate intervention (Onyema *et al.*, 2022). In recent years, educational data mining (EDM) has gained increasing popularity (Onyema *et al.*, 2022). This field utilizes machine learning to analyze academic data, predicting grades, identifying potential dropouts, and assessing academic risk (Shafiq *et al.*, 2022;

Department of Industrial Technology Management, Faculty of Industrial Technology, Muban Chom Bueng Rajabhat University, Ratchaburi, Thailand, 70150. E-mail: cwt@wzbc.edu.cn, ORCID: 0009-0008-5163-5344; Adisaksan@mcru.ac.th, ORCID: 0000-0002-6231-4000; noppadolamd@mcru.ac.th, ORCID: 0000-0002-2413-1430

* *Corresponding author*

DOI: https://doi.org/10.55766/sujst12036

Suranaree J. Sci. Technol. 33(5):030411(1-14)

Amdee, 2024). It uses historical academic records to support data-driven decision-making, ultimately enhancing student retention and overall academic performance. Related capabilities can be directly implemented in smart campus scenarios, such as academic early warning systems, academic monitoring dashboards, and adaptive learning platforms. Based on this, this paper uses multi-semester academic records as input and focuses on predicting the final cumulative grade point average (CGPA). CGPA is an important indicator of academic achievement and subsequent development potential, and its accurate prediction also has practical value for educational management and policymaking (Nachouki & Abou Naaj, 2022).

Deep learning, particularly long short-term memory networks (LSTMs), has demonstrated strong performance on educational time series data, capturing long-term dependencies (Lindemann *et al.*, 2021; Alwarthan *et al.*, 2022; Al-Azazi & Ghurab, 2023; Alnasyan *et al.*, 2024). However, the ultimate performance of LSTM is highly dependent on hyperparameter settings (Greff *et al.*, 2016; Ahmed, 2024); if the tuning and search efficiency are insufficient, the model's accuracy and stability will be limited. Therefore, heuristic search has attracted widespread attention. Compared with GA-LSTM, Bayesian optimization, and AutoML frameworks, particle swarm optimization (PSO) offers better search efficiency and scalability in high-dimensional hyperparameter spaces. It can dynamically adjust LSTM hyperparameters during training, thereby improving prediction accuracy and stability. Compared with traditional grid/random search, it also offers superior computational efficiency (Yu & Zhu, 2020; Kumar *et al.*, 2022; Ruma *et al.*, 2023; Salem *et al.*, 2024). Based on existing research, common algorithms used for academic prediction cover traditional machine learning and deep learning: artificial neural networks (ANN) (Farissi *et al.*, 2020; Sandoval-Palis *et al.*, 2020; Yağcı, 2022; Kukkar *et al.*, 2023), logistic regression (LR) (Hassan *et al.*, 2019; Cabo, 2021; Nabil *et al.*, 2021), decision trees (DT) (Sandoval-Palis *et al.*, 2020; Yağcı, 2022), random forests (RF) (Hassan *et al.*, 2019; Farissi *et al.*, 2020), support vector machines (SVM) (Hassan *et al.*, 2019; Nabil *et al.*, 2021; Venkatachalam & Sivanraju, 2024), K-nearest neighbor (KNN) (Yağcı, 2022), and time series models such as LSTM (Wu *et al.*, 2019; Kukkar *et al.*, 2023). Comprehensive evaluations also show that ANN, LR, RF, KNN, and SVM are still the most widely used traditional baselines (Almasri *et al.*, 2019). At the same time, LSTM's advantages in processing academic sequences have gradually expanded its application (Wu *et al.*, 2019). In terms of class imbalance, real-world data often presents a skewed distribution with ‘medium CGPA samples accounting for the majority’ (Alsariera *et al.*, 2022; Baashar *et al.*, 2022; Cu *et al.*, 2024). As a result, the model tends to favor the mainstream class, leading to poor recall for the minority class (high-risk) and compromised fairness (Al-Azazi & Ghurab, 2023; Kukkar *et al.*, 2023). To alleviate this problem, existing work has extensively utilized SMOTE oversampling to enhance minority class coverage on structured tabular data, offering superior interpretability and lower computational cost compared to some generative approaches (Chawla *et al.*, 2002; Scott & Plested, 2019; Sharma *et al.*, 2022; Venkatachalam & Sivanraju, 2024).

The data in this paper comes from 48,758 academic records from a university's computer science program from September 2018 to December 2024, encompassing 722 students and 28 courses. The data were structured into a student-course performance matrix, where each row represents an individual student and each column corresponds to either a course grade, a semester GPA, or a demographic attribute. This matrix was subsequently transformed into a temporal sequence format to support LSTM modeling. We systematically evaluate the performance of academic prediction using two complementary settings: Experiment 1, "semester-by-semester GPA prediction," predicts subsequent semester GPA based on completed courses at different academic stages (Tasks DSTU1–DSTU4); Experiment 2, "year-by-year cumulative CGPA prediction," predicts final CGPA based on academic performance in the first, first two, and first three years, respectively (Tasks DSTU5–DSTU7). This dual-perspective design encompasses both short-term fluctuations and long-term trajectories, facilitating easier comparison of prediction capabilities across different time scales.

Against this backdrop, we propose a PSO-LSTM framework, where LSTM is utilized to model academic sequence dependencies, PSO is employed for hyperparameter optimization, and SMOTE is combined to alleviate class imbalance, thereby balancing accuracy and fairness. The research objectives are: (1) to verify whether PSO-driven hyperparameter search can steadily improve the prediction accuracy and robustness of LSTM on multiple tasks (DSTU1–DSTU7) in an imbalanced scenario; (2) to quantify the improvements in overall accuracy and fairness indicators such as macro-average F1 and minority class recall by comparing with strong baselines (DT, RF, SVM, KNN) and deep baselines (ANN, standard LSTM); (3) to compare the information content at the

semester-level and grade-level time scales, and to test the advantages of long-term cumulative academic trajectories over short-term GPA snapshots in CGPA prediction. Furthermore, the framework has a strong system implementation and can be embedded in learning management systems (LMS) and educational data platforms to support continuous tracking and personalized intervention in the learning process. This enables data-driven decisions, such as curriculum construction, resource allocation, and student management, thereby promoting educational equity and personalized development.

Materials and Methods

This section provided a detailed explanation of the overall research method framework used to predict student grades in this study. First, the data collection and preprocessing process was introduced to ensure the authenticity and integrity of the data. Secondly, the overall process of model training and testing was clarified for the experimental design. Given the significant imbalance in student grade data, this study designed a solution to address it. In addition to addressing the issue that the LSTM model's predictive performance is sensitive to hyperparameters, the particle swarm optimization (PSO) algorithm was introduced for automatic hyperparameter optimization, and the PSO-LSTM framework was proposed. On this basis, several classic models were selected for comparative experiments, and their predictive performance and generalization abilities were systematically evaluated using multiple evaluation indicators to ensure the scientific effectiveness of the proposed method.

