Journal of Information Systems Engineering and Management
2025, 10(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Hybrid RNN-BiLSTM Approach for Student Success Estimation in Online Social Networks

[1]S. Senthamaraiselvi, [2]Dr. K. Meenakshi Sundaram, [3]Dr. J. Vandarkuzhali

[1]Research Scholar (Part Time), Department of Computer Science, Erode Arts and Science College, Erode, Tamilnadu, India.
Email: sunruks_senthu@yahoo.co.in
[2]Research Supervisor, Former Associate Professor and Head, Department of Computer Science, Erode Arts and Science College, Erode, Tamilnadu, India.
Email: lecturerkms@yahoo.com
[3]Assistant Professor, Department of Computer Science, Erode Arts and Science College, Erode, Tamilnadu, India.
Email: j.vandarkuzhali@easc.ac.in

ARTICLE INFO

Received: 02 Oct 2024

Revised: 05 Dec 2024

Accepted: 22 Dec 2024

ABSTRACT

Predicting student academic performance is a crucial task in educational data mining, enabling early identification of at-risk students and informed instructional interventions. This paper proposes a hybrid RNN-bi-LSTM method, combining the strengths of recurrent neural networks (RNNs) and bidirectional long short-term memory (bi-LSTM) networks, to model student learning behaviors and predict student performance. The hybrid approach leverages sequential dependencies and contextual information to improve prediction accuracy. Experiments on real-time high dimensional OSN user dataset demonstrate the effectiveness of the hybrid RNN-bi-LSTM method, outperforming baseline models in predicting student performance of Excel and Vivekanandha classes using neural network methods was tested. This study contributes to the development of accurate and reliable student performance prediction models, supporting data-driven decision-making in education and enhancing student success. The proposed hybrid classification performance strategy yielded an accuracy of 99.87 percent and an F1-score of 99.25 percent, according to the experimental findings obtained from the OSN User dataset.

Keywords: Data mining, Classification, RNN, BiLSTM, OSN.

I. INTRODUCTION

With the rise of online learning, social networks have become an essential tool for students to connect, collaborate, and share resources. However, instructors and researchers have long sought to understand how online social network interactions relate to student performance. Recent advances in data mining and machine learning have made it possible to analyze large-scale online social network data and identify patterns that correlate with student success [1]. By leveraging neural networks, researchers can now predict student performance with unprecedented accuracy, providing valuable insights for instructors, administrators, and students themselves. This paper has the potential to revolutionize the way to approach online learning, enabling personalized interventions, early warning systems, and data-driven instruction. By harnessing the power of online social network data, we can unlock the full potential of online learning [2-3].

Early prediction of student performance is crucial for identifying at-risk students and enabling instructors to implement targeted interventions, such as additional courses or assignments, to enhance student outcomes before the course concludes [4]. Therefore, it is essential to develop prediction models that can forecast student performance early in the course. Moreover, most studies have employed a dichotomous approach to predicting student performance (pass/fail), which neglects the potential for students to achieve excellent grades and maximize the benefits of the course [5].

Online social networks (OSNs) have emerged as a powerful tool for predicting student performance, enabling educators to recognize at-risk students and provide appropriate interventions. Among the most popular OSNs for educational purposes are Facebook, YouTube, and Twitter, which offer a wealth of data and insights on student behavior, engagement, and learning patterns. By leveraging machine learning algorithms and data analytics techniques, researchers can harness the power of OSNs to predict student performance, detect early warning signs of academic struggle, and develop targeted interventions to support student success. Through the analysis of social media data, educators can gain a deeper understanding of student learning behaviors, identify knowledge gaps, and develop personalized learning plans to enhance student outcomes [6]. By harnessing the predictive power of OSNs, educators can create a more supportive and inclusive learning environment, tailored to the unique needs and abilities of each student.

Predicting student's academic performance is a vital concern in educational data mining, with significant advancements made in this area [7]. By forecasting students' performance, educators can identify at-risk learners and offer timely support. To achieve this, researchers gather data on students' demographic characteristics, past academic achievements, and learning behaviors, and then develop prediction models using various methods such as decision trees, logistic regression, and K-Nearest Neighbor (KNN) [8]. Following the success of deep neural networks in image recognition, natural language processing, and anomaly detection [9], researchers have started exploring their application in educational data mining [10].

The following objectives are raised in order to meet the paper's aim,

- Data Preprocessing: Clean up and prepare the dataset by handling missing values, scaling features, and normalizing it.
- Feature Engineering: Extract appropriate features from the dataset, including age group, education status, social media account ownership, social media platforms used frequency of social networking site usage, studies affected by social media, reduction in exam performance, social media usage at night, and college affiliation.
- Model Development: Develop a hybrid RNN with bi-LSTM classification model, consisting of an RNN layer, a bi-LSTM layer, and a hybrid layer that concatenates the outputs from both layers.
- Training and Validation: Evaluate the model's performance using measures like accuracy, precision, recall, and F1-score after training it on a subset of the dataset.

