mathematics

MDPI

Article

LTBoost: A New High-Precision Method for Academic Early Warning and Prediction

Hailong Sun [1], Shenbing Fei [2], Mengdi Ma [1], Zhiqi Yan [1] and Wei Wang [1,*]

[1] Aeronautical Engineering Institute, Civil Aviation University of China, Tianjin 300300, China
[2] School of Transportation Science and Engineering, Beihang University, Beijing 102206, China
* Correspondence: wangwei@cauc.edu.cn

Abstract

Currently, the research on academic early warning assessment and prediction for college students under the credit system in colleges and universities is mainly based on methods such as machine learning. However, the existing prediction models often have problems such as difficulty in network structure, parameter selection, and extraction of context time series information. In response to these issues, this study, based on students' historical academic performance, proposes LTBoost, a novel framework of the XGBoost classification prediction model. The proposed model framework integrates the BiLSTM module to handle the time series information in the original data. It also integrates the proposed Enhanced Transformer module to perceive global information and obtain enhanced features. Through experiments, the LTBoost prediction model was compared with four other machine learning algorithms. The accuracy of the proposed LTBoost classification prediction model increased by 0.7% to 99.4%, demonstrating a good prediction effect on whether students are at risk of prolonging their studies. It provides a new paradigm and path for the construction of talent cultivation plans in credit-based universities.

Keywords: BiLSTM model; enhanced transformer model; SSA-XGBoost model; academic early warning; binary classification prediction

MSC: 00A35

1. Introduction

Many scholars have found that some universities are confronted with the problem that students cannot complete their studies within four years due to academic difficulties or subjective and objective factors [1]. The academic early warning mechanism is an effective measure adopted to address the possibility of students being demoted or expelled after the credit system reform due to weak course foundations, insufficient self-control and other reasons [2]. Therefore, against the background of the credit system in colleges and universities, how to help students prevent and overcome difficulties in the learning process and do a good job in academic early warning and assistance has become an urgent problem that colleges and universities need to solve.

At present, academic early warning work mainly falls into two aspects. One is institutional early warning, which is carried out based on certain triggering conditions, implementation processes, handling methods, etc. However, in actual implementation, there are problems such as the lack of significant pre-warning work, the difficulty in effectively integrating in-class and out-of-class collaboration, and the decision on the

Check for updates

Academic Editor: Junseok Kim

Received: 24 March 2026
Revised: 24 April 2026
Accepted: 27 April 2026
Published: 6 May 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

implementation of academic early warning is often made based on the results of course assessment. The second is academic performance prediction, which is the mainstream approach to studying students' academic early warnings. At present, academic prediction mainly employs techniques such as statistical modeling, data mining, and comprehensive algorithms [3–5] to conduct in-depth analysis of students' historical grades, in order to discover the existing patterns, connections, and development trends, and thereby make academic predictions. This type of method aims to identify possible situations in students' academic studies in advance through scientific data analysis means, providing a scientific basis for taking targeted support and intervention measures [6–9].

At present, the BP neural network model, SVM algorithm and other classification prediction models such as LGBM are mainly used to predict whether an academic warning will occur.

BP neural network is a kind of reverse transfer neural network and has good generalization ability. Yang proposed to use the GA-BP neural network model for prediction [10], who selected the grades of a course as the original dataset and optimized the parameters of the BP neural network with the GA genetic algorithm. Ji proposed to apply the SVM-RFE algorithm for academic early warning prediction [11]. The grades of five subjects were selected as the original dataset. SVM was used for prediction to extract the feature importance degrees. The least important features were deleted by the RFE algorithm, and the remaining features were predicted by SVM again. This process was repeated. In addition, Duong, Tran et al. also proposed using the LGBM model for academic early warning prediction [12]. The LBGM prediction model is an integrated model based on decision trees and enhanced ensemble techniques, which has a relatively fast running speed.

However, when training BP neural networks, it is prone to getting stuck in the predicament of local optimal solutions, and it takes a long time to train. The BP neural networks are difficult to select the network structure, and are highly sensitive to data preprocessing. In complex cases with multi-disciplinary features, the accuracy of this method may not be particularly high. The SVM algorithm is not suitable for situations with large data samples [13,14], and it is rather sensitive to the selection of parameters and kernel functions [15]. In the case where academic early warning is extended to all students in the future, the sample size will become very large, so the effect of the SVM algorithm model will become less than ideal. However, the LGBM model will have a relatively serious overfitting phenomenon in the case of complex data and high data noise [16].

To address the shortcomings of the above models, this paper proposes a novel framework for XGBoost prediction models, namely the LTBoost model. The proposed model has two prominent innovations. Firstly, its framework is novel. It is the first time that the combination of the BiLSTM model, the Enhanced Transformer model and the SSA-XGBoost model has been applied to an academic early warning system. Secondly, in the Enhanced Transformer, not only do there exist random deactivation and weight decay mechanisms, but a novel global feature extraction method has been set. The process of this model is as follows. Firstly, the data is preprocessed. According to the difficulty level of the course, the data is divided into two groups and, respectively, input into the BiLSTM model for processing. The BiLSTM can capture the local bidirectional temporal information in the data through the parallel structure of the forward LSTM and the backward LSTM. Compared with the traditional RNN model, the memory module of BiLSTM is more suitable for feature processing of long-term sequential problems, and its bidirectional parallel structure is able to perceive local and global information compared to the traditional LSTM model. Then, the obtained results are input into the proposed Enhanced Transformer model, which perceives global information within the entire dataset. The enhanced features are obtained through the output result of the model's encoding layer and combined with the original

data features to improve the prediction accuracy. The combination of these two models can effectively handle the short-term and long-term information of feature data. Finally, we optimized the parameter selection of XGBoost through SSA to complete the prediction task. The combination of SSA-XGBoost can quickly find the training parameter values with the smallest error compared to the single XGBoost model, significantly improving the prediction efficiency. Based on the above work, the main contributions of this paper are summarized as follows:

(1) By using the BiLSTM module to process two sets of data, respectively, it can effectively handle strong time series data and improve the accuracy of subsequent predictions.
(2) By adopting the proposed Enhanced Transformer model, the enhanced features are added on the basis of the original features to increase data information and improve the prediction accuracy.
(3) The prediction model adopting SSA-XGBoost can simultaneously take into account the historical influence of both long-term and short-term data, reduce the risk of overfitting, and improve the accuracy of classification prediction.

2. LTBoost Model and Algorithm

At present, in the traditional binary classification prediction model for academic early warning, there are mainly the following three problems:

(1) The temporal relationship between the data is strong, and the traditional architecture makes it difficult to simultaneously connect the full text information above and below, which affects the prediction accuracy.
(2) The traditional architecture is difficult to take into account, both the historical influence of long-term data and short-term data simultaneously, resulting in a decrease in the accuracy of classification prediction.
(3) Under conditions of large data capacity and complex data, the prediction effect of traditional models is poor, and the prediction effect of the models is sensitive to the selection of parameters.

Therefore, the LTBoost model we proposed, which integrates the BiLSTM model and the Enhanced Transformer model with the SSA-XGBoost prediction model. This proposed model can effectively alleviate the above predicaments and improve the accuracy of the final academic early warning classification prediction. Next, we will discuss the principles of the three main models in the proposed LTBoost model framework.