Data Collection and Preprocessing

The data used in this study were obtained from the academic performance records of computer science students at a Chinese university, spanning the academic years 2018 to 2024. This data not only contained basic information about students (such as gender, class, year of admission, etc.) but also included course grades for multiple semesters, semester grade point averages (GPAs), and final cumulative grade point averages (CGPAs), providing rich feature information for subsequent prediction tasks. In data preprocessing, missing values and abnormal data were removed, and 722 valid student records were selected to improve the quality of the basic data for model training. The dataset comprises 45 features, primarily categorized into three groups: demographic information (x1-x11), course grades (x12-x39), and GPA-related data (x40-x45), as detailed in Table 1.

Table 1. Feature Description of The Dataset

Attributes	Description	Value
x1	Class	TEXT
x2	Gender	1 F, 2 M
x3	Grade Level	2018-2021
x4	Major	TEXT
x5	Political Affiliation	TEXT
x6	Ethnicity	TEXT
x7	Hometown	TEXT
x8	Dormitory Type	1,2,3,4,6
x9	Off-Campus Accommodation	True, False
x10	Counselor Name	TEXT
x11	Class Advisor Name	TEXT
x12-x39	Final Exam Scores	0-100
x40-x45	GPA (1st-6th Semesters)	0-5
y	Final CGPA	0-5

Data Cleaning and Transformation

Initial data cleaning was performed to remove records with missing values or outliers, ensuring the integrity of the predictive model. For feature transformation, categorical features (x1-x11) were converted into numerical values using one-hot encoding for non-ordinal variables and label encoding for ordinal ones. Course grades (x12-x39) and GPA-related features (x40-x45) were standardized using Z-score normalization to ensure that all features contributed equally to the model without scale-induced bias. The final cumulative GPA (CGPA) was discretized into ordinal categories: below 2.0 as E (0), 2.0-2.49 as D (1), 2.5-3.49 as C (2), 3.5-3.99 as B (3), and 4.0 or higher as A (4).

LSTM Input Sequence Construction

To adapt the structured data for the LSTM network, we transformed the 'student-course' matrix into a three-dimensional temporal sequence. We explicitly defined one time step t as one academic semester, where the sequence length T varies by task (e.g., T=6 for DSTU7). At each time step t, the input vector x_t is composed of:

Dynamic Features (d_t): The standardized exam scores (x12-x39) and semester GPA (x40-x45) relevant to semester t.

Static Features (s): The time-invariant demographic data (x1-x11).

To capture the interaction between background factors and academic progression, we employed a repeated concatenation method, in which the static vector s is concatenated with the dynamic vector dt at each time step t. Formally,

the input is expressed as $X_t = d_t \oplus s$, where $\oplus$ denotes the concatenation operator. This ensures that the LSTM maintains contextual awareness of the student's profile throughout the hidden state updates.

Correlation Analysis

A comprehensive correlation analysis was conducted to examine the relationships between the input features and their influence on the target variable, the final cumulative GPA (CGPA), denoted as y. The results were visualized using a correlation heatmap (Figure 1), which provided several key insights into the strength and direction of associations between variables.

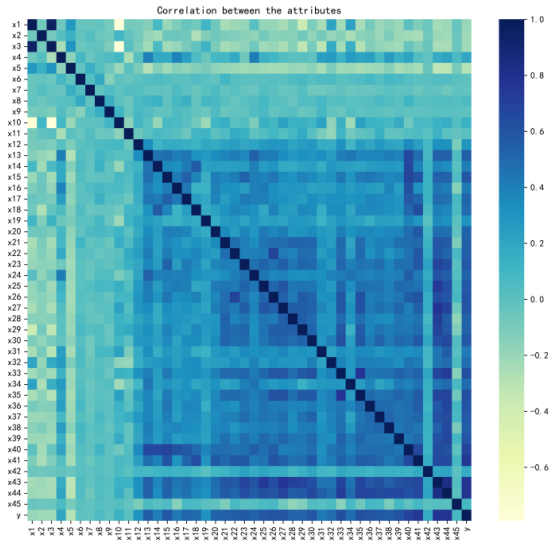


Figure 1. Correlation heatmap of input features and final CGPA (y)

As shown in Figure 1, Class (x1), Gender (x2), Grade Level (x3), and Political Affiliation (x5) exhibit weak negative correlations with the final CGPA, suggesting that these demographic factors have a minimal influence on academic performance. In contrast, Major (x4) shows a moderately positive correlation, approximately four times stronger than the variables mentioned above, indicating that students from specific academic majors may have a relative advantage. Nevertheless, the impact of Major is less significant than the strong influence of course grades.

Course grades (x12-x39) show strong positive correlations with the final cumulative GPA (CGPA), underscoring their substantial predictive power for academic performance. In contrast, among the semester-specific GPA features (x40-x45), the third (x42) and sixth semesters (x45) exhibit relatively weaker correlations with the final CGPA. This may reflect variations in course difficulty or fluctuations in students' academic engagement across different semesters.

The figure shows the Pearson correlation coefficients for each attribute in the dataset (including demographic data, course grades, and semester GPAs) with the final cumulative GPA (CGPA). Higher positive values (shown in darker shades) indicate stronger positive correlations, particularly between individual course scores and CGPA. Gender, political affiliation, and dormitory type show weak or negligible correlations.

Handling of Data Imbalance

To address class imbalance, particularly in predicting lower and higher CGPA ranges, the Synthetic Minority Over-sampling Technique (SMOTE) was employed. SMOTE was used to balance the distribution of CGPA categories by generating synthetic instances for underrepresented classes, thereby mitigating model bias toward majority groups. This preprocessing step enhances both predictive accuracy and classification fairness by ensuring a more equitable representation across all CGPA levels. Notably, as shown in Figure 2, most students fall into Category 2 (GPA 2.5-3.5), indicating class imbalance in the original dataset.

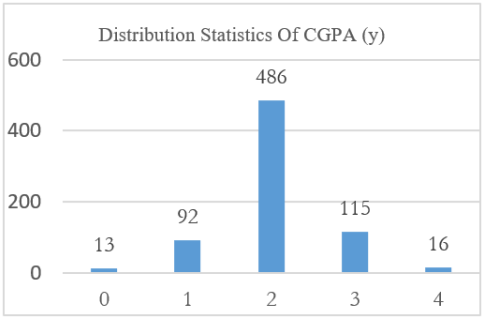


Figure 2. Distribution Statistics of CGPA (y)

Experimental Design

To comprehensively evaluate the predictive effect on students' academic performance, we designed two types of experiments: one for semester GPA and the other for academic-year CGPA, to explore the changing rules and predictability of academic performance across different time scales.