II. RELATED WORK

Guang-Yu et al. [11] explored the relationship among student behavior data and learning outcomes, suggesting that students' behavior patterns in campus life reflect their learning styles and habits. By quantitatively evaluating various behavior indicators, researchers can gain insights into students' learning status and identify patterns that correlate with academic performance, a critical metric for assessing student learning. Leveraging data from the "Four PIN" education system at Beihang Shoue College, employed the XGBoost gradient boosting decision tree algorithm to analyze students' study life and social work participation. Notably, the model achieved a prediction accuracy of 73%, demonstrating the potential of behavior data analysis in predicting student academic outcomes.

Saputra et al. [12] conducted a study to predict student evaluation outcomes in e-learning based learning systems. They utilized log data from 641 student learning activities extracted from a Learning Management System (LMS). To predict evaluation results, they employed a neural network method optimized by swarm particle optimization, which addressed the challenge of optimizing large data in neural networks. The results showed that the neural network method achieved an accuracy of 95.47 percent and an AUC value of 97.90 percent.

Giannakas et al. [13] proposed a DNN framework for early prediction of software engineering team performance, employing binary classification with two hidden layers. The framework was rigorously evaluated using various activation functions and optimizers, and the researchers also utilized SHAP to interpret the framework's decisions and identify the most influential features affecting its predictions. This study showcases the potential of DNNs in predicting team performance and demonstrates the utility of SHAP in uncovering the key factors driving these predictions.

Baashar et al. [14] highlighted the vast potential of technological advancements in educational displacement, particularly in the application of data mining. Focusing on machine learning (ML) approaches, specifically Artificial Neural Networks (ANNs), this study reviewed existing literature to identify areas for improvement and surveyed current research on ANN methods used to predict student performance. The selection of input variables varied widely depending on the study context and data availability, and few studies demonstrated tangible improvements in student outcomes, performance, and achievement. This highlights the need for further research to identify effective ANN techniques and algorithms that can be applied in real-world educational settings to enhance student success.

Rufai Aliyu Yauri et al. [15] developed a predictive model that accurately identifies students at risk of failure or success using a machine learning algorithm. Descriptive statistics were employed to establish the features that significantly impact students' academic performance. A neural network model was built with 12 input variables, two hidden layers, and one output layer, trained using the back propagation learning algorithm. The authors demonstrated the potential of machine learning algorithms in predicting student academic performance, enabling early interventions to support at-risk students.

According to S. Senthamaraiselvi and K. Meenakshi Sundaram [16], research has shown a significant correlation between social media usage and academic achievement, revealing that excessive social media use negatively impacts students' academic performance. This study investigated the relationship among students' social media usage patterns, including frequency, type, and duration, and their overall academic achievement, shedding light on the impact of social media on students' academic success.

S. Senthamaraiselvi and K. Meenakshi Sundaram [17] investigated the social network usage habits and motivations of VIIMS students, providing insights into the methodology and findings of the research. The study employed a survey questionnaire to examine usage patterns, including the use of social media for educational purposes. After analyzing the survey data collected over a 14-day period, the researchers identified the most popular social network among college students for both academic and recreational purposes, based on 140 survey responses. The study's findings shed light on how students utilize social media, enabling a comparison of usage patterns and informing strategies for effective social media integration in teaching and learning.

S. Senthamaraiselvi and K. Meenakshi Sundaram [18] introduced an advanced feature selection technique, IMIFPC, combined with an ERB classification method, to predict student performance. IMIFPC integrates Pearson's correlation and mutual information methods to select relevant features. The proposed ERB approach was tested on a real-time OSN user dataset to predict student performance, categorized into two classes. The experimental results showed that the ERB classification method achieved an impressive accuracy of 98.00% and an F1-score of 97.99%, demonstrating the effectiveness of the proposed approach in predicting student performance.

S. Senthamaraiselvi, K. Meenakshi Sundaram and J. Vandarkuzhali [19] proposed a novel Hybrid Convolution Neural Network with Long Short-Term Memory (HCNN-LSTM) approach, which synergistically combines the spatial feature extraction capabilities of CNNs with the temporal feature extraction capabilities of LSTMs. To further enhance the model's performance, dropout layers and batch normalization were incorporated. The proposed hybrid CNN-LSTM approach was evaluated on a real-time high-dimensional OSN user dataset to predict students' academic success in Excel and Vivekanandha classes using neural network methods.

III. METHODS

This part derives the hybrid deep learning model of RNN with BiLSTM classification process, which is used in the suggested study approach to execute the OSN user data classification utilizing data preprocessing. Figure 1 describes the OSN user data classification procedure, which takes into account the overall process flow diagram.

