Predicting Student Performance Using Mental Health and Linguistic Attributes with Deep Learning

Bakyalakshmi Venkatachalami¹*, Kanchana Sivanraju²

¹ Department of Computer Science, PSG College of Arts and Science, Coimbatore 641014, India
² Department of Software Systems, PSG College of Arts and Science, Coimbatore 641014, India

Corresponding Author Email: vbakyalakshmi@gmail.com

https://doi.org/10.18280/ria.370408

Received: 1 May 2023
Revised: 21 May 2023
Accepted: 29 May 2023
Available online: 31 August 2023

Keywords: educational data mining, student performance, academic achievement, social media, mental health, linguistics, deep learning

ABSTRACT

Early prediction of students' academic performance is a critical research topic in educational data mining. ML models have been developed to predict academic performance, but it has become difficult to extract high-quality handcrafted attributes due to the large dataset. To solve this issue, a Deep Neural Network (DNN) was presented to automatically extract attributes from students' multi-source data. However, it didn’t deal with the student’s mental health and their mood changes (i.e., physiological attributes), which were also essential to increase the prediction performance. Thus, this article proposes a Student Accomplishment prediction using the Distinctive Deep Learning (SADDL) model. First, the student’s academic and demographic attributes are collected along with the posts shared by them about academic performance on online social networks. A Latent Dirichlet Allocation (LDA) is applied to extract the physiological attributes from the online posts data. Then, all the attributes of students' data are given to the SADDL network, which comprises three modules: (i) a Long Short-Term Memory (LSTM) module to learn the temporal dependency; (ii) a multidimensional Deep Convolutional Neural Network (DCNN) module to learn the correlation attributes and (iii) a Multi-Layer Perceptron (MLP) module to predict the students’ academic performance. Finally, the experimental results show that the SADDL can predict students’ academic performance with an accuracy of 91.71% while using 80% training and 20% test dataset, which is 9.91% improved than the existing ML models. Similarly, the SADDL has 89.3% while using 70% training and 30% test dataset, which is 10.36% higher than the existing ML models.

1. INTRODUCTION

Learning achievement is a crucial metric for assessing learning performance. Research has revealed that students who achieve poor grades are more likely to experience stress and hopelessness, and their probability of self-immolation [1] is significantly greater than that of students who perform well academically [2, 3]. Academies must recognize students at risk promptly to motivate their academic performance and provide early assistance [4, 5]. Teachers can recognize students who require extra classes, assignments, or encouragement to prevent undesirable behaviors [6, 7]. Educational data mining has been used to predict a student's learning performance [8]. Early identification of students with poor academic performance creates opportunities for research to improve those students' academic performance and guarantee their achievement [9]. High school students are the ideal target group for this research field due to their academic achievement having an impact on their future educational opportunities [10]. The dataset gathered from high school students contains socioeconomic and educational data about the students [11]. However, determining the academic achievement of all students is an extremely challenging task due to the large number of students and inadequate resources [12]. To solve this problem, numerous scholars have claimed in the literature that different types of ML models have been utilized to predict students' academic success [13-16].

Yağcı [17] presented a novel system based on ML models to predict students’ final exam grades by considering their midterm exam grades as the source data. A Multi-Layer Perceptron Neural Network (MLPN) was used by Hamadneh et al. [18] to predict students’ performance in blended learning and reduce academic failure rates. In the study [19], the authors aimed to predict a student's performance and provide support for academic guidance using a supervised ML.

Mostly, these studies demonstrated the strong correlation between different behavior factors and academic performance to offer guidance for teachers and enhance students’ academic achievements. But the manual extraction of attributes has difficulty while considering large-scale datasets. The quality and number of attributes directly influence the model training. They cannot completely define the temporal dependency of the time-series data of students. Also, a correlation among multi-source data needs to be further extracted. So, Li et al. [20] developed an end-to-end DNN model that automatically extracts attributes from students’ multi-source data to predict academic achievement. But they did not consider other factors such as the physiological factors of students to predict their academic achievements.

Hence, in this paper, the SADDL model is proposed by using diverse attributes of students, including physiological attributes (mental health and mood changes from the previous day), demographic attributes, and academic attributes. The main contributions of this study are the following:
• Initially, the students’ academic and demographic attributes are collected. Also, API services of Twitter, Facebook and Instagram are used to get the information shared by students. The collected information is pre-processed using NLP techniques.

• Then, the feature extraction is performed using the LDA scheme, which extracts the attributes regarding mental health and mood changes of students. These attributes, along with others, are learned by the SADDDL model, which ensembles the LSTM, DCNN and MLP models.

• The LSTM network is used to extract the time-series attributes of all data variants. The temporal dependencies are converted into a feature tensor and fed to the multidimensional DCNN to learn the correlation between multiple attributes.