Experiment 1: Semester-Based GPA Prediction

First, we conducted a prediction experiment using time-series data on students' GPAs each semester. Specifically, the data were divided by

semester, and the grouping method is detailed in Table 2. To maintain consistency in data processing, we used the same discretization method as CGPA to convert GPA into classification labels, which served as the model's prediction target. The starting point of this experiment was to explore the impact of students' early academic performance on their GPA in subsequent semesters, that is, whether the model could reasonably predict future performance trends when existing academic information is limited. Through this design, we aim to gain a deeper understanding of students' performance dynamics and to provide data support for personalized academic interventions.

Table 2. Student-related attributes

Dataset	Data Used	Target
DSTU1	First and second semesters (x12-x20)	3rd semester GPA
DSTU2	First to third semesters (x12-x24)	4th semester GPA
DSTU3	First to fourth semesters (x12-x29)	5th semester GPA
DSTU4	First to fifth semesters (x12-x34)	6th semester GPA
DSTU5	First year (first and second semesters) (x12-x20)	Final CGPA
DSTU6	First and second years (first to fourth semesters) (x12-x29)	Final CGPA
DSTU7	First three years (first to sixth semesters) (x12-x39)	Final CGPA

Experiment 2: Year-Based CGPA Prediction

Considering the importance of long-term academic performance (CGPA) in evaluating students' comprehensive abilities, the second experiment focused on predicting CGPA. In this experiment, we divided the data by academic year, and the specific grouping method is also shown in Table 2. By using academic data, such as students' course grades across different academic years, the model attempted to predict their final cumulative GPA. We aimed to investigate a core question: as academic data accumulate, will the model's CGPA prediction improve significantly in accuracy? Additionally, this experiment focuses on the impact of students' early grades on their final comprehensive performance, providing a theoretical basis for early academic warning and support mechanisms.

Dataset Partitioning

In the above two experiments, we constructed multiple sub-datasets (DSTU1-DSTU7) using different feature sets for prediction tasks at different stages. To ensure the scientific nature of model evaluation and the comparability of results, all data were strictly preprocessed before training, including outlier removal, handling of missing values, and standardization. In addition to reasonably evaluating the model's generalization ability, all datasets were randomly split into training and test sets at an 80:20 ratio, ensuring the model also exhibits good predictive performance on unseen data.

The core of this experiment was the trend in student performance over time, so the time order of the data was strictly maintained when dividing the data set to avoid future information leakage into the training set. In this way, we aimed to demonstrate the predictive value of early academic performance for subsequent performance and to validate the feasibility of predicting academic performance using time-series analysis.

PSO-LSTM Framework Design

Considering that the performance of the LSTM (Long Short-Term Memory) neural network depends mainly on the reasonable setting of hyperparameters, such as weights, thresholds, number of hidden neurons, learning rate, etc., this study introduced the particle swarm optimization (PSO) algorithm in the model design to achieve automatic optimization of LSTM hyperparameters. In the past, hyperparameters were often adjusted manually in practical applications. Still, this method relies on researchers' experience, lacks systematicity, and makes it difficult to obtain the optimal global solution. Therefore, the LSTM model combined with PSO (PSO-LSTM) is expected to effectively improve the accuracy and stability of academic performance prediction.

Figure 3 shows the overall process of the PSO-LSTM framework, which includes the following five main steps:

Step 1: Initialize parameters. In the particle swarm, each particle represents a set of possible LSTM parameter configurations (e.g., weights, thresholds, number of hidden neurons). The particle swarm optimizes the fitness function through collaborative search in the multi-dimensional parameter space, thereby minimizing the LSTM model's loss function.

Step 2: Data preprocessing. Before formally training the model, the original data were integrated as necessary, including unifying data files from different sources or formats into a standard structure for subsequent processing.

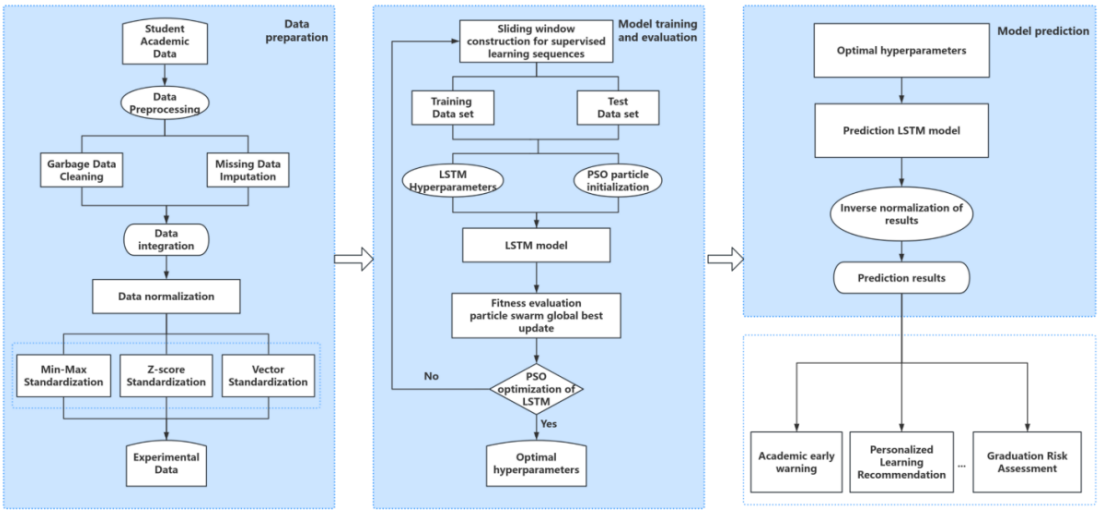


Figure 3. PSO-LSTM framework for student performance prediction

At the same time, because the feature value ranges vary widely during model training, this can easily lead to problems such as gradient imbalance. To address this, the data were normalized to the [0,1] interval. In addition, to further improve the convergence speed and stability of the model, some data were also standardized to adjust the mean and variance of the features, making them more suitable for the input requirements of the LSTM network.

Step 3: Optimize the Fitness Function. This paper uses the LSTM model's mean squared error (MSE) during training as the fitness function to guide the particle swarm algorithm's effective search in the parameter space. The particle swarm dynamically adjusted its search direction based on the fitness function, allowing each generation of particles to evolve toward minimizing the error. Through repeated iterations, the algorithm found the parameter configuration that minimizes the training error.

Step 4: Update Particle Swarm Parameters. In particle swarm optimization of LSTM parameters, the speed and position of the particles must be dynamically adjusted according to the feedback from the fitness function, and both the individual optimal solution (pBest) and the global optimal solution (gBest) must be recorded. After each iteration, the PSO updated each particle's state based on its fitness, gradually converging the parameters toward the optimal solution.

