Hybrid Deep Learning for Academic Achievement Prediction Using Spatio-Temporal and Behavioral Data in Higher Education

Asim Seedahmed Ali Osman
College of Computer Science and Engineering, University of Hafr Al Batin, Hafr Al Batin, Saudi Arabia

***Abstract*—Accurate prediction of student academic performance is essential for enabling timely and effective educational interventions. Many existing prediction approaches focus either on academic outcomes or behavioral trends, without fully capturing the interaction between spatial performance indicators and their temporal evolution. To address this limitation, this study proposes a hybrid deep learning model that integrates spatio-temporal information for forecasting student achievement in higher education. The proposed framework combines a Convolutional Neural Network (CNN) to extract spatial features from normalized academic performance data with a Long Short-Term Memory (LSTM) network to model temporal patterns in student behavioral attributes, such as attendance and participation. In addition, FOX optimization is applied to adaptively tune the learning rate, improving training stability and predictive performance. The model is evaluated using student academic and behavioral datasets, and its performance is compared with commonly used baseline models. Experimental results show that the proposed CNN–LSTM approach achieves an accuracy of 97.18 per cent, outperforming standalone LSTM and Support Vector Machine (SVM) models. Furthermore, the model effectively classifies students into low, medium, and high academic risk categories, supporting early identification of at-risk students and facilitating timely intervention in higher education environments.**

Keywords*—Hybrid deep learning; CNN; LSTM; student performance; FOX optimization

I. Introduction

Student success prediction has become an important topic in contemporary education, as it enables early identification of students who may be at risk of academic failure and supports timely and targeted interventions [1]. With the increasing availability of educational data, several predictive models have been proposed to analyze student performance and guide academic decision-making. However, most traditional prediction approaches rely primarily on academic achievement indicators, while important behavioral factors such as attendance, participation, and engagement with learning activities are often neglected [2],[5]. This limitation reduces the ability of existing models to fully represent the complex nature of student learning behavior [3].

In addition, student performance is inherently dynamic and evolves, influenced by changes in learning behavior, academic workload, and progression across courses or semesters. Many conventional models fail to capture this dynamic nature, as they focus mainly on static features and ignore temporal information related to student behavior and performance [6],[7]. Although some machine learning techniques have shown satisfactory results, they do not adequately model the time-based evolution of student data, which restricts their effectiveness in supporting early educational interventions [7].

To address temporal dependencies, recurrent neural networks, particularly Long Short-Term Memory (LSTM) models, have been widely adopted for sequential prediction tasks in educational data [8], [36]. While LSTM-based approaches are effective in modeling temporal patterns, they are not designed to capture spatial relationships among academic and behavioral features [9]. Conversely, Convolutional Neural Networks (CNNs) have been successfully applied to extract spatial patterns from educational data, such as correlations among performance indicators and behavioral attributes [10]. However, CNN-based models lack the capability to model the temporal variation of student behavior and academic progress [10]. As a result, existing approaches that consider either spatial or temporal aspects in isolation are insufficient for accurately predicting student performance in real-world educational settings [11].

Motivated by these limitations, this study addresses the problem of developing a unified framework that can jointly model both spatial and temporal characteristics of student data [4]. The main objective of this research is to forecast student academic performance using a hybrid deep learning model that integrates CNN and LSTM architectures [12], [13]. Specifically, the proposed framework utilizes normalized academic achievement data and standardized behavioral observations collected across multiple time points. CNN is employed to extract spatial patterns related to academic performance and student behavior, while LSTM networks are used to capture the temporal dynamics of these patterns over time [12], [13]. By combining the complementary strengths of CNN and LSTM, the proposed approach aims to improve prediction accuracy, enable more responsive classification of students into low-, medium-, and high-risk categories, and support timely, data-driven interventions for students who are struggling or at risk of dropout [14], [15].

Although the proposed framework is evaluated using higher education data, it is designed to be flexible and extensible to different academic environments. Structured institutional features—such as credit-based course systems, GPA and CGPA assessment schemes, prerequisite constraints, and semester-based progression policies—can be incorporated into the hybrid

CNN–LSTM model. In this extended setting, semester-to-semester academic progression and credit accumulation are modeled as temporal sequences, while grade and credit distributions are treated as spatial features, enabling early detection of academic risk in both undergraduate and postgraduate programs.

The remainder of this study is organized as follows: Section II reviews related work and discusses existing approaches and their limitations. It presents the problem formulation and highlights the research gap addressed in this study. Section III describes the proposed hybrid CNN–LSTM methodology for student performance prediction. Section IV introduces the FOX optimization technique and its role in improving model performance. Experimental results are presented in Section V, followed by a discussion in Section VI. Finally, Section VII concludes the study and outlines directions for future research.

II. Related Works

Shou et al. [16] have emphasized how internet usage behavior strongly reflects student performance, and it is showing how the digital traces will promise to predict the performance. The application of deep learning techniques under the use of convolutional architectures has been used to automatically acquire important learning features, with the potential of being better than the previous handcrafted features. Combining multi-source behavioral, demographic, and test data and combining them with hybrid deep learning has been found to perform better in estimating the results of students and providing an opportunity to intervene early. A study conducted by Liu et al. [17] showed that internet behaviors could be effectively applied in determining vulnerable students, and this proves the effect of behavior data in determining performance forecasts. CNN structures have been utilized to learn advanced learning characteristics automatically with reduced dependence on features handcrafting. Nafea et al. [18] discussed sensor-based solutions to human activity detection and the suitability of wearable sensors in the detection of motion dynamics. Hybrid networks, which combine CNN with RNN, have been shown to perform better as they can capture the space and time dimensions of the activities. Follow-up models have been centered on learning multi-resolution features and the optimal representation of data, and high levels of performance have been reached on standard HAR datasets. Xu et al. [19] studied classroom video analysis as an indicator of teacher behaviors and student interactions in classrooms and showed the utility of automated identification in the classroom. In this instance, deep learning techniques, specifically CNN, have been used to identify spatial characteristics of classroom interaction and do it correctly.

Yağcı et al. [20] explored the use of ML models to forecast achievement in education and have shown that test results can be reliable predictors of academic performance. All of them have been extensively applied to educational prediction problems due to their high classification performance. Zheng et al. [21] pinpointed the role of personal effectiveness in the outcomes of learners, which is of high predictability in learning environments. Scholarly studies have found that intrinsic and extrinsic motivators are potent in the participation and persistence of learning activities. As Nabil et al. [22] emphasized the use of deep learning to forecast achievement in higher education was discovered to be more efficient than the traditional models to predict at-risk kids. Having less precise but more explainable results than neural networks, they were commonly used in learning-based data mining. Early academic prediction has also been done using logistic regression and support vector classifiers, but in a linear and less flexible way. In managing the class imbalance, SMOTE and ADASYN resampling techniques have been proposed to strengthen classifiers. Katagiri et al. [23] emphasized how early motor skills serve as predictors of future academic and psychosocial success, thereby making them worthy of use as indicators of educational adjustment. It has been demonstrated that fine motor development is what facilitates early literacy and numeracy by optimizing the efficiency of tasks and participation in the classroom. Social interaction and regulation of emotions associated with Gross motor skills have been attributed to affect peer relationships and happiness.

Cagliero et al. [24] elaborated on the interpretability of early prediction models of student performance, and the interpretable classifiers may be useful in a specific intervention. Associative classifiers have rule-based explanations that enable the teacher to audit the model output and exhibit usable inferences. According to the meta-analysis done by Madigan et al. [25], student burnout is strongly negatively related to academic performance, with the least efficacy. Repeated fatigue research has proved that chronic fatigue impairs concentration and learning capacity. The model of predicting the success of the online courses in terms of the students is the development of the artificial intelligence-based model by Jiao et al. [26], where the focus was made on the combination of the data on the learning process and summative measures. To drive predictive models to optimality, evolutionary computation algorithms have been used to achieve competitive accuracy with respect to benchmark machine learning. Research has collected evidence that active involvement, knowledge gained and test results were better predictors of learning as compared to prerequisite knowledge. Martin et al. [27] emphasized the predictive role of grit to academic performance, especially through persistence of effort as an enabler of self-driven learning patterns. The empirical data indicate that individual performance and internal drive have a significant impact on determining the GPA of students because it influences the sense of confidence and endurance in learning. The relationship between grit and performance is mediated by cognitive strategy exploitation and self-regulation.