A. Dataset Preprocessing

One of the most important steps in data mining is data preprocessing, which turns raw data into an appropriate form by performing tasks such as data integration, data cleaning, and discretization, enabling the creation of high-quality data that can be used to train and improve the accuracy of deep learning models that we have previously discussed in [19]. The real-time dataset utilized in this study was collected from Excel College and Vivekanandha College in Tamil Nadu, India, and is stored in a CSV (Comma Separated Values) file format. The initial dataset, comprising 10 columns, underwent cleaning and discretization processes as part of preprocessing, resulting in the final preprocessed data, illustrated in Figure 2, which includes independent variables such as age group, education status, social media account ownership, social media platforms used, frequency of social networking site usage, studies affected by social media, reduction in exam performance, social media usage at night, and college affiliation, all gathered from students across various age groups and educational backgrounds, to predict students' academic performance.

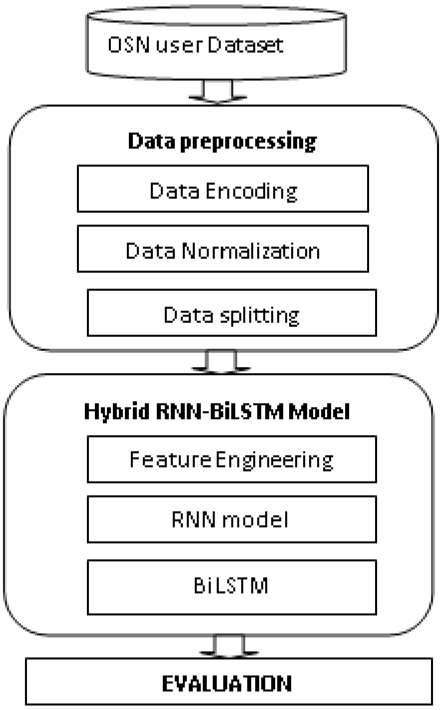
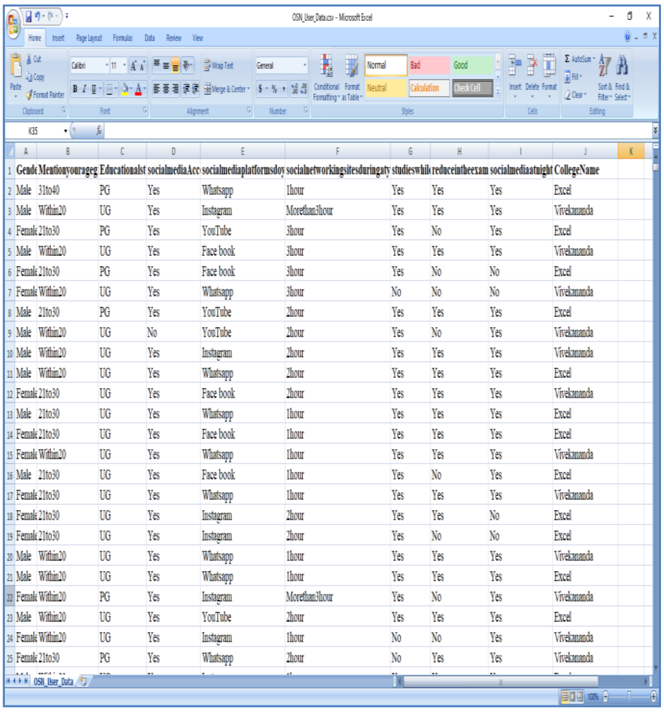


Fig.1. proposed flow diagram

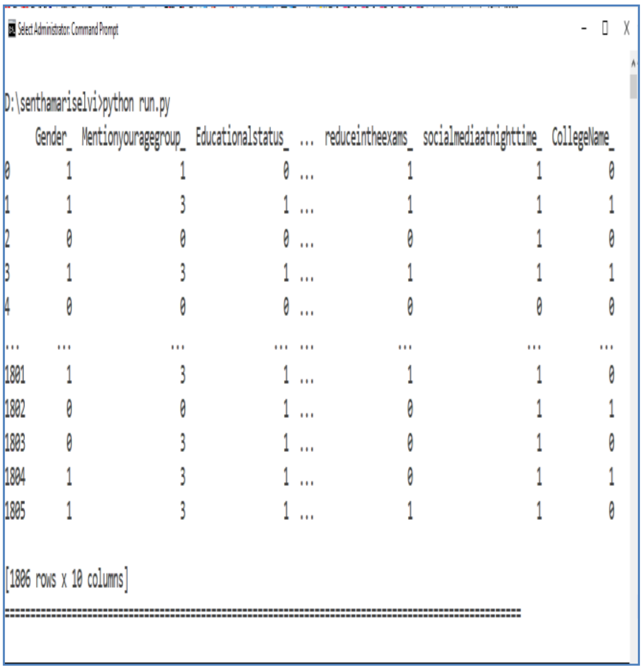
Figure 2 describes the OSN has 1806 incomplete, inconsistent, and noisy samples that are included in the proposed OSN dataset, which has 10 features.