Step 5: Build a PSO-LSTM prediction model. When the PSO algorithm's fitness function meets the preset convergence criterion or the maximum number of iterations is reached, we can determine a set of LSTM parameter combinations with the best performance. After obtaining the optimal parameters, this paper retrained the LSTM model using them to ensure the rationality and optimality of the parameter selection.

By introducing the particle swarm optimization (PSO) algorithm to automatically optimize the LSTM model's hyperparameters, the subjective problem caused by manual parameter adjustments based on experience was effectively avoided. Compared to traditional hyperparameter optimization methods (such as grid search or random search), PSO achieved faster convergence and higher search efficiency while effectively avoiding local optima, thereby enhancing the model's final generalization performance.

PSO-LSTM Hyperparameter Optimization

To ensure the robustness and reproducibility of the proposed framework, the hyperparameter search strategy for the PSO-LSTM model is explicitly defined. We utilized the Particle Swarm Optimization (PSO) algorithm to automatically navigate the high-dimensional hyperparameter space, identifying the optimal configuration that maximizes the model's predictive performance.

Optimized Parameters and Search Space

We selected five critical hyperparameters for simultaneous optimization, establishing their search boundaries based on empirical evidence and computational constraints. Specifically, the number of LSTM units (H) was explored within the range of [16, 128] (integer values) to ensure sufficient representation capacity while maintaining manageable computational overhead. To mitigate the risk of overfitting, the dropout rate (d) was constrained to 0.1-0.5. The learning rate (η) was

searched over the interval [10^{-4}, 10^{-2}] to facilitate stable convergence. Additionally, the batch size (B) and the number of training epochs (E) were searched within the ranges of [16, 128] (integer values) and [10, 200], respectively.

To further enhance generalization, an L1 regularization term (coefficient = 0.001) was applied to the LSTM layer. The output layer employed a Softmax activation function to support the five-class ordinal CGPA classification task.

PSO Configuration and Fitness Function

The PSO algorithm was implemented with a swarm size (N) of 10 and a maximum of 50 iterations (I). To ensure the stability and reproducibility of the optimization process, the inertia weight (w) and acceleration coefficients (c1, c2) were maintained at their default values of 0.5, as defined in the pyswarm library.

During optimization, each particle represents a candidate set of LSTM hyperparameters. At each iteration, particles update their positions based on their individual best solution (pBest) and the global best solution (gBest), enabling effective exploration and convergence toward the optimal region of the search space.

The fitness function was defined as the classification accuracy on a validation set. Since PSO is formulated as a minimization problem, the fitness value was computed as the negative accuracy (-Accuracy). For each particle, the corresponding LSTM model was trained on the training set and evaluated on the validation set to compute the fitness value. This strategy ensures that the optimization process prioritizes generalization over overfitting.

Comparative Models

To comprehensively evaluate the actual effect of the proposed PSO-LSTM model on academic performance prediction, this paper selected a variety of traditional machine learning and deep learning methods widely used in related fields as comparison benchmarks. These benchmark models included Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and the standard LSTM model without Particle Swarm Optimization (PSO).

This article uses accuracy, recall, and F1-score as the primary evaluation indicators to measure the model's performance from multiple perspectives. Accuracy reflects the proportion of correct classifications by the model, while recall focuses on the model's ability to identify positive classes (such as excellent academic performance). The F1-score balances accuracy and recall, making it an ideal metric for evaluating datasets with class imbalance.

Evaluation Metrics

This paper selects accuracy, precision, recall, and F1-score as the primary evaluation indicators to comprehensively evaluate the model's classification performance in predicting student scores. These indicators are calculated through four basic classification results, including true positives (TP), that is, the number of students correctly identified as 'score warnings' by the model; true negatives (TN), that is, the number of students correctly identified as 'normal scores'; false positives (FP), that is, the number of students who are normal but are mistakenly judged as warnings; and false negatives (FN), that is, the number of students who are grade warnings but are missed. The calculation formulas for each evaluation indicator are as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Pr\,e\,cision = \frac{TP}{TP+FP} \quad (2)$$

$$Re\,c\,all = \frac{TP}{TP+FN} \quad (3)$$

$$F1-Score = 2 * \frac{Pr\,ecision * Re\,call}{Pr\,ecision + Re\,call} \quad (4)$$

The K-fold Stratified Cross-Validation method was used to improve the stability and reliability of model evaluation, which divided the dataset into K parts. While K-1 selects parts of the data for training each time, the remaining data is used for testing, and this is repeated K times. Finally, we calculated the average performance indicators of all folds. The stratified strategy can maintain the proportions of samples across categories in each fold, especially on datasets with imbalanced categories. This method helped to improve the fairness and credibility of model evaluation.

Results and Discussion

This section presents two series of experimental results: semester-based GPA prediction and year-based CGPA prediction. Additionally, we compared the performance of the PSO-LSTM model with that of benchmark models, including decision tree (DT), random forest (RF), support vector machine (SVM), K-nearest neighbor (KNN), and standard LSTM. The datasets used in the experiment include DSTU1-DSTU7, and model performance was evaluated using accuracy, precision, recall, and F1-score. Additionally, we examined the impact of data imbalance processing

methods on model performance and investigated how various data balancing techniques can improve predictive performance.

Experimental Setup

The experimental dataset was split into 80% for training and 20% for testing, as explained in Section 3.2. The data growth is mainly reflected in two aspects:

DSTU1 to DSTU4: Based on the expansion of semesters, each dataset contains more semester information.

DSTU5 to DSTU7: Each dataset includes additional academic year data on the original basis, expanding the years covered.

The experiments aim to analyze how including more academic data improves model performance in predicting GPA and CGPA.

Table 3 presents the configuration parameters for the various machine learning models used in this study. These parameters are set to ensure that each model can fully adapt to data growth, thereby optimizing the prediction of students' academic performance.

Table 3. Parameter configuration of Models

Model	Parameters
DT	DecisionTreeClassifier(Criterion='gini', Spliter='best', random_state=42)
RF	RandomForestClassifier(N_estimator=50, criterion='gini', random_state=42)
SVM	SVC(random_state=42, C=1.0, kernel='rbf', degree=3, gamma='scale')
KNN	KNeighborsClassifier(random_state=42, n_neighbors=5, weights='uniform')
ANN	(hidden_layer_sizes=(64,10), activation='relu') (optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
LSTM	LSTM(64, activation='relu') Dropout(0.1) Dense(5, activation='softmax') (optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
PSO-LSTM	Units (H): Optimized via PSO (Range: 16–128); Dropout (d): Optimized via PSO (Range: 0.1–0.5); Learning Rate (η): Optimized via PSO (Range: 10^{-4}-10^{-2}); Batch Size (B): Optimized via PSO (Range: 16–128); Epochs (E): Optimized via PSO (Range: 10–200); Activation: 'relu' (LSTM), 'softmax' (Dense); Regularization: L1 (0.001)

Effect of Dataset Size and Structure

Semester-Based Prediction (DSTU1 to DSTU4)

The PSO-LSTM model shows the impact of cumulative academic data on the prediction effect when predicting semester-based GPA (DSTU1 to DSTU4). Overall, the more semester data is added, the higher the accuracy and F1 score the model achieves, indicating that historical academic performance has significant predictive value for GPA prediction. However, the results also show that the impact of data from different semesters on model performance fluctuates, especially in DSTU2 and DSTU4, whose prediction accuracy is lower than that of DSTU3, suggesting that the model's adaptability to these datasets differs.