Heppt et al. [28] have pointed out the predictive significance of books-at-home measures on student academic performance, especially in the area of language comprehension. The trend of consistency in findings has been that print access boosts literacy development and moderates the effects of socioeconomic status on achievement. Home literacy environment, as reported by parents, typically compares better with self-reported ones of students and is more correlated with academic outcomes. The researchers also indicate that reading traditional print materials, including books by parents, leads to increased development of language as compared to using computer materials like e-books. Xiong et al. [29] also indicated that there are two-way relationships between parent involvement and higher education performance of adolescents, which indicate how the measure of

student achievement can actually mediate parenting. Studies have continually indicated that parental engagement does have a positive relationship with increased student engagement, student motivation, and higher education success. The relationship between family involvement and the learning outcomes has also proved to be mediated by academic engagement, especially behavioral engagement. It is further implied that these relations may not be gender similar in terms of their force and direction, as they may have variation in mediation effects between girls and boys. Comprehensively, parental involvement is a strong but multifaceted factor in the academic performance of adolescents. Al-Abyadh et al. [30] have pointed out the decisive role played by personal effectiveness and personal organization to academic success of students, in that they stressed the interdependent roles of both on learning success. It has also been indicated in research that students with personal organization skill can plan, monitor, and control their learning behaviors. Personal performance has long been considered a legitimate predictor of motivation, tenacity and performance in learning settings of various kinds. Comparisons involving other cultures indicate that these psych variables operate at the same level in aiding academic performance. Generally, the combination of personal organization strategies and the building of individual effectiveness is the most significant in the attainment of university students.

Online traces and patterns of internet use are discovered to screen student achievement very well with deep learning, graph-based, and recurrent models, providing more precise outcomes based on hybrid approaches. Other behavioral information and neural networks, such as spiking models, also boost the early identification of at-risk learners. The use of sensors with CNN and BiLSTM to combine spatial and temporal variations is a good approach to research [31], [35], but attention-based video classification of the classroom is a good predictor of engagement and participation. Machine learning techniques remain good predictors of academics, with deep learning techniques using resampling techniques being more stable. Personal effectiveness, motivation, self-regulation, and grit are psychological predispositions that have direct involvement in achieving results [32], where autonomous learning and interest often mediate these effects. They are powerful predictors of both academic and social adaptation, whereas parental involvement reciprocates and affects both achievement and is determined by gender-specific factors. Burnout is an adverse outcome factor, the greatest impact of which is reduced efficacy and disengagement. Process and summative information combined in AI models enhance forecasts in e-learning, whereas explainability and trust are provided by explainable models. Collectively, the evidence reiterates the fact that the set of technological, psychological, behavioral, and environmental factors is combined and supported by sophisticated AI and machine learning to enhance predictive accuracy and intervention in academic performance.

III. Proposed Methodology

The idea of academic performance prediction is based on integrated and systematic spatial and temporal learning based on an advanced model of deep learning. This starts with the data collection process, where the performance and behavior data of the students are collected in the form of normalized academic scores and standardized behavior scores. Once the data is gathered, there are activities involved in preprocessing, which include handling of missing data, normalization of academic marks, normalization of behavior features, and identification of temporal features. The processed data is then sent to the Hybrid CNN and LSTM model, in which the CNN detects the spatial patterns of data and the LSTM detects the time dependency of data. The identified features of CNN and LSTM are joined under a fusion layer and fed to the SoftMax function to categorize the students into one of the three categories: low, medium, or high risk. Besides this, the FOX optimization method is used to adjust the learning rates, which hastens the training process and enhances the convergence by fine-tuning gradients with the help of fractional-order derivatives. The performance metrics to ensure that the model can predict the results of students based on the past data. Fig. 1 demonstrates the hybrid deep learning architecture to predict the success of the students based on spatio-temporal and behavioral data.

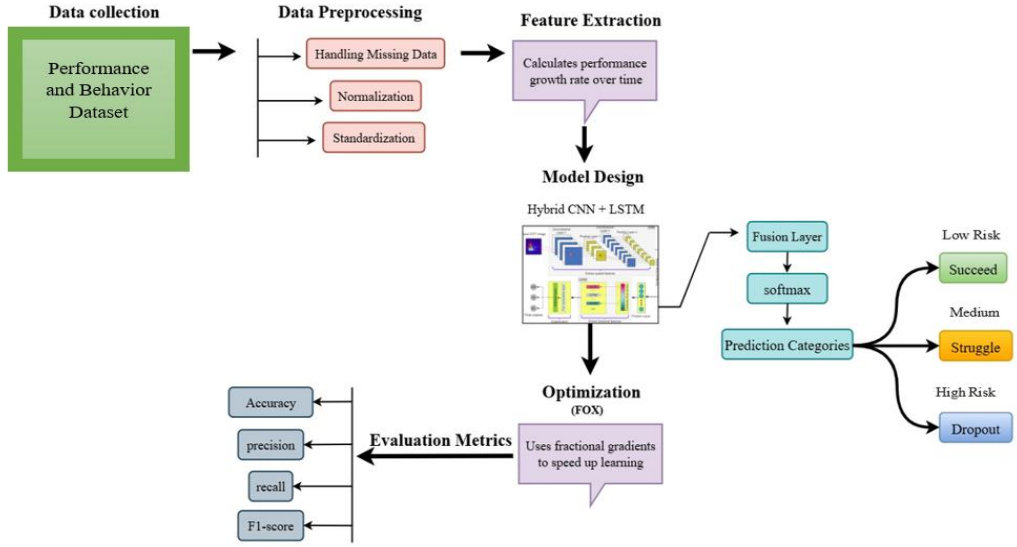

Fig. 1. Hybrid deep learning framework for predicting student success using spatio-temporal and behavioral data.

A. Data Collection

The academic history and learning behavior of the students in the Performance and Behavior Dataset obtained on Kaggle is well-rounded, as it consists of exam and assignment in-subject scores, attendance, classroom and study behavior, disciplinary behavioral indicators, and socio-demographic characteristics such as parental education, gender, and family background. The data set captures longitudinal academic tracks of students in several subjects, and this will be used to study the performance of the students longitudinally. To deploy the framework at the university-level, it additionally helps to combine the information system data of students, including course enrollment, credit completion, GPA/CGPA, and academic standing, with the logs of the learning management system, including the frequency of logins, content access, time-on-task and assignment submission behavior. Normalization, categorical encoding, and missing value are preprocessing steps that provide high-quality inputs that can be easily modeled spatio-temporally to predict early risks by academic performance and learning behavior.

B. Data Preprocessing

Preprocessing ensures that the behavioral and spatio-temporal data are consistent, cleaned, and prepared to be analyzed. The data treatment, academic scores normalization, behavioral indicators standardization, and time learning patterns extraction are missing in the operations involved. The conversions ensure that the data from various sources and scales may be compared and modeled in a meaningful way.

1) Handling missing data: The norm in datasets that cover the practical education is missing values due to attendance not mentioned, assignment not submitted, behavioral records missing or data entry errors. In the absence of such missing entries, analysis will be biased, and model performance will be worse as deep learning models need full and consistent input. To solve this, numeric attributes (e.g., test scores, percentage of class attendance, proportion of participation) are imputed with a mean imputation method, which replaces a missing value with the mean of all the available values of the attribute. This guarantees that the value added to it is representative of the frequency of the data, and it does not distort the general trend. The imputation equation can be written as in Eq. (1):

$$x_i' = \frac{\sum_{j=1}^{n} x_j}{n} \quad (1)$$

where, x_i' is the imputed value for the missing observation, x_j is the observed value of the feature, n is the number of valid observations of the feature. Mode imputation strategy is used in categorical attributes (e.g., type of study habit, disciplinary status, level of participation in classes). It is done by replacing the missing item with the most frequent category to keep the consistency in class distributions. These imputation techniques would ensure that the data set is not lost and the useful records are retained. This preserves the sample size and mitigates the systematic loss of data bias, and preserves all spatio-temporal and behavioral characteristics to conduct additional analysis. Finally, this approach produces a balanced platform of model training in which no feature is unfairly skewed due to the absence of data.

2) Normalization of academic scores: In education datasets, the marks of different subjects like mathematics, reading and writing usually have dissimilar marking systems that make these marks incomparable and hard to merge. An example of this is that one of the subjects will have a mark between 0 and 50, and the other one will have a mark between 0 and 100. Min-Max Normalization is used in order to have marks with an equal range. The process standardizes the data in a common range to facilitate easier analysis and comparisons among the individuals without the overbearing influence of one person because of the large range of scores. The scores are transformed through min-max normalization with the highest and the lowest numbers in the dataset. The normalized score (x_i') for a given raw score (x_i) is calculated in Eq. (2):

$$x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

where, x_i' is the normalized score, x_i is the original score of the student, $x_{\min}$ is the lowest score observed in the data of that subject, $x_{\max}$ is the highest score observed for that subject.

