International Journal of Computational and Experimental Science and ENgineering
(IJCESEN)

Vol. 11-No.2 (2025) pp. 1683-1692
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Deep Learning Fusion for Student Academic Prediction Using ARLMN Ensemble Model

B.Vaidehi[1*], K. Arunesh[2]

[1]Department of Computer Science Sri S.Ramasamy Naidu Memorial College (Affiliated to Madurai Kamaraj University, Madurai) Sattur – 626 203, Tamil Nadu, India.
* **Corresponding Author Email:** vaipri21@gmail.com - **ORCID:** 0009-0004-0281-3766

[2]Department of Computer Science Sri S.Ramasamy Naidu Memorial College (Affiliated to Madurai Kamaraj University, Madurai) Sattur – 626 203, Tamil Nadu, India.
Email: arunesh_naga@yahoo.com - **ORCID:** 0000-0001-7160-3691

Article Info:

DOI: 10.22399/ijcesen.734
Received : 22 January 2025
Accepted : 17 March 2025

Keywords :

Education,
Academic Prediction,
Student Success,
Recurrent Neural Network,
Long Short-Term Memory.

Abstract:

The realization of accurate student performance prognostication within the educational domain presents a critical capability for the timely implementation of intervention strategies and supplementary support mechanisms. This research proposes the Adaptive Recurrent Logistic Memory Network (ARLMN), an innovative approach for student academic prediction. The ARLMN combines Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) network, and Sigmoid Plus - Adaptive Activation Function(S-AAF). The integrated system achieves an impressive accuracy of approximately 98%. By incorporating these methodologies, this model captures temporal dependencies and patterns in student data, including academic, demographic, and emotional information. Pre-processing involves standardizing features before feeding them into the RNN and LSTM models, which are then combined using S-AAF classifier for robust predictions. Experimental results demonstrate the effectiveness of this approach, achieving high accuracy in forecasting student academic performance. By identifying factors that impact performance, this model empowers educators to proactively intervene and ensure student success.

1. Introduction

The education sector prioritizes achieving quality education, recognizing it as a key driver for reaching the ambitious goals laid out in the Sustainable Development Agenda. Quality education goes beyond mere knowledge acquisition; it fosters critical thinking and problem-solving skills. The research proposes a novel strategy for enhancing learning outcomes by emphasizing the use of advanced techniques and technologies to personalize education for each student. The vast amount of educational data becomes a treasure trove of insights when analyzed through data mining. These hidden patterns shed light on how students learn best. This process contributes to improving the overall quality of education [1]. In education, parents are calling for a comprehensive review of the examination system due to perceived biases and limitations in current evaluation methods. The assessment process is heavily reliant on manual input and access to students' background information, making it susceptible to human biases and external influences. To ensure fair evaluations, there's an increasing demand for a more precise metric that can assess students' academic progress and interactions while addressing inherent biases[2,3,4]. Traditional machine learning models are commonly used for predicting student achievement due to their simplicity and low computational requirements. However, they often fail to fully leverage data information, resulting in suboptimal prediction accuracy. In contrast, deep learning techniques, such as Recurrent Neural Networks [5] and Recursive Neural Networks[6] , offer improved accuracy by handling large datasets and capturing complex feature relationships. This research introduces an innovative approach to predict academic performance by combining RNN, LSTM, and S-AAF algorithms. The methodology combines various techniques to create a more comprehensive and robust solution. Specifically, the model analyzes student behavior patterns for prediction. It achieves this by using an RNN+LSTM architecture, which captures important long-term associations from the dataset. Through hidden units, RNN technique analyzes student data streams and retains past information by incorporating prior state feedback into additional hidden layers. The developed model effectively preserves essential information over extended durations using integrated gates. The proposed method has been validated through experiments, demonstrating its effectiveness in significantly improving the accuracy and reliability of prediction

1.1 Objectives of the paper

- Create an innovative deep learning model called ARLMN for prediction.
- Evaluate and compare the performance of ARLMN model to determine the most effective approach for predicting students' performance.
- Evaluate the ARLMN model using performance metrics.
- Evaluate the performance of proposed model against traditional models.

1.2 Manuscript outline

The paper is explored as follows: Section 2 reviews existing systems, highlighting research gaps. Section 3 outlines the proposed system for predicting learner success. In Section 4, experimental findings and analyses are presented. Finally, Section 5 concludes the paper and discusses future research directions.