As shown in Table 4, PSOs-LSTM outperforms all benchmark models in the semester GPA prediction task, demonstrating that PSO is effective at optimizing hyperparameters. The trend shows that the model's prediction accuracy generally improves when the semester data increases. Furthermore, DSTU3 achieves the highest accuracy (80%), indicating that the data from three semesters effectively balances data volume and complexity. However, DSTU2 (in early semesters) and DSTU4 (in later semesters) exhibit lower accuracy, highlighting the challenges in predicting early academic performance due to student adaptation, as well as the increased complexity and variability in coursework and student trajectories that characterize later semesters. Additionally, semester-based GPA prediction is inherently less accurate than final CGPA prediction, as cumulative academic trends influence CGPA. In contrast, short-term fluctuations have a greater impact on semester GPA. These findings suggest that academic performance prediction models should prioritize long-term cumulative data while incorporating additional contextual features to improve semester GPA predictions.

Year-Based CGPA Prediction (DSTU5 to DSTU7)

The CGPA prediction results based on academic-year data (DSTU5 to DSTU7) indicate that the model's predictive performance improves steadily as more academic years are added. This indicates that accumulated academic data is highly valuable for predicting long-term academic performance.

Table 4. Semester-Based GPA Prediction Results

Dataset	Model	Accuracy	Precision	Recall	F1-Score
	DT	0.5586	0.4367	0.4814	0.4471
	RF	0.6759	0.5234	0.4976	0.4926
	SVM	0.6759	0.6566	0.518	0.5577
DSTU1	KNN	0.6345	0.6345	0.5218	0.5552
	ANN	0.6276	0.4767	0.4855	0.4626
	LSTM	0.6552	0.5327	0.5343	0.5151
	PSO-LSTM	0.7172	0.5933	0.5734	0.5654
	DT	0.5448	0.4214	0.4041	0.4104
	RF	0.6414	0.6591	0.4019	0.4334
	SVM	0.6483	0.5145	0.3767	0.3952
DSTU2	KNN	0.5655	0.4315	0.3381	0.4345
	ANN	0.6483	0.5823	0.459	0.4959
	LSTM	0.6207	0.5098	0.4499	0.4726
	PSO-LSTM	0.7103	0.7414	0.4984	0.5552
	DT	0.6414	0.5573	0.5026	0.5157
	RF	0.7724	0.5194	0.4567	0.4665
	SVM	0.7724	0.5478	0.4474	0.4631
DSTU3	KNN	0.6897	0.642	0.4927	0.5259
	ANN	0.7517	0.5663	0.5099	0.5254
	LSTM	0.731	0.6581	0.5366	0.5586
	PSO-LSTM	0.8	0.6992	0.6319	0.6535
	DT	0.5586	0.4719	0.5733	0.5017
	RF	0.6966	0.6877	0.4735	0.5328
	SVM	0.6621	0.3127	0.3007	0.2959
DSTU4	KNN	0.6345	0.3429	0.2949	0.3014
	ANN	0.6069	0.3582	0.3981	0.3609
	LSTM	0.6552	0.4347	0.4535	0.4386
	PSO-LSTM	0.7172	0.5021	0.4835	0.4897

Compared to a GPA prediction based on a single semester, a CGPA prediction is more accurate because aggregating grades across multiple semesters can reduce the impact of short-term fluctuations, making the prediction results more stable. As shown in Table 5, the PSO-LSTM model outperforms the baseline model in all datasets, achieving the highest accuracy and F1 score. Specifically, the model's accuracy increased from 87.59% in DSTU5 to 95.17% in DSTU7, indicating that as the amount of historical academic data increased, the model's stability improved further, and its generalization ability also increased.

These results highlight that traditional machine learning models (DT, RF, SVM, KNN) struggle to match the performance of deep learning-based approaches, particularly in DSTU7, where PSO-LSTM substantially improves recall and F1-score. The strong performance of PSO-LSTM compared to standard LSTM confirms that PSO-driven hyperparameter optimization enhances predictive accuracy and model generalization. The findings suggest that long-term academic trends are more predictable than short-term semester-based performance variations, making CGPA a more stable and robust target for academic forecasting models.

Impact of Data Imbalance Handling

In the CGPA prediction task, data category imbalance may cause the model to overpredict categories with higher proportions, thereby affecting overall prediction performance. To address this issue, this study employed the SMOTE method to balance the dataset, ensuring equal representation across CGPA categories and mitigating the impact of category distribution on model learning.

Before applying SMOTE, due to the high proportion of medium CGPA values in the dataset, the PSO-LSTM model is more likely to be biased towards this category when making predictions, resulting in a high misclassification rate for students with extremely high or extremely low CGPA. After SMOTE processing, the data volume for each CGPA category reached 486, thereby eliminating the imbalance problem. This uniform distribution enabled the model to learn a more balanced decision boundary, thereby improving the ability to distinguish students of different CGPA levels and making the prediction results fairer and more accurate.

Table 5. Year-Based CGPA Prediction Results

Dataset	Model	Accuracy	Precision	Recall	F1-Score
DSTU5	DT	0.7586	0.5808	0.5616	0.5593
	RF	0.8621	0.4936	0.4948	0.4938
	SVM	0.8276	0.4991	0.4558	0.4604
	KNN	0.7724	0.6349	0.4846	0.5352
	ANN	0.7862	0.5051	0.5137	0.5038
	LSTM	0.8	0.5464	0.5401	0.5424
	PSO-LSTM	0.8759	0.8544	0.7427	0.7442
DSTU6	DT	0.7586	0.5244	0.5868	0.5322
	RF	0.8621	0.6982	0.5865	0.6246
	SVM	0.8552	0.6950	0.5793	0.6187
	KNN	0.8207	0.6736	0.5317	0.5803
	ANN	0.8345	0.7736	0.6208	0.6514
	LSTM	0.8413	0.7098	0.7344	0.7067
	PSO-LSTM	0.9103	0.8338	0.7829	0.7989
DSTU7	DT	0.8345	0.7678	0.7237	0.6837
	RF	0.8897	0.7223	0.7076	0.7133
	SVM	0.8897	0.5174	0.5217	0.5164
	KNN	0.7655	0.6213	0.5035	0.5435
	ANN	0.8690	0.8978	0.7503	0.7786
	LSTM	0.9034	0.9168	0.8422	0.8622
	PSO-LSTM	0.9517	0.9542	0.8989	0.9139