2.1. BiLSTM Module

Observing the original data, it can be found that the original data scores are sorted by the semester time. Selecting the professional course scores at the end of each semester, the difficulty of the major increases with the passage of time. The professional scores in the later period are related to the proficiency in the earlier period; that is, they are related to the professional scores obtained in the earlier period. The overall dataset shows a strong temporal relationship. The LSTM model has a good performance for such data processing. However, the traditional LSTM model only relates to the above information at runtime, which may lead to the loss of information capture and affect the accuracy of the final prediction. This paper adopts a bidirectional LSTM model for the preliminary processing of data. When processing the data, it can simultaneously connect with the information of the context to improve the accuracy of prediction.

LSTM introduces three main gating mechanisms, namely the forget gate, the input gate, and the output gate. Each gate consists of a Sigmoid neural network layer and a pointwise multiplication operation. The Sigmoid layer has a value range, which is used

to describe how much of each part can pass through. "0" represents that no quantity is allowed to pass through, and "1" represents that any quantity is allowed to pass through.

The forgetting gate is used to determine how many memory states there were at the previous moment, that is, how much information has been discarded.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

where W_f and b_f represent the weight and bias parameter of the forget gate, f_t is the output of the forget gate, x_t is the input at the current moment, and h_{t-1} is the hidden state at the previous moment, σ is the Sigmoid activation function.

The input gate is used to determine how much new information should be remembered at the current moment.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

where W_i and b_i represent the weight and bias parameter of the input gate, i_t is the output of the input gate, and $\widetilde{C}_t$ is the candidate memory state, W_C and b_C represent the weight and bias parameter of the candidate memory.

Update the memory state based on the results of the forgetting gate and the input gate.

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C}_t \tag{4}$$

where C_t represents the memory state at the current moment, C_{t-1} represents the memory state at the previous moment, and $\odot$ represents element-by-element multiplication.

The output gate controls how much information at the current moment should be output to the next moment.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

where W_o and b_o represent the weight and bias parameter of the output gate, o_t is the output of the output gate, and h_t is the hidden state at the current moment.

The LSTM algorithm flow is shown in the following Figure 1:

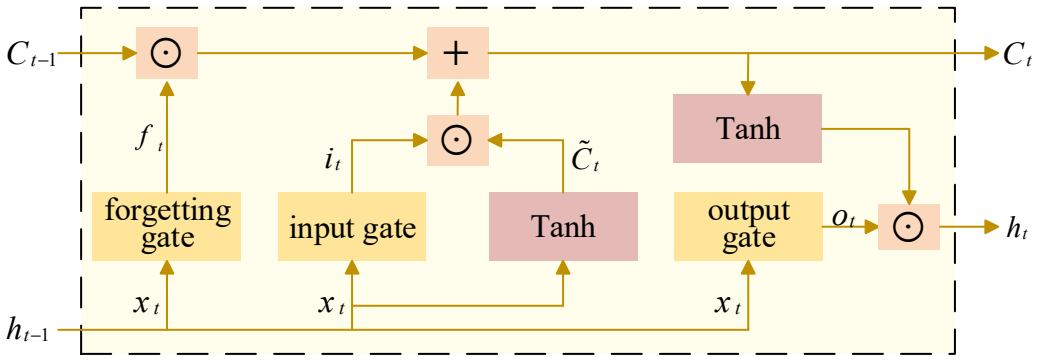

Figure 1. Flowchart of LSTM algorithm.

BiLSTM mainly consists of two independent LSTM layers running in parallel, shown as Figure 2. During operation, one processes data from front to back, while the other processes data from back to front. At the end of running the two LSTM layers, the outputs are merged in a concatenated manner [17].

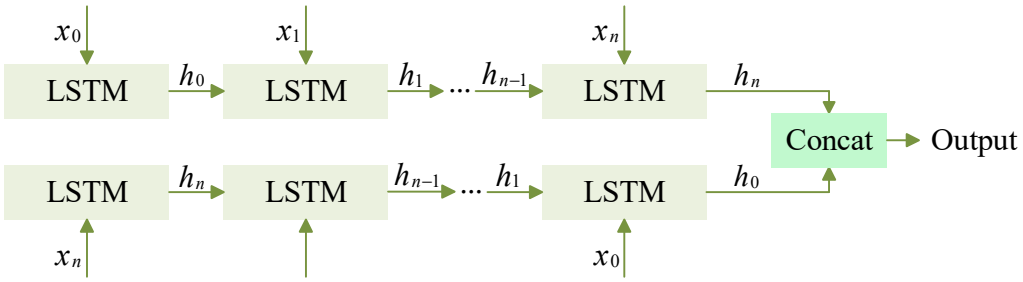

Figure 2. Flowchart of the BiLSTM algorithm.

In this paper, the original data is normalized first. The processing principle follows Equation (7), and the training set and test set are divided in a ratio of 7:3.

$$x_0'' = \frac{x'' - \min(x'')}{\max(x'') - \min(x'')} \tag{7}$$

Among them, x_0'' represents the normalized feature value, x'' represents the feature value before normalization, $\min(x'')$ represents the minimum value within the feature to which x'' belongs (i.e., the same column in the dataset), and $\max(x'')$ represents the maximum value within the feature to which x'' belongs.

The features are divided into two groups according to professional courses and basic courses, which are x_1 and x_2. Input these two sets of data into the BiLSTM model, respectively, and obtain two outputs with a feature number of 10. Combine them to obtain dataset x' with a feature number of 20. The temporal features extracted by BiLSTM are then fed into the Enhanced Transformer module described next.

2.2. The Proposed Enhanced Transformer Module

The proposed Enhanced Transformer model further conducts global perception of the context information, ultimately obtaining enhanced features, which are combined with the original features for prediction. This model could improve the accuracy of the results. The flow of the Enhanced Transformer model is shown in Figure 3.

This study characterizes temporal features through positional encoding. The position encoding expression is

$$\begin{aligned} PE_{pos,2u} &= \sin\left(pos/10000^{2u/d_{model}}\right) \\ PE_{pos,2u+1} &= \cos\left(pos/10000^{2u/d_{model}}\right) \end{aligned} \tag{8}$$

where PE is a matrix of position encoding $pos \in [0, \max_len)$, which represents a specific position. $\max_len$ is a sample length of the input data, $u \in [0, d_{model}/2)$, which represents a specific dimension, $2u$ is an even position in the vector, $2u + 1$ is an odd position in the vector, and the time series vectors obtained from the position encoding are merged with x' to get x''. d_{model} is the total dimensionality of the input sequence.

The Enhanced Transformer module utilizes the multi-head self-attention mechanism, which can avoid the attention from focusing too much on itself. Take x'' as input. Firstly, we need to construct the query matrix Q, the key matrix K and the value matrix V. The empirical formulas can be used to obtain the weight $\left\{W_e^Q, W_e^K, W_e^V\right\}$ of each matrix for each channel. $Q_e = x_2'W_e^Q$, $K_e = x_2'W_e^K$, $V_e = x_2'W_e^V$ of each channel are obtained according to the weights. e represents the labeling of the head. Then, the information matrix $head_e$ of each header is obtained by the softmax algorithm, that is, z_1, z_2, z_3, z_4 in Figure 3. They

are spliced together and multiplied with the output weight matrix to get the final output matrix O:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{9}$$

$$head_e = Attention\left(x_2'W_e^Q, x_2'W_e^K, x_2'W_e^V\right) \tag{10}$$

$$O = MultiHead\left(\left\{x_2'W_e^Q\right\}, \left\{x_2'W_e^K\right\}, \left\{x_2'W_e^V\right\}\right) = Concat(head_1, \ldots, head_h)W^O \tag{11}$$

$$W_e^Q \in \mathbb{R}^{d_{model} \times d_k}, W_e^K \in \mathbb{R}^{d_{model} \times d_k}, W_e^V \in \mathbb{R}^{d_{model} \times d_v}, W^O \in \mathbb{R}^{hd_v \times d_{model}} \tag{12}$$

where d_q, d_k, and d_v are the number of corresponding matrix columns, respectively. Define $d_q = d_k = d_v = d_{model}/h$, and W^O is the output weight matrix. Concat can splice the matrices together, and $h = 4$.