• Then, a unified attribute vector is obtained by fusing the temporal dependencies, correlation attributes and student demographic attributes.

• Further, the MLP classifier is used to predict students’ academic performance. Thus, this SADDDL model enhances the efficiency of predicting students who are at risk in their academics.

The remaining article is prepared as follows: Section 2 presents the studies associated with the prediction of student learning achievement using ML models. Section 3 discusses the proposed algorithm and Section 4 shows its performance compared to the existing models. Section 5 concludes this study and suggests future enhancements.

2. LITERATURE SURVEY

One of the foremost essential problems in higher education is predicting students’ learning achievements. To tackle this problem, many scholars have employed educational data mining applications in decision making and predicting learning performance. This section reviews the related works from the aspects of predicting students’ performance and early identification of students with high risk using ML and deep learning models.

2.1 Student performance prediction using ML and deep learning models

The studies based on ML and deep learning models typically formulate the academic performance prediction problem as a classification process, intended at predicting students’ performance. Several studies were developed for predicting students’ performance using different data sources and models. Some of them are the following:

Chui et al. [21] presented an Improved Conditional Generative Adversarial Network-based Deep Support Vector Machine (ICGAN-D SVM) system for predicting students’ performance. But accuracy was less for large-scale datasets since the DSVM performed only well in a low-data volume scenario. Alshanqiti and Namoun [22] presented an intelligent hybrid model by combining cooperative filtering, fuzzy set rules, and Lasso linear regression methods to predict student academic grades and their influential factors. But the model’s reliability was not guaranteed because a publicly available, limited dataset did not contain every possible factor that may influence student accomplishments.

Li et al. [23] designed a combined Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and K-means models to find the correlation between students’ behavioral patterns and performance grades. But it was complex to apply this model to high-dimensional multisource behavior characteristics. Also, additional meaningful factors were needed to enhance the prediction performance.

Abdelkader et al. [24] presented the K-Nearest Neighbor (KNN) and SVM classifications to assess satisfaction levels with online learning for higher education students during COVID-19. But the KNN’s performance was degraded due to the slow training and the SVM has a high degree of complexity. Saidani et al. [25] suggested Extreme Gradient Boosting (XGBoost), Category Boosting (CatBoost) and Light Gradient Boosted Machine (LightGBM) to predict students’ performance. But other relevant student characteristics, like socio-demographic features and grades achieved in registered courses, were needed to improve the prediction accuracy.

Bujang et al. [26] suggested a multiclass prediction framework to predict students’ performance grades using J48, KNN, Naïve Bayes (NB), SVM, Linear Regression (LR) and Random Forest (RF). But it could be analyzed with an ensemble or deep learning models to evaluate the efficiency of the prediction framework. Li and Liu [27] presented DNN to extract informative features with corresponding weights in predicting students’ performance. But accuracy was not effective due to the use of less number of features. Hussain et al. [28] designed a regression model to predict the student’s academic performance utilizing deep learning. But it only considered the academic records of the students who had already completed their course, whereas other factors related to students’ behavior were essential to improve the accuracy.

Pu et al. [29] developed a graph convolutional network to predict students’ performance in the following semester according to their past final exam results. But the accuracy was very low because of the vague correlation in the graph or adjacency matrix, i.e., the correlation between relevant courses was poor. Poudyal et al. [30] developed a hybrid 2D CNN by integrating two distinct 2D CNN models to predict academic performance. But it did not analyze the influence of every attribute on the student’s performance. Also, the accuracy was less due to the small dataset. Yousafzai et al. [31] presented an attention-based bidirectional LSTM network to predict student performance from historical data. But it considered only significant features, whereas students’ demographic attributes, physiological factors, etc., may increase the prediction accuracy.

2.2 Early identification of at-risk student using ML and deep learning models

The student performance prediction can be also utilized to identify at-risk students, i.e., determine whether a student belongs to a low-performance group. It might be quite useful for early warning and feedback to be delivered to at-risk students before their final exam weeks. Many studies were developed for identifying at-risk students earlier with the aid of multiple data sources and models.

Chui et al. [32] developed a Reduced Training Vector-based SVM (RTV-SVM) to predict at-risk and marginal students. But accuracy was less and also needed to predict students’ performance. Sahlaoui et al. [33] applied ensemble methods such as extra trees and XGBoost with Shapley additive explanations to predict students’ performance and identify at-risk students. But the performance could be enhanced by using...
datasets with more attributes.

Adnan et al. [34] presented a predictive framework based on RF, SVM, KNN, extra tree, AdaBoost, gradient boosting and Artificial Neural Network (ANN) to characterize students’ learning behaviors, predict their performance scores and identify at-risk students. But additional textual attributes related to the student’s feedback were needed to increase prediction accuracy.

