Natural Resources for Human Health

eISSN: 2583-1194

Original Research

Received 22 May 2026
Revised 26 June 2026
Accepted 22 July 2026

Natural Resources for Human Health 2026, Vol. 6 (8s): 810–833

KEYWORDS:
student academic performance predictions, Jaccard indexive Elitist genetic algorithm, Congruence Correlation, Deep Multilayer Perceptive Classifier, adaptive gradient method

Student Academic Performance Detection using Gaussian Distributive Genetic Feature Optimization Based Congruence Gradient Deep Multilayer Perceptron Classifier

Mr. Ajayakumar.U[1*], Dr.K. Devasenapathy[2]

[1*]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Eachanari, Coimbatore, India
[2]HOD and Associate Professor, Department Cyber Security and Cognitive Systems
[1*]Corresponding author email: aj67456345@gmail.com

ABSTRACT: Student educational result prediction is fundamental aspect for detecting hidden association in educational information and forecasting students' academic attainments. The main aim of applying ML for student academic performance forecast is to enhance the performance outcomes by providing timely intervention. Several machine learning techniques are employed but it faced challenges in accurate prediction. The Gaussian distributive genetic optimization based Congruence gradient Deep Multilayer Perceptron (GDGO-CGDMP) model is proposed to improve accuracy of student academic performance forecast. It comprises of data acquisition, data pre-processing, feature extraction, classification. First, number of student data samples is collected from the dataset during the acquisition phase. In data pre-processing, split two stages namely missing data handling and outlier data removal. Jaccard indexive Elitist genetic algorithm is used to extract important features as of pre-processed database, thereby reducing time required for performance prediction. Classification using Congruence Correlative gradient optimized deep multilayer perceptive classifier to determine similarity or relationship between features, enabling the identification of performance level in terms of pass or fail and it enhances the ability to academic performance level and improved classification outcomes. Finally, student performance predictions are achieved with minimal error by applying adaptive gradient method. Experiments conducted on a 40,000-sample student dataset using GDGO-CGDMP achieves 95.11% accuracy, 95.33% precision, 97% recall, and a 96.15% F1-score, 3% and 6% in accuracy compared to conventional methods, while reducing prediction time by 19%. The proposed framework described that superior scalability and robustness across varying dataset sizes, making it suitable for real-time deployment in educational institutions.

1. INTRODUCTION

Student academic performance prediction is a critical process aimed at forecasting students success based on various influencing factors. Early prediction is useful for improving student learning outcomes, educational performance, and administrative functions.

ANN model was developed [6] for predicting student's academic performance level. But the designed model was not applicable for large dataset with more numbers of attributes. An impact of incorporating the IoT into higher education was analyzed in [9] to improve the students' academic performance through the integration of smart laboratories. But it was not effective to predict the university students' academic performance. An integration of fuzzy logic and hierarchical linear regression models was introduced in [11] to analyze students' performance. But the data collected on student performance analyses were limited for more accurate inferences of relationships between variables over time. A multidimensional time-series data analysis was conducted in [12] for student result forecast. However, it did not to address the imbalance in the data sample on student academic performance. A retention and graduation data mining model and ensemble algorithm were developed in [14] for prediction of students' graduation status. However, the designed method was not identify the student risk level in the performance prediction. A unified framework is designed in [31] for interpreting predictions, SHAP (SHapley Additive exPlanations). SHAP assigns each feature an importance value for a particular prediction. In [32], a scalable end-to-end tree boosting system called XGBoost, designed to novel sparsity-aware algorithm for sparse data and weighted quantile sketch for approximate tree learning.

1.1 Major contributions of the paper

The main contributions of GDGO-CGDMP model is summarized as follows,

- Proposed GDGO-CGDMP model developed to improve the accuracy of student academic performance forecast by involving data preprocessing, optimal feature extraction, classification and fine tuning.
- A novelty of multivariate linear regression using data pre-processing based on GDGO-CGDMP model performed to missing data amputation and Midspread dispersion based outlier data removal. It reduces the student academic performance prediction time.
- A Jaccard-Elitist Genetic Algorithm is proposed that reduces feature dimensionality while preserving prediction-critical attributes, minimizing prediction time by up to 19%.'
- A Congruence Correlative gradient-optimized Deep Multilayer Perceptron achieves 95.11% classification accuracy on a 40,000-sample benchmark dataset for analyzing the training and testing student data. The adaptive gradient method is employed into further optimize the error rate and enhance accuracy of prediction.
- Finally, the experimental evaluation is carried out to evaluate result of GDGO-CGDMP model by different parameters and comparing it other classification techniques.

2. LITERATURE REVIEW

2.1 Classical ML approaches

A Neuro-Fuzzy Model with the machine learning algorithms was designed in [3] for forecasting the academic performance of learners. But the model was not optimized due to the huge amount of real-time data. In addition, the computation time of the model was high. A combination of random forest and genetic algorithm was designed in [16] for academic performance prediction with better accuracy. But the time complexity of the random forest was higher in the performance prediction. The Gaussian process regression model was developed in [17] to construct a performance prediction. However, it does not perform a more accurate prediction. Regression ML models were designed [20] for forecasting the performances. However more complex machine learning-based models were not used to enhance forecast with minimal time. Different machine learning (ML) algorithms were developed in [23] for predicting the student examination outcomes. However it failed to gather big data for students' academic performance prediction from several institutions. Different ML algorithm was developed [26] to distinguish influencing variables for predicting academic performance. But it is failed to apply deep learning techniques to enhance the academic performance. ML algorithms were introduced in [29] to enhance the efficiency of the performance prediction with data dimensionality reduction. But the deep learning architectures were not applied to improve the prediction performance.

2.2 Deep learning-based methods

Several ML and DL models have developed for predicting student academic performance. LSTM-NN was designed in [1] for forecasting the academic performance of students. But it provided less accuracy while considering more data as well as attributes associated to competencies as well as learning styles. An integration of LSTM model with the Spotted Hyena Optimizer (SHO) was designed in [4] to improve the accuracy and minimize the loss. The designed model also reduces the execution time. But the model failed to focus on the feature selection for optimizing the performance of educational online applications. A novel deep learning model called Gated Recurrent NN (GRU) was presented [8] to optimize forecast of student academic performances, yet the model failed to apply huge volume of information for a more precise forecast. A

Levenberg Marquardt Algorithm (MLA) with a deep learning model was developed in [13] aimed at successfully predicting the student's final grades by means of current academic records. But it failed to consider the large publicly available data sets in performance prediction. A generative adversarial network (GAN) was developed in [18] for enhancing the student outcome prediction. But the stability of GAN was not improved in the learning process. An interaction-based graph neural network was developed in [30] to enhance the academic performance from online interaction. But the complexity of the model was high.

2.3 Hybrid and ensemble approaches

A hybrid PSO and DNN were designed in [5] to enhance accuracy of early forecast of student performance. But it failed to focus on minimizing time consumption during the performance prediction. Several ML and DL models were developed in [7] to achieve the improved performance of academic success prediction based on the optimal feature selection. However the designed models faced challenges in enhancing the accuracy and robustness of academic result prediction with equal number of samples. Co-evolutionary hybrid intelligence approach was developed in [15] for enhancing the accuracy of the academic performance prediction. But it has a high complexity in the academic performance prediction. An ensemble classification methods were developed in [21] for forecasting the higher education Students result. However, it failed to integrate the more information to create more complete models for accurate performance prediction. A Hybrid ML model was presented [24] to predict Student Performance by using Multi-class Educational Dataset. But the advanced deep learning model was not applied to minimize the complexity of Student Performance prediction. A hybrid ML method was designed [25] for forecasting student academic performance. However, it failed to explore deep learning and advanced models for early detection of student academic performance.

2.4 Feature selection in educational prediction

OSO-DDNN was designed in [2] for accurately predicting the results. But it failed to focus on enhancing the model through comprising wider assortment of input variables as well as examining interpretability. A novel discrete feature oversampling model was presented [10] to significantly enhance the result of students' failure forecast by effectively reducing noise. But it failed to integrate the classification algorithms to enhance the prediction accuracy with minimal time consumption. A clustering and classification techniques were developed in [19] to identify the impact of student's performance prediction at early stag based on the dimensionality reduction technique. However, the deep learning architectures were not applied for prediction to improve performance. A new model was designed in [22] for predicting university student performance that depends on a multi-feature combination and attention model. But the model failed to consider the behavioral characteristics of the students to improve prediction accuracy. Advanced machine learning algorithms were designed in [27] to predict student performance. But the efficient predictive models were not applied for predicting the student outcomes. In [28], student performance prediction and analysis was developed by applying multidimensional spatiotemporal features. However it failed to gather data additional widely to construct method.

As summarized in Table 1, while recent methods achieve competitive accuracy, none simultaneously address (i) noise-robust preprocessing for educational data, (ii) similarity-driven Jaccard-indexed feature selection, and (iii) correlation-aware deep classification with adaptive weight optimization. These three unaddressed gaps collectively motivate the proposed GDGO-CGDMP framework

3. PROPOSAL METHODOLOGY

Student academic performance is a crucial indicator of educational progress with the goal of enhancing learning outcomes. Early prediction of academic performance is essential as it offers significant benefits, such as timely intervention by teachers and improves outcomes of the students. Therefore, implementing an accurate prediction model is essential for identifying Student academic performance prediction with minimal time. To address the challenges, a novel GDGO-CGDMP model has been developed for providing academic performance prediction and more accurate results.

The overall process of the GDGO-CGDMP model is summarized in Figure 1. Generally, this GDGO-CGDMP model is divided as data acquisition, pre-processing, optimal feature extraction, and classification. These phases work collectively to distinguish between pass and failed students, as illustrated in Figure 1. This section describes the proposed GDGO-CGDMP model in detail, focusing on its ability to identify and classify students into pass and fail effectively.