	A	B	C	D	E	F	G	H	I	J
1	Gende	Mentionyourageg	Educationalst	socialmediaAcc	socialmediaplatformsdoy	socialnetworkingsitesduringaty	studieswhil	reduceintheexam	socialmediaatnight	CollegeName
2	Male	31to40	PG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Excel
3	Male	Within20	UG	Yes	Instagram	Morethan3hour	Yes	Yes	Yes	Vivekananda
4	Femal	21to30	PG	Yes	YouTube	3hour	Yes	No	Yes	Excel
5	Male	Within20	UG	Yes	Face book	3hour	Yes	Yes	Yes	Vivekananda
6	Femal	21to30	PG	Yes	Face book	3hour	Yes	No	No	Excel
7	Femal	Within20	UG	Yes	Whatsapp	3hour	No	No	No	Vivekananda
8	Male	21to30	PG	Yes	YouTube	2hour	Yes	Yes	Yes	Excel
9	Male	Within20	UG	No	YouTube	2hour	Yes	No	Yes	Vivekananda
10	Male	Within20	UG	Yes	Instagram	2hour	Yes	Yes	Yes	Vivekananda
11	Male	Within20	UG	Yes	Whatsapp	2hour	Yes	Yes	Yes	Excel
12	Femal	21to30	UG	Yes	Face book	2hour	Yes	Yes	Yes	Vivekananda
13	Male	21to30	UG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Excel
14	Femal	21to30	UG	Yes	Face book	1hour	Yes	Yes	Yes	Excel
15	Femal	Within20	UG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Vivekananda
16	Male	21to30	UG	Yes	Face book	1hour	Yes	No	Yes	Excel
17	Femal	21to30	UG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Vivekananda
18	Femal	21to30	UG	Yes	Instagram	2hour	Yes	Yes	No	Excel
19	Femal	21to30	UG	Yes	Instagram	2hour	Yes	No	No	Excel
20	Male	Within20	UG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Vivekananda
21	Male	Within20	UG	Yes	Whatsapp	1hour	Yes	Yes	Yes	Excel
22	Femal	Within20	PG	Yes	Instagram	Morethan3hour	Yes	No	Yes	Vivekananda
23	Male	Within20	UG	Yes	YouTube	2hour	Yes	Yes	Yes	Excel
24	Femal	Within20	UG	Yes	Instagram	1hour	No	No	Yes	Vivekananda
25	Femal	21to30	PG	Yes	Whatsapp	2hour	No	Yes	Yes	Vivekananda

Fig.2. OSN User Dataset

The next stage involves encoding labels in the datasets after loading. Since deep neural networks require numerical values, the non-numerical labels in each dataset must be converted. Specifically, the One-Hot Encoder is used to transform the data features and Excel and Vivekananda class labels into numerical values. Following this, the data is normalized to maximize within-range characteristics. As shown in Figure 3, this preprocessing step translates each full feature into attribute relation values, enabling the model to process the data effectively.



```
Select Administrator Command Prompt

D:\senthamariselvi>python run.py
      Gender_  Mentionyouragegroup_  Educationalstatus_  ...  reduceintheexams_  socialmediaatnighttime_  CollegeName_
0           1                     1                   0  ...                  1                        1             0
1           1                     3                   1  ...                  1                        1             1
2           0                     0                   0  ...                  0                        1             0
3           1                     3                   1  ...                  1                        1             1
4           0                     0                   0  ...                  0                        0             0
...       ...                   ...                 ...  ...                ...                      ...           ...
1801        1                     3                   1  ...                  1                        1             0
1802        0                     0                   1  ...                  0                        1             1
1803        0                     3                   1  ...                  0                        1             0
1804        1                     3                   1  ...                  0                        1             1
1805        1                     3                   1  ...                  1                        1             0

[1806 rows x 10 columns]
==========================================================================================
```

Fig.3. OSN User Preprocessed data

B. *Hybrid RNN-BiLSTM model*

The paper employs a hybrid deep learning approach, combining RNNs and Bi-LSTM networks, to predict student performance. The methodology consists of data preprocessing, feature engineering, model development, training and validation, hyper-parameter tuning, and testing. The dataset is first cleaned and preprocessed, followed by data features and scaling. A hybrid RNN with bi-LSTM classification model is then developed, consisting of an RNN layer, a bi-LSTM layer, and a hybrid layer that concatenates the outputs from both layers. The model is trained and validated using a portion of the dataset, with hyper-parameter tuning performed to optimize performance. Finally, the model is evaluated on a separate test dataset to ensure generalizability, aiming to leverage the strengths of both RNNs and bi-LSTM networks to capture complex patterns in the data and improve student performance prediction accuracy.

Feature Engineering process involves extracting relevant features from the data, including age group (A), education status (E), social media account ownership (S), social media platforms used (P), frequency of social networking site usage (F), studies affected by social media (T), reduction in exam performance (R), social media usage at night (N), and college affiliation (C). These features are then normalized using Min-Max Scaler and converted into numerical variables using One-Hot Encoding. Next, RNN component takes in feature data and outputs a hidden state and an output at each time step. The hidden state is computed using the current input, the previous hidden state, and the learnable weights and biases. The output is computed using the hidden state and the learnable weights and biases.