3) Standardization of behavioral features: Behavioral indicators (attendance rate, frequency of participation, frequency of submitting assignments, etc.) in learning data sets may vary in range and units of measurement. As an example, attendance could be measured as a percentage, participation frequency (e.g., low, medium, high), and rate of submission of assignments as a number, or as a percentage. These range variations and units of measurement may present difficulties in analyzing these features together, in that some variables may possibly overshadow or bias analysis just because of their range. Z-score Standardization is done in order to make these features of behavior comparable across subjects. The operation transforms all features in such a manner that the distribution has a mean of 0 and a standard deviation of 1. The standardization equation is available in Eq. (3):

$$x_i' = \frac{x_i - \mu}{\sigma} \quad (3)$$

where, x_i' is the original value of the feature of the i-th data point, x_i is the standardized value of the feature of the i-th data point, μ is the average of each student's feature, and σ is the parameter's variance of all students.

4) Applying SMOTE for class imbalance: After the features (X) and labels (y) have been preprocessed, it is time to use SMOTE to solve the problem of class imbalance in the training dataset. The SMOTE process involves creating synthetic examples for classes with underrepresented data to ensure balanced training data. This not only helps the model but also directs the model's attention towards the minority classes in terms of predictions.

C. Temporal Feature Extraction

Multi-scale temporal sequences are used to model student performance to learn both short-term dynamics as well as academic growth over time. Within-semester time series, including weekly variation in academic scores, attendance, and

engagement, indicate instant learning behavior, whereas semester-to-semester variations indicate the long-term signals of performance change, credit accumulation, and academic status shift. Temporal trends are measured in terms of the rate of growth in performance between successive time points, which denote an improvement, stagnation or reduction in academic performance. The student trajectories can be harmonized over long durations of education, and analyzed on a scale at this dual level of time-modelling. The growth rate in performance is expressed in Eq. (4):

$$\text{GrowthRate}_t = \frac{\text{Score}_t - \text{Score}_{t-1}}{\text{Score}_{t-1}} \times 100 \quad (4)$$

where, Score_t is the student's performance (e.g., exam score, assignment score, or overall grade) at time t (current period), Score_{t-1} is the student's performance at the previous time point (e.g., previous exam, week, or term), The result is multiplied by 100 to express the growth rate as a percentage.

D. Model Design Hybrid CNN and LSTM

It is based on an advanced deep learning model to classify student performance based on spatial and temporal data. CNN identifies spatial patterns with the help of such features as scores, attendance, and behavior, whereas LSTM identifies temporal changes over time. The CNN and LSTM output features are sent through a fully connected layer to acquire abstract patterns and features. Last, the output layer relocates the students to either low, medium, or high risk depending on the patterns learnt. Fig. 2 indicates the layout of the hybrid CNN and LSTM.

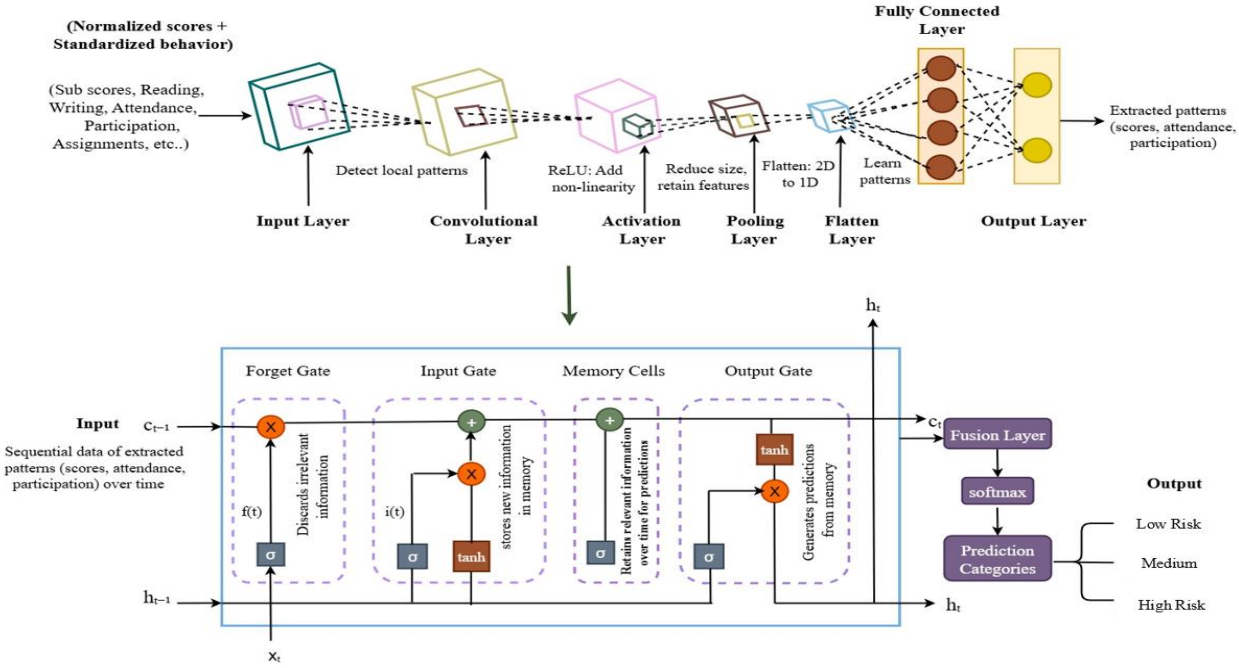

Fig. 2. Architecture of a hybrid CNN and LSTM.

1) Convolutional Neural Network: CNN identifies spatial patterns in the input features of performance grades of students, attendance, and participation. The CNN scans the input data to find significant patterns, including attendance-performance correlations or correlations between subjects, by applying the application of filters (kernels). These patterns are then captured in the form of feature maps and further processed by the activation and pooling layers so that only important features are stored, and complexity is minimized such that they may be utilized in the final classification and analysis.

a) Input layer: The first CNN layer operates on features such as academic scores, attendance, participation, and other behavioral traits that have been normalized. As a means to the end of predictive power, the model can also, alongside other variables like student demographics, LMS activity logs, and so forth, login frequency and time-on-task information, to mention a few, the model can also give more precise prediction outcomes by incorporating these features since they are often indicators of academic outcomes and thus provide a clearer view of the learning behavior. This additional input not only increases the feature space but also makes the model more robust by ensuring that it captures the patterns that perhaps only the grades alone would have otherwise missed. This enhancement makes the model more inclusive, leading to improved student performance prediction reliability.

b) Convolutional layer: The convolutional layer calculates the input data through filtering (or kernels). A filter is a small matrix that is passed over the input data (a sliding window) and multiplies and adds the products element-wise. This is used to identify significant spatial patterns of the input data, i.e. subject performance patterns or attendance patterns. This may be formulated mathematically as Eq. (5):

$$F(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} I(i+m-1, j+n-1) \cdot K(m,n) \quad (5)$$

where, $F(i,j)$ is the output feature map at position (i,i), $I(i,j)$ is the input data (score/attendance at position i,j), $K(m,n)$ is the filter used to identify patterns, M and N are the

filter's dimensions. After passing through all of the input data, the filter generates feature maps that highlight several significant features, such as attendance trends or scoring patterns.

Another way to improve model efficiency is by using a 1D-CNN structure instead of a conventional 2D-CNN. A 1D-CNN is highly applicable for time-oriented academic records like scores, attendance, or behavioral patterns because it processes one-dimensional data through its filters. Compared to 2D-CNN with its image-like grids, a 1D-CNN is concerned with the temporal aspect of the data, thus using fewer parameters and consequently incurring lesser computational cost. Not only does this lighter architecture manage to capture important trends in student performance, but it also enhances training speed and efficiency of the whole model. Such an alternative provides an additional CNN design more consistent with the temporal nature of academic data.

c) Activation Layer (ReLU): The convolution operation is followed by the activation layer (usually ReLU) that adds a nonlinear behavior to the network needed to understand intricate patterns. ReLU is just a mere conversion of the negative values into zero to enable the network to concentrate on positive values to accelerate training. This working can be mathematically expressed, as in Eq. (6):

$$f(x) = \max(0, x) \tag{6}$$

When the input value x is negative, ReLU will make it 0; when the value is positive, it does not change. This aids the CNN to learn and identifying more elaborate patterns since it gets to erase the adverse influences that may not be significant.

d) Pooling layer: The pooling layer will have the responsibility of down-sampling both the spatial height and width of the feature maps produced by the convolutional layer without eliminating the larger features. Max pooling is the most common technique, where the maximum value of a small region of the feature map is selected with the aim of minimizing the amount of data at the expense of the most valuable data. This operation can be expressed in Eq. (7):

$$P(i, j) = \max_{m,n}\{F(i + m, j + n)\} \tag{7}$$

where, $P(i, j)$ is the pooled value at the location (i, j), $F(i + m, j + n)$ is the feature map generated from the preceding layer. The max pooling operation effectively sums up the region it scans, without losing the essential detail, as it compresses the data.

e) Flatten layer: The flatten layer transforms the 2D feature maps acquired following the pooling to 1D vectors. This is due to the requirement of the next fully connected layers of having the input to be a 1D array. The flatten operation is merely the transfer of the matrix of the pooled features to the one-dimensional features in the form of a dense layer.

f) Fully connected layer: The one-dimensional vector is flattened and then sent to the fully connected layer, where every node is connected to every other node in the same layer. This learns complex, high-level features, which combine data from the CNN spatial feature extraction. In essence, it learns predictive patterns of student attainment, taking into consideration all the features the CNN has identified.