Zhao et al. [35] developed a framework called Augmented Education (AugmentED) to predict academic performance depending on the multisource and multi-feature behavior data of students. First, the characteristics defining dynamic changes in temporal life patterns were mined by the LSTM. Then, the RF, Gradient Boost Regression Tree (GBRT), KNN, SVM, and XGBoost were used to forecast academic performance. But it needs additional attributes about the student’s activities for effective prediction.

Nabil et al. [36] developed the DNN-based predictive model using the database gathered from an open 4-year university for students’ learning achievement prediction in new subjects according to their grades in earlier subjects and at-risk students’ identification. But it could be integrated with more semester data to increase the model's efficiency. Yang et al. [37] presented the CNN to predict at-risk students by transforming students’ course involvements into images. But it did not consider more complex factors that impact the prediction accuracy.

The above-discussed works predict students’ learning achievements based on different ML and deep learning models. But they have considered only specific kinds of attributes, whereas additional attributes are needed to enhance the prediction performance. Also, the prediction efficiencies mostly depend on the number of attributes mined manually by skilled experts, which makes the ML models difficult to perform when using a large-scale dataset with various kinds of student attributes. On the other hand, deep learning models did not learn temporal dependency among various attributes. So, this study focuses on developing a novel deep learning algorithm to automatically extract temporal dependency and correlation attributes among different kinds of attributes from the large-scale dataset for students’ performance prediction.

3. PROPOSED METHODOLOGY

This section explains the SADDL model for student performance prediction, which involves four major modules, as portrayed in Figure 1: data acquisition, pre-processing, feature extraction, and deploying a deep learner for prediction.

3.1 Data acquisition

In this study, the student records from the government and self-financed engineering colleges around Coimbatore, Tamil Nadu, are collected as a dataset. These records include academic and demographic attributes of the students who studied in the period from June 2022 to December 2022. The academic attributes include student number, course name, college category (i.e., government or self-financed), grades in each subject, study materials (i.e., lecture notes & book materials), way of teaching, the number of students in a course, an allowance of smartphones, etc. The demographic attributes are students’ name, age, sex, home place (i.e., rural, urban, or semi-urban), family category (i.e., nuclear or joint), career, academic skills of family aspirants, the guidance of parents for homework, the number of friends, time spent on watching TV, internet access at home, etc. This dataset contains 126 attributes (i.e., columns), 5000 instances (rows) and 1 class attribute (i.e., either distinction, pass, high-distinction, or fail). After obtaining this dataset, preprocessing is used to convert the textual/categorical data into numeric form using one-hot encoding. Using this dataset, the academic and demographic attributes of the students are directly extracted.

In addition to this dataset, the number of conversations and posts shared by the students about their academic experiences on Twitter, Facebook and Instagram are acquired as a students’ social media dataset. This dataset is preprocessed and used to extract the students’ physiological attributes, which are described in the following section.

3.2 Students’ physiological attribute extraction

3.2.1 Pre-processing

The post and conversation information in the students’ social media dataset is preprocessed using the following NLP techniques.

- Eliminate punctuation, emojis, unique typescripts and excess spacing.
- Remove duplicate characters from the string (e.g., “Gooood Mormrrninggg” becomes “Good Morning”, etc.).
- Delete numeric values and translate words to lowercase.
- Enlarge contractions (e.g., “I’m” becomes “I am”, “I’ve” becomes “I have”, “can’t” becomes “cannot”, and so on).
- Eliminate stopwords (e.g., the, an, let, with, and so on) using the NLTK dictionary that includes 40 stop words.
- Lemmatize words using the WordNet to translate the words to their root form (e.g., “studied” becomes “study”, “drawn” becomes “draw”, and so on). This study uses lemmatization rather than stemming since it has higher accuracy for context analysis.

These preprocessing steps can reduce the total amount of information in the student’s social media dataset by removing redundant data. This preprocessed dataset is used to extract the students’ physiological attributes.

3.2.2 LDA feature extraction

LDA is the other feature extraction scheme utilized to create topic models by allocating a topic of each word after extracting it in a dataset. It is a probabilistic model that extracts latent topics from a group of documents. The major aim is that the student database is defined as an arbitrary combination of latent topics, which are represented by a distribution over words (e.g., physiological terms often used by the students). For a given dataset D containing M documents, with document d having N_{d} words (d ∈ 1, …, M), the LDA models D based on the below generation procedure.