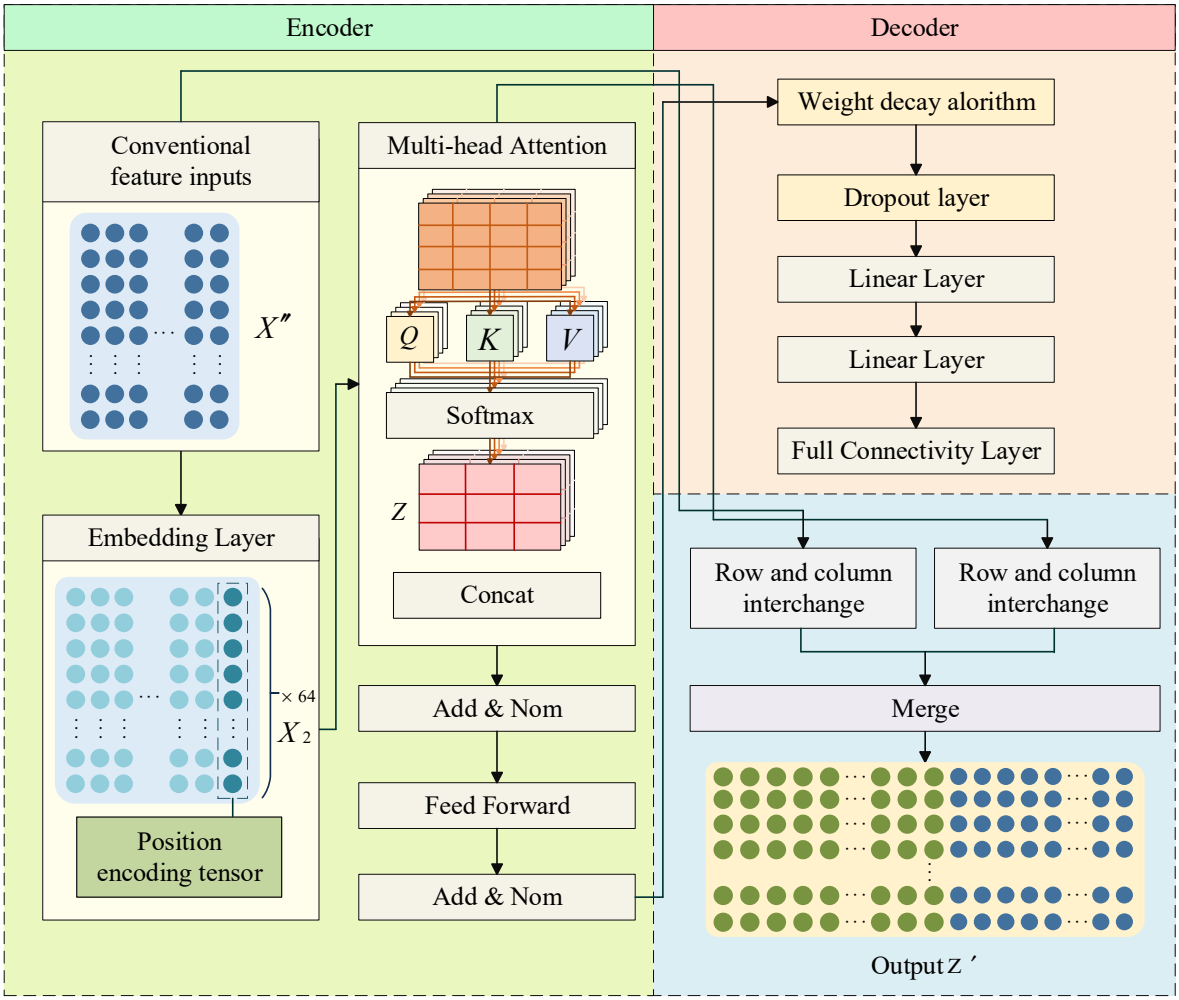

Figure 3. Flowchart of the Enhanced Transformer module.

Then, in order to reduce the computational processing capacity of the Transformer model, a weight reduction algorithm and a Dropout layer were added before the linear layer in the decoding part of the traditional Transformer model. Eventually, by enhancing the encoding results of the Transformer module, 16 enhanced features were obtained based on the context information of the entire dataset. Compared with the original data, these results have more concise feature information and are beneficial for reducing the processing complexity in subsequent predictions. The original input data x' and the output matrix O obtained by the Enhanced Transformer module are swapped with the row and column,

and then stacked horizontally to obtain z', which will be used as the input dataset for the subsequent SSA-XGBoost regression prediction.

In this study, the weight attenuation algorithm and Dropout layer are added to the decoding part of the Enhanced Transformer model, which can effectively reduce the complexity of the model, prevent overfitting of the model, and improve the generalization ability of the model.

2.3. The Proposed SSA-XGBoost Regression Prediction Module

The LTBoost model selects the SSA-XGBoost prediction module to conduct academic warning predictions on the processed student performance data. The XGBoost model is an optimized distributed gradient boosting model based on decision trees. Compared with traditional prediction models, it is more flexible and efficient. The regularization technique can control the complexity of the model and is more suitable for handling nonlinear systems. Furthermore, the XGBoost model can effectively handle high-dimensional time series features, and its predictions have good generalization and stability, with high prediction accuracy. The SSA algorithm can enhance the efficiency and stability when searching for the core parameters of the model, improving the prediction accuracy. The XGBoost model used in this paper for prediction has the significance of parameter selection optimization.

This paper adopts the SSA algorithm to optimize the parameter selection of the XGBoost model, the flowchart shown as Figure 4. The SSA algorithm can enhance the efficiency and stability in finding the core parameters of the XGBoost model and improve the prediction accuracy.