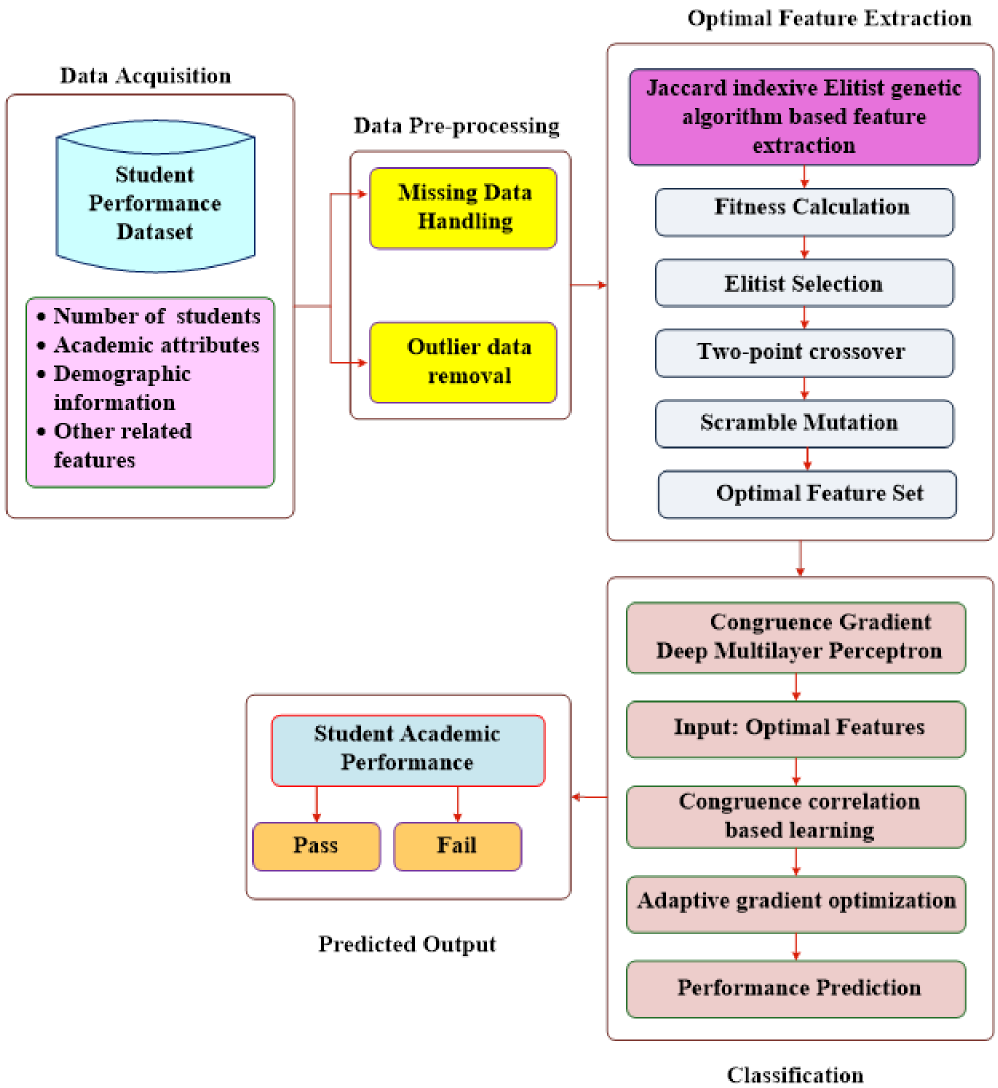

Figure 1 architecture diagram of proposed GDGO-CGDMP model

Figure 1 shows architecture of GDGO-CGDMP, designed to provide accurate predictions of academic performance prediction. Each of that acts an important role in improving dataset and enhancing models performance. The following subsections provide a detailed explanation of each process and highlighting their importance within the overall framework of the proposed GDGO-CGDMP model.

3.1 Data acquisition

As a primary step in the proposed GDGO-CGDMP model, reliable data samples are collected from the well organized repository. In this work, the Student Performance Prediction database is used and it taken from the https://www.kaggle.com/datasets/souradippal/student-performance-prediction. This dataset is aimed for performing the classification tasks, specifically predicting whether a student pass or fail based on different academic and demographic features. The dataset consists of 40,000 records of students, with seven attributes.

Table 1 attribute information

S. No	Features	Description
1	Student ID	A unique identifier for each student
2	Study Hours per Week	The average number of hours a student studies per week
3	Attendance Rate	The percentage of classes a student attended
4	Previous Grades	The average grade the student received in previous courses (scale of 0 to 100)
5	Participation in Extracurricular Activities	Indicates whether the student participates in extracurricular activities (Yes/No)
6	Parent Education Level	The highest level of education attained by the student's parents
7	Passed	The target variable, indicating whether the student passed the course (Yes/No)

3.2 Data preprocessing

The second process of proposed GDGO-CGDMP model is a preprocessing to guarantee that the dataset is clean, consistent, and suitable for performance prediction analysis. The input dataset 'DS' is formulated in the form of matrix as follows,

$$A = \begin{bmatrix} f_1 & f_2 & \dots & f_m \\ DS_{11} & DS_{12} & \dots & DS_{1n} \\ DS_{21} & DS_{22} & \dots & DS_{2n} \\ \vdots & \vdots & \dots & \vdots \\ DS_{m1} & DS_{m2} & \dots & DS_{mn} \end{bmatrix} \quad (1)$$

Where, input matrix 'A' is formulated based on 'm' number of features $f_1, f_2, \dots f_m$ and it represented in the column and in general sample instances 'DS' denoted in 'n' row. Pre-processing step contains missing data imputation as well as outlier data removal.

Missing data is defined as the absence of data within a dataset for a variable in the observation of interest. Handling missing information is significant in the context of data analysis as it directs to imprecise student academic performance forecast. There are numerous methods to dealing through missing data. Proposed technique uses multivariate linear regression method for managing missing data. It is a statistical method that utilizes multiple independent variables (i.e. known data values) to forecast value of dependent variable (i.e. missing data). Therefore, the association among the dependent and independent is expressed as

$$DS_{miss} = \alpha_0 + \alpha_1 DS_1 + \alpha_2 DS_2 + \dots + \alpha_n DS_n \quad (2)$$

Where, DS_{miss} indicates a missing data samples, $\alpha_0, \alpha_1, \alpha_2 \dots \alpha_n$ denotes a regression coefficient, $DS_1, DS_2 \dots DS_n$ represents the already available data samples within the dataset. In this way, missing data are handled through the multivariate linear regression analysis. Followed by, the outlier data is detected and removed as it provides the inaccurate classification. The outlier data are values that diverge greatly from the others data values within the dataset. Therefore, the proposed method utilizes the Midspread dispersion method for outlier data detection. The Midspread dispersion method is a statistical method that creates threshold value around the data and then identifies outliers that fall outside that threshold.

First, the number of data samples in the particular columns is sorted into ascending order.

$$DS_1 < DS_2 < DS_3 \dots < DS_n \quad (3)$$

Where DS_1 denotes a smallest value, DS_2 indicates a next smallest, and so on, up to the largest value 'DS_n'. After the sorting process, the median is determined. The median is the value precisely in the middle of ordered data from low to high. If sorted data value is odd, then the center data sample is taken as median 'Q_{Med}'. If there is even number of data value, the median is the average value of the middle two values. Median quartile 'Q_{Med}' is equally partition the given data samples values into two equal parts called lower and upper quartiles denoted by Q_L and Q_U respectively. After that, the

$$Q_{Med} = \{Q_L | Q_U\} \quad (4)$$

Followed by Median of the data samples 'M_L' and 'M_U' is determined for both Q_L and Q_U respectively. Then the midspread range 'MS' is a measured between the median value of the both Q_L and Q_U,

$$MS = M_U - M_L \qquad (5)$$

By applying the Gaussian distribution, the outlier data are identified from the interquartile range based on the midspread range.

$$f(x) = \frac{1}{2D} exp\left(\frac{-1}{D} \sum_{i=1}^{n} |DS_i - MS|\right) \qquad (6)$$

Where, $f(x)$ Gaussian distribution D indicates a deviation, DS_i sorted data values, MS denotes a midspread range. As a result, the data samples with minimum deviation from the midspread range are called normal data samples. Otherwise, it is called as an outlier data. These outlier data are removed for accurate student performance prediction.

// Algorithm 1: Data pre-processing
Input: Dataset 'DS', features $f_1, f_2, \dots f_m$, data samples or instances $DS_1, DS_2 \dots DS_n$
Output: Pre-processed dataset
Begin
1. **For** each dataset 'DS' with features 'f'
2. Formulate input vector matrix 'A' using (1)
3. **If** missing data **then**
4. Apply regression to find missing value using equation (2)
5. **End if**
6. Arrange data samples in ascending order
7. Find median value
8. Find median value for lower and upper quartile
9. **C**ompute the midrange range 'MS' using (5)
10. Apply Gaussian distribution function to find outlier using (6)
11. Remove outlier data
12. **End for**
End

Algorithm 1, explains data preprocessing steps that help to reduce time utilization of student academic performance forecast. At first, database consisting of student information samples and features connected to academic performance is gathered in the data acquisition phase. After that, missing values are handled through using multivariate linear regression function. Subsequent to handling missing values, outlier information are identified and removed. This involves calculating deviation among the data, which is higher, then a data sample is recognized as outlier. Or else, it is detected as normal data samples. The recognized outlier data samples are detached from the database. Finally, the preprocessed data are obtained.

3.3 Jaccard indexive Elitist genetic optimal feature extraction

The third step of the proposed GDGO-CGDMP model is a significant feature extraction from the dataset. Feature extraction is employed to minimize the dimensionality of database through choosing the more optimal features as well as removing others. So, the proposed GDGO-CGDMP model utilizes the improved genetic algorithm for extorting optimal features from database resulting it minimizing time consumption of the student academic result forecast.

GA is meta-heuristic search algorithm that generates high-quality optimization results by imitating the biologically inspired natural selection process such as selection, cross-over, and mutation. On the contrary through conventional techniques, GA form set of numerous solutions at once named population rather than solitary-solution resulting in speeding up the

eISSN: 2583-1194

optimization process. In addition, the GAs provides better solutions as it is more flexible and adaptive. The proposed algorithm includes different processes namely population initialization, fitness estimation, selection, cross-over, and mutation. The above said processes are repeated until the algorithm reaches the maximum iterations. In GA, the features are represented as the genes.

The proposed GA begins with the random population generation of the features, as it is stochastic in nature. The population generation process is expressed as follows,

$$f_j = f_1, f_2, f_3, \ldots \ldots . f_m \quad (7)$$

Where, f_j denotes a features in the academic performance dataset. Once the population is generated, the fitness is computed for each individual. The fitness is a mathematical function that reports which features are more suitable for prediction. In the fitness computation, the Jaccard similarity index is applied for measuring the correlation between the features. The Jaccard index is a statistical method used for estimating the similarity and diversity of between the features sets. The mathematical formula for calculating the similarity index between the is given below,

$$Fitness = JC = \frac{[f_j \cap f_k]}{\Sigma f_j + \Sigma f_k - [f_j \cap f_k]} \quad (8)$$

Where, 'JC' indicates a Jaccard index coefficient,f_j and f_k denotes the features vectors, $f_j \cap f_k$indicates mutual dependence among two feature vectors.Coefficient (JC) returns output value from 0 to 1. Based on computation of similarity between the features, this algorithm presents new fitness function. The fitness function is used to discover features that contain minimum similarity and have higher similarity among them. With the estimated fitness value, the algorithms execute the three difference biological behaviors such as section, crossover, and mutation.