- Choose a multinomial distribution \( \phi \), for the topic \( t \), (\( t \in \{1, \ldots, T \} \)) from a Dirichlet distribution with a
variable $\beta$;

- Select a multinomial distribution $\phi_d$ for $d$, ($d \in \{1, \ldots, M\}$) from a Dirichlet distribution with a variable $\alpha$;
- For a word $W_n$, ($n \in \{1, \ldots, N_d\}$) in $d$
  - Choose a topic $Z_{dn}$ from $\phi_d$;
  - Select $W_n$ from $\varphi_{Z_{dn}}$;

According to this generation procedure, words in documents are only observed variables, whereas others are latent variables ($\varphi, \phi$) and hyperparameters ($\alpha, \beta$). To infer these variables, the probability of the observed data $D$ is calculated and improved by

$$P(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\phi_d|\alpha) \left( \prod_{n=1}^{N_d} \sum_{Z_{dn}} p(Z_{dn}|\phi_d) p(W_{dn}|Z_{dn}, \beta) \right) d\phi_d$$

(1)

$\alpha$ parameters of the topic Dirichlet prior and the distribution of words over topics, which are obtained from the Dirichlet distribution and provided by $\beta$. Here, $N$ denotes the size of vocabulary. The Dirichlet multinomial pair for dataset-level topic distributions ($\alpha, \beta$) is taken. The Dirichlet multinomial pair for topic-word distributions is provided by $\beta$ and $\varphi$. Variables $\phi_d$ are the document-level variables. $Z_{dn}$ and $W_{dn}$ are the word-level variables, which are samples for all words in all documents.

Figure 2 illustrates an example of LDA-based physiological feature extraction. It is noted that the LDA provides the words in the topics with their distribution using Dirichlet distribution. A topic is defined as a weighted list of words (i.e., students’ physiological attributes). Thus, the LDA scheme is performed to extract the physiological attributes related to the student’s mental health and mood changes from the student social media database.

### 3.3 SADDL model for prediction

As illustrated in Figure 3, the SADDL-based prediction model comprises four major stages: (i) Input: different attributes of student physiological, academic, and demographic data are the input of the DDL model; (ii) Temporal analysis: the LSTM is applied to independently model each category of the attribute to learn their temporal dependency; (iii) Correlation analysis: a multidimensional DCNN is applied on the tensor converted from the temporal dependency of each data type to learn the correlation among multiple attributes; (iv) Output: the student’s physiological, academic, demographic, temporal, and correlation attributes are fused as the input of the MLP classifier for predicting learning achievements.

Figure 2. Example of LDA for students’ physiological feature extraction

![Figure 2: Example of LDA for students’ physiological feature extraction](image)

Figure 3. Structure of SADDL for student’s performance prediction

![Figure 3: Structure of SADDL for student’s performance prediction](image)
3.3.1 Input
This network model considers the different kinds of student information such as academic background, demographic data and physiological data. Each kind of student’s data is a time-series, i.e., all records have a timestamp, yet different students hold distinct attributes. Here, $X_i = (x_{1i}, ..., x_{li}, ..., x_{Ni})$ is the N categories of multi-source attributes of student i, $x_{ij} = [x_{1ij}, ..., x_{lij}, ..., x_{Nij}]$ is the jth attribute of student i, $x_{ij} (1 \leq t \leq T_{ij})$ is the vector defining single event record information at period t, including single consumption record, single gateway login record, where $T_{ij}$ is the length of the jth attribute of student i.

3.3.2 Temporal analysis of attributes using LSTM network
LSTM is a kind of recurrent network, which is used to process a sequence of values. It can efficiently diminish the complexity of extracting long-term dependencies from long-range sequences. To extract temporal dependency in each attribute from the students’ data, the LSTM is applied. The LSTM network layer comprises the input gate, forget gate and output gate. The forget gate is used to calculate a degree of forgetting a feature preceded by the current LSTM unit as Eq. (2):

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

(2)

In Eq. (2), $W_f, b_f$ are the weight vector and bias value of the forget layer, respectively. $\sigma$ is the sigmoid activation function, $x_t$ is the input feature in the input gate, $f_t$ is the forget gate, and $h_t$ is the result of a previous hidden state.

The input gate is used to determine how much present input feature is contained in the image as Eqns. (3), and (4):

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

(3)

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

(4)

Here, $W_i, W_c$ are the weight vector of the input gate and neuron condition vector, respectively. $b_i, b_c$ are the bias values of the input gate and neuron condition vector, respectively. $\tanh$ is the hyperbolic tangent activation function. $i_t, \tilde{C}_t$ are the input gate, and the updated new cell state, respectively.

Once the features traverse via the input and forget gates, the LSTM fine-tunes their units to determine the outcome of the current LSTM unit and pass it to the consecutive LSTM unit as Eq. (5):

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

(5)

In Eq. (5), $C_t$ is the current cell state, and $C_{t-1}$ is the old cell state. The output gate merges the present input and LSTM unit to compute the result of the present LSTM unit as:

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

(6)

$$h_t = o_t \ast \tanh(C_t)$$

(7)

In Eqs. (6) and (7), $h_t$ represent the hidden state that serves as the solution of the block over t, $o_t$ is the output gate, $W_o$ and $b_o$ are the weight vector and bias value of the output gate, respectively.