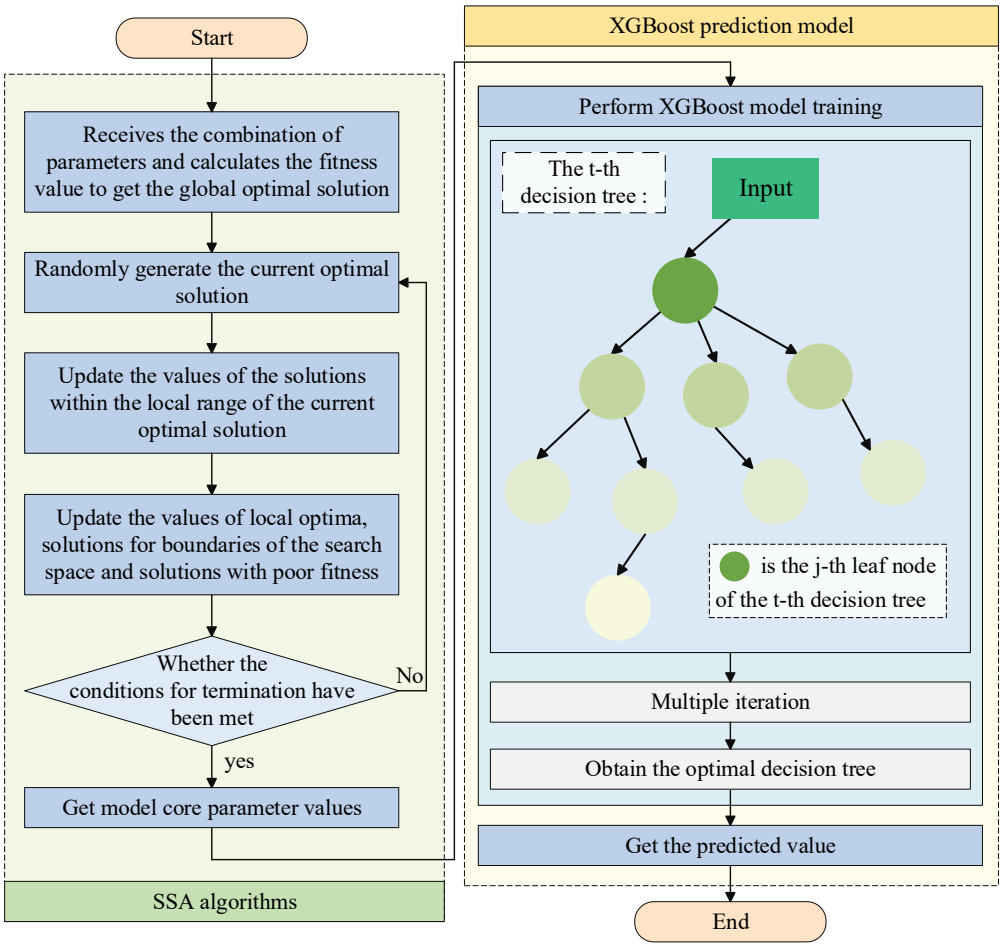

Figure 4. Flowchart of the SSA-XGBoost module.

First, initialize the parameters of the XGBoost model (the learning_rate α, the number of weak evaluators ϑ, the proportion of samples taken during random sampling δ, the maximum depth of the tree ℓ and the value of the random sampling feature ratio ξ). Each piece of data in the parameter input matrix X is denoted as $x_{u,v}$, where u and v represents the v-th solution of the u-th parameter. There are a total of U parameters and V solutions for each parameter.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1v} & \dots & x_{1V} \\ x_{21} & x_{22} & \dots & x_{2v} & \dots & x_{2V} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{u1} & x_{u2} & \dots & x_{uv} & \dots & x_{uV} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{U1} & x_{U2} & \dots & x_{Uv} & \dots & x_{UV} \end{bmatrix}$$

Its solution is updated after initialization:

$$x_{u,v} = low_v + rand \times (up_v - low_v), v = 1, 2, \dots, V \tag{13}$$

where $x_{u,v}$ is the each data point in the input matrix X, up_v is the upper position boundary. low_v is the lower position boundary, and $rand$ takes the value between 0 and 1.

Randomly generate the current optimal solution and update the value of the current optimal solution according to Equation (14).

$$X_{u,v}^{t+1} = \begin{cases} X_{u,v}^{t} \cdot \exp\left(-\dfrac{u}{\alpha \cdot iter_{\max}}\right), & R_2 < ST \\ X_{u,v}^{t} + Q \cdot L, & R_2 \geq ST \end{cases} \tag{14}$$

where X_{uv}^{t} is the v-th solution of the u-th parameter in the current iteration, t is the number of iteration steps, $iter_{\max}$ is the maximum number of iterations, $\alpha \in (0, 1]$, which is a random number, $R_2 \in [0, 1]$, which is the warning value, $ST \in [0.5, 1]$, which is the safety value, Q is a random number that follows a normal distribution, L is the matrix of $1 \times V$, in which each element is 1.

Update the values of the solutions within the local range of the current optimal solution:

$$X_{u,v}^{t+1} = \begin{cases} Q \cdot \exp\left(-\dfrac{X_{worst} - X_{u,v}^{t}}{u^2}\right), & u > \dfrac{U}{2} \\ X_{p}^{t+1} + \left|X_{u,v}^{t} - X_{p}^{t+1}\right| \cdot A^{+} \cdot L, & \text{otherwise} \end{cases} \tag{15}$$

$$A^{+} = A^{T}\left(AA^{T}\right)^{-1} \tag{16}$$

where X_p is the current optimal solution, X_{worst} is the current global worst solution, and A is a matrix of $1 \times V$, in which each element is randomly assigned a value of 1 or -1.

Updating the values of local optimal solutions, solutions at the boundaries of the search space, and solutions with very poor fitness according to Equation (17) can effectively avoid errors in local optimal solutions.

$$X_{u,v}^{t+1} = \begin{cases} X_{best}^{t} + \beta \cdot \left|X_{u,v}^{t} - X_{best}^{t}\right|, & f_u > f_g \\ X_{u,v}^{t} + K \cdot \left(\dfrac{\left|X_{u,v}^{t} - X_{worst}^{t}\right|}{(f_u - f_g) + \varepsilon}\right), & f_u = f_g \end{cases} \tag{17}$$

where X_{best} is the current global optimal solution, β is the step size control parameter, which is a random number that follows the standard normal distribution, $K \in [-1, 1]$, which is a

random number, f_u is the fitness value of X_{uv}^t, f_g is the fitness value of the current global optimal solution, f_w is the fitness value of the current global worst solution, and ε is the minimum constant, avoiding the situation where the denominator is 0.

The optimal solution of the XGBoost model parameters is obtained through the SSA algorithm and then brought into the XGBoost model for prediction. Define the model as a t decision tree and the model's predicted value at the c-th tree is

$$\hat{y}_s = \sum_{c=1}^{t} f_c(z_s') \tag{18}$$

where s is the s-th sample, n is the total number of samples, $s \in n$, and f_c is the decision function of the c-th tree.

The objective function of the t-th decision tree can be denoted as

$$L(\Phi) = \sum_s l\left(y_s, \hat{y}_s^{(t-1)} + f_t(z_s')\right) + \sum_c \Omega(f_c) \tag{19}$$

where l is the loss function. $\Omega(f_c)$ is the regularity term, which represents the model complexity of the c-th tree. y_s is the predicted value of the s-th sample z_s' and $\hat{y}_s^{(t-1)}$ represents the predicted value of the previous $t-1$ decision trees for the sample s.

Then, we performed a second-order Taylor expansion of the objective function and remove the constant term.

$$L^{(t)} = \sum_{s=1} \left[l\left(y_s, \hat{y}_s^{(t-1)}\right) + g_s f_t(z_s') + \frac{1}{2} h_s f_t^2(z_s') \right] + \sum_c \Omega(f_c) \tag{20}$$

where $g_s f_t(z_s')$ is the first-order derivative of the residual. $\frac{1}{2} h_s f_t^2(z_s')$ is the second-order derivative of the residual. We regularized expansion and removed the constant term to obtain:

$$L^{(t)} = \sum_{s=1}^{n} \left[g_s f_t(z_s') + \frac{1}{2} h_s f_t^2(z_s') \right] + \Omega(f_c) \tag{21}$$

where we define the regular term $\Omega(f_c)$ as Equation (22):

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{22}$$

Combine primary term coefficients and secondary term coefficients:

$$L^{(t)} = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2 \right] + \gamma T \tag{23}$$

$$G_j = \sum_{r \in I_j} g_r \quad H_j = \sum_{r \in I_j} h_r \tag{24}$$

where T is the total number of leaf nodes. I_j represents the sample of the j-th leaf node. G_j is the sum of first-order partial derivatives contained in leaf node j. H_j is the sum of second-order partial derivatives contained in leaf node j. w_j is the score of leaf node j. The optimal solution for leaf node w_j^* is obtained by derivation of Equation (19).

$$w_j^* = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \tag{25}$$

The objective function is obtained by bringing w_j^* into Equation (21).

$$L^{(t)} = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2 \right] + \gamma T \tag{26}$$

The objective function obtained through optimization and iteration can be used to optimally divide the decision tree. The tree that has been divided each time is taken as the next tree to be optimally divided. Finally, the leaf node scores of each decision tree are summed up to the final prediction value y_{pre}.