This layer is capable of learning the correlation of the features extracted, i.e. how one subject in the performance of one class may also be correlated with the engagement in some other activity. In order to give final predictions, the dense layer makes a weighted summation of inputs and feeds them through an activation function.

- Output Layer: The learned features of the fully connected layer are fed into the Output Layer, which converts them into output. It converts the characteristics into patterns extracted, such as scores, participation, and attendance. The prediction patterns are the predictive ones and are the higher aspects of student behaviors and performances.
- Ablation of CNN:
  - To check how well the model's performance is without spatial feature extraction, eliminate or comment out the layers related to CNN.
  - Evaluate the model after removing the CNN in terms of accuracy and other performance metrics.

2) Long Short-Term Memory: The LSTM networks are based on the RNN; they can take in sequential data and therefore can be applied in situations where time matters in order of data, such as the case of tracking students over time. LSTMs can recall the relevant information over a long period of time, forget the irrelevant information, and change the state of memory using a combination of gates to control the flow of data. These characteristics enable LSTMs to identify more advanced patterns of time-series data and also provide predictions based on past input.

This is because the LSTM takes sequential data in the shape of normalized scores (i.e., mathematics, reading, writing) and standardized behavioral scores (i.e., attendance, participation, assignment completion). All these data points are synchronized across several time steps, which enables the learning of the LSTM in order to determine how the performance and behavior of the students change. The time series nature of the data allows the model to measure the patterns and trends with time. This feed will be helpful when estimating the future performance and categorizing student performance.

a) Forget gate: The forget gate of the model then determines which part of the information at the previous step is to be forgotten. It does this by generating values between 0 and 1 of every value in the memory cell. These values determine the extent of the information of the past that will be required to make the current prediction, and which will not be needed and will be forgotten. The forget gate can be defined as in Eq. (8):

$$\mathrm{f_t} = \sigma(\mathrm{W_f} \cdot [\mathrm{h_{t-1}}, \mathrm{x_t}] + \mathrm{b_f}) \tag{8}$$

where, f_t is the forget gate output (which determines which portion of the previous memory to forget), The linear activated function is denoted by σ, which maps the output between 0 and 1, W_f is the forget gate's strength array, h_{t-1} is the hidden state

from the previous time step, x_t *is the* input data at the current time step, b_f is the bias term for the forget gate.

The forget gate applies the logistic activation function to produce outputs in the range of 0 to 1; 0 signifies forget everything, and 1 remember everything. This makes the model retain only the relevant past data to make predictions of the future and forget the non-relevant portions of the memory.

b) Input gate: Based on the forget gate, the input gate identifies the new information that has to be added to the memory cell. It creates a perfect memory cell update with the help of a tanh function and is based on a sigmoid equation to determine what values should be modified. This will enable the LSTM to add new useful information, without forgetting the most useful past environment. Input gate output, as shown in Eq. (9):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (9)$$

where, i_t is the input gate output (which determines which parts of the input will be updated in the memory), W_i represents the input gate's weight matrix, h_{t-1} symbolizes the previous time step's concealed state, x_t refers to the current time step's input data, b_i is the input gate's bias term, σ stands for the logistic activation function.

The sigmoid function provides the values of 0 to 1 to denote the percentage of the new data that is to be considered as relevant to be stored in the memory, and the tanh function provides the values that can be stored in the memory cell; it is normalized and added to the memory cell.

c) Memory cells: The memory cell includes valuable data in the long term after the input gate. It does it by combining the last memory state and the input candidate memory, multiplied by the input gate output. The current memory cell stores information that is useful in the next time step, and the LSTM can remember the useful information of the past and apply it in the next prediction. The memory cell adds the previously transferred memory of the previous time step, C _(t-1) and the current time step input C^t. This combination is done in such a way that only important data will be retained in the memory cell to be utilized again in the future, and the model has the ability to retain important patterns but forget information that is not very critical with time.

d) Output gate: The output gate of the LSTM determines which data can be traced out of the memory cell to form the final secret state (or output). It then transfers the result over a tanh activation process after computing an activation based on the current state of the memory cell. The result enhances the outcome of the gate activation and dictates a carryover of the concealed state to the next time period. When the input has gone through the output gate, the data is fed into the LSTM output, which is then used to combine the hybrid model output, and this is done through a fusion layer.

e) Attention mechanism: The LSTM, although it does capture the temporal dependencies, still suffers from the fact that not all time steps are equally important for predicting the student's performance. By introducing the attention mechanism after the LSTM layer, this problem is solved. The attention layer looks at all the hidden states that have been generated throughout the time sequence and gives each time step a weight according to its importance. In this manner, the model will be able to concentrate more on the pivotal moments in the student's behavior like a sudden drop in attendance or a big shift in academic scores at the same time, it will be able to reduce the impact of the less relevant periods. The attention score for each time step t is expressed in Eq. (10):

$$e_t = v^T \tanh(W_h h_t + b_h) \quad (10)$$

where, e_t is the raw attention score for time step *t*, h_t *is the* LSTM hidden state at time step *t,* W_h is the Weight matrix that transforms the LSTM hidden state for attention scoring, b_h is the bias term added during the transformation, v^T is the trainable weight vector used to convert the transformed hidden state into a single importance score, tanh is the activation function that introduces nonlinearity. The incorporation of attention eventually results in a model that is more stable, more accurate, and more interpretable. In this case, the educators can now know the periods that had the most influence on the prediction of low, medium, or high risk. The raw scores are then converted into attention weights through the application of a softmax function, which is presented in Eq. (11):

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T} \exp(e_k)} \quad (11)$$

where, α_t is the attention weight for time step *t,* e_t is the raw attention score at time step t, $\sum_{k=1}^{T} \exp(e_k)$ is the Normalization term that sums the exponentiated scores of all *T* time steps, *T* is the Total number of time steps in the sequence. The aggregated data is then passed to the softmax layer, which determines the probability distribution of all possible outcomes. In order to enhance the efficiency and prevent overfitting of the hybrid CNN–LSTM model, regularization techniques are integrated into the training process. The methods make the model less noise-sensitive and thus improve its capability to generalize to new student data.

- Dropout is implemented between the layers that randomly turn off a fraction of the neurons. This will help to avoid the network from depending excessively on specific nodes, and also, to make it learn more powerful and shared representations.
- Batch Normalization is utilized to control the distribution of layer activations, thereby yielding smoother and quicker training. It lessens internal covariate shift and aids the model to stay strong even when faced with variations or noise in the behavioral and academic inputs.

Dropout and batch normalization together significantly bolster the hybrid model's robustness, reduce the chances of overfitting, and deliver more accurate classifications of students into the categories: low, medium, and high risk. Finally, but not least, the SoftMax output is employed to generate the predicted categories, which determine the performance or risk of the student. All these assist the LSTM to predict dynamically and time-dependently with the aid of sequential pattern (LSTM) and spatial information (CNN).

The hybrid CNN-LSTM system can be applied to actual circumstances in universities by integrating student information

systems and learning management systems data without making any changes to the fundamental design. The indicators specific to the university, including credit accumulation, GPA/CGPA threshold, course failures, academic probation, and graduation eligibility, are modeled as structured inputs, and academic and credit patterns as spatial features represented with CNN, and semester-based progression and learning behavior as temporal sequences with the standard academic calendars. To aid in transparent decision-making, explainable AI methods are used to uncover the most significant academic and behavioral predictors to make accurate classifications of the probability of probation, course failure, and delayed graduation risks as low, moderate, and high-risk risks and inform academic advising.

- Ablation of LSTM:
  - To discover how the model reacts without the LSTM layers involved in temporal sequence modeling, either remove them or comment them out.
  - This will indicate how much LSTM contributes to the capturing of time dependencies in the data.

TABLE I. STUDENT PERFORMANCE PREDICTION CATEGORIES

Category	Description
Low Risk	According to the model, the pupil will probably do well and be successful.
Medium	According to the model, the pupil is likely to struggle and encounter obstacles.
High Risk	The model predicts a high risk of the student dropping out.