3.3.3 Correlation analysis of attributes using DCNN
Since multi-source performance information is from a similar student, a relationship among various attributes should exist. To learn the correlation among multiple attributes, a tensor approach is applied to convert the temporal dependency vector of every attribute into a 2D DCNN is adopted to extract the correlation among multiple attributes. This DCNN is utilized to mine image characteristics, where an image is defined by a tensor $(o,h,c)$, where o, h and c denote the width, height and number of channels, respectively. As depicted in Figure 2, the temporal dependency vectors of N kinds of attributes are converted into a 2D tensor, where $o=\text{h}=\text{c}=M$, $M$ is the dimension of the temporal dependency vector. Based on the tensor, the 2D DCNN is performed.

3.3.4 Output or Multi-Layer Perceptron (MLP) classifier
In this study, the learning performance prediction is modeled as a categorization process, the network result is $y \in \{0: \text{Distinction}, 1: \text{Fail}, 2: \text{High Distinction}, 3: \text{Pass}\}$, and the grading process can be performed as follows:

A student’s academic performance is typically measured by Grade Point Average (GPA) of continuous numerical values. This work defines the prediction of academic performance as the classification task, i.e., estimating whether the student’s performance is a distinction, fail, high distinction, or pass. So, the GPA is split into discrete academic grades. First, each student in the database is ranked based on the GPA in descending manner, then the top k% of students’ achievements such as 85-100% are defined as high distinction and the bottom k% of students’ achievements such as 0-49% are defined as failure. The other student’s achievements such as 75-84% are defined as distinction, and 50-64% are defined as pass.

The MLP classifier is trained by fusing the students’ academic, demographic, physiological, temporal, correlation and demographic attributes to provide the learning grade. In the MLP classifier (as shown in Figure 4), the input layer has n neurons representing the number of attributes. The second layer is the hidden layer, in which the number of neurons should be less than twice the size of the input layer. The Rectified Linear Unit (ReLU) activation function is used for both input and hidden layers. The final layer is the output layer, which comprises the softmax activation for final prediction.

The Softmax $(P(y,b))$ is a probability of attributes belonging to a given grade y. It is determined by Eq. (8).

$$P(y,b) = \frac{P(b,y) \times P(y)}{\sum_{y=1}^{N} P(b,y) \times P(y,b)}$$

(8)
In Eq. (8), \( P(y) \) is grade probability, \( y \) is the total number of grades. Eq. (8) is rewritten as Eq. (9):

softmax = \( P(y, b) = \frac{e^{\beta y[b]}}{\sum_{b=1}^{B} e^{\beta N[b]}} \) \hspace{1cm} (9)

Where \( \beta y[b] = \ln[p(b, y) \times P(y)] \)

Figure 5 presents a flow chart of the SADDL model for students’ performance prediction, which helps to understand how the model works.

### 3.3.5 Model configuration

Table 1 presents an entire configuration of the SADDL model. The initial layer denotes the input containing \( N \) kinds of attributes, where \( T_i \) and \( F_i \) are the data sequence length and attribute number of \( i^{th} \) student. The second layer applies the LSTM on all kinds of attributes and provides a vector having 32 attributes. In the 3rd, 4th and 5th layers, the concatenation, reshape and permute layers are used to translate the temporal dependency of \( N \) types of attributes into a tensor of \((o, h, 32)\). Because there are only some kinds of physiological behaviors, more 2D convolutional layers may tend to overfit; so, only 2 convolutional layers are configured in the 6th and 7th layers, with the kernel number of 32 and 64, respectively. The kernel sizes of (2,2) and the step sizes of (1,1), as well as, the 6th layer use filling mode to maintain the tensor size unchanged.

The tenth layer considers the students’ academic, demographic and physiological attributes as input. In the 11th layer, such attributes are concatenated with the temporal dependency and correlation attributes, where \( L_7 \), \( L_8 \) and \( L_1 \) denote the lengths of the output vectors in the 7th, 8th and first layers, respectively. The concatenated attribute vector is given as input to the MLP classifier in the 12th layer, where 4 denotes the grades of academic performance and the activation function is softmax. The dropout layer is assigned before the MLP layer to prevent overfitting and the dropout rate is set to 0.5.

### 3.3.6 Handling class imbalance problem using weighted cross-entropy error

To handle class imbalance problems, under-sampling, over-sampling and weighted loss functions are available. On the other hand, under-sampling may reduce the number of samples, which degrades the efficiency of SADDL training and over-sampling is computationally inefficient for high-dimensional datasets. Compared to these schemes, the weighted loss function needs fewer computing resources to handle class imbalance problems. As a result, the weighted cross-entropy error is considered as the loss function in this paper. The weighted cross-entropy error is defined as follows:

\[
loss = \frac{1}{N} \sum_{k=1}^{N} \sum_{c=1}^{M} w_c y_c^k \log(p_c^k) \tag{11}
\]

Where \( w_c = \frac{N}{M+N_c} \)

In Eqs. (11) and (12), \( w_c \) is the weight of label \( c \), \( N \) indicates the total amount of student data instances, \( N_c \) denotes the amount of student data instances belonging to \( c \), \( M \) denotes the amount of class labels, \( y_c^k \) indicates the actual grade of \( k^{th} \) data instances belonging to \( c \), and \( p_c^k \) defines the estimated grade probability.