2.4. Algorithm Pseudocode and Complexity Analysis

To facilitate reproducibility, Algorithm 1 presents the complete training procedure of LTBoost.

Algorithm 1: LTBoost Training Procedure

Input: Historical course score matrix X, labels y
Output: Trained LTBoost model

1. Normalize X to [0, 1] range
2. Split features into two groups: X_gen (general courses) and X_spec (specialized courses)
3. Split data into training (70%) and test (30%) sets temporally
4. // BiLSTM stage
5. h_gen ← BiLSTM(X_gen) // forward + backward LSTM
6. h_spec ← BiLSTM(X_spec)
7. h_temporal ← Concat(h_gen, h_spec)
8. // Enhanced Transformer stage
9. h_global ← EnhancedTransformer(h_temporal) // multi-head self-attention
10. X_aug ← Concat(h_temporal, h_global) // augmented feature set
11. // SSA-XGBoost stage
12. Initialize XGBoost parameters (lr, n_estimators, subsample, max_depth, colsample)
13. θ* ← SSA_optimize(XGBoost, X_aug, y_train) // sparrow search
14. model ← train_XGBoost(X_aug, y_train, θ*)
15. Return model

Computational Complexity Analysis:

The overall time complexity of LTBoost is dominated by three modules:

BiLSTM: $O(T \cdot L \cdot H^2)$, where T is sequence length, L is number of layers, H is hidden dimension.

Enhanced Transformer: $O(T^2 \cdot d_{model})$ for self-attention, plus $O\left(T \cdot {d_{model}}^2\right)$ for feed-forward layers.

SSA-XGBoost: $O(K \cdot d \cdot log\ n)$ for tree building (K: trees, d: features, n: samples), plus $O(I \cdot P \cdot D)$ for SSA optimization (I: iterations, P: population size, D: parameter dimension).

Overall complexity: $O(T \cdot L \cdot H^2 + T^2 \cdot d_{model} + K \cdot d \cdot log\ n)$. In practice, the SSA overhead is negligible ($I \leq 50$, $P \leq 30$).

3. Experiment and Result Analysis on Academic Early Warning and Prediction

The original data for the academic warning prediction comes from the school's academic administration system of the Aviation Engineering College of the Civil Aviation of China University for students of grades 18 and 19. The original data is messy and the

courses selected by different students vary, making it difficult to extract logical information. Therefore, we only selected students from the same major and used their compulsory course grades as samples to make predictions about their academic situations. A dataset consisting of 19 features was obtained, and all the data grades were arranged in chronological order from the first year to the fourth year.

The first few columns of the data mostly contain the grades of basic courses such as "Mechanical Drawing", "Theoretical Mechanics", and "Mechanical Materials". The latter columns represent the grades of more challenging specialized courses. The grades of the latter courses are partly related to the accumulation of knowledge from the former courses. Therefore, the grades of the former courses have a certain impact on the latter courses to some extent. Moreover, the academic situation is predicted based on the overall performance of the students in the following academic year. If the grades of the earlier courses are not satisfactory, the requirements for the grades in the later courses will be raised to avoid being warned. Thus, in such cases, the timing issue of the courses should also be taken into account when making predictions. The complex model proposed in this paper has well considered this point. And the warning values y of 0 or 1 in this paper, when $y = 0$, indicate that the student has received an academic warning, or when $y = 1$, indicate that the student has not received an academic warning (0). The dataset consists of 518 samples. The ratio of samples labeled as "requiring alert" to those labeled as "not requiring alert" is 8:251.

Then, we normalized the academic early warning prediction data, which alleviated the related problems of gradient descent optimization being hindered due to significant differences in the range of eigenvalues. This could reduce the model's sensitivity to specific feature scales and enhance its generalization ability on different data scales [18]. Then, the training set and the test set were divided in a ratio of 7:3, and then into two datasets according to the compulsory professional courses and the general basic courses.

The parameters in the SSA-XGBoost module mainly include the learning_rate α, the number of weak evaluators ϑ, the proportion of samples taken during random sampling δ, the maximum depth of the tree ℓ and the value of the random sampling feature ratio ξ. According to past experience articles [19], the values of δ and ξ are in the range of 0.5 to 1, and ℓ is generally in the range of 1 to 10. Then, we refer to similar experience articles on predicting nonlinear data [20] to get the optimal value with the help of the SSA algorithm.

Ultimately, we optimized the XGBoost model using the SSA algorithm and obtained the following parameter value results (Table 1).

Table 1. The result of SSA-XGBoost module.

Parameter	Learning_Rate	n_Estimators	Subsample	Max_Depth	Colsample_Bytree
value	0.19	163	0.79	7	0.77

This paper employs LSTM, CNN, XGBoost, and the LSTM + CNN model for a comparative experiment. For other machine learning network parameters, the best parameter value was selected by comparing the error obtained by bringing in different values. This paper selects four evaluation models, which are Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). The equations of the four evaluation models are, respectively, shown as follows.

$$\mathrm{MSE}(y_{pre}, y_{act}) = \frac{1}{n}\sum_{u=1}^{n}(y_{pre} - y_{act})^2 \tag{27}$$

$$\mathrm{RMSE}(y_{pre}, y_{act}) = \sqrt{\frac{1}{n}\sum_{u=1}^{n}(y_{pre} - y_{act})^2} \tag{28}$$

$$MAE = \frac{1}{n}\sum_{u=1}^{n}\left\|y_{pre} - y_{act}\right\| \tag{29}$$

$$MAPE = \frac{1}{n}\sum_{u=1}^{n}\left\|\frac{y_{pre} - y_{act}}{y_{pre}}\right\| \tag{30}$$

where y_{pre} is the predicted value of carbon emissions, y_{act} is the true value of carbon emissions from the machine tool, and n is the number of samples in the test set.

The errors of each evaluation index obtained during the optimization process are shown in the following table.

It can be seen from the above three tables, Tables 2–4, that when colsample_bytree takes 0.4, the MSE is reduced by 4.2% and 8.6% compared with 0.6, 0.5 and 0.8. The MAE was reduced by 1.9%. The MAPE was reduced by 0.1%. When the subsample was taken as 0.8, the error of MSE was reduced by 5.9%, 14.8% and 4% compared with those taken as 0.5, 0.7 and 0.9. The MAE was reduced by 5.4%, 7.0% and 1.9%, respectively. The MAPE was reduced by 0.1%. When max_depth was set at 7, the MSE was reduced by 4.6%, 1.9% and 24.6%, respectively, compared with when it was set at 5, 3 and 1. The MAEs were reduced by 5.4%, 10.3% and 34.0%, respectively. Therefore, we take colsample_bytree as 0.4, subsample as 0.8, and max_depth as 7.

Table 2. Comparison of optimization errors of colsample_bytree parameters in the XGBoost model.

Colsample_Bytree	MSE (10^{-1})	MAE	RMSE	MAPE
0.4	0.138	0.053	0.118	0.996
0.5	0.151	0.054	0.123	0.996
0.6	0.144	0.053	0.120	0.994
0.8	0.151	0.055	0.123	0.997

Table 3. Comparison of optimization errors of subsample parameters in XGBoost models.

Subsample	MSE (10^{-1})	MAE	RMSE	MAPE
0.5	0.153	0.056	0.124	0.991
0.7	0.169	0.057	0.130	0.994
0.8	0.144	0.053	0.120	0.994
0.9	0.150	0.054	0.122	0.995

Table 4. Comparison of optimization error of max_depth parameter in XGBoost model.