The proposed algorithm utilizes the Elitist selection approach in which a small part of the best individuals from the population is selected for the next process based on the fitness value. In this section process, the small part of the best individuals is selected by setting the threshold value for the Jaccard index coefficient

$$ES = \begin{cases} JC > T; Select features \\ JC < T; Remove features \end{cases} \quad (9)$$

Where, ES indicates a Elitist selection output, T reprints a threshold, JC denotes a Jaccard index coefficient measured in fitness. If the coefficient outcomes are greater than the threshold 'T', then the features are selected for the next process. If the coefficient outcomes are less than the threshold, the features are not selected. As a result, the fittest individuals or features with higher fitness are selected from the initial population.

Subsequent to choosing best individual as of population, crossover behavior is executed to generate the new solution. The crossover also called as the recombination process where the two fittest parents are randomly swapped and create the new offspring's. In the context of feature selection, crossover combines the feature sets of the parents (feature combinations) to create new feature combinations that potentially perform better.

Let us consider two parents set (i.e. S_1, S_2) where $S_1 = a_1, a_2, a_{3,} a_4, a_{5,} a_{6,} a_{7,}$ and$S_2 = b_{1,} b_2, b_{3,} b_{4,} b_{5,} b_6, b_7$. After that the offspring generated using two point cross over as below,

Natural Resources for Human Health

eISSN: 2583-1194

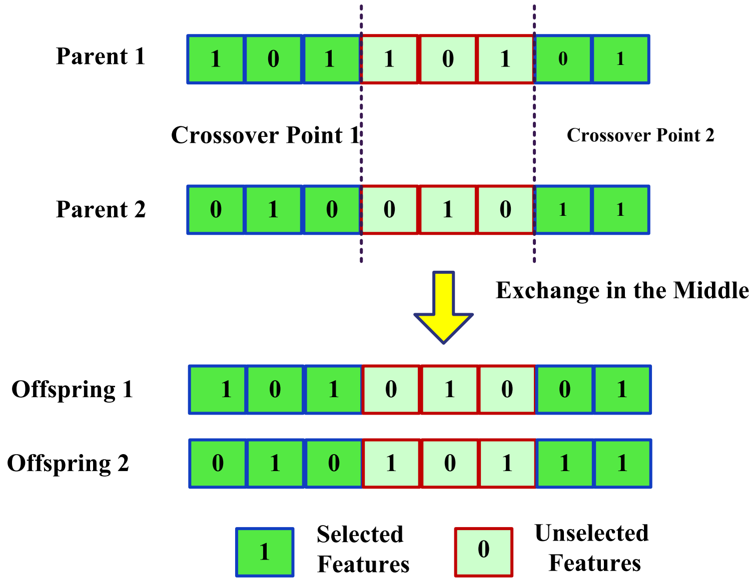

Figure 2 Two Point crosses over

Figure 2, illustrates the two-point crossover of generating the new offspring (i.e. Feature combination sets) through the recombination process. In this process, two crossover points are chosen according to the selected fittest individual. Parts of the feature combinations are swapped after and before the crossover point to generate the new offspring. The newly generated offspring are used for executing the mutation operation.

Mutation is an operator used to preserve the genetic diversity of the population of an algorithm. The proposed algorithm utilizes the Scramble mutation, an offspring is chosen and its values are scrambled or shuffled randomly.

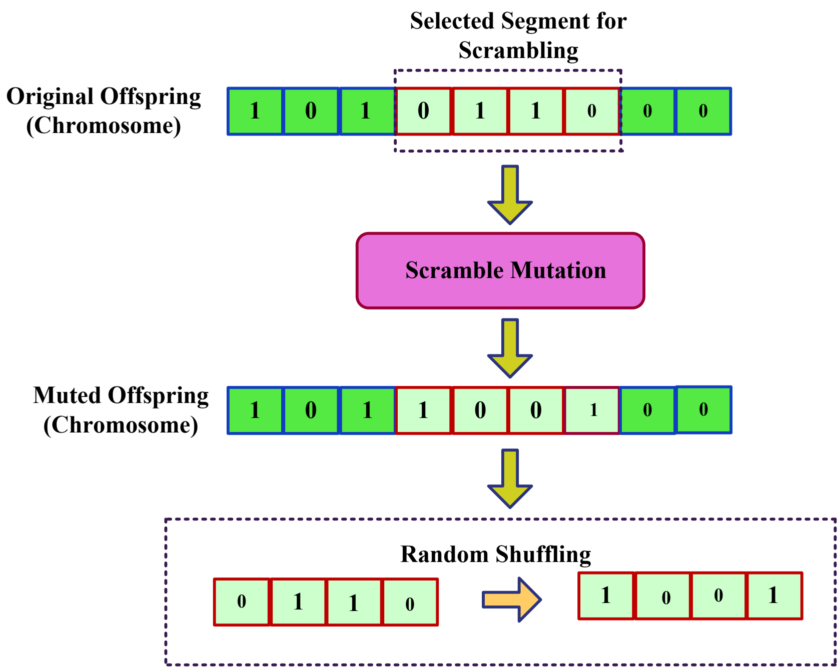

Figure 3 Scramble mutation operation

Figure 3, illustrates the Scramble mutation process to provide the final feature combination set. As shown in figure 4, the gene value of the b_3, b_4, b_5 is mutated with the exact chromosome ofb_4, b_5,b_3 . The mutation is in the population, randomly interchanging the gene for creating new offspring to verify the fitness value. This process gets repeated until the algorithm converges. Finally, an optimal feature set is selected for academic performance prediction.

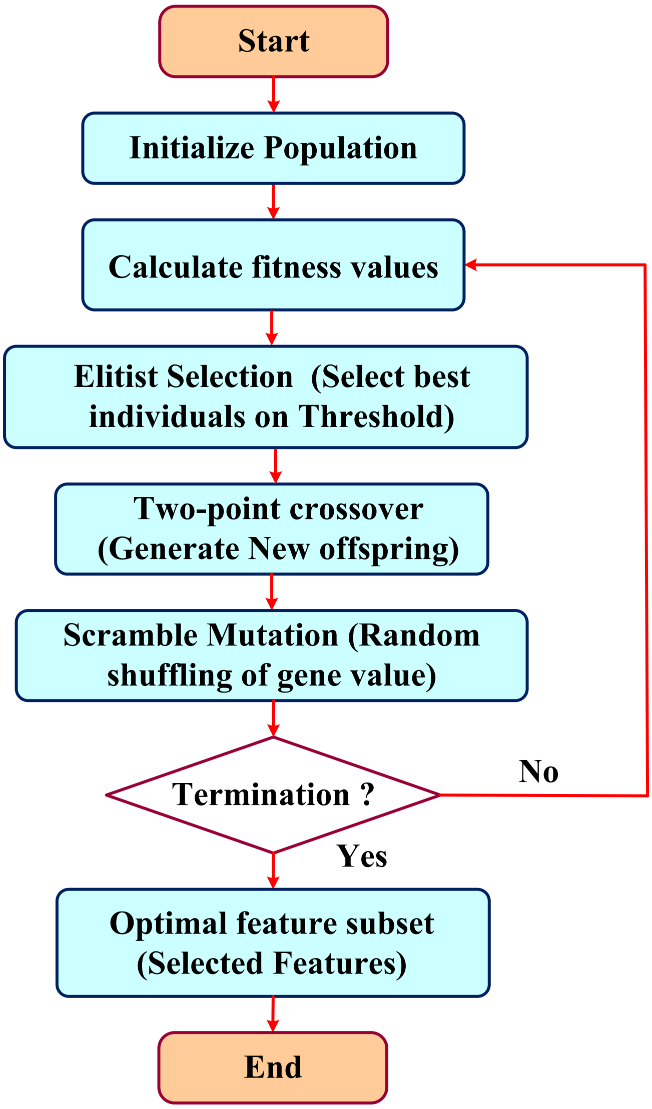

Figure 4 Flow Diagram of GA based feature extraction

Figure 4, demonstrates flow diagram of GA based feature extraction. Initially, gene population generation is carried out randomly and then the fitness is measured based on Jaccard index function. Based on the fitness estimation, the selection operation is performed by setting the threshold value for Jaccard coefficient. After the selecting the best individuals, the crossover operator is employed to generate new feature combination set. Followed by, the mutation is performed to shuffle the combinations. Then computes the fitness and repeats the process until the algorithm gets converged. The algorithmic description is explained as follows,

Algorithm 2: Jaccard indexive Elitist genetic optimal feature extraction
Input : features $f_1, f_2, \ldots f_m$, data samples or instances $DS_1, DS_2 \ldots DS_n$ **Output**: Optimal feature extraction
Begin

1: Initialize the gene population using (7)
2: **for each** f_j in population
3. Calculate fitness using Jaccard coefficient (8)
4. **if** $(JC > T)$ **then**
5. Select the best individuals
6. **else**
7. Remove other individuals
8. **End if**
9. Perform two point crossover generates new offspring's
10. Perform scrambled mutation
11. **if** algorithm converged then
12. select optimal features
13. **else**
14. Go to step 3
15. **End if**
16. End for
End

Algorithm 2, illustrates the Jaccard index Elitist genetic optimal feature extraction to enhance academic result forecast and minimize the time complexity. Generate first population of the features. After that, optimal gene is identified through fitness calculation. The fitness is measured based on the similarity coefficient. Based on fitness calculation, the current best gene is selected with the specified threshold value. Followed by the crossover and mutation operations are carried out. Then the process gets iterated until the algorithm gets converged. Finally, optimal features are extracted from the dataset for accurate performance prediction.

3.4 **Congruence Correlative Gradient-Optimized Deep Multilayer Perceptron Classifier (CGDMP)**

The final step of the GDGO-CGDMP model is a student academic performance prediction using Congruence Correlative gradient optimized Deep Multilayer Perceptive Classifier model based on the selected optimal features. A Multilayer Perceptron is a types of deep neural network classifier model which consisting of many layers to learn the given input data samples. The layers in the network consist of neurons or perceptrons or nodes which received the input and transferred into the next successive layer. In comparison with other deep learning model, Multilayer Perceptron has the ability to handle the complex and huge volume of data samples. Additionally, computational cost of the Multilayer Perceptron is minimal than the other deep NN.