The RNN layer is composed of a single recurrent neuron with a tanh activation function. The neuron takes in the input vector *Xt* at time step *t* and the previous hidden state $h\{t\text{-}1\}$. To compute the current hidden state h_t using the following equation:

$$h_t = \tanh(W_h * h_{t-1} + U_h * X_t + bias_h)^d \qquad (1)$$

Where X_t = (A, E, S, P, F, T, R, N, C) ∈ R^9 (input vector at time step t), W_h and $bias_h$ is learnable weights and biases.

Softmax is a mathematical function used in neural networks (NNs) for multi-class classification problems. It's used in the output layer to predict probabilities of each class. The neuron computes the output rnn_t using the following equation:

$$rnn_t = softmax(W_l * h_t + bias_l) \qquad (2)$$

Where, rnn_t is the output of the RNN layer at time step *t*, W_l and $bias_l$ is learnable weights and biases, h_t is hidden state of RNN at time step *t*, *softmax* is the function, which is used to normalize the output to ensure it's a valid probability distribution.

The result layer takes in the output rnn_t from the RNN layer and computes the final output using the softmax activation function. The final output is a probability distribution over the two classes (Excel and Vivekanandha).

The bi-LSTM layer is a type of recurrent neural network (RNN) layer that processes the input data in both forward and backward directions, capturing both past and future contexts. In the context of the dataset, the bi-LSTM layer takes in the input vector X_t at time step t and outputs two hidden states: Forward and Backward.

The forward hidden state (Forward) encapsulates information from earlier time steps, while the backward hidden state (Backward) encompasses information from future time steps. By concatenating these two hidden states, the model combines both past and future context, resulting in a comprehensive final output that integrates information from all time steps.

The bi-LSTM layer is particularly useful for modeling student prediction data with long-term dependencies, such as the dataset which contains features like social media usage and exam performance. The Forward LSTM gates of input data X_t is defined as,

Input Gate:

$$i_t fwd = sigmoid(W_i fwd * X_t + U_i fwd * h_{t-1} fwd + bias_i fwd) \quad (3)$$

Forget Gate:

$$f_t fwd = sigmoid\left(W_f fwd * X_t + U_f fwd * h_{t-1} fwd + bias_f fwd\right) \quad (4)$$

Output Gate:

$$o_t fwd = sigmoid(W_o fwd * X_t + U_o fwd * h_{t-1} fwd + bias_o fwd) \quad (5)$$

Cell state update:

$$c_t fwd = sigmoid(W_c fwd * X_t + U_c fwd * h_{t-1} fwd + bias_c fwd) \quad (6)$$

The final hidden state update of Forward LSTM the below equation,

$$h_t fwd = \ o_t fwd * \tanh(c_t fwd) \quad (7)$$

The Backward LSTM gates of input data X_t is defined as,

$$i_t bwd = sigmoid(W_i bwd * X_t + U_i bwd * h_{t-1} bwd + bias_i bwd) \quad (8)$$

Forget Gate:

$$f_t bwd = sigmoid\left(W_f bwd * X_t + U_f bwd * h_{t-1} bwd + bias_f bwd\right) \quad (9)$$

Output Gate:

$$o_t bwd = sigmoid(W_o bwd * X_t + U_o bwd * h_{t-1} bwd + bias_o bwd) \quad (10)$$

Cell state update:

$$c_t bwd = sigmoid(W_c bwd * X_t + U_c bwd * h_{t-1} bwd + bias_c bwd) \quad (11)$$

The final hidden state update Backward LSTM the below equation,

$$h_t bwd = \ o_t bwd * \tanh(c_t bwd) \quad (12)$$

Where, W, U, and *bias* are learnable parameter, h_t is hidden state, $i_t fwd$, $i_t bwd$ is input of forward and backward hidden state, *sigmoid* is the sigmoid activation function, which maps the input to a value between 0 and 1.$f_t fwd$, $f_t bwd$ is forward forget and backward gate, $o_t fwd$, $o_t bwd$ is output of forward and backward gate, $c_t fwd$, $c_t bwd$ is cell state of forward and backward gate, $h_t fwd$, $h_t bwd$ is final LSTM of hidden forward and backward gate.

The proposed bi-LSTM layer takes in the input features and outputs two hidden states: $h_t Fwd$ and $h_t bwd$. The forward hidden state captures information from the past, while the backward hidden state captures information from the future. The output of the bi-LSTM layer is the concatenation of the forward and backward hidden states:

$$bilstm_t = softmax(W_b * h_t fwd + W_{hb} * h_t bwd + bias_b] \quad (13)$$

Where, W_b and $bias_b$ are learnable parameters.

The proposed hybrid layer of RNN with bi-LSTM is defined as,

$$HBlayer = softmax(W_{hb} * [rnn_t, bilstm_t] + bias_{hb}) \quad (14)$$

The hybrid model combines the strengths of both RNN and bi-LSTM by capturing feature dependencies with RNN and contextual information with bi-LSTM. The result of hybrid layer is then used as input to the next layer (e.g. a classification layer) to predict the target variable.