Table 6. Effect of Data Balancing on Model Performance

Dataset	Model	Accuracy	Precision	Recall	F1-Score
DSTU5	PSO-LSTM (Before)	0.8759	0.8044	0.7427	0.7442
	PSO-LSTM (After)	0.8966	0.8107	0.8226	0.8003
DSTU6	PSO-LSTM (Before)	0.9103	0.8338	0.7829	0.7989
	PSO-LSTM (After)	0.9172	0.8203	0.8457	0.8169
DSTU7	PSO-LSTM (Before)	0.9517	0.9542	0.8989	0.9139
	PSO-LSTM (After)	0.9655	0.9418	0.9381	0.9356

The results in Table 6 demonstrate that applying SMOTE significantly improves the predictive performance of the PSO-LSTM model, particularly in addressing the uneven distribution of CGPAs. Among them, the prediction accuracy on the DSTU5 dataset increases by 2.1%, and the recall rate increases by 8.0%, indicating that data balancing has a particularly significant effect on predicting early academic data. In addition, the accuracy of DSTU6 and DSTU7 increases by 0.7% and 1.4%, respectively, further verifying SMOTE's generalization ability across different datasets. The improvement in recall rates across datasets indicates that, after balancing, the model's ability to identify high-achieving and academically at-risk students has been enhanced, overcoming the previous bias towards the medium CGPA category.

From Table 6, we can see that the accuracy of each dataset has improved to varying degrees, with the DSTU7 group achieving the highest score of 96.55%. This phenomenon demonstrates that data balancing enhances model performance, particularly by improving recall and optimizing prediction accuracy. The application of the SMOTE method to structured data sets is evident, effectively enhancing the stability of the CGPA prediction model.

Results Interpretation and Discussion

The experimental results show that the PSO-LSTM model outperforms traditional machine learning and basic deep learning models in semester-based GPA prediction (DSTU1-DSTU4) and year-based CGPA prediction (DSTU5-DSTU7). As shown in Figure 4, with more academic data, the accuracy of CGPA prediction increases gradually and is significantly higher than that of semester-based GPA prediction. This indicates that long-term cumulative academic trends are more effective at predicting students' academic performance than short-term semester fluctuations and further supports the notion that CGPA is a more stable target for academic prediction.

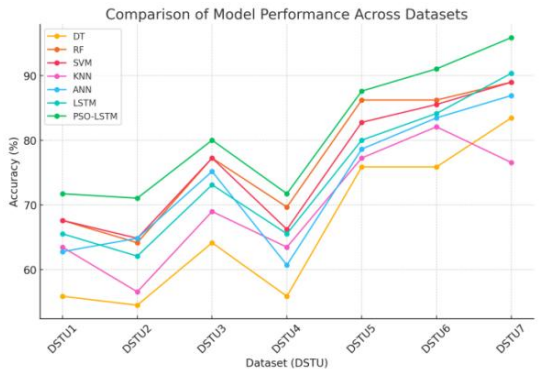


Figure 4. Comparison of Model Performance Across Datasets

As shown in the results, the performance of traditional machine learning models (DT, RF, SVM, KNN) declines as the dataset grows, whereas that of deep learning models (ANN, LSTM, PSO-LSTM) remains high. As shown in Table 4 (semester-based GPA prediction) and Table 5 (year-based CGPA prediction), DT consistently achieves the lowest accuracy, indicating that its ability to handle complex temporal relationships in academic performance data is limited. RF and SVM perform better but still struggle to handle long-term dependencies. For instance, in DSTU5–DSTU7, as the dataset grows, the accuracy of CGPA prediction remains relatively low compared to the other models.

Among deep learning models, ANNs and standard LSTMs outperform traditional models, demonstrating that neural networks are superior at capturing complex relationships among academic performance characteristics. However, the results of the two prediction tasks, DSTU6 and DSTU7, show that the LSTM achieves higher accuracy than the ANN when processing long-term academic data. The PSO-LSTM model further improves the performance of the standard LSTM by optimizing hyperparameters, achieving the highest accuracy and F1 score across all datasets.

As illustrated in Figure 5, the prediction accuracy of various models differs markedly in semester-based GPA prediction tasks. In contrast, model performance in year-based CGPA prediction appears relatively more stable. This observation suggests that, as academic data accumulates over time, deep learning models, particularly PSO-LSTM, demonstrate enhanced modelling and predictive capabilities, especially when processing data with long-term dependencies. Conversely, traditional machine learning models struggle to capture these temporal dynamics effectively. Considering the data, DSTU1 through DSTU4 correspond to semester-based GPA prediction, while DSTU5 through DSTU7 are designed for cumulative CGPA prediction. Notably, prediction accuracy improves significantly in the latter, reinforcing the advantage of deep learning approaches in handling aggregated, longitudinal data.

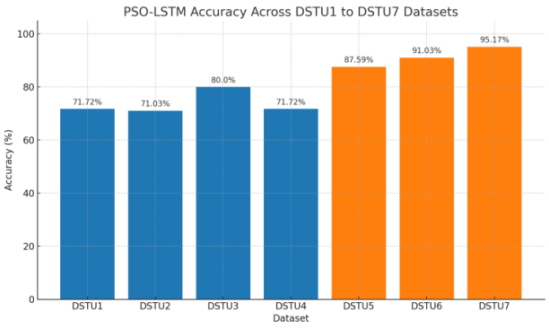


Figure 5. PSO-LSTM accuracy on DSTU1–DSTU7 datasets

This study also revealed the significant impact of data category imbalance on the accuracy of academic performance prediction. When category imbalance is not addressed, the PSO-LSTM model exhibits a bias towards the medium CGPA category, which accounts for the largest proportion, leading to a high misclassification rate when the model encounters students with extreme CGPAs (e.g., high or low grades).

As shown in Figure 6, the model trained on data that SMOTE has not balanced tends to classify most students into the main CGPA range. After SMOTE processing, the model's recognition performance in the extreme CGPA category with a small sample size has been significantly improved, effectively alleviating category imbalance bias and improving overall prediction reliability. The left panel shows results without handling class imbalance, while the right panel shows results after applying SMOTE. Rows indicate actual CGPA categories, and columns represent predicted categories. Categories 0-4 correspond to GPA bands: E (<2.0), D (2.0-2.5), C (2.5-3.5), B (3.5-4.0), and A (>4.0). SMOTE improves recall for minority classes and enhances fairness in classification.