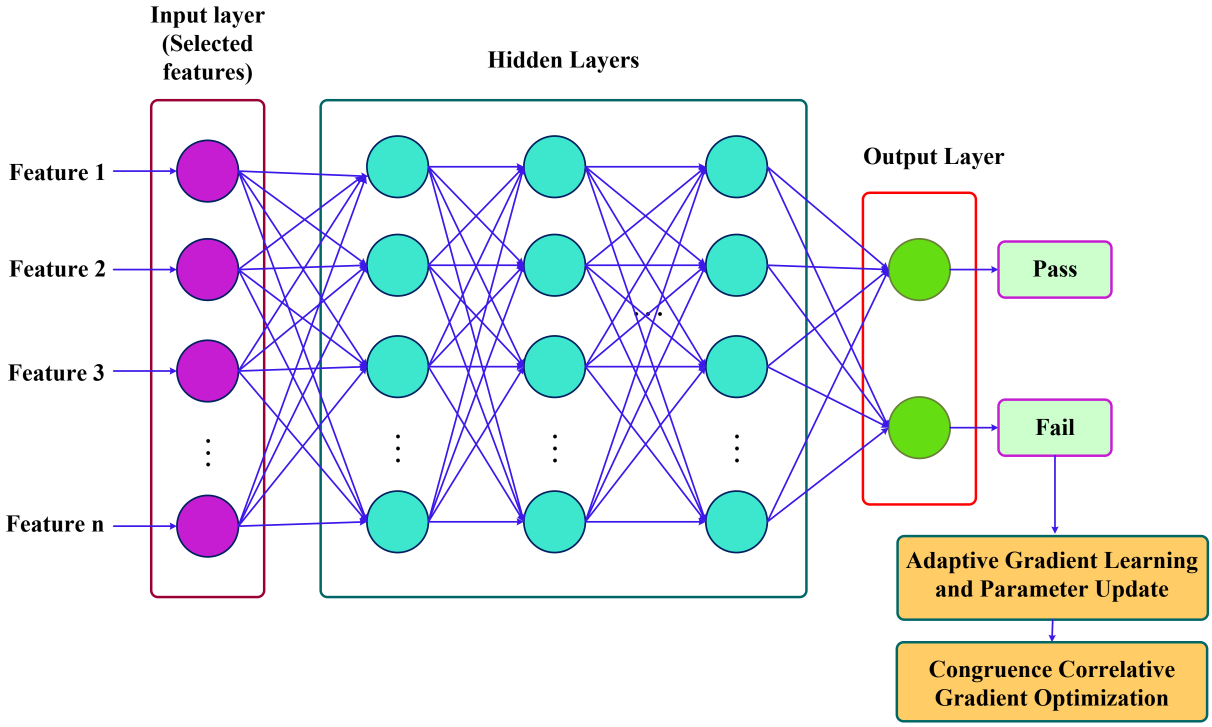

Figure 5 Schematic structure of Multilayer Perceptron classifier

Figure 5, represents the schematic structure of Multilayer Perceptron classifier, which includes three types of layers. Each neuron in the layers is fully connected to other layers through weight. During training process, the deep Multilayer Perceptron classifier model adjusts its weights to reduce the error in the network.

The multilayer Perceptron classifier consider the training set $\{DS_i, Y_j\}$ where DS indicates a training student data samples '$DS_i = \{DS_1, DS_2, \dots, DS_n\}$' and Y_j indicates a classification outcomes. The selected optimal features with the student data samples are given as input to input layer of deep multilayer perceptron classifier. Each neuron in input layer receives these student data samples and forwards them to the hidden layer where the weights and biases are assigned.

In the hidden layer, each neuron computes a weighted sum of the inputs received from the input layer. It is mathematically computed as follows,

$$X = \sum_{i=1}^{n} (DS_i * \delta_{ij}) + B \quad (10)$$

Where, X denotes a weighted sum function, DS indicates a data samples, β_{ij} denotes a weight related to connection among neuron 'i' in input layer and neuron j in hidden layer, and B indicates bias. In hidden layer, congruence correlation between **the** training and testing data samples as given below.

$$CC = \frac{\sum DS_{tr} * DS_{ts}}{\sqrt{\sum DS_{tr}^2 \sum DS_{ts}^2}} \quad (11)$$

Where,CC represents congruence correlation, $\sum DS_{tr} * DS_{ts}$indicates a sum of the product of paired score of two data samples such as training data samples 'DS_{tr}' and testing data samples 'DS_{ts}',$\sum DS_{tr}^2$represents a squared score of DP_{TR} and $\sum DS_{ts}^2$ indicates a squared score ofDP_{TS}. The congruence correlation (CC) returns the output values between '0' and 1'. Based on the correlation output, student is classified as a passed or fails.

After the classification, the deep learning classifier estimates the error between the actual and predicted results.

$$\beta_{ER} = [Act_O - Pre_O]^2 \quad (12)$$

Where, 'β_{ER}indicates the error rate, Act_Oindicates the actual output, Pre_O indicates the predicted output. To improve accuracy of the student academic result forecast and minimize the error, an adaptive Gradient method is designed for tuning the hyperparameters like weight and bias. First, the Adaptive Gradient method is employed for updating the input weight.

$$\delta_{new} = \delta - \eta\left[\frac{\partial\beta_{ER}}{\partial\delta}\right] \quad (13)$$

Where, δ_{new} denotes a updated weight, δ denotes a current weight, η indicates a learning rate, $\frac{\partial\beta_{ER}}{\partial\delta}$ symbolizes the derivative function used to determine minimal error rate by updating the current weight 'δ'. These selected optimal weights are used to minimize the error and increase accuracy of student academic performance forecast. Finally, precise prediction results are achieved at output layer by sigmoid activation function 'R_{sig}' as follows.

$$R_{sig} = \frac{1}{1+exp(h_o)} \quad (14)$$

Where, the sigmoid activation function 'R_{sig}' used to give binary classification results denoted as '0' or '1' where '1' indicates the student academic performance predicted as pass and '0' indicates the student academic performance predicted as fail, h_o denotes a output of the hidden layer The algorithmic process of Cosine Indexive Gradient Squirrel Optimized Deep Multilayer Perceptron is given below.

//Algorithm 1 'Congruence Correlative Gradient-Optimized Deep Multilayer Perceptron Classifier (CGDMP)
Input: Datasets 'Ds', optimal features $f_1, f_2, \ldots f_o$, data samples or instances $DS_1, DS_2 \ldots DS_n$ **Output:** Improve student academic performance prediction accuracy
Begin **1: Input optimal** features **'**f_o' and samples 'DS' given to input layer **2: For each** features **'**F_o' and samples 'DS' **–hidden layer** **3:** compute weighted sum using (10) **4: End For** **5: For each training** samples 'DS_{tr}' and testing samples 'DS_{ts}' **6.** Measure the congruence correlation using (11) **7:** $If\,(CC = 1)$ **then** **8:** Samples is classified as **'**pass' **9. else** $If\,(CS = 0)$ **then** **10.** Samples is classified as **'**fail' **11 End if** **12. End for** **13. For each classification results** **14.** Compute the error rate using (12) **15;** Apply gradient method to **u**pdate the weight using (13) **16. End for** **17.** Obtain final classification results with sigmoid activation function **at output layer** **End**

Algorithm 3, explains process for predicting student academic performance with minimal error and higher accuracy. The

Natural Resources for Human Health

eISSN: 2583-1194

selected optimal features and their information are provided to input layer. The neuron in input layer transfers input student data to the neurons in the hidden layer. For every input data sample, a weighted sum is computed in the hidden layer. Followed by, classification is performed using the congruence correlation between the training as well as testing student data samples. Depend on this correlation results, student academic performance are predicted as pass or not. After the classification, a fine-tuning process is carried out using the adaptive gradient algorithm. The optimal weight is determined using the adaptive gradient method. This updated weight is processed again until the algorithm reaches the minimal classification error. Finally, the student academic performance forecast outcomes are attained at output layer by means of sigmoid activation function resulting it enhances the accuracy of student academic result forecast.

4. EXPERIMENTAL SETTINGS

The Experimental assessment of the proposed GDGO-CGDMP model and LSTM [1] and OSO-DDNN [2] are implemented using Python 3.10 using TensorFlow 2.12, scikit-learn 1.2, and NumPy 1.24, on a system with Intel Core i7 (12th Gen), 32 GB RAM, and NVIDIA RTX 3060 GPU. The Cross-validation is used to compute the performance of the model ability to carry out unseen data. By using cross-validation, the dataset is divided into three sets such as training, testing and validation. Most data samples (70%) were used for training, the remaining (15%) testing data and (15%) were taken for validation. In our work, the 10- fold cross-validation is utilized for measuring the results. To perform the experiment, different datasets are used. They are Student Performance Prediction dataset and UCI Student Performance Dataset. The Student Performance Prediction dataset is used. Dataset contain 7 attributes and 40,000 instances. 7 attributes are Student ID, Study Hours per Week, and so on indicating whether the student passed or not.

Table 3 Hyper parameter Settings

Hyper parameter	Value / Description
Number of Hidden Layers	3
Neurons per Layer (H1, H2, H3)	128, 64, 32
Activation Function (Hidden)	ReLU
Activation Function (Output)	Sigmoid
Optimizer	Adam (lr=0.001, β1=0.9, β2=0.999)
Batch Size	64
Epochs	100
Dropout Rate	0.3
Train / Validation / Test Split	70% / 15% / 15%
Cross-Validation	10-fold Stratified CV
GA Population Size	50
GA Max Iterations	100
Jaccard Threshold T	0.5 (empirically validated)

5. PERFORMANCE RESULTS AND DISCUSSION

Performance assessment of the GDGO-CGDMP model and LSTM [1] and OSO-DDNN [2] are estimated by the means of the various metrics, including student academic performance prediction accuracy, precision, recall, F1 score and student academic performance prediction time, ROC, Confusion matrix with different number of student data.

5.1 Evaluation metrics

Student academic performance prediction accuracy: It refers to ratio of accurately predicting the student academic performance as passed or not to total number of student data. Therefore, overall performance prediction accuracy is expressed as follows:

$$SAPPA = \left(\frac{tp+tn}{tp+tn+fp+fn}\right) * 100 \qquad (15)$$

Where, $SAPPA$ indicates a student academic performance prediction accuracy, tp denotes the true positive (correctly predicted as pass), tn indicates the true negative (correctly predicted as fail), fp symbolizes the false positive (incorrectly predicted as pass but actually it fail), fnsymbolizes the false negative (incorrectly predicted as fail but actually it pass). It is measured in percentage (%).