Algorithm 1: Hybrid RNN-BiLSTM

Input: Input Dataset D, Train Set T, Batch Normalization *BN,* Number of Epochs, Learning rate.

Output: prediction (*P*)

Preparation:

1. Data Preprocessing
2. Feature Engineering
3. Model Development
4. hybrid RNN with BiLSTM Model

Steps:

While (T)

1. $D_p \leftarrow$ Data preprocessing // Data Encoding, Data Normalization and Data splitting
2. $F \leftarrow$ Feature Engineering (D_p)
3. $RNN_t \leftarrow$ Perform RNN(F)
4. $BiLSTM_t \leftarrow$ Perform BiLSTM(F)
5. $Hybrid_t \leftarrow$ Concatenate of RNN_t amd $BiLSTM_t$
6. P $\leftarrow$ Hybrid model to predict the student classification using $Hybrid_t$

End While

The proposed HRNN-BiLSTM (Hybrid RNN with BiLSTM) technique combines the strengths of RNNs and BiLSTM networks to improve the accuracy of predictions in the OSN (Online Social Network) dataset. By leveraging both RNN and BiLSTM, HRNN-BiLSTM can capture long-term dependencies, model complex user behavior, and enhance feature extraction, leading to improved accuracy and generalizability. This hybrid approach is particularly effective for OSN datasets, which exhibit complex, sequential, and long-term dependent behavior, allowing HRNN-BiLSTM to better represent user behavior and adapt to new data, ultimately contributing to more accurate predictions and a deeper understanding of OSN user behavior.

IV. RESULTS AND DISCUSSION

The proposed Hybrid RNN with BiLSTM (HRNN-BiLSTM) technique was used to predict the results. With Windows 10 operating system, 8GB main memory, 3.28GHz x64-based Intel I5-6500U series CPUs, and Python 3.8 simulations, the results are achieved. The resulting parameters of OSN dataset overall HRNN-BiLSTM training accuracy with loss is described in below Table 1, figures 4 and 5,

Table 1. Training and validation of loss and accuracy of the proposed Hybrid RNN-BiLSTM model for OSN User dataset classification

Epochs	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
Jan-30	0.4318	0.3811	2.1918	2.7888
Feb-30	0.714	0.4833	1.3557	2.5394
Mar-30	0.7928	0.5403	0.964	2.2411
Apr-30	0.8585	0.6424	0.7038	1.9577
May-30	0.9022	0.6523	0.532	1.6982
Jun-30	0.9271	0.698	0.4198	1.4742
Jul-30	0.9373	0.7525	0.3497	1.3016
Aug-30	0.9424	0.7976	0.3037	1.0783
Sep-30	0.9485	0.8389	0.263	0.8447
Oct-30	0.9581	0.8369	0.2322	0.7221
Nov-30	0.9625	0.8919	0.2016	0.5497
Dec-30	0.9688	0.9175	0.1763	0.42
13/30	0.9721	0.9391	0.1566	0.3462
14/30	0.9742	0.9509	0.1424	0.2774
15/30	0.9777	0.9411	0.1229	0.2387
16/30	0.9821	0.9627	0.1097	0.192
17/30	0.9797	0.9548	0.1073	0.18
18/30	0.979	0.9627	0.111	0.1826
19/30	0.9849	0.9666	0.0849	0.1588
20/30	0.9867	0.9725	0.0799	0.1388
21/30	0.9878	0.9764	0.0708	0.1178
22/30	0.986	0.945	0.0713	0.2152
23/30	0.9869	0.9745	0.0719	0.11
24/30	0.9884	0.9666	0.0625	0.1068
25/30	0.9876	0.9646	0.064	0.1141
26/30	0.9908	0.9745	0.0495	0.1039
27/30	0.9869	0.9489	0.0623	0.1727
28/30	0.9928	0.9627	0.0472	0.1287
29/30	0.9935	0.9745	0.0386	0.0886
30/30	0.9948	0.9725	0.0401	0.0879

Fig. 4. Proposed Hybrid RNN-BiLSTM Training model Accuracy

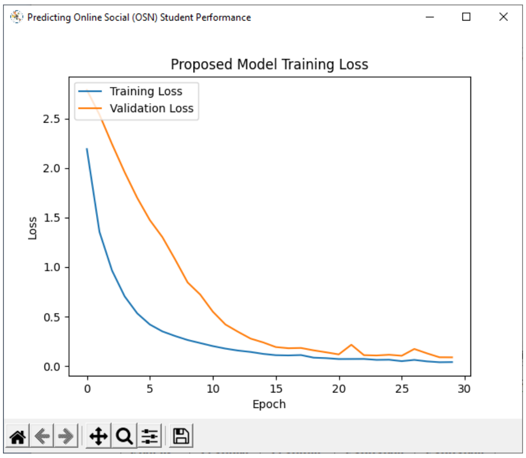


Fig. 5: Proposed Hybrid RNN-BiLSTM Training and validation loss

The proposed method trains a learning model using the fit() method, specifying training data, batch size (128), number of epochs (30), validation data, and a callback (history) to record training history. During a random initialization the propoed method network weights are typically initialized randomly. So the network starts with random guesses, and it takes some time for the network to learn and adjust the weights to make accurate predictions. During each epoch, the model processes training data in batches, trains on each batch using the Adam optimizer and categorical_crossentropy loss, evaluates on validation data, records training history, and prints a summary of training progress, including loss and accuracy on training and validation data is described in table 1.