2. Literature Review

Deep learning has evolved into a potent machine learning approach, harnessing multiple layers of features or data representations to achieve cutting-edge results. Deep learning is increasingly applied in educational data mining, utilizing statistical methods, data mining, visualization, and machine learning tools to analyze and process data from various sources. The generated study analytics aim to investigate institutional databases, while learning management frameworks interpret information to enhance learning processes and environments. [7,8]. In prediction, the utilization of RNN models, has demonstrated remarkable efficacy. RNNs, prized for their capacity to capture sequential patterns and temporal dependencies within data, have emerged as pivotal tools in accurately forecasting student academic outcomes. Hassan et al. employed LSTM deep model to predict student dropout[9]. M. Wasif et.al referenced various models to forecast based on pass or fail outcomes using demographic information [10].Tomasevic et al. discovered that Artificial Neural Networks excelled in predicting student test results using past performance and interaction data, with demographic factors contributing little to accuracy[11]. Okubo et al. utilized RNN in forecasting student performance and the findings highlight RNN's superior 90% prediction accuracy over traditional regression method[12].Yanbai et al. suggested a hybrid RNN-GRU neural network to forecast students' academic outcomes, utilizing demographic data and assessment records to differentiate between passing and failing [13]. Adnan et al. utilized a Deep Feed Forward Neural Network to predict final outcomes, achieving the highest accuracy of 72% when incorporating all available features [14]. Su et al. employed an RNN model on log data from a learning management system to predict applicants' academic performance, reporting a 90% prediction accuracy [15]. Research across various papers underscores the extensive range of technologies utilized in technology-supported learning. Integration of Deep Learning into Educational Data Mining enables effective handling of diverse educational data, notwithstanding the non-linearity and substantial variability inherent in Deep Learning based systems. The work diverges from conventional approaches by incorporating statistical and sequential features via an RNN+LSTM model. Additionally, demographic, academic and emotional data are incorporated to predict student performance.

3. Methodology

The research methodology aimed at developing and evaluating predictive models for student performance. First, dataset will undergo meticulous preparation;the prepared dataset will be divided for further analysis. Following data splitting, ARLMN model is constructed and assessed for predictive model. Additionally, the same dataset will be evaluated for comparative analysis.

3.1. Dataset Acquisition

The dataset comprises extensive information on students' academic performance, demographics, Academics and lifestyle factors. This dataset offers a holistic view of factors influencing student success, facilitating detailed analysis and predictive modeling. The dataset is available for download from (https://doi.org/10.24432/C5TG7T).

3.2. Data Pre-processing

The dataset is separated into features and the target variable and also into training and testing sets. The features in the dataset are standardized, which is essential for neural networks. LSTM/RNN models require 3D data: (samples, time steps, features). Here, the input data is reshaped accordingly.

3.3. Model Training and Evaluation

Two models, one using LSTM and another using RNN, are constructed. These models consist of multiple layers of LSTM and RNN cells followed by densely connected layers. The models are trained on the training data. Predictions are generated using both LSTM and RNN models on the testing data.A custom classification model is instantiated and trained on the combined predictions from both LSTM and RNN models. Finally, the accuracy of the model is evaluated, as illustrated in Figure 1.

3.4. Proposed Model

The working of the proposed ARLMN Model is illustrated in Figure 2. This work introduces a Deep Learning model that predicts student academic performance based on RNN+LSTM. The Features are transformed into feature vectors representing demographic, academic and others. These vectors are combined to create a comprehensive representation of a student's historical characteristics (Fy). Then, S-AAF are used to forecast using this dataset.The objective of this is to streamline the dataset by selecting relevant features and reducing irrelevant ones, facilitated by RNN+LSTM layers which capture temporal correlations among student actions and feature connections. An RNN (Recurrent Neural Network) maintains hidden states while taking input x and generates both hidden states H and output states O. The equations of this process are:

$$O_t = \sigma(W_{Ht}O_t + b_t) \quad (1)$$

$$H_t = \sigma(w_{Ht-1}\ H_{t-1} + W_{xtXt} + b_{Ht}) \quad (2)$$

Here, $W_{Ht}O_t$ represents the hidden-to-output weight vector, H_{t-1} signifies the hidden unit from the previous sequence, $w_{Ht-1}\ H_t$denotes the hidden-to-hidden weight vector, and b_{Ht} and $b_t$are biases. This process effectively identifies relevant temporal relationships among features.

It begins by ingesting a sequence of inputs denoted as $x = x_1, x_2, \dots x_n$Through its computational process, the RNN sequentially processes each input element and generates corresponding hidden states $H = H_1, H_2, \dots H_n$and output states$O = O_1, O_2, \dots O_n$. This iterative process allows the RNN to capture temporal dependencies within the input sequence, thereby providing valuable insights into the data's temporal dynamics.

The following sections detail the architecture and workflow of the ARLMN Ensemble Model, illustrating how each component contributes to the overall predictive power of the system.

Input Layer

The input layer comprises multiple features representing different types of data, such as academic performance, demographic information, and other relevant attributes. These inputs are denoted as $x_1, x_2, \dots x_t$

LSTM Layer

The LSTM layer processes the sequential input data. LSTM networks are highly effective due to their unique architecture, which includes gates that regulate the flow of information. In the proposed method, the RNN cells are categorized as LSTM units. These LSTM units form a stack, allowing them to capture temporal sequence characteristics through forget gate, input gate, and output gate.