Table I of the predicted categories shows the input categories of the model, which involve using CNN and LSTM. The models include: low risk, in which the model student performs well and is an academic success; medium, which refers to the fact that the student will struggle and he/she may require more help to be a better student; and high risk, which refers to students who are at a high risk of disengagement and can drop out. The predictions will be made based on trends within the space observations (e.g., scores, attendance) and time trends (e.g., behavior over time), which makes the model categorize the students into such groups to provide them with an individualized intervention and guidance. Algorithm 1 shows hybrid deep learning for academic achievement prediction.

Algorithm 1

Input:

1. Spatio-Temporal Data:
   - Normalized Academic Scores: Scores from various subjects (e.g., mathematics, reading, writing).
   - Standardized Behavioral Features: Attendance, participation, assignment completion rates, etc.
   - Time-Series Data: Data over multiple time steps (e.g., weeks, terms).

Output:

- Predicted Categories:
  - Low Risk: The student is likely to perform well.
  - Medium: The student is likely to face academic challenges.
  - High Risk: The student is at risk of dropping out.

Step 1: Data Preprocessing

1. Normalization:
   - Apply Min-Max normalization to academic scores to bring them to a comparable range (e.g., [0, 100]).
   - Apply Z-score standardization to behavioral features like attendance, participation, and homework completion.
2. Handle Missing Data:
   - Use techniques such as mean imputation or data interpolation to handle missing values in the dataset.

Step 2: Convolutional Neural Network for Spatial Feature Extraction

1. Input Layer:
   - Input normalized academic scores and standardized behavioral features as multi-dimensional data.
2. Convolutional Layer:
   - Find geographic patterns in input data, use filters (kernels) (e.g., relationships among scores and attendance).
3. Activation Layer (ReLU):
   - Apply the ReLU activation function to introduce nonlinear behavior.
4. Pooling Layer:
   - Minimize spatial dimensions without sacrificing important features, use max pooling.
5. Flatten Layer:
   - In order to use the feature maps from the pool as inputs for the fully linked layer, flatten them into a 1D vector.
6. Fully Connected Layer:
   - Learn complex patterns from the extracted features and pass the learned features to the next stage.

Step 3: LSTM for Temporal Modeling

1. Input Layer:
   - Sequential data of extracted features (scores, attendance, participation) over time is passed into the LSTM model.
2. Forget Gate:
   - Based on its applicability to future projections, the forget gate determines whether data from previous steps should be ignored.
3. Input Gate:
   - To guarantee that only pertinent data is retained, the input gate determines which fresh data must be put to the memory cell.
4. Memory Cell:
   - Memory cells store the relevant information from the current and previous time steps. On the basis of the input and forget gates, they update.
5. Output Gate:
   - The output gate passes the current input through an activation mechanism and generates the outcome depending on the modified memory cell.

Step 4: Feature Fusion Layer

1. Fusion Layer:

- Combine the outputs of the CNN (spatial features) and LSTM (temporal features) to form a unified representation.

Step 5: Final Prediction Layer

1. Fully Connected (Dense) Layer:
   - This layer learns from the fused features and classifies the student risk into one of the categories: low, medium, or high risk.
2. Softmax Layer:
   - Utilizing the softmax activation function, a distribution of chances across the categories is generated.

Output:

- The ultimate classification is determined by using the softmax output (predicted categories: low, medium or high).

IV. FOX OPTIMIZATION

The FOX optimizer (Fractional Order optimizer with Exponential Convergence) is a superior optimization method that optimizes the rate of convergence and accuracy of deep learning models. Unlike the traditional optimizers, FOX uses fractional-order derivatives to learn how to update the learning rate. The novel approach to gradient calculation adjustment with the use of fractional exponents has been utilized in the novel method to enhance the stability and efficiency of deep learning models during training. FOX accelerates convergence by the fact that it addresses the problems of the local minima and sluggishness of training, which are intrinsic to the classical optimization methods. FOX is an optimal and more effective way of updating parameters since it employs fractional orders in updating the gradients. The most essential thing about the FOX optimizer is its two-step process, which is gradient update and fractional-order scaling. The first step is the formula of gradient update that utilizes the derivatives of the model parameters in a fractional order to update them. This is represented in Eq. (12):

$$\theta_t = \theta_{t-1} - \alpha_t \cdot g_t^{\gamma} \tag{12}$$

where, θ_t is the parameter at time step t, α_t is the learning rate at time t, g_t is the gradient at time t, and γ is the fractional order used to scale the learning rate. This update would speed convergence, scaling the gradient with a fractional exponent, which gives more accurate parameter updates. Step 2 is fractional-order scaling, in which the gradient scale is scaled by its L 2 norm. This action keeps the amount of the update proportionate to the importance of every parameter, enhancing the efficiency of the optimizer with complex, multi-layer models like CNN + LSTM. The fractional-order scaling takes the form of Eq. (13):

$$\theta_t = \theta_{t-1} - \alpha_t \cdot (\|g_t\|^{\gamma}) \tag{13}$$

where, $\|g_t\|$ represents the L2 norm of the gradient at time t, making sure the optimizer modifies the learning rate in accordance with the gradient's strength. The fractional order γ maximizes the value of each gradient to its optimal value, thus allowing the optimizer to respond better to the dynamic gradient with time.

A. Comparative Analysis: FOX vs. Adam Optimizer

The comparison with the Adam optimizer, which is one of the most popular deep learning optimizers, is sought for a better understanding of the advantages of FOX. Adam makes use of momentum and adaptive learning rates derived from first and second-order moment estimates to update parameters. This results in fast convergence and allows Adam to deal with noisy gradients effectively.

FOX is significantly different from Adam in two main aspects. Firstly, FOX employs fractional-order gradient scaling, which gives more flexible and accurate control over parameter updates, in contrast to Adam, which uses fixed exponential averages. Secondly, FOX updates its step size according to the L2 norm of the gradient, which enables it to breathe life into flat regions and slow-converging plateaus that often hinder Adam. Consequently, the path to convergence for FOX could be shorter and its reliability higher, especially with the intricate hybrid structures such as CNN + LSTM.

Nonetheless, the Adam algorithm is still a robust reference point mainly due to its ease of use, less intensive computations, and wide acceptability through its performance in various tasks. The comparison of both optimizers simultaneously uncovers the learning dynamics, convergence behavior, and robustness of the model under consideration more clearly.

B. Ablation of FOX

- Detect the absence of FOX and install the simpler optimizer (Adam), then check if the removal of FOX resulted in the slowing down of convergence and the reduction of model stability.
- The above-mentioned assessment will facilitate determining whether or not the tailor-made FOX optimizer is a necessity or if a standard optimizer like Adam can yield the same result.

V. RESULTS

The outcomes of the proposed hybrid CNN and LSTM structure are provided in this section. The architecture was written in Python with popular deep learning libraries based on TensorFlow and Keras being used to perform the computations. Training and testing data entail scholastic achievement information of the students, as well as normalized scores on subjects like mathematics, reading, and writing, and standardized behavioral characteristics like participation, attendance, and submission of assignments. The data is spread across the time steps, depicting the performance of the students on several terms or weeks of the year, to enable the model to learn the spatial and temporal pattern of knowledge. Subsequent parts include general performance metrics, including the rate of correctness and an exactness measure, as well as the rate of sensitivity and the balance score and graphical depictions such as a confusion matrix, ROC curve and precision-recall curve. The results demonstrate how the framework can be used to make predictions of the student performance categories, which explains its effectiveness in a classroom setting.

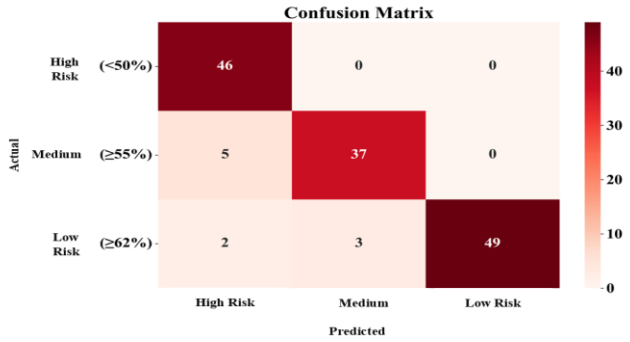

Fig. 3. Confusion matrix.

The accuracy of the model in predicting the performance of pupils is measured using the confusion matrix in Fig. 3. It highlights the differences between risk categories (low, medium, and high risk) and projected categories based on the model's projections. True positives are observed on the diagonal boundary, such that the model made the right prediction as to the respective sets of student performance. The off-diagonal values are poor predictions where the model predicted one category in another one. In general, there is excellent predictive confidence in the low and medium risk groups that is represented in the matrix.