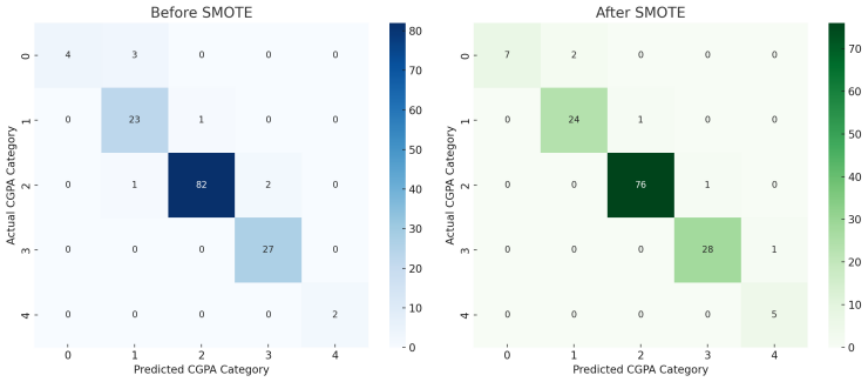


Figure 6. Confusion matrices of CGPA prediction using the PSO-LSTM model on the DSTU7 dataset before and after SMOTE

As shown in Figure 7, the bar chart illustrates the impact of SMOTE on four key performance metrics: Accuracy, Precision, Recall, and F1-score. After SMOTE preprocessing, the model's overall performance improved notably, particularly on the DSTU7 dataset, which had the most severe class imbalance. Specifically, accuracy increased by 1.4%, while recall improved by 3.92%. The significant enhancement in recall indicates that the model has become more sensitive in identifying students at both ends of the academic spectrum, those with outstanding performance and those at academic risk, thereby mitigating bias toward the majority. Moreover, the concurrent improvement in the F1-score suggests that SMOTE not only boosts classification accuracy but also enhances the model's balance and robustness in recognizing minority-class students. These improvements collectively contribute to greater fairness and reliability in the model's predictive outcomes.

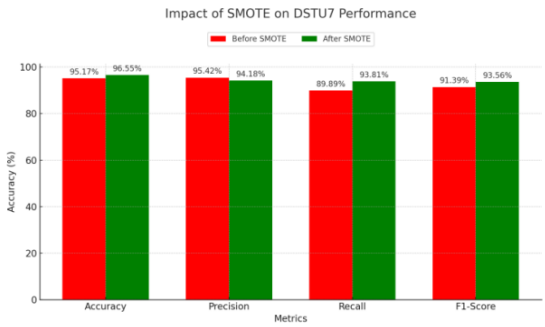


Figure 7. Comparison of PSO-LSTM performance on DSTU7 before and after SMOTE

In summary, the experimental results indicate that the model's predictive performance continues to improve as semester data accumulate. This indicates that long-term learning process data significantly enhanced predictive performance. At the same time, using data preprocessing methods such as SMOTE effectively alleviates class imbalance, thereby improving the model's ability to identify different student types. In addition, the LSTM-based deep learning model outperforms the traditional machine learning method, indicating that the time-series model has unique advantages for revealing the laws of change in students' academic performance. Overall, the PSO-LSTM model demonstrates excellent prediction stability and accuracy in this study, thereby enhancing the accuracy of academic performance prediction and providing adequate technical support for personalized tutoring and early warning mechanisms for students, which reflects a high practical application value.

Conclusions

This study proposes an improved prediction framework, PSO-LSTM, that combines particle swarm optimization (PSO) with a long short-term memory (LSTM) network to enhance the accuracy and fairness of predicting college students' academic performance. Through systematic experiments on multiple subtasks, the results show that the model outperforms traditional machine learning models (such as RF, SVM, and KNN) and unoptimized standard LSTM models in predicting semester GPA (DSTU1–DSTU4) and annual average CGPA (DSTU5–DSTU7). For example, on the DSTU7 dataset, PSO-LSTM improved the F1 score by 5.17% and the recall rate by 5.67% compared to the standard LSTM, reflecting a stronger ability to recognize the minority class.

The study also finds that the CGPA prediction based on annual cumulative data is more accurate

than that based on single-semester GPA data, particularly in terms of recall rate. This finding indicates that features derived from long-term academic performance are more representative and can effectively improve the model's stability and robustness.

To alleviate the problem of data imbalance, the study introduced the SMOTE oversampling strategy to amplify the minority class samples, which achieved good results in the CGPA prediction task, significantly improved the model's ability to identify marginal groups (such as middle-level students), and verified the critical value of the data balance strategy in improving the fairness of prediction.

Although the PSO-LSTM model performed well in the experiment, it still faces challenges in practical applications. For example, although the global search mechanism of the PSO algorithm can optimize the model parameters, it also brings high training overhead, especially when facing large-scale or dynamically updated data, its efficiency may become a constraint factor; in addition, the model's adaptability in integrating multi-source heterogeneous data (such as psychological behavior records, classroom interaction logs, etc.) also needs to be strengthened.

Furthermore, this study is subject to certain limitations. Specifically, the dataset is derived from a single computer science program at one institution, which may limit the generalizability of the findings. Although the proposed PSO-LSTM framework achieved strong performance within this context, its effectiveness may vary across different academic disciplines and institutional environments.

Overall, PSO-LSTM offers a practical and scalable modeling paradigm for enhancing the accuracy and fairness of college student performance prediction, with significant theoretical value and promising practical applications. Future research should focus on validating the proposed framework across diverse academic programs and multiple institutions, improving computational efficiency, and extending the model to incorporate multi-source heterogeneous educational data.

Funding

The authors received no specific financial support for the research, authorship, and/or publication of this article.

Author Contributions

Conceptualization: W.C., A.S., N.A.; Methodology: W.C., A.S., N.A.; Validation: N.A.; Formal Analysis: W.C.; Investigation: W.C., A.S., N.A.; Data Curation: W.C.; Writing – Review & Editing: W.C., N.A. All authors have read and agreed to the published version of the manuscript.

Ethics Statement

Ethical approval was not required for this study because it did not involve human participants, animals, identifiable personal data, or materials requiring institutional ethical review.

Declaration of Generative AI Use

The authors declare that no Generative AI tools were used in the preparation of this manuscript.

Conflict of Interest

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

Data Availability Statement

No new data were generated or analyzed in support of this research.

Acknowledgement

Data from the School of Information and Engineering, Wenzhou Business College, supported this work. The authors thank Wenzhou Business College and Muban Chom Bueng Rajabhat University for their support.