Pseudocode1: Sigmoid Plus – Adaptive Activation Function

1:Input:

X: Combined feature vector derived from the outputs of RNN and LSTM layers

y: Vector of corresponding class labels

T: Threshold value for the sigmoid function

2:Output

Learned weight coefficients w

Bias term b

Algorithm

3: Initialize

w: Vector of weight coefficients

b: Bias term

m: Number of training examples

4:For iteration in range (num_iterations):

Compute the linear combination of input features and parameters: z=X*w+ b

5:Apply S-AAf Activation function

$$\text{S-AAF(z,T)} = \frac{1}{e^{\ln(e^{(z+T)} + e^{(z+T)^2} + 1) - (z+T))}}$$

6: Compute the gradient of the cost function:

dw= (1/m)*X.T *(a-y)

db=(1/m)*sum(a- y)

7: Update the parameters using gradient descent:

w = w - learning_rate * dw

b=b-learning_rate *db

8. Iteration

Repeat steps 3 to 5 for a specified number of iterations or until convergence

9. Prediction:

Once the model is trained, use the trained parameters (w,b) and the S-AAF activation function to make predictions for new input data:

Prediction = S –AAF(X*w+b,T)

Input Gate

The input gate determines the current input x_t and the previous hidden state H_{t-1} should be stored in the cell state C_t. σdenotes the sigmoid activation function, W_{xi}andW_{hi} are the weight matrices. $b_i$is the bias vector. The output $i_t$represents the proportion of information to be added to the cell state.

$$i_t = \sigma(W_{xi}.x_t + W_{hi}.h_{t-1} + b_i) \quad (3)$$

Forget Gate

Determines which characteristics to retain or discard from the previous h_{t-1} and x_t. Again, it uses a activation function to generate values between 0 and 1.It has its own weight matrices (W_{xf}, W_{hf}) and bias vectorb_f.

$$f_t = \sigma(W_{xf}.x_t + W_{hf}.h_{t-1} + b_f) \quad (4)$$

Initial Layers

Similar to the input gate, the prediction in initial layers gate determines how much information from x_t and the H_{t-1} should be stored in the cell state C_t. It follows the same structure as the input gate, with its own set of weight matrices $(W_{xp,}W_{hp})$and bias vector b_p.

$$p_t = \sigma(W_{xp}.x_t + W_{hp}.h_{t-1} + b_p) \quad (5)$$

Output Gate

The output gate determines which parts of the cell state C_t should be included in the output h_t. It regulates how much of the internal state is revealed in the output at the current time step. It follows the same structure as the previous gates, with its own weight (W_{xo}, W_{ho}) and bias vector b_o.

$$o_t = \sigma(W_{xo}.x_t + W_{ho}.h_{t-1} + b_o) \quad (6)$$

Output Layer

After processing through LSTM , the data is passed into dense layer. This layer helps refine the features extracted by the LSTM and prepares the data for the final classification.

Classification

The final phase involves classifying the student based on the processed data. The classification is typically performed using S-AAF in the output layer to generate a probability score.

Detailed Workflow

1	Input Data(x_t)	:	The input features at time step t
2	Previous Hidden State (h_{t-1})	:	The hidden state from the previous time step.
3	Previous Cell State (C_{t-1})	:	The cell state from the previous time step.
4	Forget Gate (f_t)	:	Decides what information to discard from the cell state.
5	Input Gate (i_t)	:	Decides which new information to store in the cell state.
6	Cell State Candidate (C_t)	:	Potential values to add to the cell state.
7	Updated Cell State (C_t)	:	Combines previous cell state, forget gate, and input gate information.
8	Output Gate (o_t)	:	Decides what information from the cell state to output.
9	Fully Connected Layer	:	Further processes the hidden states to prepare for the final classification.
1	Output Layer	:	Uses a S-AAF to generate a probability score.
1	Decision Rule	:	Classifies the student based on the probability score.

Algorithm 1 : ARLMN Model

1. Data Collection and Preprocessing:

- Collect academic, demographic, and other relevant data.
- Preprocess data: Handle missing values, normalize, and split into training and testing sets.

2. Input Layer:

- Prepare input features $x_1, x_2, \dots x_t$for each time step t

3. LSTM Layer:

- Initialize LSTM parameters W_f, W_i, W_c, W_0 and biases b_f, b_i, b_c, b_0
- Perform forward pass through LSTM to compute:

$$f_t = \sigma(W_{xf}.x_t + W_{hf}.h_{t-1} + b_f)$$

$$i_t = \sigma(W_{xi}.x_t + W_{hi}.h_{t-1} + b_i)$$

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c$$

$$C_t = f_t.C_{t-1} + i_t.\tilde{C}_t$$

$$o_t = \sigma(W_{xo}.x_t + W_{ho}.h_{t-1} + b_o)$$

$$h_t = O_t.\tanh(C_t)$$

4. Fully Connected (Dense) Layer:

- Pass the final hidden state $h_t$through a dense layer to compute:

$$z = W_{dense}.h_t + b_{dense}$$

5. Ensemble Learning:

- Train model on the training data.
- Aggregate predictions using weighted average or majority voting.

6. Adaptive Learning Mechanism:

- Dynamically update learning parameters based on performance using gradient descent:

7. Output Layer and Classification:

- Use S-AAF on the dense layer output to computeClassification.(Defined in Pseudocode1)

8. **Evaluation:**

- Assess performance using accuracy, precision, recall, and F1 score.

4. Results and Analysis

4.1. Performance metrics

The effectiveness of system is assessed with four metrics, outlined [16] as follows.