The HRNN-BiLSTM Classification results for each model include the F1-score, precision, recall, and accuracy. The comparison between conventional methods and precision, recall, accuracy, and F1-score measurements is presented in Table 2 and Table 3. In this table 2 values are obtained by using an OSN dataset to train and test each model. The dataset is split into training and testing sets, and the models are trained on the training set. Then, the models are evaluated on the testing set, and the resulting true positives, true negatives, false positives, and false negatives are used to calculate the precision, recall, accuracy, and F1-score.

$$Precison = \frac{TP}{TP + FP} \tag{15}$$

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (17)$$

$$F1score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (18)$$

Table 2. Values of TP, TN, FP and FN

Methods	**TP**	**TN**	**FP**	**FN**
DT	35	17	5	0
Random Forest	39	20	3	0
Naïve Bayes (NB)	111	28	3	1
ERB [18]	120	40	3	0
Hybrid CNN-LSTM [19]	126	62	1	0
Proposed HRNN-BiLSTM	**700**	**53**	**1**	**0**

Table 3. A comparative analysis of F1-score metrics, precision, recall, and accuracy.

Methods	**Precision**	**Recall**	**Accuracy**	**F1-score**
DT	87.5	100	91.22	93.33
Random Forest	92.85	100	95.16	96.29
Naïve Bayes (NB)	97.36	99.1	97.2	97.2
ERB [18]	97.56	100	98.15	98.35
HCNN-LSTM	99.2	100	99.5	99.6
Proposed HRNN-BiLSTM	**99.85**	**100**	**99.87**	**99.92**

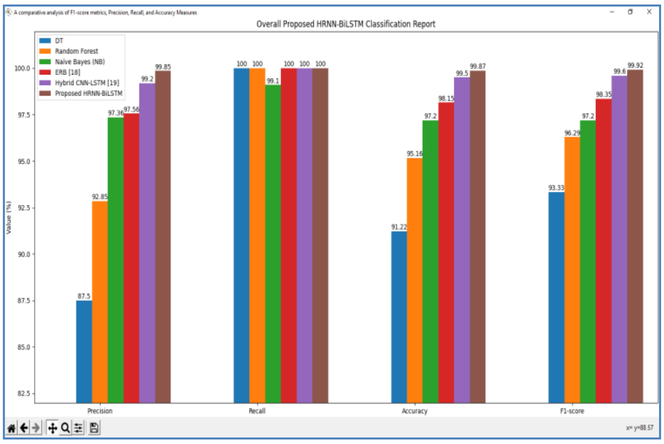


Fig. 6: Performance Analysis

As illustrated in figure 6, the values are likely obtained through a classification task, where each model is trained to predict a specific class or label. The proposed HRNN-BiLSTM model has the highest accuracy (99.87%) and F1-score (99.25%), indicating that it performs the best among all the models listed.

V. CONCLUSION AND FUTURE WORK

The paper examined the HRNN-BiLSTM classification method, which combines a Recurrent Neural network and Bidirectional long short-term memory, on high-dimensional real-time OSN datasets. The hybrid RNN-bi-LSTM method has demonstrated promising results in capturing feature dependencies and contextual information in the dataset. The combination of RNN and bi-LSTM has improved the accuracy of the model and provided a more comprehensive understanding of the data. The hybrid method has also shown the ability to handle long-term dependencies and provide increased interpretability. According to the findings, the proposed offered the best accuracy. Using the OSN dataset, we were able to achieve 99.48 % accuracy with fewer epochs. Future work should focus on hyper-parameter tuning, regularization techniques, multimodal fusion, online learning, interpretable models, scalability, and real-world applications to further improve and extend the hybrid RNN-bi-LSTM method.

REFERENCES

[1] Ahmad, Wasim. Higher Education for Persons with Disabilities in India: Challenges and Concerns, Journal of Disability Management and Rehabilitation, 2017, 2, 1, 1-4.

[2] Yssel, Nina, Natalya Pak, Jayne Beilke. A door must be opened: Perceptions of students with disabilities in higher education. International Journal of Disability, Development and Education 63(3), 2016, 384-394

[3] Arsad, Pauziah Mohd, Norlida Buniyamin. A neural network students' performance prediction model (NNSPPM). In 2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA) IEEE 2013, 1-5.

[4] Conijn R., Snijders C., Kleingeld A., Matzat U. Predicting student performance from LMS data: A comparison of 17 blended courses using Moodle LMS. IEEE Trans. Learn. Technol. 2016, 10, 17–29.