Max_Depth	MSE (10^{-1})	MAE	RMSE	MAPE
7	0.104	0.035	0.102	1.005
5	0.109	0.037	0.105	1.004
3	0.106	0.039	0.103	0.997
1	0.138	0.053	0.118	0.996

For LSTM models, the purpose of optimizing the model is mainly achieved by adjusting its hidden layer functions. The results of the model optimization process are shown in the following table.

Obviously, according to the results of Table 5, when the number of hidden layers is four, compared with six, five and eight, the MSE is reduced by 4.5% and 5.2%, respectively. The RMSE has been reduced by 2.6%. The MAE was reduced by 1.9%. Therefore, we set the number of hidden layers to four. Finally, we conducted hyperparameter optimization on the CNN model. This paper studied three widely used excitation functions, including Sigmoid, Tanh, and ReLU. The results predicted by the model under the three excitation functions are shown in the following figure.

Table 5. Error comparison of different hidden_sizes in LSTM models.

Hidden_Size	MSE (10^{-1})	MAE	RMSE	MAPE
5	0.134	0.050	0.116	1.322
6	0.133	0.052	0.115	1.302
8	0.134	0.051	0.116	1.307
4	0.127	0.051	0.113	1.339

Based on the results of Figure 5, we select the ReLU function as the excitation function of the CNN model. Based on the above conclusions, we can determine the values of all the hyperparameters for model optimization. In the CNN + LSTM model, the MSE obtained by using the ReLU excitation function is the smallest.

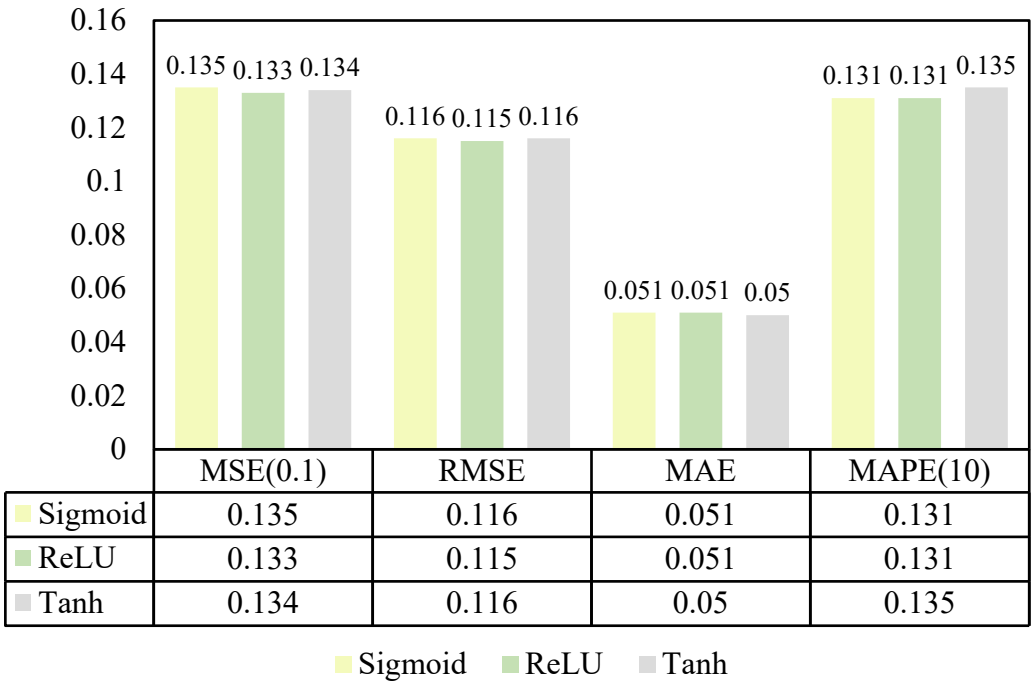

Figure 5. Comparison of MSE values for three excitation functions of CNN models.

Set the training steps of the five classification prediction models to 150 times. Divide all the samples in the dataset into rows and divide them into the training set and test set in a 7:3 ratio for the experiment. As a binary classification prediction problem, in this paper, the situation where the prediction result is less than 0.5 is defined as 0; that is, no academic warning occurs. Otherwise, it is defined as 1; that is, an academic warning occurs.

To verify the effectiveness of the BiLSTM module, the Enhanced Transformer model and SSA-XGBoost, this study conducted ablation experiments.

Table 6 shows that the MSE of the LTBoost proposed in this paper is 27.9% lower than that of BiLSTM, 7.7% lower than that of BiLSTM + SSA-XGBoost, and 4.8% lower than that

of the Enhanced Transformer + SSA-XGBoost. The MAE of LTBoost is 41.3% lower than that of BiLSTM, 36% lower than that of BiLSTM + SSA-XGBoost, and 12.8% lower than that of the Enhanced Transformer + SSA-XGBoost. The RMSE of LTBoost is 13% lower than that of BiLSTM, 3.7% lower than that of BiLSTM + SSA-XGBoost, and 2.4% lower than that of the Enhanced Transformer + SSA-XGBoost. The MAPE of LTBoost is 29.2% lower than that of BiLSTM, 11% lower than that of BiLSTM + SSA-XGBoost, and 1.1% lower than that of the Enhanced Transformer + SSA-XGBoost.

Table 6. LTBoost Model Abandonment Experiment.

	MSE (10^{-1})	MAE	RMSE	MAPE
BiLSTM	0.133	0.0506	0.1153	1.2969
BiLSTM + SSA-XGBoost	0.112	0.0487	0.1058	1.1147
Enhanced Transformer + SSA-XGBoost	0.109	0.0404	0.1044	1.0150
LTBoost	0.104	0.0358	0.1020	1.0038

This article compares LTBoost with the commonly used CNN, LSTM, XGBoost and CNN + LSTM, and the results are shown in the table below:

Table 7 shows that the MSE of the LTBoost proposed in this paper is 30.8%, 28.8%, 27.9%, and 29.8% lower than that of CNN, LSTM, XGBoost, and CNN + LSTM, respectively; the MAE of LTBoost is 39.1%, 38.5%, 38.5%, and 40.2% lower than that of CNN, LSTM, XGBoost, and CNN + LSTM, respectively; the RMSE of LTBoost is 14.4%, 13.3%, 13.1%, and 30.7% lower than that of CNN, LSTM, XGBoost, and CNN + LSTM, respectively; and the MAPE of LTBoost is 42.4%, 34%, 30.7%, and 40.4% lower than that of CNN, LSTM, XGBoost, and CNN + LSTM, respectively.

Table 7. LTBoost Model comparison Experiment.

	MSE (10^{-1})	MAE	RMSE	MAPE
CNN	0.136	0.0498	0.1167	1.4290
LSTM	0.134	0.0496	0.1156	1.3454
XGBoost	0.133	0.0496	0.1154	1.3118
CNN + LSTM	0.135	0.0502	0.1162	1.4090
LTBoost	0.104	0.0358	0.1020	1.0038

The accuracy rate is adopted as the evaluation model, as shown in Equation (31). To present the prediction situation more intuitively, this paper also incorporates a confusion matrix as an evaluation index.

$$Acu = \frac{TP + TN}{Total} \tag{31}$$

where TP represents the number of the actual value is 1 and the predicted value is also 1, TN represents the number of the actual value is 0 and the predicted value is also 0, and $Total$ represents the number of samples in the test set.