• **Accuracy**: This metric evaluates how often a machine learning model correctly predicts outcomes. It's calculated by dividing the number of correct predictions by the total number of predictions, as per Equation 7.

$$Accuracy = \frac{PE + NE}{WP + PE + NE + WN} \quad (7)$$

• **Precision:** It quantifies the proportion of correctly identified positive instances relative to all instances predicted as positive. The calculation is performed using Eq. 8.

$$Precision = \frac{PE}{WP + PE} \quad (8)$$

• **Recall**: It calculates the proportion of true positive out of all predicted positive cases, as depicted in Equation 9.

$$Recall = \frac{PE}{PE + WN} (9)$$

• **F-measure:** This measure combines accuracy and recall, focusing solely on positive predictive data points. Equation 10 computes it.

$$F - Measure = \frac{2PE}{2PE + PE + WN} \quad (10)$$

The proposed system's performance is evaluated through three experiments, comparing models integrating RNN and LSTM architectures with diverse classification algorithms to determine the most effective configuration for accurate predictions.

4.2. Experimental results of proposed Model with different Classification Models

The assessment of machine learning models that integrate RNN and LSTM with different classifiers reveals that ,as presented in Table 1 and illustrated in Figure 3, the RNN + LSTM + S-AAF algorithm outperforms the rest, achieving the highest accuracy (98.73%). Although the RNN + LSTM + SVM model excels in precision and recall, its overall accuracy is lower. Models using Logistic Regression and K-Nearest Neighbors with RNN/LSTM exhibit the poorest performance across all metrics. This underscores the superior predictive capability of combining RNN/LSTM with the S-AAF algorithm.

4.3 Experimental results of proposed Model with different Classifiers

The comparative analysis of various classifiers, as presented in Table 2 and illustrated in Figure 4, highlights the performance of proposed work, which achieves an accuracy of 98.73%. These metrics demonstrate its robustness in accurately identifying positive cases while minimizing errors. In contrast, traditional classifiers such as Decision Tree, Naïve Bayes, and SVM show lower performance, particularly in recall. Logistic Regression presents the weakest performance with 81% accuracy. Overall, the proposed model's exceptional metrics underscore its advanced ability to handle complex data patterns and provide highly accurate and reliable predictions, making it superior to traditional classifiers for predictive analytics.

4.4. Experimental results of proposed Model with different Deep Learning Models

The comparative analysis,as presented in Table 3 and illustrated in Figure 5, reveals the proposed model's superior performance, achieving 98.73% accuracy, 98.11% precision, 1.0 recall, and a 99.05% F1 score. This model outperforms others in handling complex data patterns. The RNN+GRU+S-AAF and CNN+LSTM+S-AAF models also perform well but fall short with accuracies of 96.20% and similar precision and recall metrics. The ANN+GRU+S-AAF model shows balanced performance with 93.67% accuracy, while the ANN+LSTM+S-AAF achieves 92.40% accuracy. The CNN+GRU+S-AAF model records the lowest performance with 88.60% accuracy. Overall, the proposed model demonstrates exceptional predictive abilities, outperforming all other deep learning models in key metrics. Recurrent Neural Network is interesting method and used in different applications [17-20].