### 3.3.7 Preventing overfitting issue

A huge gap between the number of model parameters and the number of student data made the SADDL model prone to overfitting. To prevent overfitting, this study applies early stopping and dropout during the model training. If the error value is less than 0.001 for 40 consecutive epochs, the training is terminated and the parameter setting of dropout is given in Table 1.

#### Table 1. Entire configuration of SADDL model

<table>
<thead>
<tr>
<th>Layer</th>
<th>Kernel Value</th>
<th>Kernel Dimension</th>
<th>Result Size</th>
<th>Activation</th>
<th>Linked Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>-</td>
<td>-</td>
<td>((T, F, e)\times N)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>LSTM</td>
<td>-</td>
<td>-</td>
<td>((32, 1)\times N)</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>Concatenate</td>
<td>-</td>
<td>-</td>
<td>((32, N))</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>Reshape</td>
<td>-</td>
<td>-</td>
<td>((32, o))</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>Permute</td>
<td>-</td>
<td>-</td>
<td>((o, h, 32))</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>Conv2D</td>
<td>32</td>
<td>(2,2)</td>
<td>((o, h, 32))</td>
<td>ReLU</td>
<td>5</td>
</tr>
<tr>
<td>Conv2D</td>
<td>64</td>
<td>(2,2)</td>
<td>((o-1, h-1, 64))</td>
<td>ReLU</td>
<td>6</td>
</tr>
<tr>
<td>Flatten</td>
<td>-</td>
<td>-</td>
<td>(((o-1)*(h-1)\times 64))</td>
<td>ReLU</td>
<td>4</td>
</tr>
<tr>
<td>Flatten</td>
<td>-</td>
<td>-</td>
<td>((32\times N))</td>
<td>-</td>
<td>8</td>
</tr>
<tr>
<td>Concatenate</td>
<td>-</td>
<td>-</td>
<td>((L_7+L_8+L_1))</td>
<td>-</td>
<td>1, 7, 8</td>
</tr>
<tr>
<td>Dropout (0.5)</td>
<td>-</td>
<td>-</td>
<td>((L_7+L_8+L_1))</td>
<td>-</td>
<td>9</td>
</tr>
<tr>
<td>MLP</td>
<td>-</td>
<td>-</td>
<td>(4)</td>
<td>Softmax</td>
<td></td>
</tr>
</tbody>
</table>
4. EXPERIMENTAL RESULTS

The effectiveness of the SADDL model is measured and evaluated with the existing ML models using the considered student dataset (discussed in Section 3.1) in MATLAB 2019b. For performance analysis, two different data split ratios are used:

(i) 4000 (i.e., 80%) data instances (1000 from each class) are used for training and 1000 (i.e., 20%) data instances (250 from each class) are used for testing.

(ii) 3500 (i.e., 70%) data instances (875 from each class) are used for training and 1500 (i.e., 30%) data instances (375 from each class) are used for testing.

The existing models considered for performance analysis are SVM [24], LightGBM [25], RF [26], DNN [27] and GBRT [35].

4.1 Parameter settings

Table 2 lists the parameters settings for proposed SADDL and existing models: SVM [24], LightGBM [25], RF [26], DNN [27] and GBRT [35] to compare the efficiency of academic performance prediction.

<table>
<thead>
<tr>
<th>Models</th>
<th>Parameters</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBRT [22]</td>
<td>Number of trees</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Learning rate</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>Maximum tree depth</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Minimum Split gain</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Minimum Leaf weight</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>Row sampling</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Column sampling</td>
<td>1.0</td>
</tr>
<tr>
<td>RF [24]</td>
<td>Number of trees in the forest</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Maximum tree depth</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Number of hidden layers</td>
<td>3</td>
</tr>
<tr>
<td>DNN [26]</td>
<td>Number of neurons at kth hidden layer</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>Number of neurons at output layer</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Batch size</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>Learning rate</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Activation function</td>
<td>ReLU</td>
</tr>
<tr>
<td></td>
<td>Optimizer</td>
<td>Adam</td>
</tr>
<tr>
<td></td>
<td>Number of epochs</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Loss function</td>
<td>Cross-entropy</td>
</tr>
<tr>
<td></td>
<td>Weight regularization</td>
<td>0.001</td>
</tr>
<tr>
<td>SVM [27]</td>
<td>Kernel type</td>
<td>Linear</td>
</tr>
<tr>
<td></td>
<td>Kernel degree</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Penalty</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>Gamma</td>
<td>0.01</td>
</tr>
<tr>
<td>LightGBM [29]</td>
<td>Number of boosted trees to fit</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Minimum sum of instance weight</td>
<td>4.1</td>
</tr>
<tr>
<td></td>
<td>Maximum tree depth</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Minimum loss reduction</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>L1-regularization on weights</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>L2-regularization on weights</td>
<td>4.2</td>
</tr>
<tr>
<td></td>
<td>Learning rate</td>
<td>0.02</td>
</tr>
<tr>
<td>Proposed SADDL</td>
<td>Learning rate</td>
<td>0.0001</td>
</tr>
<tr>
<td></td>
<td>Batch size</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>Loss function</td>
<td>Weighted cross-entropy</td>
</tr>
</tbody>
</table>