The confusion matrices of LTBoost and the four comparison models are shown in the following figure:

It can be seen from the confusion matrices in Figures 6 and 7 that LTBoost only predicted one sample originally "1" as "0", while CNN, LSTM and CNN + LSTM all predicted two samples originally "1" as "0", and the XGBoost model predicted a sample originally "0" as "1". Predict a sample that was originally "1" as "0". It can be seen from Table 8 that the accuracy of the LTBoost model has increased by 0.7% compared with

the other four comparison models. From this, it can be seen that LTBoost has the best prediction effect.

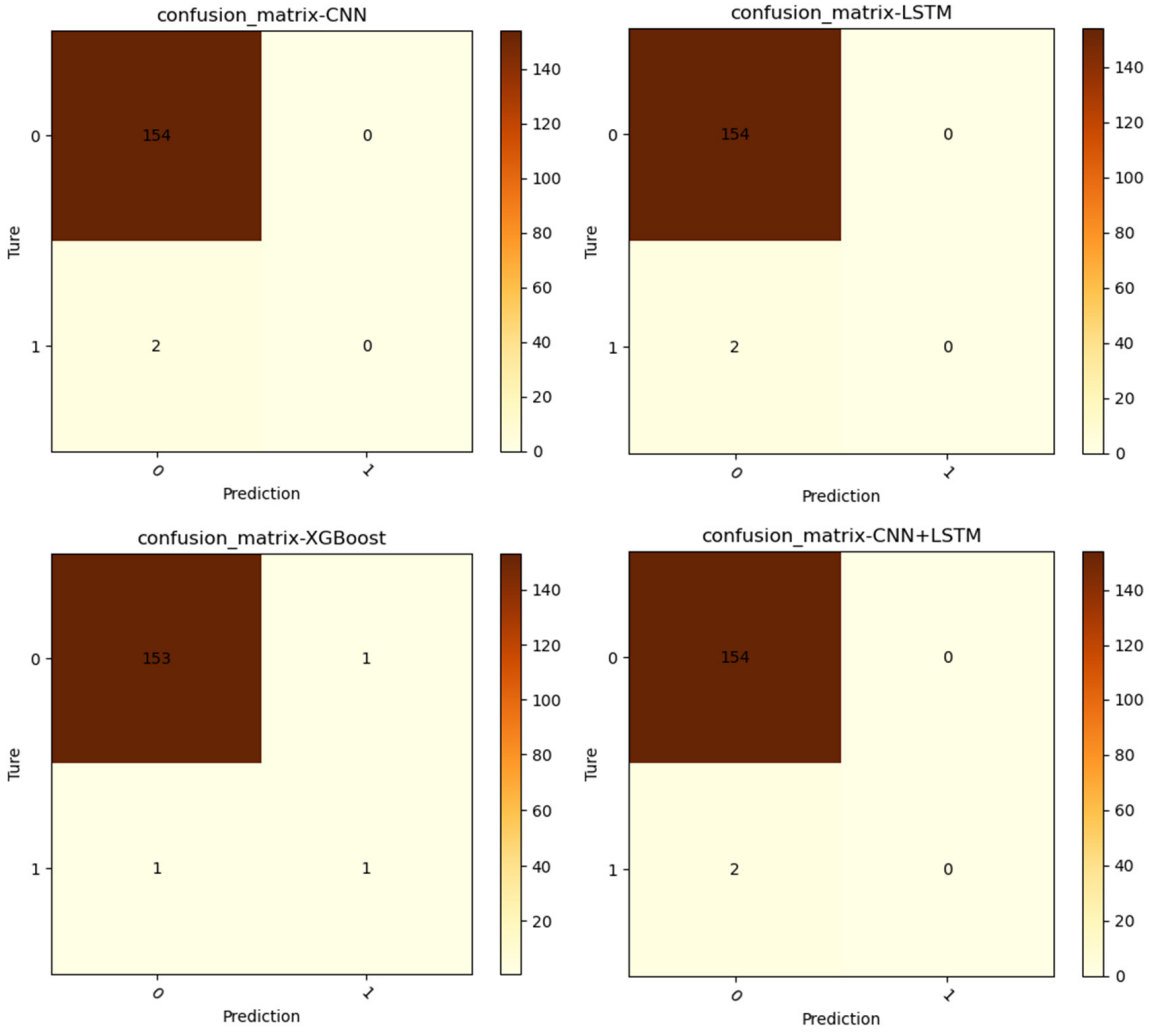

Figure 6. The confusion matrices of the four comparison models.

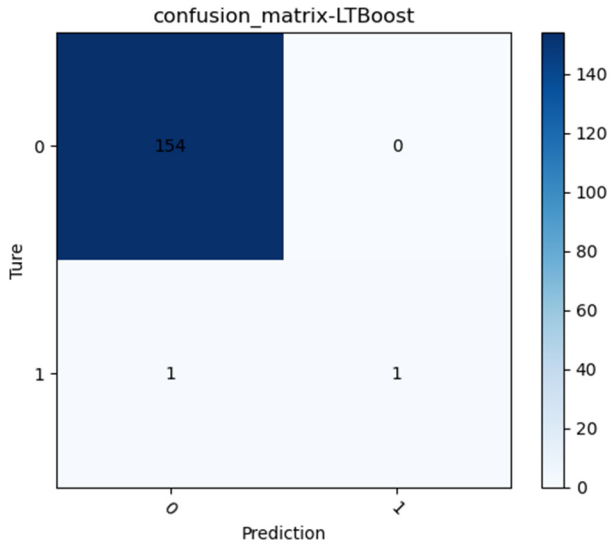

Figure 7. The confusion matrix of the LTBoost model.

Table 8. The accuracy of the LTBoost model compared with four comparison models.

Acu	CNN	LSTM	XGBoost	CNN + LSTM	LTBoost
value	0.987	0.987	0.987	0.987	0.994

In order to make the experimental results more convincing, we also incorporated 5-fold cross-validation in the experiment.

As can be seen from Table 9, after 5-fold validation, the average RMSE value of the LTBoost model decreased by 12.06%, 12.06%, 9.05% and 14.73%, respectively, compared to the CNN model, LSTM model, XGBoost model and CNN + LSTM model. From this, it can also be proven that LTBoost has the best prediction effect.

Table 9. The average RMSE of the LTBoost model compared with four comparison models after 5-fold cross-validation.

	CNN	LSTM	XGBoost	CNN + LSTM	LTBoost
RMSE	0.1692	0.1692	0.1636	0.1745	0.1488

The data of academic early warning has a strong temporal correlation before and after. Just as the data features of the earlier time series are often general education courses, the values of these features may affect the feature values of the professional courses that are later in the time series. As can be seen from the prediction results shown above, for LSTM, CNN and CNN + LSTM models, it is impossible to capture the context information simultaneously, and the prediction effect is not very good in comparison, indicating that the underfitting phenomenon in these predictions has occurred. The XGBoost model may still have insufficient fitting due to the fact that the data captured by the model from a long time ago has little connection with the data prediction at this moment.

The LTBoost model incorporates the BiLSTM module and the Enhanced Transformer module. By connecting the BiLSTM module with context information to extract temporal patterns, the prediction accuracy is improved. Additionally, the Enhanced Transformer module is integrated into it to increase the feature information of the prediction data and the prediction feature conditions, making the prediction results closer to the true values. Further improve the accuracy of predictions.

4. Conclusions

Against the background of the credit system in colleges and universities, the academic early warning classification prediction model can help students make academic plans in advance, adjust their learning state in a timely manner, and reduce the probability of failing grades. Based on the high temporal characteristics of various subject performance data, this paper proposes the LTBoost prediction model. This model uses BiLSTM to preprocess the data, extract the upper and lower time series information, adopts the Enhanced Transformer module to increase the number of feature parameters, and uses the SSA-XGBoost module for prediction.