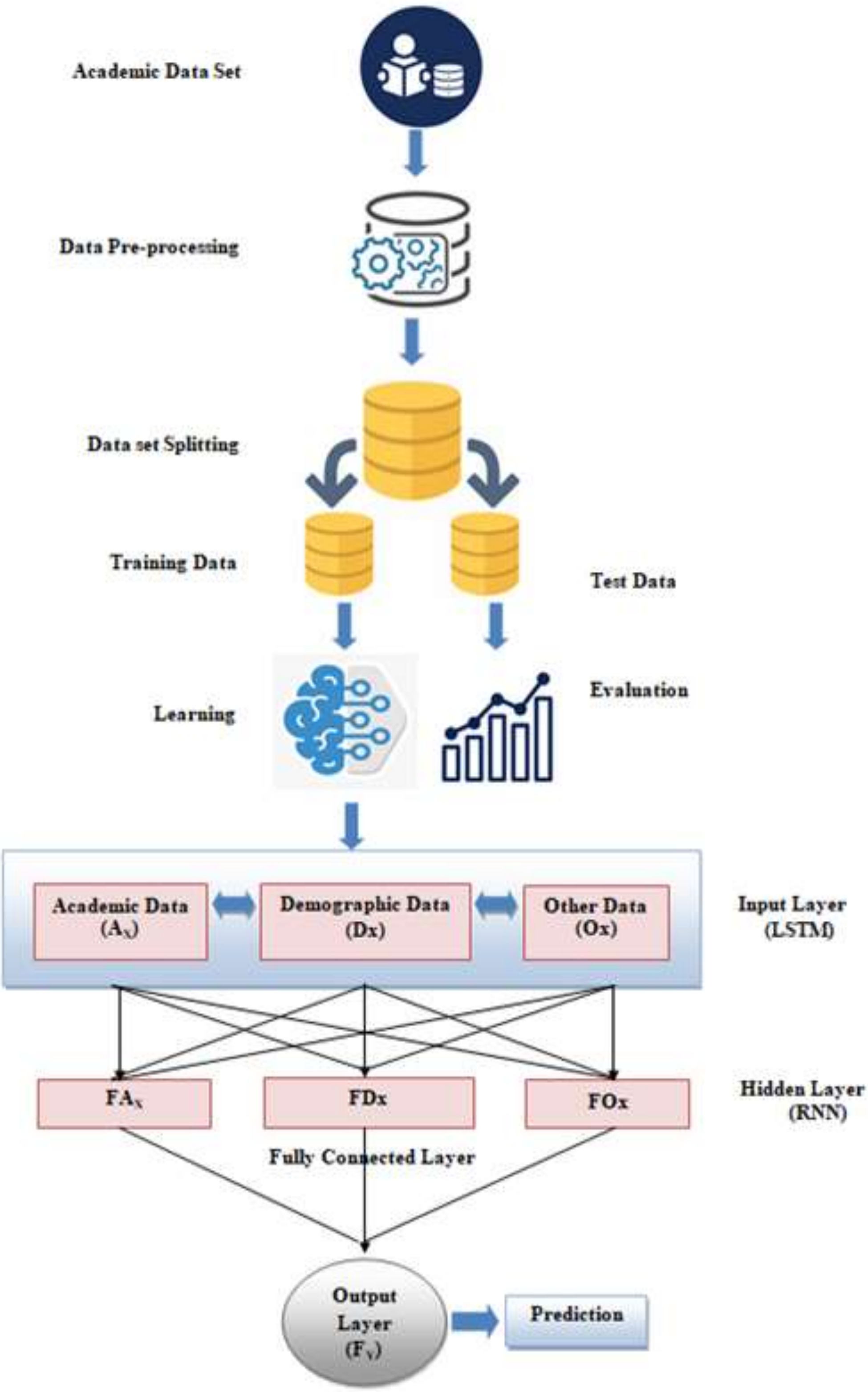

Figure 1. *The accuracy of the model*

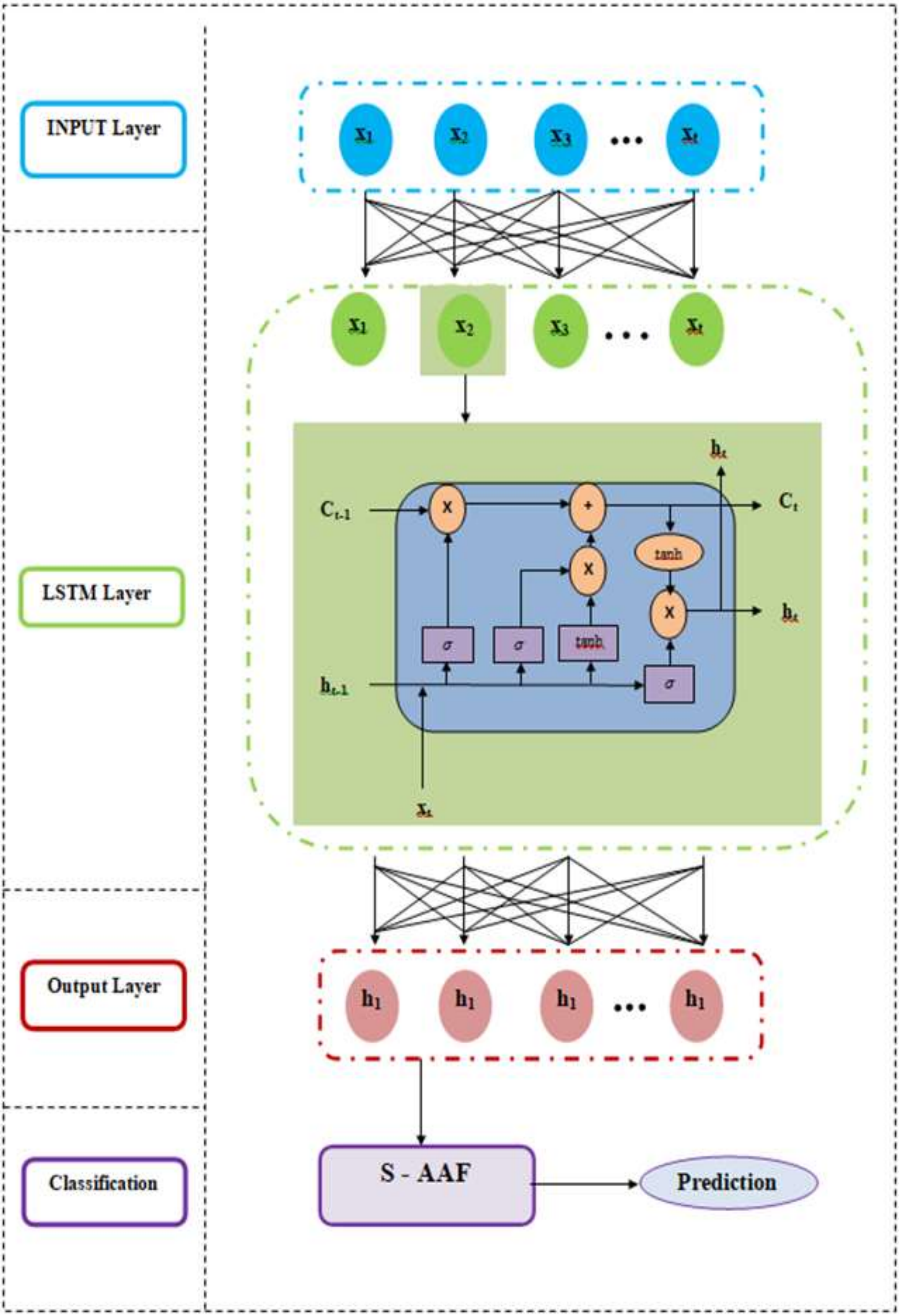

Figure 2. *Proposed ARLMN Model*