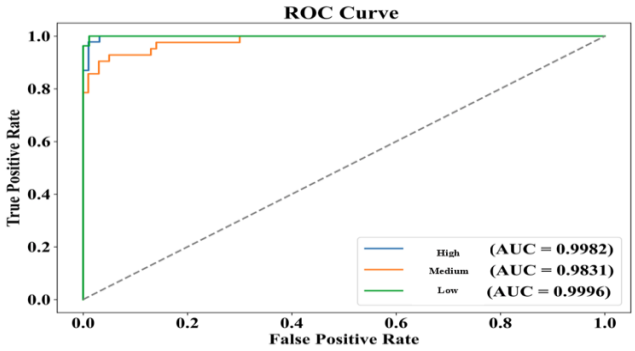

Fig. 4. ROC curve.

In the Fig. 4 of the ROC curve, the capacity of the framework in predicting the result of the pupil in three groups, that is, high risk, medium, and low risk, is evident. The trade-off in the curve is the true positive rate (sensitivity) and false positive rate (1-specificity) for each group. Accuracy is very high with the area under the curve of the model being 0.9982 in the case of high risk, 0.9831 in the case of medium and 0.9996 in the case of low risk, demonstrating that the model is a fantastic one in the differentiation of the different classes of students. The green curve that provides the best AUC is the Succeed curve, and it has high predictive power on the set of students who will succeed, although the other two classes also have good predictive power.

The model is described by a plot, which shows its training and validation accuracy at various epochs in Fig. 5. The accuracy of training is blue and the accuracy of validation is orange. The training and validation accuracies increase with the number of epochs, and the training accuracy is always equal to 1.0. The validation accuracy also increases significantly but unequally, which may be a manifestation of overfitting or minor variation in the performance of the training and the validation sets. Overall, the model has justifiable positive results in learning the plot and good generalization between epochs.

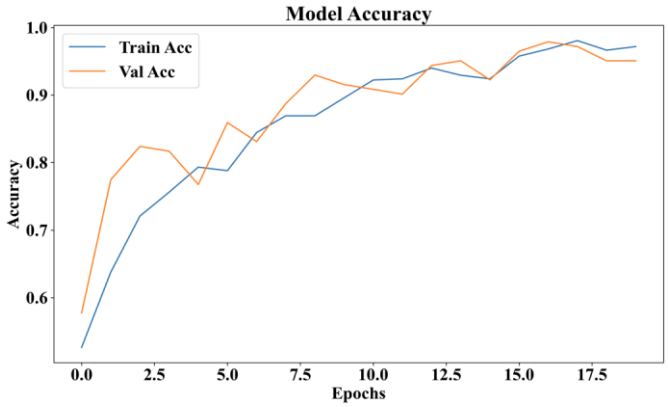

Fig. 5. Model accuracy.

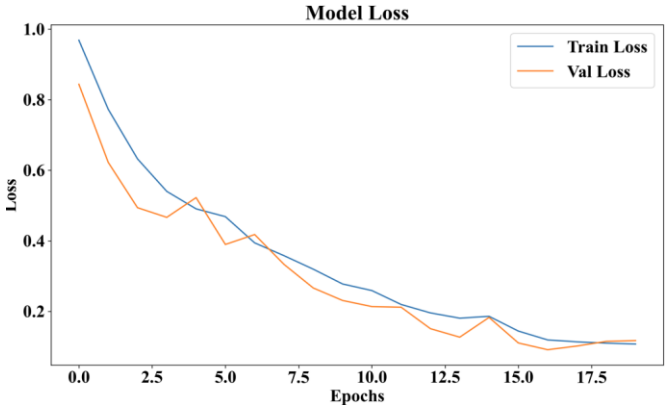

Fig. 6. Model loss.

In Fig. 6, the plot of the training and validation loss of the model at various epochs is presented. Blue and orange lines represent the training and validation loss, respectively. With the increase of the epochs, the losses become less and less, which means that the model learns and minimizes the errors. However, loss on validation hits some point, and this signifies that the model is generalizing very well. This indicates that the score of the verification set remains constant in the course of training, and this means that training is efficient and there is minimal overfitting, although the algorithm itself is still learning.

A. Performance Analysis

The results of the performance analysis section indicate how efficient the proposed Hybrid CNN and LSTM model is in terms of predicting the academic success of students. Fig. 7, Fig. 8, and Fig. 9 show important performance indicators that are the False Negative vs. False Positive Rates and the Precision-Recall curves and the overall performance of the model that has high precision, recall, and accuracy. The outcome demonstrates that the model has a high degree of accuracy in predicting different categories of students, and it is superior to regular models such as LSTM and SVM. The better results of an enhanced deep learning model using various evaluation measures are also confirmed in Table II, confirming its use to predict with high precision in academic studies.

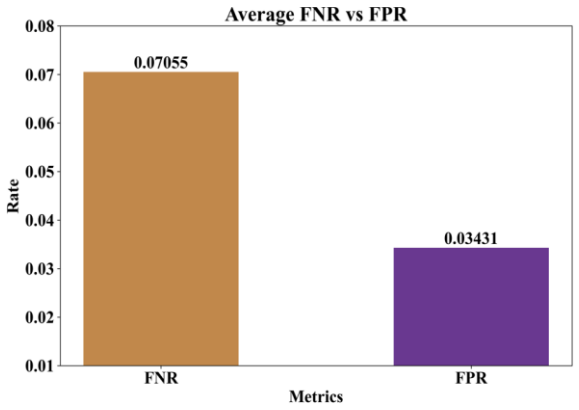

Fig. 7. Average FNR vs. FPR.

In Fig. 7, the average predictions of the model of FNR versus FPR are compared using the bar chart. FNR has a brown bar with a measure of 0.070555, and FPR has a purple bar with a measure of 0.03431. As indicated in the graph, the model creates a greater FNR than the FPR, which implies that the model incorrectly classifies students as not being at risk more than the rate of the inaccurate classification of the students as being at risk.

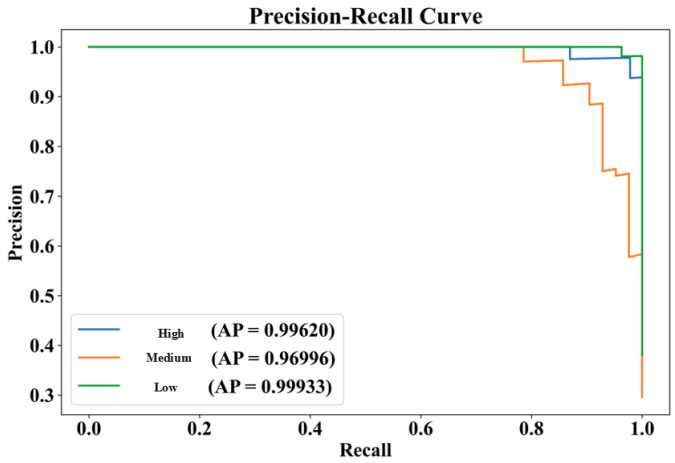

Fig. 8. Precision-recall curve.

The model performance for the different classes: high risk, medium, and low risk is shown in precision-Recall curves in Fig. 8. Each curve will indicate the precision-recall trade-off of each class as well as the area under the curve, which will indicate the performance of the model. The highest AP belongs to succeed 0.999333, which gives a high recall and precision, followed by 0.99620 that gives high recall but low precision, and 0.96996 that gives low recall but high precision. These values reflect the fact that the model predicts successful pupils with phenomenal precision, and is fairly good at predicting dropouts, but somewhat reduced in the medium prediction.

The performance values of the hybrid model are excellent results of the performance on several performance metrics in Fig. 9. The prediction of the model was 97.18 per cent, which is a good overall result. Precision and recall are also quite high at 97.02 and 97.35, respectively, which shows that the model did not misclassify the positive instances and had the lowest number of FPR and FNR. F1-score is also 97.14, which indicates a recall and precision balance. These results indicate the predictive ability of the model for student success in the three risk levels.

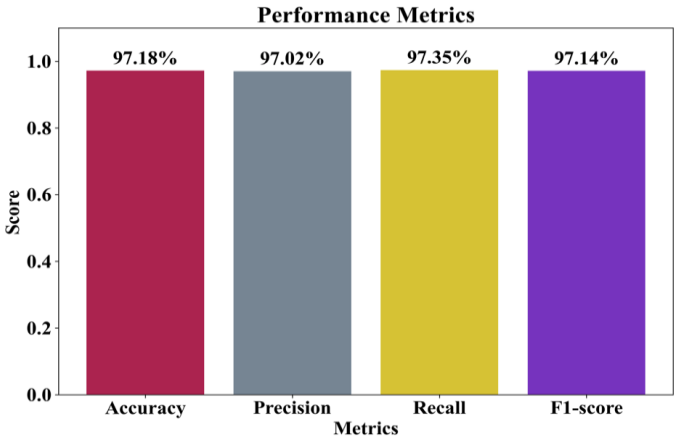

Fig. 9. Performance metrics.