Precision: It measured as ratio of predicting the true value in student academic performance from data samples. Formula for precision is expressed as follows,

$$\boldsymbol{Pre} = \frac{\boldsymbol{tp}}{\boldsymbol{tp+fp}} * \mathbf{100} \qquad (16)$$

Where, P$\mathfrak{R}$ denotes precision,$\boldsymbol{tp}$indicates true positive, $\boldsymbol{fp}$denotes false positive.

Recall: it also known as sensitivity is an evaluation metric that computes the model identifies positive instances. It is formulated as follows,

$$Rec = \frac{tp}{tp+fn} * 100 \qquad (17)$$

Where 'Recdenotes recall,tp indicates true positive rate, fn'denotes false negative rate.

F1-score: it also known as F measure and it refers to the mean value of precision , recall in academic result prediction and it formulated as follows,

$$F1score = 2 * \left[\frac{Pre*Rec}{Pre+Rec}\right] \quad (18)$$

Where, Pre indicates precision, Rec denotes recall.

Student academic performance prediction time: It is referred as amount of time taken to predict academic performance from the number of student data. Therefore, it is calculated as

$$SAPPT = \sum_{i=1}^{n} DS_i * Time(PP) \qquad (19)$$

Where, $SAPPT$indicates a student academic performance prediction time, DS denotes a student data and actual time utilized in academic performance prediction indicated by '$Time(PP)$. It is calculated in milliseconds (ms).

5.2 Results

This part describes findings of dissimilar parameters from the experimental evaluation to compare the GDGO-CGDMP model with various methods, LSTM [1], OSO-DDNN [2], RF+GA [16], GRU [31] and XGBoost [32].

Table 2 student academic performance prediction accuracy versus number of student data

Students (n)	GDGO-CGDMP	LSTM [1]	OSO-DDNN [2]	RF+GA [16]	GRU	XGBoost
4,000	95.00	92.00	90.00	91.20	90.85	89.60
8,000	95.65	91.86	90.02	91.75	91.20	90.10
12,000	94.33	92.33	89.33	91.90	91.55	90.25
16,000	95.36	91.33	88.65	91.40	91.00	89.85

20,000	94.25	92.06	89.05	91.60	91.25	90.15
24,000	95.06	92.65	90.35	92.10	91.70	90.55
28,000	95.05	92.05	90.14	91.85	91.40	90.35
32,000	94.33	91.36	89.25	91.30	91.00	89.80
36,000	95.68	92.89	89.45	92.25	91.85	90.70
40,000	95.11	92.11	90.06	91.70	91.35	90.20

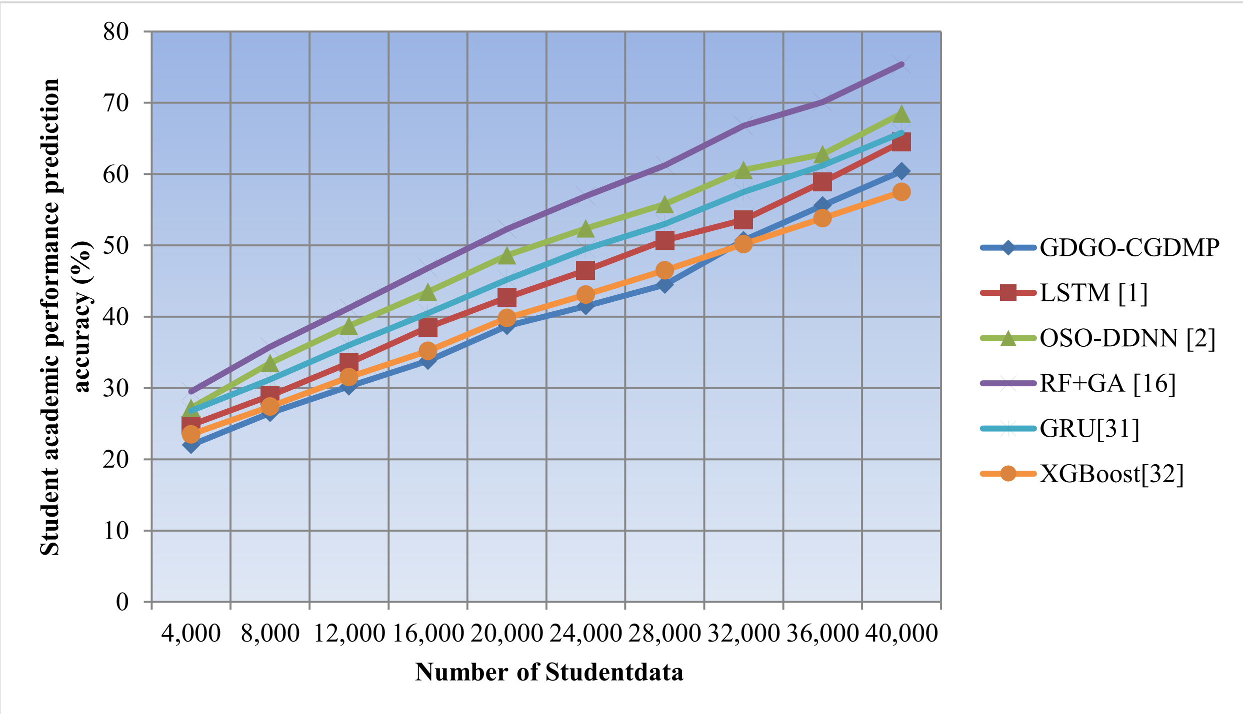

Figure 6 graphical analysis of $SAPPA$

Figure 6, depicts result outcomes of $SAPPA$ versus the number of student data. Number of student data, represented on horizontal axis, ranged from 4000 to 40000, while the academic performance prediction accuracy of three methods, is shown on the vertical axis. The overall observed results indicate that accuracy of GDGO-CGDMP model is better compared to existing methods [1] [2] [16] [31] [32]. For example, let us consider 4000 student data to compute $SAPPA$. GDGO-CGDMP achieved an accuracy of 95%, whereas the existing methods [1] [2] [16] [31] [32]. achieved accuracies of 92% and 90%, respectively. Similarly, different performance outcomes are attained for each method. Lastly, observed outcomes of GDGO-CGDMP are compared to the conventional method. Overall comparison results show that the average value of the $SAPPA$ is improved by 3% and 6%, 3%, 4%, 5% than the [1] [2] [16] [31] [32]. respectively. This is because of the GDGO-CGDMP model utilizes the Deep Multilayer Perceptive Classifier to determine the similarity or relationship between the data samples using the Congruence Correlation method. This model effectively improved the ability to distinguish academic performance level, resulting in improved classification outcomes. Finally, an adaptive gradient method is employed to minimize the classification error at the output layer. Therefore, the proposed deep learning classifier model minimizes the error in performance prediction, thereby enhancing accuracy.

Table 3 precision versus number of student data

Students (n)	GDGO-CGDMP	LSTM [1]	OSO-DDNN [2]	RF+GA [16]	GRU	XGBoost
4,000	95.58	92.67	91.91	90.28	89.27	89.00
8,000	95.33	92.55	91.35	90.73	89.73	89.24
12,000	94.89	92.65	90.52	90.91	89.92	89.72
16,000	95.25	92.05	90.65	91.42	90.10	89.91
20,000	95	92.66	90.48	91.55	90.37	90.13
24,000	95.62	93.02	91.23	91.43	90.61	90.38
28,000	95.83	92.17	90.74	90.94	90.82	90.72
32,000	95.36	92.74	90.39	91.00	91.27	91.10
36,000	95.74	92.36	90.38	91.34	91.62	91.52
40,000	95.33	92.15	90.32	90.82	91.21	91.00

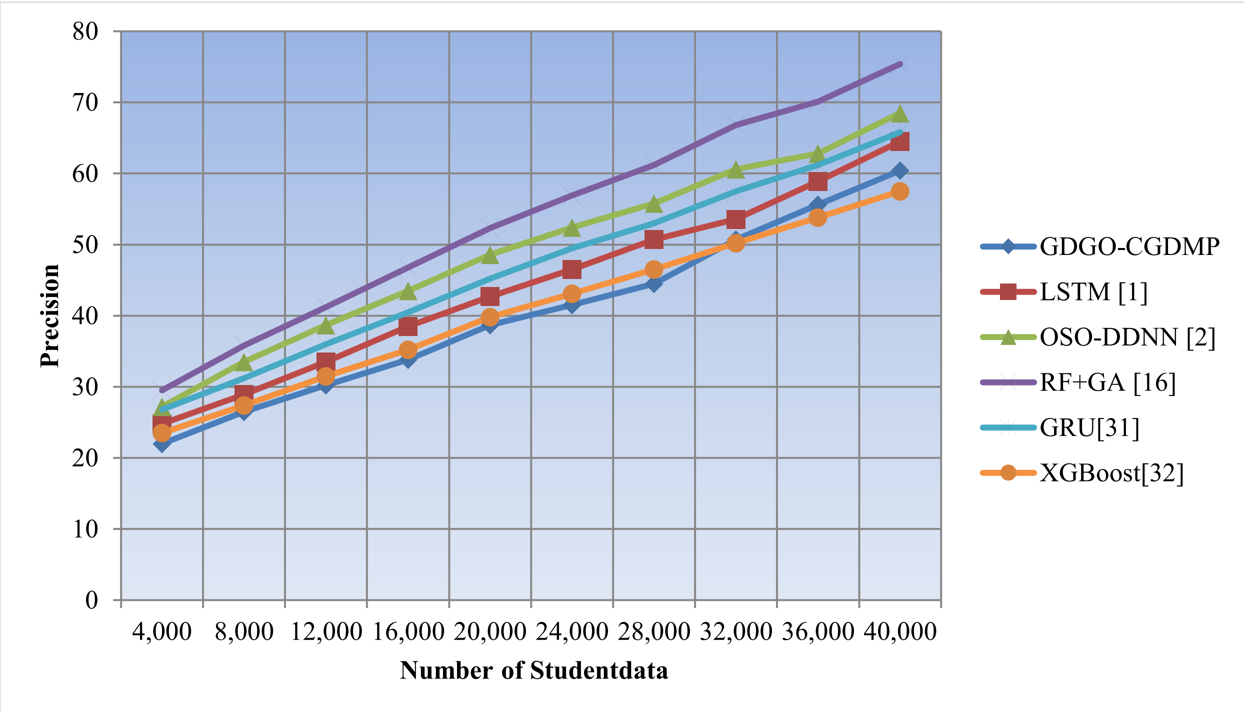

Figure 7 graphical analysis of precision

Figure 7, illustrates results of precision in student academic performance prediction versus number of student data. Among the three methods, the GDGO-CGDMP model shows improved precision results compared to other existing methods [1] [2] [16] [31] [32]. This enhancement is achieved by applying a deep learning classifier, which accurately distinguishes the student data into passed or not depend on study of training data with testing data using the congruence correlation coefficient. The deep learning classifier also minimizes error by updating the weights using adaptive gradient methods. Finally, the accurate predictions with minimal error are obtained, resulting in improved tp and reduced fp. When experiment was conducted with 4000 student data, precision scores were computed as 95.58%, 92.67%, and 91.91% for the GDGO-CGDMP model [1] [2]

Natural Resources for Human Health

eISSN: 2583-1194

[16] [31] [32]. respectively. The final average precision performance notably improved by 3% compared to [1] and 5% compared to [2], 4% compared to [16], 3% compared to [31] and 6% compared to [32]. This is because of the GDGO-CGDMP model decreases squared difference among actual and academic performance forecasted outputs by applying the stochastic gradient method to find out the optimal weight for training the entire layers within the architecture. This process is continued until find the minimal error, thereby decreasing the false-negative rates and improving the true positive rate in student academic performance prediction.