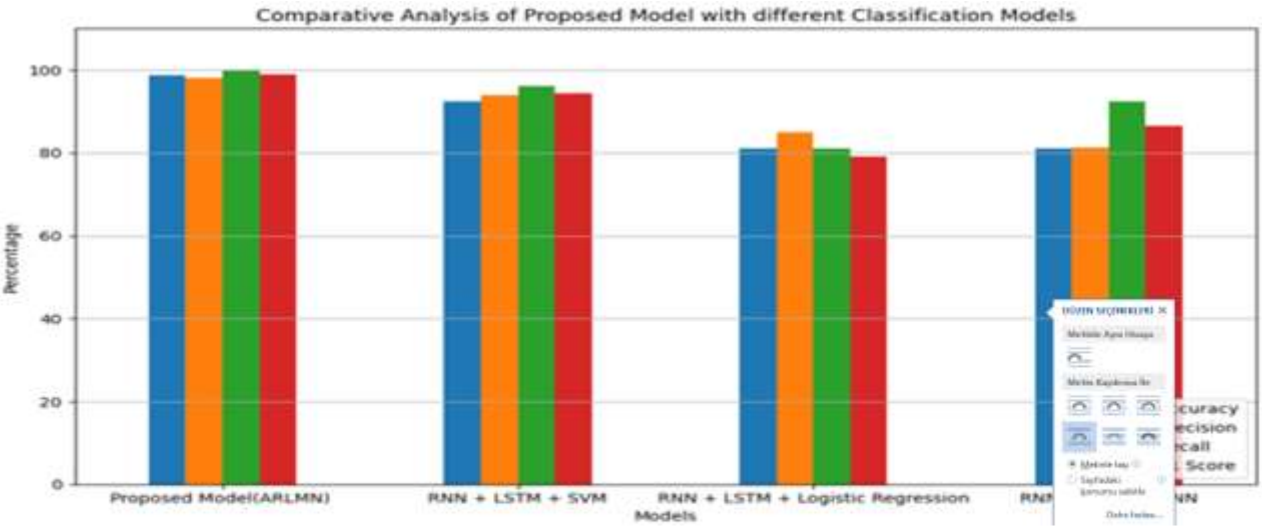

Figure 3. *The assessment of machine learning models that integrate RNN and LSTM with different classifiers reveals*