References

Ahmed, E. (2024). Student performance prediction using machine learning algorithms. *Applied Computational Intelligence and Soft Computing, 2024*(1), 4067721. https://doi.org/10.1155/2024/4067721

Al-Azazi, F. A., & Ghurab, M. (2023). ANN-LSTM: A deep learning model for early student performance prediction in MOOC. *Heliyon, 9*(4), e15382. https://doi.org/10.1016/j.heliyon.2023.e15382

Almasri, A., Celebi, E., & Alkhawaldeh, R. S. (2019). EMT: Ensemble meta-based tree model for predicting student performance. *Scientific Programming, 2019*(1), 3610248. https://doi.org/10.1155/2019/3610248

Alnasyan, B., Basheri, 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, 7*, 100231. https://doi.org/10.1016/j.caeai.2024.100231

Alsariera, Y. A., Baashar, Y., Alkawsi, G., Mustafa, A., Alkahtani, A. A., & Ali, N. (2022). Assessment and evaluation of different machine learning algorithms for predicting student performance. *Computational Intelligence and Neuroscience, 2022*(1), 4151487. https://doi.org/10.1155/2022/4151487

Alwarthan, S. A., Aslam, N., & Khan, I. U. (2022). Predicting student academic performance at higher education using data mining: A systematic review. *Applied Computational Intelligence and Soft Computing, 2022*(1), 8924028. https://doi.org/10.1155/2022/8924028

Amdee, N. (2024). A comparative study of the applicability of regression models in predicting student academic performance. *Naresuan University Engineering Journal, 19*(1), 39-49. https://ph01.tci-thaijo.org/index.php/nuej/article/view/255799

Baashar, Y., Alkawsi, G., Mustafa, A., Alkahtani, A. A., Alsariera, Y. A., Ali, A. Q., Hashim, W., & Tiong, S. K. (2022). Toward predicting student's academic performance using artificial neural networks (ANNs). *Applied Sciences, 12*(3), 1289. https://doi.org/10.3390/app12031289

Cabo, C. (2021). Use of machine learning to identify predictors of student performance in writing viable computer programs with repetition loops and methods. In *2021 IEEE Frontiers in Education Conference (FIE)* (pp. 1-9). IEEE. https://doi.org/10.1109/FIE49875.2021.9637302

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357. https://doi.org/10.1613/jair.953

Cu, N. G., Nghiem, T. L., Ngo, T. H., Nguyen, M. T. L., & Phung, H. Q. (2024). Increment of academic performance prediction of at-risk student by dealing with data imbalance problem. *Applied Computational Intelligence and Soft Computing, 2024*(1), 4795606. https://doi.org/10.1155/2024/4795606

Farissi, A., Dahlan, H. M., & Samsuryadi. (2020). Genetic algorithm-based feature selection with ensemble methods for student academic performance prediction. *Journal of Physics: Conference Series, 1500*(1), 012110. https://doi.org/10.1088/1742-6596/1500/1/012110

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems, 28*(10), 2222-2232. https://doi.org/10.1109/TNNLS.2016.2582924

Hassan, S. U., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S., & Herrera, F. (2019). Virtual learning environment to predict withdrawal by leveraging deep learning. *International Journal of Intelligent Systems, 34*(8), 1935-1952. https://doi.org/10.1002/int.22129

Kukkar, A., Mohana, R., Sharma, A., & Nayyar, A. (2023). Prediction of student academic performance based on their emotional wellbeing and interaction on various e-learning platforms. *Education and Information Technologies, 28*(8), 9655-9684. https://doi.org/10.1007/s10639-022-11573-9

Kumar, G., Singh, U. P., & Jain, S. (2022). An adaptive particle swarm optimization-based hybrid long short-term memory model for stock price time series forecasting. *Soft Computing, 26*(22), 12115-12135. https://doi.org/10.1007/s00500-022-07451-8

Lindemann, B., Müller, T., Vietz, H., Jazdi, N., & Weyrich, M. (2021). A survey on long short-term memory networks for time series prediction. *Procedia CIRP, 99*, 650-655. https://doi.org/10.1016/j.procir.2021.03.088

Nabil, A., Seyam, M., & Abou-Elfetouh, A. (2021). Prediction of students' academic performance based on courses' grades using deep neural networks. *IEEE Access, 9*, 140731-140746. https://doi.org/10.1109/ACCESS.2021.3119596

Nachouki, M., & Abou Naaj, M. (2022). Predicting student performance to improve academic advising using the random forest algorithm. *International Journal of Distance Education Technologies, 20*(1), 1-17. https://doi.org/10.4018/IJDET.296702

Onyema, E. M., Almuzaini, K. K., Onu, F. U., Verma, D., Gregory, U. S., Puttaramaiah, M., & Afriyie, R. K. (2022). Prospects and challenges of using machine learning for academic forecasting. *Computational Intelligence and Neuroscience, 2022*(1), 5624475. https://doi.org/10.1155/2022/5624475

Ruma, J. F., Adnan, M. S. G., Dewan, A., & Rahman, R. M. (2023). Particle swarm optimization-based LSTM networks for water level forecasting: A case study on Bangladesh river network. *Results in Engineering, 17*, 100951. https://doi.org/10.1016/j.rineng.2023.100951

Salem, H., Mead, M., & El-Taweel, G. (2024). Particle swarm optimization-based hyperparameters tuning of machine learning models for big COVID-19 data analysis. *Journal of Computer and Communications, 12*(3), 160-183. https://doi.org/10.4236/jcc.2024.123010

Sandoval-Palis, I., Naranjo, D., Gilar-Corbi, R., & Pozo-Rico, T. (2020). Neural network model for predicting student failure in the academic leveling course of Escuela Politécnica Nacional. *Frontiers in Psychology, 11*, 515531. https://doi.org/10.3389/fpsyg.2020.515531

Scott, M., & Plested, J. (2019). GAN-SMOTE: A generative adversarial network approach to synthetic minority oversampling for one-hot encoded data. In *Proceedings of the 26th International Conference on Neural Information Processing (ICONIP 2019)* (Vol. 15, No. 2, pp. 29-35).

Shafiq, D. A., Marjani, M., Habeeb, R. A. A., & Asirvatham, D. (2022). Student retention using educational data mining and predictive analytics: A systematic literature review. *IEEE Access, 10*, 72480-72503. https://doi.org/10.1109/ACCESS.2022.3188767

Sharma, A., Singh, P. K., & Chandra, R. (2022). SMOTified-GAN for class imbalanced pattern classification problems. *IEEE Access, 10*, 30655-30665. https://doi.org/10.1109/ACCESS.2022.3158977

Venkatachalam, B., & Sivanraju, K. (2024). Enhanced student performance prediction using data augmentation with ensemble generative adversarial network. *Indian Journal of Science and Technology, 17*(44), 4619-4632. https://doi.org/10.17485/IJST/v17i44.2833

Wu, N., Zhang, L., Gao, Y., Zhang, M., Sun, X., & Feng, J. (2019). CLMS-Net: Dropout prediction in MOOCs with deep learning. In *Proceedings of the ACM Turing Celebration Conference-China* (pp. 1-6). Association for Computing Machinery. https://doi.org/10.1145/3321408.3322848

Yağcı, M. (2022). Educational data mining: Prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments, 9*(1), 11. https://doi.org/10.1186/s40561-022-00192-z

Yu, T., & Zhu, H. (2020). *Hyper-parameter optimization: A review of algorithms and applications*. arXiv. https://doi.org/10.48550/arXiv.2003.05689