B. Model Explainability with SHAP

In order to improve the interpretability of the model, utilized SHAP to identify the important factors that influence the predictions. SHAP values reveal the contribution of each feature (e.g., academic scores, attendance, participation) to the decision of the model for a particular prediction. By attributing the prediction to the contributions from each feature, SHAP provides the reasoning behind the prediction of a student risk level as high, medium, or low.

Through the application of SHAP values, it is possible to notice that attendance and participation are the most determining factors in the prediction of the "medium" category. A student might be predicted as a "Struggler", and the prediction may be influenced by behavioral factors like low participation, even though the student has high academic scores. SHAP makes it clear and transparent. The force plot below shows the contributions of features for the student, who has been predicted as a Struggler, individually. The contributions that are positive push the prediction in the direction of "medium", while the negative contributions retract it from the prediction. Such insights are especially helpful for the education sector as they not only provide practitioners with an understanding of the predictions of the model but also enable them to customize interventions according to the particular reasons that determine a student's performance.

TABLE II. PERFORMANCE EVALUATION TABLE

Method	Accuracy	Precision	Recall	F1-Score
LSTM [33]	87	87	88	88
SVM [34]	82	93	84	89
GRU	92.34	91.00	90.30	90.65
Bi-LSTM	93.50	92.10	91.50	91.80
Transformer	94.75	94.00	93.90	93.95
TCN	95.20	94.60	94.20	94.40
Hybrid CNN and LSTM	97.18	97.02	97.35	97.14

Table II provides a detailed view of how different models, including LSTM, SVM, GRU, Bi-LSTM, Transformer, TCN, and the proposed Hybrid model, performed compared to one another. The Hybrid CNN and LSTM model got the best performance ever recorded with the highest accuracy (97.18 per cent) and also surpassed all remaining models according to precision, recall, and F1-score. Additionally, the two models, i.e., Transformer and TCN, in this case, have exhibited great capability, especially when it comes to long-term dependency issues, as their maximum accuracy reached 94.75 per cent and 95.20 per cent, respectively. On the other hand, while GRU and Bi-LSTM have yielded close results, the hybrid model is still the most powerful in predicting students' performance.

VI. Discussion

The outcomes from the comparison with the models such as GRU, Bi-LSTM, and those based on Transformer and TCN kept confirming that the hybrid CNN-LSTM-FOX model is superior to traditional models like LSTM and SVM in terms of accuracy, scoring a remarkable 97.18 per cent. The hybrid model has been very effective in combining the strengths of both CNN and LSTM, enabling the former to extract spatial patterns while the latter recognizes temporal ones, eventually providing a very holistic understanding of student performance. In the case of models like Transformer and TCNs, they offer certain advantages over LSTM in capturing long-term dependencies and processing sequential data, but still, the hybrid CNN-LSTM-FOX model, when judged strictly on performance, is the best. Conversely, GRU is less accurate than LSTM and the hybrid model, but is still a good choice in resource-constrained environments due to its simpler architecture and lower computational time. In general, the findings indicate that the hybrid model with FOX is always superior to comparative architectures as it successfully learns both spatial and temporal learning patterns. Besides higher education, the structure can be easily incorporated into the institutions of higher learning where the academic progression is stipulated by credit accruals, GPA/CGPA and prerequisites. It is appropriate in academic risk identification in undergraduate and postgraduate programs due to its capability to model heterogeneous and semester-wise academic trajectories. The academic risk is determined based on the higher-education outcomes, like course failure, GPA/CGPA, academic probation, and delayed graduation, which allows grouping low-risk, moderate-risk, and high-risk groups.

A. Focused Error Analysis

While the hybrid CNN-LSTM model exhibits a robust performance, it continues to be necessary to take care of the situations in which the model performs poorly. A concentrated error analysis indicates that the model is particularly in trouble when it comes to students who are academically very good but have low attendance, resulting in misclassification between "low" and "medium". The model often makes these errors with the pupils who occasionally attend classes, and hence, the model might not be able to pick up on this engagement of the student's very well. Furthermore, sometimes the students with irregular behavior patterns are wrongly classified, which implies that the model is not able to grasp the full complexities of behavioral data over time. The above-mentioned weaknesses signal the areas for possible model refinement. Later on, one of the approaches could be developing the data preprocessing, for instance, through better management of attendance and participation data or combining it through more sophisticated features that track student engagement over time. Without doubt, the model that is more sensitive to the mentioned factors could lead to a more accurate and reliable prediction system.

B. Practical Implications

The proposed framework is a kind of early warning mechanism allowing academic advisors and instructors to recognize students who are at risk at the early stage of their academic life. Predictions on the model are converted to risk scores that can be acted upon and early warning groups to enable advisors to prioritize students who need urgent attention. Advisors can use explainable outputs to emphasize important academic and behavioral risk factors (such as a drop in GPA, inadequate credit accumulation, failure in courses, low attendance, and decreased LMS engagement) to develop specific interventions, such as tutoring, mentoring, probation support, or individualized learning plans. This is a timely and data-driven decision support which improves retention of students, efficient use of institutional resources and also an ongoing monitoring of the student progression within the university academic systems.

To guarantee scalability and generalization, the proposed framework will be tested on a variety of colleges and academic subjects to take into consideration the differences in the curriculum structure, grading policy, and credit requirements. The cross-program evaluation helps to evaluate the model in terms of its robustness and transferability to prove that it can be applied not only in one institution or discipline.

C. Ethical and Privacy Considerations

To enable the adoption of responsible deployment in the context of universities, the framework will utilize ethical and privacy-sensitive tools, such as role-based access control, data minimization, and bias monitoring. There is restricted access to student data based on institutional roles (e.g., advisors, instructors), and prediction is based on only necessary academic and behavioral characteristics. Also, the results of predictions are reviewed periodically within demographic and academic subgroups to address and reduce possible bias to secure, equitable, and reliable decision support of academic advising.

VII. Conclusion and Future Work

The study introduces a hybrid CNN-LSTM that is used to determine the student risk level results of being low, medium, or high. CNN is used to process student data, such as academic grades and personal attributes, to identify spatial patterns, whereas LSTM is used to track progress over semesters. The model was found to have an accuracy of 97.18 per cent and was better than other traditional models like LSTM and SVM; the interpretability was also guaranteed through SHAP to determine the key predictive variables. It could be assessed as an early-warning system that helps teachers to notice at-risk students and intervene timely. The framework can apply to a wide range of educational contexts with the introduction of university-specific characteristics such as credit systems, GPA/CGPA, prerequisites, and semester-wise progression, although tested on a single-institution dataset, which limits the scope of

generalization of the results. Further studies could involve larger and multi-institutional groups of data and other socio-economic and learning-environment variables to further enhance the predictive strength. The framework consists of ethical protections like role-based access, data minimization, and bias monitoring that can be deployed in a higher education system in a privacy-aware and responsible manner.

The proposed framework will be tested in the future on a variety of colleges and academic fields to determine its strength and cross-program extensibility in the context of different higher education institutions.

DECLARATIONS

Data Availability: https://www.kaggle.com/datasets/mahmoudelhemaly/students-grading-dataset

REFERENCES

[1] Y. A. Alsariera, Y. Baashar, G. Alkawsi, A. Mustafa, A. A. Alkahtani, and N. A. Ali, "Assessment and evaluation of different machine learning algorithms for predicting student performance," Computational Intelligence and Neuroscience, vol. 2022, no. 1, Art. no. 4151487, 2022, doi: 10.1155/2022/4151487.

[2] Y. Wan, R. Li, W. Li, and H. Du, "Impact pathways of AI-supported instruction on learning behaviors, competence development, and academic achievement in engineering education," Sustainability, vol. 17, no. 17, p. 8059, Jan. 2025, doi: 10.3390/su17178059.

[3] Q. Meng and Q. Zhang, "The influence of academic self-efficacy on university students' academic performance: The mediating effect of academic engagement," Sustainability, vol. 15, no. 7, p. 5767, 2023, doi: 10.3390/su15075767.

[4] P. Gui, G. M. Alam, and A. B. Hassan, "Whether socioeconomic status matters in accessing residential college: Role of RC in addressing academic achievement gaps to ensure sustainable education," Sustainability, vol. 16, no. 1, p. 393, Jan. 2024, doi: 10.3390/su16010393.

[5] N. Yavuzalp and E. Bahcivan, "A structural equation modeling analysis of relationships among university students' readiness for e-learning, self-regulation skills, satisfaction, and academic achievement," Research and Practice in Technology Enhanced Learning, vol. 16, no. 1, p. 15, 2021, doi: 10.1186/s41039-021-00162-y.