4.2 Performance metrics

The accuracy, precision, recall and f-measure provide better insights into the prediction compared to the other metrics. It helps to understand the efficiency of prediction models when considering new unlabeled data. So, this study considers these metrics for performance analysis. The performance metrics used to evaluate the proposed and existing models are described below:

- **Accuracy**: It is the percentage between a proper prediction of students’ grades and the total number of predictions performed.

\[
\text{Accuracy} = \frac{\text{True Positive (TP)} + \text{True Negative (TN)}}{\text{TP} + \text{TN} + \text{False Positive (FP)} + \text{False Negative (FN)}}
\] (13)

TP refers to the amount of positive data that are predicted as positive, TN refers to the amount of negative data that are predicted as negative, FP refers to the amount of negative data that are predicted as positive, and FN refers to the amount of positive data that are predicted as negative.

- **Precision**: It is the percentage of properly predicted students’ grade classes at TP and FP rates.

\[
\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}
\] (14)

- **Recall**: It is the percentage of properly predicted students’ grade classes at TP and FN rates.

\[
\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}
\] (15)

- **F-measure**: It is determined in Eq. (17):

\[
F - \text{measure} = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}
\] (16)
- RMSE: It is the square root of the average of the square of every error. It is the variance of each observed data \( (y_g) \) and correctly predicted data \( (p_g) \).

\[
RMSE = \sqrt{\frac{\sum (y_g - p_g)^2}{N}} \quad (17)
\]

In Eq. (17), \( N \) is the total number of instances.

Table 3 demonstrates the confusion matrix of the SADDL for student academic performance prediction using 20% test dataset. It is a most well-known representation of the prediction model performance. It illustrates the number of correctly and incorrectly predicted grades, compared to the actual outcomes (target result) in the test data. Using this matrix, TP, FP, FN and TN values for each class are measured, which are given in Table 4. So, Table 4 is used to calculate the accuracy, precision, recall and f-measure values of SADDL model using 20% test dataset.

<table>
<thead>
<tr>
<th>Actual</th>
<th>Class</th>
<th>Distinction</th>
<th>Fail</th>
<th>High-distinction</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted</td>
<td></td>
<td>229</td>
<td>7</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Distinction</td>
<td>7</td>
<td>230</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>High-distinction</td>
<td>6</td>
<td>7</td>
<td>229</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Pass</td>
<td>8</td>
<td>6</td>
<td>7</td>
<td>229</td>
</tr>
</tbody>
</table>

Table 4. Detailed statistics by each class for SADDL using 80% training and 20% test dataset

<table>
<thead>
<tr>
<th>Class</th>
<th>TP</th>
<th>FP</th>
<th>FN</th>
<th>TN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinction</td>
<td>229</td>
<td>21</td>
<td>21</td>
<td>729</td>
</tr>
<tr>
<td>Fail</td>
<td>230</td>
<td>20</td>
<td>20</td>
<td>730</td>
</tr>
<tr>
<td>High-distinction</td>
<td>229</td>
<td>21</td>
<td>21</td>
<td>729</td>
</tr>
<tr>
<td>Pass</td>
<td>229</td>
<td>21</td>
<td>21</td>
<td>729</td>
</tr>
</tbody>
</table>

Table 5 demonstrates the confusion matrix of the SADDL for student academic performance prediction using 30% test dataset. Using this matrix, TP, FP, FN and TN values for each class are measured, which are given in Table 6. So, Table 6 is used to calculate the accuracy, precision, recall and f-measure values of SADDL model using 30% test dataset.