By comparing with four prediction model algorithms, namely LSTM, CNN, CNN + LSTM, and XGBoost, the following conclusions can be drawn:

(1) This study found that the data on academic early warning problems have strong time series characteristics. The BiLSTM module, which prioritizes the processing of raw data, can well connect the time series relationships among the data and improve the prediction accuracy of classification.

(2) The Enhanced Transformer module can obtain enhanced features associated with the original data. The added weight attenuation algorithm and Dropout layer can prevent overfitting of the model, significantly improving the prediction accuracy of the model.
(3) The prediction accuracy of the LTBoost prediction model is higher than that of the other models. Compared with traditional prediction models, it can perceive global information and mine time series information, and the integrated SSA search algorithm further improves the accuracy of model prediction and the convenience of optimizing parameters. Its accuracy rate has increased by 0.7%. Therefore, the LTBoost prediction model has the highest accuracy rate in the academic early warning classification prediction task.

The LTBoost prediction model can effectively fulfill the academic early warning and prediction tasks for college students. It can help students set study plans in advance, reduce the probability of failing grades, and also provide data analysis support for colleges to make decisions on adjusting teaching plans.

The LTBoost model is not only applicable to binary classification prediction of academic early warning, but also has potential for prediction and classification tasks of other similar strong time series data. In the future, we plan to collect more data from different directions to verify our conjecture and study methods that can effectively reduce the running time of the Enhanced Transformer module.

Author Contributions: Conceptualization, H.S. and S.F.; Formal Analysis, H.S., S.F. and Z.Y.; Funding Acquisition, Z.Y. and H.S.; Methodology, H.S. and S.F.; Resources, Z.Y.; Software, W.W. and M.M.; Supervision, W.W.; Validation, W.W.; Visualization, S.F.; Writing—Original Draft, S.F., W.W. and M.M.; Writing—Review and Editing, H.S., S.F. and Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Key Laboratory of Infrastructure Durability and Operation Safety in Airfield of CAAC (Funding Number: MK202402, Funder: Zhiqi Yan), the Tianjin Natural Science Foundation (Funding Number: 24JCQNJC00220, Funder: Zhiqi Yan) and the Central Universities Civil Aviation University of China Special (Funding Number: 3122025005, Funder: Hailong Sun).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bai, X.; Yuan, S.; Du, X.; Sun, S.; Han, Y.; Liu, Y. A Preliminary Study on the Warning Factors of Academic Failure among College Students. *J. Tianjin Norm. Univ.* **2022**, 70–76. [CrossRef]
2. Zhai, M.; Wang, S.; Wang, Y.; Wang, D. An interpretable prediction method for university student academic crisis warning. *Complex Intell. Syst.* **2022**, *8*, 323–336.
3. Kotsiantis, S.; Pierrakeas, C.; Pintelas, P. Pre-dicting Students' Performance in Distance Learning Using Machine Learning Techniques. *Appl. Artifi. Intell.* **2004**, *18*, 411–426. [CrossRef]
4. Asif, R.; Merceron, A.; Ali, S.A.; Haider, N.G. Analyzing undergraduate students' performance using educational data mining. *Comput. Educ.* **2017**, *113*, 177–194. [CrossRef]
5. Ramanathan, K.; Thangavel, B. Minkowski sommon feature map-based densely connected deep convolution network with LSTM for academic performance prediction. *Concurr. Comput. Pract. Exp.* **2021**, *33*, e6244. [CrossRef]
6. Feng, J.; Lian, X. Reflections on and Exploration of Academic Early Warning Management and Support for Students in Colleges and Universities. *Appl. Math. Nonlinear Sci.* **2024**, *9*, 1–19. [CrossRef]
7. Young, E.L.; Moulton, S.E.; Julian, A. Integrating social-emotional-behavioral screening with early warning indicators in a high school setting. *Prev. Sch. Fail.* **2021**, *65*, 255–265. [CrossRef]
8. Mccallum, J.; Duffy, K.; Hastie, E.; Ness, V.; Price, L. Developing nursing students'decision making skills: Are early warning scoring systems helpful? *Nurse Educ. Pract.* **2013**, *13*, 1–3. [CrossRef] [PubMed]

9. Hudson, W.E., Sr. Can an early alert excessive absenteeism warning system be effective in retaining freshman students? *J. Coll. Stud. Retent. Res. Theory Pract.* **2016**, *7*, 217–226. [CrossRef]
10. Yang, X. Optimization and implementation of management technology integrated with data analysis for college students' course evaluation and academic early warning. *Syst. Soft Comput.* **2025**, *7*, 200255. [CrossRef]
11. Ji, B. Research on the Application of SVM-RFE Algorithm in the Construction of Students' Academic Early Warning Model. *Int. J. Inf. Syst. Model. Des.* **2024**, *16*, 1–21. [CrossRef]
12. Duong, H.T.H.; Tran, L.T.M.; To, H.Q.; Van Nguyen, K. Academic performance warning system based on data driven for higher education. *Neural Comput. Appl.* **2022**, *35*, 5819–5837. [CrossRef] [PubMed]
13. Duong, H.T.H.; Tran, L.T.M.; To, H.Q.; Van Nguyen, K. Robust regression using support vector regressions. *Chaos Solitons Fractals* **2021**, *144*, 110738. [CrossRef]
14. Khyathi, G.; Indumathi, K.P.; Jumana Hasin, A.; Lisa Flavin Jency, M.; Krishnaprakash, G.; Lisa, F.J.M. Support Vector Machines: A Literature Review on Their Application in Analyzing Mass Data for Public Health. *Cureus* **2025**, *17*, e77169. [CrossRef] [PubMed]
15. Karamizadeh, S.; Abdullah, S.M.; Halimi, M.; Shayan, J.; Javad Rajabi, M. Advantage and drawback of support vector machine functionality. In *Proceedings of the 2014 International Conference on Computer, Communications, and Control Technology (I4CT), Langkawi, Malaysia, 2–4 September 2014*; IEEE: Piscataway, NJ, USA, 2014; pp. 63–65. [CrossRef]
16. Chen, S.; Jin, H.; Li, L. Analysis and Comparison of House Price Prediction Based on XGboost and LightGBM. *Adv. Econ. Manag. Political Sci.* **2023**, *46*, 55–61. [CrossRef]
17. Fan, Y.; Tang, Q.; Guo, Y.; Wei, Y. BiLSTM-MLAM: A Multi-Scale Time Series Prediction Model for Sensor Data Based on Bi-LSTM and Local Attention Mechanisms. *Sensors* **2024**, *24*, 3962. [CrossRef] [PubMed]
18. Sun, Z.; Wang, X.; Huang, H.; Yang, Y.; Wu, Z. Predicting compressive strength of fiber-reinforced coral aggregate concrete: Interpretable optimized XGBoost model and experimental validation. In *Structures*; Elsevier: Amsterdam, The Netherlands, 2024; Volume 64.
19. Ding, N.; Ruan, X.; Wang, H.; Liu, Y. Automobile Insurance Fraud Detection Based on PSO-XGBoost Model and Interpretable Machine Learning Method. *Insur. Math. Econ.* **2025**, *120*, 51–60.
20. Kamali, H.A. Impact of liquid injector pressure on gas flow characteristics and evaporation rates: A combined Eulerian-Lagrangian approach and PSO-optimized XGBoost model. *Int. Commun. Heat Mass Transf.* **2025**, *164*, 108822. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.