[6] X. Tang, M. T. Wang, F. Parada, and K. Salmela-Aro, "Putting the goal back into grit: Academic goal commitment, grit, and academic achievement," Journal of Youth and Adolescence, vol. 50, no. 3, pp. 470–484, Mar. 2021, doi: 10.1007/s10964-020-01360-1.

[7] Y. Baashar, G. Alkawsi, A. Mustafa, A. A. Alkahtani, Y. A. Alsariera, A. Q. Ali, et al., "Toward predicting student's academic performance using artificial neural networks (ANNs)," Applied Sciences, vol. 12, no. 3, p. 1289, Jan. 2022, doi: 10.3390/app12031289.

[8] Z. Wu, T. F. Spreckelsen, and G. L. Cohen, "A meta-analysis of the effect of values affirmation on academic achievement," Journal of Social Issues, vol. 77, no. 3, pp. 702–750, Sep. 2021, doi: 10.1111/josi.12460.

[9] E. M. Onyema, K. K. Almuzaini, F. U. Onu, D. Verma, U. S. Gregory, M. Puttaramaiah, and R. K. Afriyie, "Prospects and challenges of using machine learning for academic forecasting," Computational Intelligence and Neuroscience, vol. 2022, no. 1, Art. no. 5624475, 2022, doi: 10.1155/2022/5624475.

[10] Z. Xu, Y. Zhao, B. Zhang, J. Liew, and A. Kogut, "A meta-analysis of the efficacy of self-regulated learning interventions on academic achievement in online and blended environments in K-12 and higher education," Behaviour & Information Technology, vol. 42, no. 16, pp. 2911–2931, 2023, doi: 10.1080/0144929X.2022.2135680.

[11] S. Hussain and M. Q. Khan, "Student-performulator: Predicting students' academic performance at secondary and intermediate level using machine learning," Annals of Data Science, vol. 10, no. 3, pp. 637–655, 2023, doi: 10.1007/s40745-021-00376-5.

[12] E. Goh and H. J. Kim, "Emotional intelligence as a predictor of academic performance in hospitality higher education," Journal of Hospitality & Tourism Education, vol. 33, no. 2, pp. 140–146, 2021, doi: 10.1080/10963758.2020.1843197.

[13] S. Poudyal, M. J. Mohammadi-Aragh, and J. E. Ball, "Prediction of student academic performance using a hybrid 2D CNN model," Electronics, vol. 11, no. 7, p. 1005, Mar. 2022, doi: 10.3390/electronics11071005.

[14] J. S. Ryu, H. R. Chung, B. M. Meador, Y. Seo, and K. O. Kim, "The associations between physical fitness, complex vs simple movement, and academic achievement in a cohort of fourth graders," International Journal of Environmental Research and Public Health, vol. 18, no. 5, p. 2293, Mar. 2021, doi: 10.3390/ijerph18052293.

[15] X. Wu and M. Guo, "Higher education expansion and the changing college wage premium in Hong Kong, 1976–2016," Chinese Journal of Sociology, vol. 8, no. 3, pp. 28–47, 2022, doi: 10.1177/2057150X221103221.

[16] Z. Shou, M. Xie, J. Mo, and H. Zhang, "Predicting student performance in online learning: A multidimensional time-series data analysis approach," Applied Sciences, vol. 14, no. 6, p. 2522, Mar. 2024, doi: 10.3390/app14062522.

[17] C. Liu, H. Wang, Y. Du, and Z. Yuan, "A predictive model for student achievement using spiking neural networks based on educational data," Applied Sciences, vol. 12, no. 8, p. 3841, Apr. 2022, doi: 10.3390/app12083841.

[18] O. Nafea, W. Abdul, G. Muhammad, and M. Alsulaiman, "Sensor-based human activity recognition with spatio-temporal deep learning," Sensors, vol. 21, no. 6, p. 2141, Mar. 2021, doi: 10.3390/s21062141.

[19] T. Xu, W. Deng, S. Zhang, Y. Wei, and Q. Liu, "Research on recognition and analysis of teacher–student behavior based on a blended synchronous classroom," Applied Sciences, vol. 13, no. 6, p. 3432, Mar. 2023, doi: 10.3390/app13063432.

[20] M. Yağcı, "Educational data mining: Prediction of students' academic performance using machine learning algorithms," Smart Learning Environments, vol. 9, no. 1, p. 11, 2022, doi: 10.1186/s40561-022-00192-z.

[21] B. Zheng, C. Chang, C. H. Lin, and Y. Zhang, "Self-efficacy, academic motivation, and self-regulation: How do they predict academic achievement for medical students?" Medical Science Educator, vol. 31, no. 1, pp. 125–130, 2021, doi: 10.1007/s40670-020-01123-2.

[22] A. Nabil, M. Seyam, and A. Abou-Elfetouh, "Prediction of students' academic performance based on courses' grades using deep neural networks," IEEE Access, vol. 9, pp. 140731–140746, 2021, doi: 10.1109/ACCESS.2021.3119334.

[23] M. Katagiri et al., "Fine and gross motor skills predict later psychosocial maladaptation and academic achievement," Brain and Development, vol. 43, no. 5, pp. 605–615, May 2021, doi: 10.1016/j.braindev.2020.12.009.

[24] L. Cagliero, L. Canale, L. Farinetti, E. Baralis, and E. Venuto, "Predicting student academic performance by means of associative classification," Applied Sciences, vol. 11, no. 4, p. 1420, Feb. 2021, doi: 10.3390/app11041420.

[25] D. J. Madigan and T. Curran, "Does burnout affect academic achievement? A meta-analysis of over 100,000 students," Educational Psychology Review, vol. 33, no. 2, pp. 387–405, Jun. 2021, doi: 10.1007/s10648-020-09533-1.

[26] P. Jiao, F. Ouyang, Q. Zhang, and A. H. Alavi, "Artificial intelligence-enabled prediction model of student academic performance in online engineering education," Artificial Intelligence Review, vol. 55, no. 8, pp. 6321–6344, 2022, doi: 10.1007/s10462-022-10146-9.

[27] H. Martin, R. Craigwell, and K. Ramjarrie, "Grit, motivational belief, self-regulated learning (SRL), and academic achievement of civil engineering students," European Journal of Engineering Education, vol. 47, no. 4, pp. 535–557, 2022, doi: 10.1080/03043797.2021.1935532.

[28] B. Heppt, M. Olczyk, and A. Volodina, "Number of books at home as an indicator of socioeconomic status: Examining its extensions and their incremental validity for academic achievement," Social Psychology of Education, vol. 25, no. 4, pp. 903–928, 2022, doi: 10.1007/s11218-022-09702-6.

[29] Y. Xiong, X. Qin, Q. Wang, and P. Ren, "Parental involvement in adolescents' learning and academic achievement: Cross-lagged effect and mediation of academic engagement," Journal of Youth and Adolescence, vol. 50, no. 9, pp. 1811–1823, Sep. 2021, doi: 10.1007/s10964-021-01464-3.

[30] M. H. A. Al-Abyadh and H. A. H. Abdel Azeem, "Academic achievement: Influences of university students' self-management and perceived self-efficacy," Journal of Intelligence, vol. 10, no. 3, p. 55, Jul. 2022, doi: 10.3390/jintelligence10030055.

[31] J. K. Singh and A. Kaur, "Is teaching and learning in Chinese higher education classrooms internationalized? Perspectives from international students in China," Higher Education Research & Development, vol. 42, pp. 1283–1297, 2023, doi: 10.1080/07294360.2022.2098325.

[32] V. Macakova and C. Wood, "The relationship between academic achievement, self-efficacy, implicit theories and basic psychological needs satisfaction among university students," Studies in Higher Education, vol. 47, no. 2, pp. 259–269, 2022, doi: 10.1080/03075079.2020.1739011.

[33] Z. S. Hafdi and S. El Kafhali, "A comparative evaluation of machine learning methods for predicting student outcomes in coding courses," AppliedMath, vol. 5, no. 2, p. 75, 2025, doi: 10.3390/appliedmath5020075.

[34] Y. Wang et al., "N-STGAT: Spatio-temporal graph neural network-based network intrusion detection for near-earth remote sensing," Remote Sensing, vol. 15, no. 14, p. 3611, Jul. 2023, doi: 10.3390/rs15143611.

[35] O. N. Akande, M. O. Lawrence, and P. Ogedebe, "Application of bidirectional LSTM deep learning technique for sentiment analysis of COVID-19 tweets: post-COVID vaccination era," Journal of Electrical Systems and Information Technology, vol. 10, no. 1, p. 50, 2023, doi: 10.1186/s43067-023-00082-9.

[36] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint, arXiv:1412.3555, Dec. 2014. Available: https://arxiv.org/abs/1412.3555