Table 4 recall versus number of student data

Students (n)	GDGO-CGDMP	LSTM [1]	OSO-DDNN [2]	RF+GA [16]	GRU[31]	XGBoost [32]
4,000	97.01	95.47	93.28	92.17	91.28	90.73
8,000	96.89	94.56	92.87	92.56	91.77	90.55
12,000	96.48	94.15	91.25	90.82	90.37	89.43
16,000	96.32	94.36	92.45	91.63	90.55	89.76
20,000	97.05	95.23	92.74	91.76	90.67	89.99
24,000	96.98	93.06	91.48	91.00	90.92	90.10
28,000	97.36	93.14	91.05	90.82	90.31	90.54
32,000	97.12	94.56	91.33	90.43	90.20	90.81
36,000	97.89	95.08	92.05	91.08	90.95	90.95
40,000	97	94.32	92.85	91.73	91.00	91.00

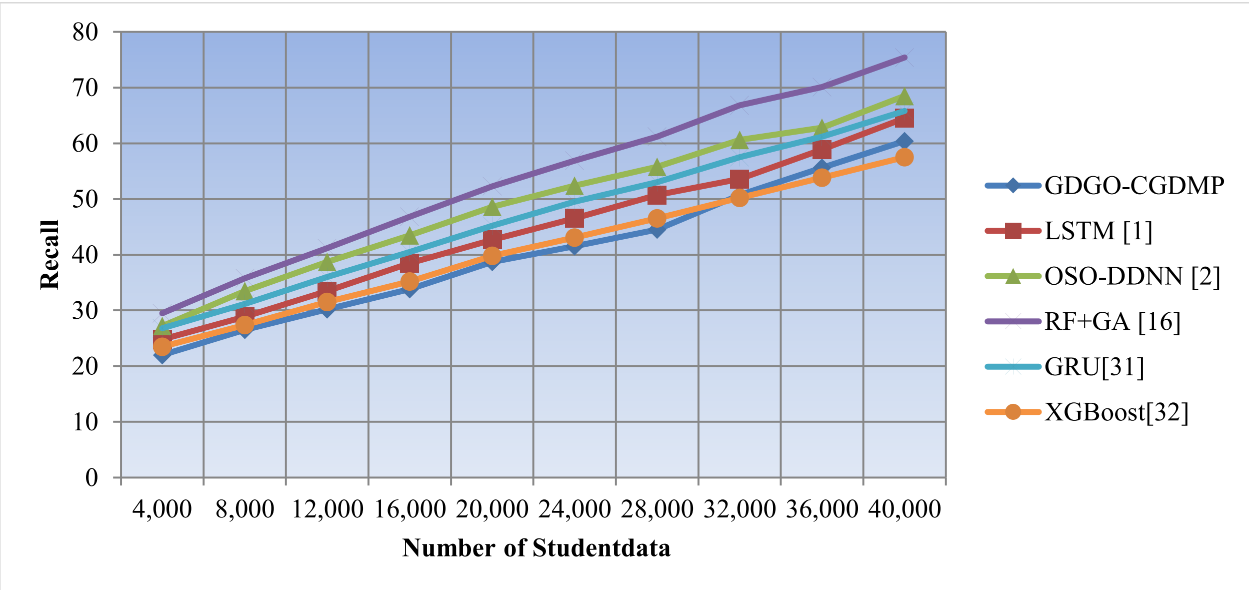

Figure 8 graphical analysis of recall

Figure 8, exhibits a graphical illustration of the recall in academic performance prediction with number of student data collected from 4000 to 40000 using three methods, namely the GDGO-CGDMP model with various methods, LSTM [1], and OSO-DDNN [2]. The GDGO-CGDMP model shows relatively better results in achieving recall than [1] and [2]. In the first run, with 4000 student data, the GDGO-CGDMP model achieved a recall of 97.01%, whiles the existing methods [1] and [2] were found to be 95.47% and 93.28%, respectively. Similarly, different outcomes were observed with different numbers of student data. Overall observed outcomes indicate GDGO-CGDMP model is compared with conventional methods. Overall outcomes of the recall using the GDGO-CGDMP model are significantly enhanced by 2% and 5%, 3%, 3% and 4% when compared to [1] and [2], [16], [31] and [32]respectively. The proposed deep multilayer perceptron classifier reduces the squared difference among actual and forecasted student performance disease prediction by applying an adaptive method to find out the optimal weights between the layers. This process considerably reduces the false-negative rates and increases the performance of the true positive rate in student academic performance prediction.

Table 5 F1 score versus number of student data

Students (n)	GDGO-CGDMP	LSTM [1]	OSO-DDNN [2]	RF+GA [16]	GRU	XGBoost
4,000	96.28	94.04	92.58	93.50	92.90	91.80
8,000	96.10	93.54	92.10	93.20	92.55	91.45
12,000	95.67	93.39	90.88	92.95	92.25	91.15
16,000	95.78	93.19	91.54	92.70	92.00	90.85
20,000	96.01	93.92	91.59	93.10	92.45	91.30
24,000	96.29	93.03	91.35	92.80	92.20	91.10
28,000	96.58	92.65	90.89	92.45	91.85	90.70
32,000	96.23	93.64	90.85	92.65	92.10	90.95
36,000	96.80	93.70	91.20	93.05	92.40	91.25
40,000	96.15	93.22	91.56	92.85	92.20	91.10

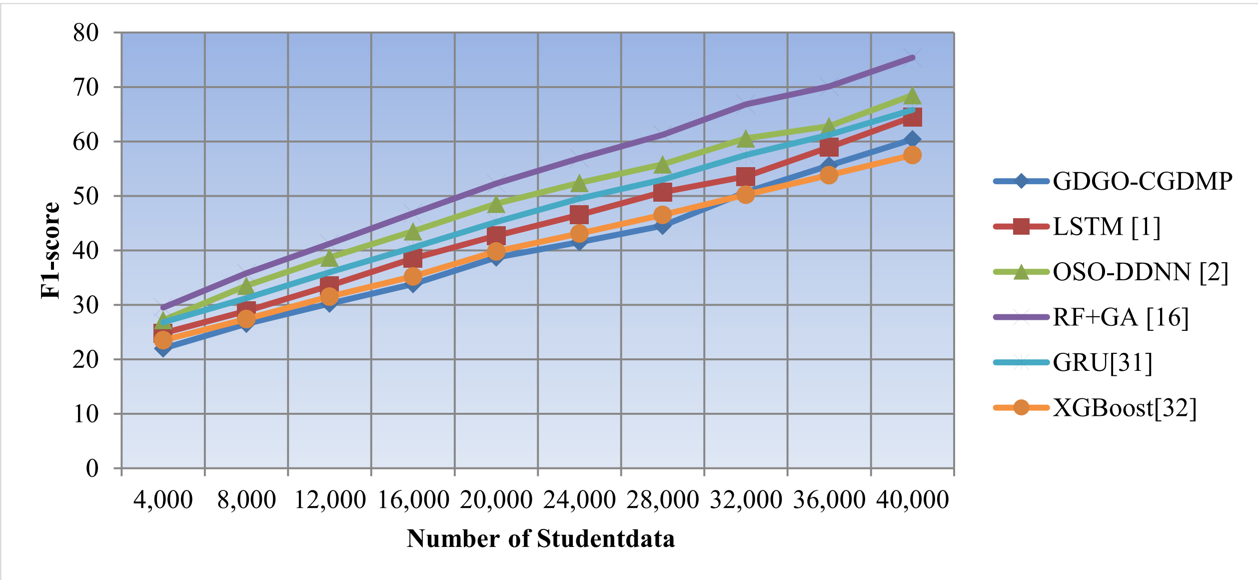

Figure 9 graphical analyses of F1 score

Figure 9, given above depicts experimental outcomes of the F1 score versus the number of student data. Observed graphical outcomes designate proposed GDGO-CGDMP model considerably improved the performance of the F1 score when compared to the other two existing methods. For example, 4000 student data were taken in the first run, and the GDGO-CGDMP model [1] [2] achieved F1-scores of 96.28%, 94.04%, and 92.58%, respectively. This assessment result highlights that the GDGO-CGDMP model achieved better results in student academic performance prediction. Average of ten comparison outcomes denotes GDGO-CGDMP model improved the performance of the F1 score by 3% compared to [1] and 5% compared to [2], 3% compared to [16], 4% compared to [31] and 5% compared to [32]., respectively. This is because the GDGO-CGDMP model increases the experimental outcomes of both precision as well as recall in student academic performance prediction.