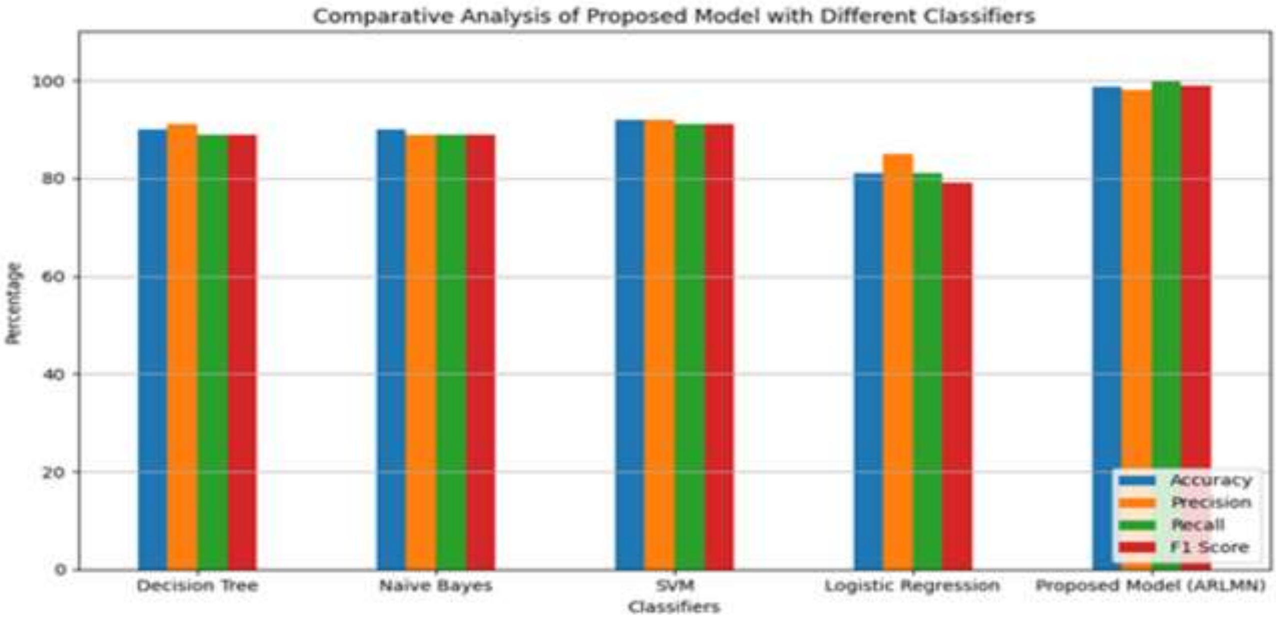

Figure 4. *The comparative analysis of various classifiers*

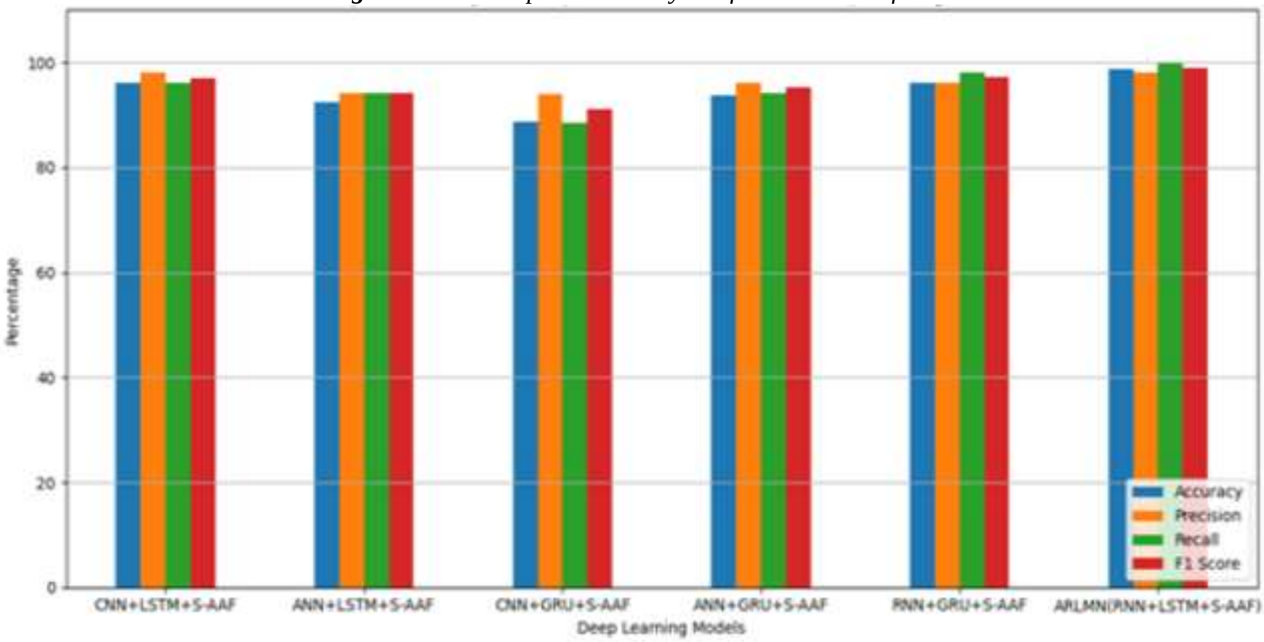

Figure 5. *The comparative analysis*

Table 1. *Comparative Analysis of proposed model with different Classification Models*

Classification Model	Accuracy	Precision	Recall	F-Measure
ARLMN(RNN+LSTM+S-AAF)	98.73	98.11	100	99.05
RNN+LSTM+SVM	92.41	94	96.15	94.3
RNN+LSTM+Logistic Regression	81	85%	81	79
RNN + LSTM + KNN	81.01	81.36	92.31	86.49

Table 2. *Comparative Analysis of proposed model with different Classifier*

Classifier	**Accuracy(%)**	**Precision(%)**	**Recall(%)**	**F1 Score(%)**
Decision Tree	90	91	89	89
Naïve Bayes	90	89	89	89
SVM	92	92	91	91
Logistic Regression	81	85	81	79
Proposed Mosel(ARLMN)	98.73%	98.11	100	99.05

Table 3. *Comparative Analysis of proposed model with different Deep Learning Models*

Deep Learning Models	**Accuracy(%)**	**Precision(%)**	**Recall(%)**	**F1 Score(%)**
CNN + LSTM + S-AAF	96.20	98.03	96.15	97.08
ANN + LSTM + S-AAF	92.40	94.23	94.23	94.23
CNN + GRU + S-AAF	88.60	93.87	88.46	91.08
ANN + GRU + S-AAF	93.67	96.07	94.23	95.14
RNN + GRU + S-AAF	96.20	96.22	98.07	97.14
Proposed Mosel(ARLMN)	98.73	98.11	100	99.05

4. Conclusions

This work introduces an approach for prediction using an RNN combined with LSTM and S-AAF models, termed ARLMN. This combined model significantly outperforms traditional Machine Learning and Deep Learning models. The high performance of the ARLMN model is due to its ability to extract both statistical and sequential features from students' academic records and demographic data. Compared with models like RNN+LSTM+SVM and RNN+LSTM+ Logistic Regression, the ARLMN model delivers notably better results. Traditional models, such as Decision Tree, Naïve Bayes, SVM, and Logistic Regression, also fall short of the performance achieved by the ARLMN model. While other Deep Learning models with S-AAF techniques, such as CNN+LSTM+S-AAF, ANN+LSTM+S-AAF, and RNN+GRU+S-AAF, show good performance, but they do not match with ARLMN. The research attributes the success of ARLMN to its comprehensive feature extraction capabilities and the robust S-AAF technique. Future research will focus on evaluating the model with various datasets and exploring a unified time-varying DL model to simplify the prediction process and improve adaptability across different educational settings.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.

- **Author contributions:** B.Vaidehi-Conceptualization, Methodology, Formal analysis, Writing- original draft. K.Arunesh-Conceptualization, Methodology, Validation, Supervision, Writing – review & editing

- **Funding information:** The authors declare that there is no funding to be acknowledged.