[5] Meier Y., Xu J.. Atan O., Van der Schaar M. Predicting grades. IEEE Trans. Signal Process. 2015, 64, 959–972.

[6] Fernando Deller. Students Performance Prediction Framework of Teachers' Features. International Journal for Innovation Education and Research 9(2), 178-196.

[7] Romero C., Ventura S. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. In Educational Data Mining and Learning Analytics: An Updated Survey; John Wiley & Sons: Hoboken, NJ, USA, 2020; Volume 10, p. e1355.

[8] Xiao W., Ji P., Hu J.J.E.R. A survey on educational data mining methods used for predicting students' performance. Eng. Rep. 2022, 4, e12482.

[9] Pouyanfar S., Sadiq S., Yan Y., Tian H., Tao Y., Reyes M.P., Shyu M.L., Chen S.C., Iyengar S.S. A survey on deep learning: Algorithms, techniques, and applications. ACM Comput. Surv. 2018, 51, 1–36.

[10] Hernández-Blanco A., Herrera-Flores B., Tomás D., Navarro-Colorado B. A systematic review of deep learning approaches to educational data mining. Complexity 2019, 2019, 1306039.

[11] Guang-Yu L., Geng H. The Behavior Analysis and Achievement Prediction Research of College Students Based on XGBoost Gradient Lifting Decision Tree Algorithm. In Proceedings of the 2019 7th International Conference on Information and Education Technology—ICIET 2019, Aizu, Japan, 29–31 March 2019;

[12] Saputra E.P. Prediction of Evaluation Result of E-learning Success Based on Student Activity Logs with Selection of Neural Network Attributes Base on PSO. J. Phys. Conf. Ser. 2020, 1641, 012074.

[13] Giannakas F., Troussas C., Voyiatzis I., Sgouropoulou C. A deep learning classification framework for early prediction of team-based academic performance. Appl. Soft Comput. 2021, 106, 107355.

[14] Baashar Y., Alkawsi G., Mustafa A., Alkahtani AA., Alsariera YA., Ali AQ., Hashim W., Tiong SK. Toward Predicting Student's Academic Performance Using Artificial Neural Networks (ANNs). Applied Sciences. 2022; 12(3):1289.

[15] Rufai Aliyu Yauri, Hassan Umar Suru, James Afrifa, and Hannatu G. Moses. A Machine Learning Approach in Predicting Student's Academic Performance Using Artificial Neural Network, Journal of Computational and Cognitive Engineering, 2023, Vol. 3(2) 203–212.

[16] Senthamaraiselvi S., and Meenakshi Sundaram K. Reprecussion of Social Media in the Area of Education, Inernational Conference on Advanced Computing (ICAC), 2023.

[17] Senthamaraiselvi S.. and Meenakshi Sundaram K. A study on social networking usage amoung students, 4th International Conference on Artificial Intelligence Trending Towards Automation, 2023.

[18] Senthamaraiselvi S., and Meenakshi Sundaram K. Predicting Online Social Network Student Performance Using Enhanced RandomBayesian Algorithm, International Journal of Intelligent Systems and Applications in Engineering, 12(2), 384–390, 2023.

[19] Senthamaraiselvi S., MeenakshiSundaram K., Vandarkuzhali J. Hybrid Deep Neural Network for Predicting Online Social Network Student Performance, International Journal of Communication Networks and Information Security, 2024, 16(3), 6701.

Mrs. S.Senthamaraiselvi Pursuing Ph.D Computer science, Part time Research scholar in Erode Arts and Science College and is presently, Assistant professor in the Department of MCA at Vivekanandha Institute of Information and Management Studies, Tiruchengode. She Has 18 years of academic experience. she has published 5 papers in International Journals and has more than 20 presentaions in International and National conferences.

Dr.K.Meenakshi Sundaram is Research Supervisor, Former Associate Professor and Head, PG and Research Dept. of Computer Science, Erode Arts and Science College (Autonomous), Erode. He received his Ph.D.(part timre) in Computer Science from D.G.Vaishnav College, Chennai, He holds his Master of Computer Applications (MCA) from PSG College of Tech, Coimbatore and Master of Science (Mathematics) from PSG College of Arts and Science, Coimbatore. He has produced 9 P.hD Research and 45 M.phil scholars in computer science. He has published papers in 37 International journals (Scopus Indexed and UGC Approved) and 35 National Journals and has 55 presentations in International and National conferences to his credit. Has 35.5 years of academic experience and has held various positions at Erode Arts and Science College (Autonomous), Erode.

Dr. J. Vandarkuzhali has completed M.Sc.,M.Phil., Ph.D. and working as an Assistant Professor in the Department of Computer Science(Aided) in Erode Arts and Science College, Erode. She has been in the academic service since 2001. She has published 15 papers in the refered international journals. She is having 8 years of experience in research. Her areas of interest are Data Science and Internet of Things.