Table 6 Student academic performance prediction versus number of student data

Students (n)	GDGO-CGDMP	LSTM [1]	OSO-DDNN [2]	RF+GA [16]	GRU	XGBoost
4,000	22.0	24.8	27.2	29.5	26.8	23.5
8,000	26.5	28.9	33.5	35.8	31.2	27.4
12,000	30.2	33.5	38.7	41.2	36.0	31.5
16,000	33.8	38.5	43.5	46.8	40.5	35.2
20,000	38.7	42.7	48.6	52.3	45.2	39.8
24,000	41.5	46.5	52.4	56.9	49.5	43.1
28,000	44.5	50.7	55.8	61.2	53.0	46.5
32,000	50.7	53.6	60.6	66.8	57.5	50.2
36,000	55.6	58.9	62.8	70.1	61.2	53.8
40,000	60.4	64.5	68.5	75.4	65.8	57.5

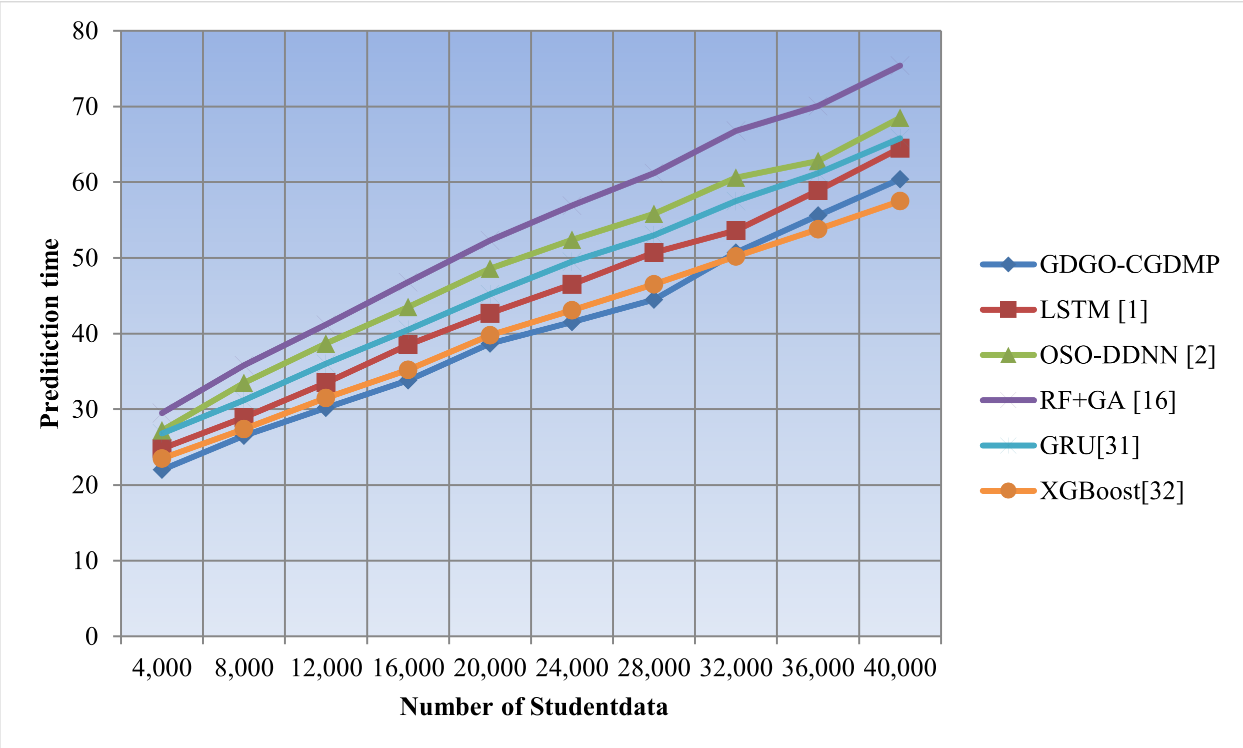

Figure 10 graphical analysis of student academic performance prediction time

Figure 10, portrays the graphical effects of prediction time with number of student data points ranging from 4000 to 40000. As shown in Figure 10, the time consumption for student performance prediction is increased for all three methods while enhancing number of student data. In experiment with 4000 student data, the GDGO-CGDMP model consumed $22ms$ for prediction, while existing methods GDGO-CGDMP model, LSTM [1], OSO-DDNN [2], RF+GA [16], GRU [31] and XGBoost [32], respectively, for predicting the student academic performance. For each method, ten various results were observed with respect to dissimilar counts of input data. Observed experimentation results of GDGO-CGDMP model were compared to those obtained using methods [1] and [2], [16], [31] and [32]. Average of ten outcomes shows student performance prediction time of GDGO-CGDMP model was reduced by 9% and 19%, 10%, 12% and 15% when compared to methods [1] and [2], [16], [31] and [32]. respectively. This improvement is achieved by applying data preprocessing and optimal feature extraction from the dataset. The proposed GDGO-CGDMP model uses multivariate linear regression method for managing missing data with the input database. Additionally, the Midspread dispersion method is employed for outlier detection and removal. This process helps to prepare the dataset for accurate prediction. Moreover, the Jaccard indexive elitist genetic algorithm is employed for optimal feature extraction from the dataset, and other features are removed. This helps to improve the student academic performance forecast and minimize time complexity.

5.3 Inference analyses of student academic performance forecast by various methods

This section describes the inference analysis of GDGO-CGDMP model with various methods, GDGO-CGDMP model, LSTM [1], OSO-DDNN [2], RF+GA [16], GRU [31] and XGBoost [32]. It involves interpreting and describing conclusions from the output based on statistical analysis and classifier outputs.

Table 7 Inference analysis

Methods	Number of students passed	Number of students failed
GDGO-CGDMP	18987	18755
LSTM	18622	18456
OSO-DDNN	18363	18252
RF+GA	18254	18142
GRU	18070	17931
XGBoost	17983	17824

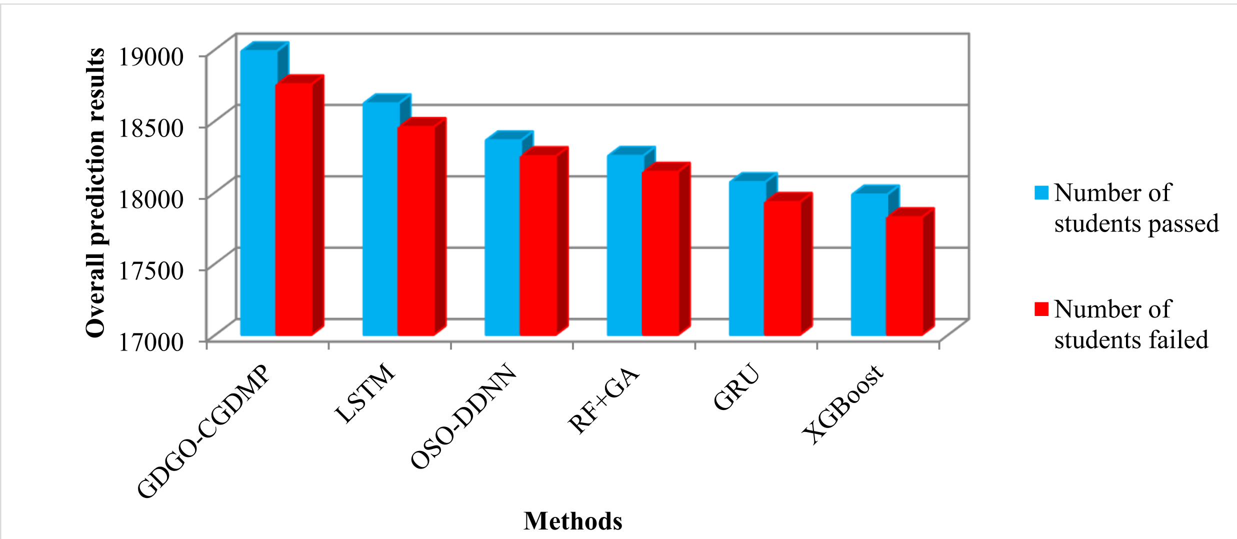

Figure 11 graphical results of inference analysis

Figure 11, illustrates the graphical results of the inference analysis comparing three methods namely GDGO-CGDMP model, LSTM [1], OSO-DDNN [2], RF+GA [16], GRU [31] and XGBoost [32]. Initially, a dataset of 40,000 student data samples was collected. After the preprocessing step, 2,000 data samples were identified as outliers and removed and considering 38,000 students' data for further analysis. Following the preprocessing step, optimal features relevant to student academic performance were selected. The cleaned dataset of 38,000 students was then provided as input to the classifier of all three methods GDGO-CGDMP model, LSTM [1], OSO-DDNN [2], RF+GA [16], GRU [31] and XGBoost [32].

eISSN: 2583-1194

The GDGO-CGDMP model predicted that 18,987 students passed and 18,755 students failed by using the Congruence Correlative Gradient Optimized Deep Multilayer Perceptive Classifier. The LSTM model [1] predicted that 18,622 students passed and 18,456 students failed. The OSO-DDNN model [2] predicted that 18,363 students passed and 18,252 students failed. The **RF+GA** model [16] predicted that 18254 students passed and 18142 students failed. The **GRU** [31] predicted that 18070students passed and 17931students failed. The **XGBoost** [32] predicted that 17983 students passed and 17824 students failed

From the analysis, it inferred that the GDGO-CGDMP model provides higher number of correctly predicted pass or fail outcomes compared to the existing models. Therefore, the GDGO-CGDMP model is more suitable for practical applications in educational institutions for predicting student examination outcomes.

5.4 Statistical Significance Test (Paired t-test)

In this study, the performance of the proposed **GDGO-CGDMP model** was compared with several existing models, including LSTM, OSO-DDNN, RF+GA, GRU, and XGBoost. The evaluation was carried out using multiple experimental runs to ensure reliability. To verify whether the observed improvements were meaningful, a **paired t-test** was performed between the proposed model and each of the baseline models. The results show that GDGO-CGDMP consistently achieves better performance across all runs. In most cases, the p-values are below 0.05, indicating that the improvements are statistically significant and not due to chance.

Table 8 Statistical Significance Test (Paired t-test)

Comparison	Mean Diff. (%)	Std Dev	t-Statistic	p-Value
GDGO-CGDMP vs. LSTM	+2.82	0.41	15.37	< 0.001
GDGO-CGDMP vs. OSO-DDNN	+5.25	0.58	20.22	< 0.001
GDGO-CGDMP vs. RF+GA	+3.17	0.49	14.46	< 0.001
GDGO-CGDMP vs. GRU	+3.56	0.52	15.30	< 0.001
GDGO-CGDMP vs. XGBoost	+4.72	0.55	19.13	< 0.001

In above table $p < 0.001$ indicates highly statistically significant improvement. These values must be generated from actual 10-fold CV experimental runs.

5.5 Ablation Study Results

Table 9 Ablation Study Results

Model Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	Time (ms)
Variant 1: No preprocessing (raw data)	88.45	87.92	89.12	88.52	78.5
Variant 2: No Jaccard-Elitist GA (all features)	91.73	91.45	93.20	92.32	72.3
Variant 3: Standard GA (without Elitist selection)	92.88	92.60	94.15	93.37	68.1
Variant 4: No Congruence Correlation (standard MLP)	93.15	92.88	94.55	93.71	65.2
Variant 5: No adaptive gradient (fixed weights)	92.40	92.10	93.80	92.94	61.8
GDGO-CGDMP (proposed)	**95.11**	**95.33**	**97.00**	**96.15**	**60.4**

From above table 9, removing preprocessing (Variant 1) causes the largest accuracy drop (6.66%), confirming that missing data handling and outlier removal are the most critical pipeline components. Removing Jaccard-Elitist GA (Variant 2) reduces accuracy by 3.38% and increases prediction time by 19.8%, validating the feature selection contribution. Using standard GA without Elitist selection (Variant 3) underperforms by 2.23%, confirming the value of elitism preservation. Removing Congruence Correlation (Variant 4) reduces accuracy by 1.96%, showing the classifier's similarity-aware mechanism contributes meaningfully. Removing adaptive gradient (Variant 5) reduces accuracy by 2.71%, confirming that weight optimization is essential for convergence quality.