- **Data availability statement:** Data supporting this work are available at Dataset 1: https://doi.org/10.24432/C5TG7T.

References

[1]P. Gou, (2023). Teaching English using mobile applications to improve academic performance and language proficiency of college students, *Educ. Inf. Technol.*, pp. 1–15, doi: 10.1007/s10639-023-11864-9.

[2]Heissel, J. A., Levy, D. J., & Adam, E. K. (2017). Stress, sleep, and performance on standardized tests: Understudied pathways to the achievement gap. *AERA Open*, 3(3), 2332858417713488.

[3]Schumacher, C., &Ifenthaler, D. (2018). Features students really expect from learning analytics. *Computers in Human Behavior*, 78, 397–407.

[4]Sievertsen, H. H., Gino, F., &Piovesan, M. (2016). Cognitive fatigue infuences students' performance

on standardized tests. *Proceedings of the National Academy of Sciences*, 113(10), 2621–2624.

[5]B.-H. Kim, E. Vizitei, and V. Ganapathi, (2018). GritNet: Student performance prediction with deep learning, *arXiv*:1804.07405.

[6]L. Zhang, X. Xiong, S. Zhao, A. Botelho, and N. T. Heffernan, ‘‘Incorporating rich features into deep knowledge tracing,’’ in *Proc. 4th ACM Conf. Learn. Scale*, Apr. 2017, pp. 169–172, doi: 10.1145/3051457.3053976.

[7]Ahmed M, Najmul Islam AKM (2019) Deep learning: hope or hype. *Ann Data Sci* 7(3):427–432. https://doi.org/10.1007/s40745-019-00237-0.

[8]Member S (2010) Educational data mining: a review of the state of the Art. *IEEE Trans Syst Man Cybern* C 40(6):601–618

[9]S.U. Hassan, H. Waheed, N.R. Aljohani, M. Ali, S. Ventura, F. Herrera, (2019). Virtual learning environment to predict withdrawal by leveraging deep learning, *Int. J. Intell. Syst.* 34 (8);1935–1952, https://doi.org/10.1002/int.22129.

[10]M. Wasif, H. Waheed, N.R. Aljohani, S.-U. Hassan, (2019). Understanding Student Learning Behavior and Predicting Their Performance, pp. 1–28, https://doi. org/10.4018/978-1-5225-9031-6.ch001.

[11]Tomasevic, N., Gvozdenovic, N., &Vranes, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, 143, 103676.

[12]Okubo, F., Yamashita, T., Shimada, A., & Ogata, H. (2017). A neural network approach for students' performance prediction. *In Proceedings of the Seventh International Learning Analytics & Knowledge Conference (LAK '17). Association for Computing Machinery, New York,* NY, USA (pp. 598–599). https://doi.org/10.1145/3027385.3029479.

[13]Y. He, et al., (2020) Online at-risk student identification using RNN-GRU joint neural networks, *OR Inf.* 11 (10);1–11, https://doi.org/10.3390/info11100474.

[14]M. Adnan, et al., (2021). Predicting at-risk students at different percentages of course length for early intervention using machine learning models, *IEEE Access* 9;7519–7539.

[15]Su, Y., Liu, Q., Liu, Q., Huang, Z., Yin, Y., Chen, E., & Hu, G. (2018). Exercise-enhanced sequential modeling for student performance prediction. *In Proceedings of the AAAI Conference on Artifcial Intelligence* (Vol. 32, No. 1). https://doi.org/10.1609/aaai.v32i1.11864.

[16]Kukkar, A., Mohana, R., Nayyar, A., Kim, J., Kang, B. G., &Chilamkurti, N. (2019). A novel deeplearning-based bug severity classifcation technique using convolutional neural networks and random forest with boosting. *Sensors*, 19(13), 2964

[17]P. Padma, & G. Siva Nageswara Rao. (2024). CBDC-Net: Recurrent Bidirectional LSTM Neural Networks Based Cyberbullying Detection with Synonym-Level N-Gram and TSR-SCSOFeatures. *International Journal of Computational and Experimental Science and Engineering*, 10(4). https://doi.org/10.22399/ijcesen.623

[18]Sunandha Rajagopal, & N. Thangarasu. (2024). The Impact of Clinical Parameters on LSTM-based Blood Glucose Estimate in Type 1 Diabetes . *International Journal of Computational and Experimental Science and Engineering*, 10(4). https://doi.org/10.22399/ijcesen.656

[19]Rajani Kumari Inapagolla, & K . Kalyan Babu. (2025). Audio Fingerprinting to Achieve Greater Accuracy and Maximum Speed with Multi-Model CNN-RNN-LSTM in Speaker Identification: Speed with Multi-Model CNN-RNN-LSTM in Speaker Identification. *International Journal of Computational and Experimental Science and Engineering*, 11(1). https://doi.org/10.22399/ijcesen.1138

[20]Olola, T. M., & Olatunde, T. I. (2025). Artificial Intelligence in Financial and Supply Chain Optimization: Predictive Analytics for Business Growth and Market Stability in The USA. *International Journal of Applied Sciences and Radiation Research*, 2(1). https://doi.org/10.22399/ijasrar.18