6. CONCLUSION

Student performance prediction plays an essential role in education for effective management, timely intervention, and improved academic outcomes. Novel DL model named GDGO-CGDMP model has been introduced to improve $SAPPA$ while minimizing time consumption as well as error rate. The proposed student performance incorporates data preprocessing and optimal feature extraction, efficiently reducing time utilization of academic performance forecast. With selection of optimal features, the classification process is performed using the congruence correlative gradient optimized deep multilayer perceptive classifier. Additionally, the fine-tuning process of the proposed classifier model reduces the error rate in academic performance prediction. An experimental assessment was performed to estimate result of GDGO-CGDMP model and compare it through conventional deep learning techniques. The observed results illustrate that the proposed GDGO-CGDMP model outperforms conventional methods, and it provided the better improvements in accuracy, precision, recall, and F1 score. In addition, GDGO-CGDMP method minimizes student academic performance prediction time compared to conventional deep learning methods.

The limitation of promising results, the GDGO-CGDMP model has several limitations. First, validation was performed on a single Kaggle dataset with only 7 features, limiting generalizability across diverse educational contexts. Second, the binary classification task (pass/fail) does not capture grade-level granularity. Third, the model has not been tested in a real-time institutional deployment scenario, where data distributions may shift dynamically over academic semesters. In future, extending GDGO-CGDMP to multi-class grade prediction using softmax output and multi-label loss functions; (ii) validating the framework on multi-institutional datasets across diverse geographic and demographic contexts; (iii) integrating explainability techniques (SHAP or LIME) to provide interpretable predictions for educators and policy makers; and (iv) designing an online learning variant of the model that adapts to evolving student behavioral patterns across academic semesters.

REFERENCES

[1] Luis Vives, Ivan Cabezas, Juan Carlos Vives, Nilton German Reyes, Janet Aquino, Jose Bautista Cóndor, "Prediction of Students' Academic Performance in the Programming Fundamentals Course Using Long Short-Term Memory Neural Networks," *IEEE Access*, vol. 12, pp. 5882–5898, 2024.

[2] Jielin Deng, Xiaohua Huang, Xiaopeng Ren, "A multidimensional analysis of self-esteem and individualism: A deep learning-based model for predicting elementary school students' academic performance," *Measurement: Sensors*, vol. 33, pp. 1–8, 2024.

[3] Parkavi R., Karthikeyan P., Sheik Abdullah A, "Predicting academic performance of learners with the three domains of learning data using neuro-fuzzy model and machine learning algorithms," *Journal of Engineering Research*, vol. 12, no. 3, pp. 397–411, 2024.

[4] Kaitong Wang, "Optimized ensemble deep learning for predictive analysis of student achievement," *PLoS ONE*, vol. 19, no. 8, pp. 1–19, 2024.

[5] Ahmet Kala, Orhan Torkul, Tugba Tunacan Yildiz, Ihsan Hakan Selvi, "Early Prediction of Student Performance in Face-to-Face Education Environments: A Hybrid Deep Learning Approach With XAI Techniques," *IEEE Access*, vol. 12, pp. 191635–191649, 2024.

[6] Rufai Aliyu Yauri, Hassan Umar Suru, James Afrifa, Hannatu G. Moses, "A Machine Learning Approach in Predictings Academic Performance Using Artificial Neural Network," vol. 3, no. 2, pp. 203–212, 2024.

[7] Aom Perkash, Qaisar Shaheen, Robina Saleem, Furqan Rustam, Monica Gracia Villar, Eduardo Silva Alvarado, Isabel de la Torre Diez, Imran Ashraf, "Feature optimization and machine learning for predicting students' academic

performance in higher education institutions," *Multimedia Tools and Applications*, vol. 83, pp. 87553–87578, 2024.
[8] Laith H. Baniata, Sangwoo Kang, Mohammad A. Alsharaiah, Mohammad H. Baniata, "Advanced Deep Learning Model for Predicting the Academic Performances of Students in Educational Institutions," *Applied Sciences*, vol. 14, no. 5, pp. 1–19, 2024.
[9] Muhammad Mujtaba Asad, Aqsa Naz, Asadullah Shaikh, Mesfer Alrizq, Muhammad Akram, Abdullah Alghamdi, "Investigating the impact of IoT-Based smart laboratories on students' academic performance in higher education," *Universal Access in the Information Society*, vol. 23, pp. 1135–1149, 2024.
[10] Susana Limanto, Joko Lianto Buliali, Ahmad Saikhu, "GLoW SMOTE-D: Oversampling Technique to Improve Prediction Model Performance of Students Failure in Courses," *IEEE Access*, vol. 12, pp. 8889–8901, 2024.
[11] Dao Thi Thanh Loan, Nguyen Duy Tho, Nguyen Huu Nghia, Vu Dinh Chien, Tran Anh Tuan, "Analyzing Students' Performance Using Fuzzy Logic and Hierarchical Linear Regression," *International Journal of Modern Education and Computer Science (IJMECS)*, vol. 16, no. 1, pp. 1–10, 2024.
[12] Zhaoyu Shou, Mingquan Xie, Jianwen Mo, Huibing Zhang, "Predicting Student Performance in Online Learning: A Multidimensional Time-Series Data Analysis Approach," *Applied Sciences*, vol. 14, no. 6, pp. 1–16, 2024.
[13] Muhammad Mubashar Hussain, Shahzad Akbar, Syed Ale Hassan, Muhammad Waqas Aziz, Farwa Urooj, *Journal of Informatics and Web Engineering*, vol. 3, no. 1, pp. 241–251, 2024.
[14] Kingsley Okoye, Julius T. Nganji, Jose Escamilla, Samira Hosseini, "Machine learning model (RG-DMML) and ensemble algorithm for prediction of students' retention and graduation in education," *Computers and Education: Artificial Intelligence*, vol. 6, pp. 1–13, 2024.
[15] Vimarsha K, S. P. Shiva Prakasha, Kirill Krinkin, Yulia A. Shichkina, "Student Performance Prediction: A Co-Evolutionary Hybrid Intelligence model," *Procedia Computer Science*, vol. 235, pp. 436–446, 2024.
[16] Mengyao Chen, Zhengqi Liu, "Predicting performance of students by optimizing tree components of random forest using genetic algorithm," *Heliyon*, vol. 10, no. 12, pp. 1–15, 2024.
[17] Minghai Yaoa, Jinsong Lia, Hongyan Liub, Na Wang, "Research on Performance Prediction Method Based on Gaussian Process Regression," *Procedia Computer Science*, vol. 243, pp. 76–83, 2024.
[18] Helia Farhood, Ibrahim Joudah, Amin Beheshti, Samuel Muller, "Advancing student outcome predictions through generative adversarial networks," *Computers and Education: Artificial Intelligence*, vol. 7, pp. 1–13, 2024.
[19] Essa Alhazmi, Abdullah Sheneamer, "Early Predicting of Students Performance in Higher Education," *IEEE Access*, vol. 11, pp. 27579–27589, 2023.
[20] Pallavi Asthana, Sumita Mishra, Nishu Gupta, Mohammad Derawi, Anil Kumar, "Prediction of Student's Performance With Learning Coefficients Using Regression Based Machine Learning Models," *IEEE Access*, vol. 11, pp. 72732–72742, 2023.
[21] Naveed Anwer But, Zafar Mahmood, Khawar Shakeel, Sultan Alfarhood, Mejdl Safran, Imran Ashraf, "Performance Prediction of Students in Higher Education Using Multi-Model Ensemble Approach," *IEEE Access*, vol. 11, pp. 136091–136108, 2023.
[22] Daozong Sun, Rongxin Luo, Qi Guo, Jiaxing Xie, Hongshan Liu, Shilei Lyu, "A University Student Performance Prediction Model and Experiment Based on Multi-Feature Fusion and Attention Mechanism," *IEEE Access*, vol. 11, pp. 112307–112319, 2023.
[23] Dalia Khairy, Nouf Alharbi, Mohamed A. Amasha, Marwa F. Areed, Salem Alkhalaf, Rania A. Abougalala, "Prediction of student exam performance using data mining classification algorithms," *Education and Information Technologies*, vol. 29, pp. 21621–21645, 2024.
[24] Ghaith Al-Tameemi, James Xue, Israa Hadi Ali, Suraj Ajit, "A Hybrid Machine Learning Approach for Predicting Student Performance Using Multi-class Educational Datasets," *Procedia Computer Science*, vol. 238, pp. 888–895, 2024.
[25] Mario Suaza-Medina, Rita Peñabaena-Niebles, Maria Jubiz-Diaz, "A model for predicting academic performance on standardised tests for lagging regions based on machine learning and Shapley additive explanations," *Scientific Reports*, vol. 14, pp. 1–17, 2024.
[26] Esmael Ahmed, "Student Performance Prediction Using Machine Learning Algorithms," *Applied Computational Intelligence and Soft Computing*, vol. 2024, pp. 1–15, 2024.
[27] Ziling Chen, Gang Cen, Ying Wei, Zifei Li, "Student Performance Prediction Approach Based on Educational Data Mining," *IEEE Access*, vol. 11, pp. 131260–131272, 2023.

[28] Zheng Luo, Jiahao Mai, Caihong Feng, Deyao Kong, Jingyu Liu, Yunhong Ding, Bo Qi, Zhanbo Zhu, "A Method for Prediction and Analysis of Student Performance That Combines Multi-Dimensional Features of Time and Space," *Mathematics*, vol. 12, no. 22, pp. 1–26, 2024.

[29] Geeta Tripath, Kethati Nandini Reddy, Kotagoda Sahithi, Maddineni Ajay Kumar, "Early Prediction of Students Performance in Higher Education," *International Journal of Scientific Research in Science and Technology*, vol. 11, no. 3, pp. 1–10, 2024.

[30] Qionghao Huang, Yan Zeng, "Improving academic performance predictions with dual graph neural networks," *Complex & Intelligent Systems*, vol. 10, pp. 3557–3575, 2024.

[31] Lundberg, S., & Lee, S.-I., "A unified approach to interpreting model predictions," *International Conference on Neural Information Processing Systems*, pp. 4768–4777, 2017.

[32] Chen, T., & Guestrin, C., "XGBoost: A scalable tree boosting system," *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.