<table>
<thead>
<tr>
<th>Actual</th>
<th>Class</th>
<th>Distinction</th>
<th>Fail</th>
<th>High-distinction</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted</td>
<td></td>
<td>333</td>
<td>12</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Distinction</td>
<td>15</td>
<td>331</td>
<td>18</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>High-distinction</td>
<td>15</td>
<td>15</td>
<td>336</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Pass</td>
<td>12</td>
<td>17</td>
<td>6</td>
<td>340</td>
</tr>
</tbody>
</table>

Table 6. Detailed statistics by each class for SADDL using 70% training and 30% test dataset

<table>
<thead>
<tr>
<th>Class</th>
<th>TP</th>
<th>FP</th>
<th>FN</th>
<th>TN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinction</td>
<td>333</td>
<td>42</td>
<td>42</td>
<td>1083</td>
</tr>
<tr>
<td>Fail</td>
<td>331</td>
<td>44</td>
<td>44</td>
<td>1081</td>
</tr>
<tr>
<td>High-distinction</td>
<td>336</td>
<td>39</td>
<td>39</td>
<td>1086</td>
</tr>
<tr>
<td>Pass</td>
<td>340</td>
<td>35</td>
<td>35</td>
<td>1090</td>
</tr>
</tbody>
</table>

Figure 6. Analysis of precision, recall, and f-measure for different models using 80% training and 20% test dataset

Figure 6 plots the precision, recall, and f-measure of different students’ academic performance prediction models on the 80% training and 20% test dataset. It notices that the SADDL achieves greater precision, recall, and f-measure compared to the other ML models due to the use of various attributes, which are learned automatically and without difficulty. The SADDL increases the precision by 17.7%, 15%, 8.5%, 8.1%, and 5.3%, in contrast with the LightGBM, GBRT, RF, SVM, and DNN, respectively. The recall of the SADDL is maximized by 16.9%, 14.1%, 8.8%, 8.3%, and 5.9% compared to the LightGBM, GBRT, RF, SVM, and DNN models, respectively. Also, the SADDL enhances the f-measure values by 17.2%, 14.5%, 8.6%, 8.1%, and 5.6% compared to the LightGBM, GBRT, RF, SVM, and DNN, respectively.

Figure 7. Analysis of precision, recall, and f-measure for different models using 70% training and 30% test dataset

Figure 7 illustrates the precision, recall, and f-measure of different students’ academic performance prediction models on the 70% training and 30% test dataset. The precision of the SADDL is increased by 19.1%, 16.45%, 12.18%, 10.17% and 5.36% compared to the LightGBM, GBRT, RF, SVM and DNN, respectively. The recall of the SADDL is increased up to 19.19%, 16.42%, 11.98%, 10.07% and 5.71% compared to the LightGBM, GBRT, RF, SVM and DNN models.
respectively. The f-measure of the SADDL is enhanced by 19.15%, 16.43%, 12.08%, 10.12% and 5.53% compared to the LightGBM, GBRT, RF, SVM and DNN, respectively.

From Figures 6 and 7, it is observed that the prediction model achieves better performance while considering more training data, i.e., when using 80% training data, the SADDL can be trained efficiently and predicted the students’ performance with better accuracy, compared to the use of 70% data for training.

In Figure 8, the impact of each category of attribute using the 80% training and 20% test dataset is drawn in terms of prediction accuracy. Even though each attribute contributes good accuracy in predicting students’ performance independently, they are not sufficient to satisfy the prediction efficiency. Because each attribute defines particular characteristics of students, e.g., academic attributes define the students’ learning behavior, demographic attributes define the students’ family background and personal behavior, whereas physiological attributes define the students’ mental health about academic activities. Usually, students’ academic performance depends on all these attributes. So, the proposed SADDL considered all attributes about students’ academic, demographic and physiological information for prediction process. It is noted that the SADDL effectively maximizes the accuracy of predicting student academic performance by considering all categories of attributes, compared to the use of each attribute separately in the previous ML models. On average, the SADDL using all attributes increases the accuracy by 15.9%, 13.4%, 8.2%, 7.7% and 5%, in contrast with the LightGBM, GBRT, RF, SVM and DNN models, respectively.

Figure 9 demonstrates the impact of each attribute category using the 70% training and 30% test dataset in terms of prediction accuracy. It is observed that the accuracy of the SADDL model on the combined attributes is increased by 15.02%, 12.92%, 9.8%, 8.45% and 6.06% compared to the LightGBM, GBRT, RF, SVM and DNN models, respectively. Thus, it is inferred that more training instances can train the model well for the prediction process compared to the less training instances.

Table 7 portrays the RMSE values obtained by the different models for predicting students’ academic performance using all kinds of attributes. It is shown that the SADDL using 20% test dataset can decrease the RMSE by 36.99%, 34.07%, 26.35%, 25.36% and 18.52%, respectively, compared to the LightGBM, GBRT, RF, SVM and DNN models. Also, the RMSE of the SADDL using 30% test dataset is reduced by 28.61%, 25.97%, 22.88%, 14.14% and 10.55% compared to the LightGBM, GBRT, RF, SVM and DNN models, respectively. Thus, it is concluded that the SADDL enhances the prediction performance by using more training instances including all kinds of students’ attributes in contrast with the existing LightGBM, GBRT, RF, SVM and DNN models.

5. CONCLUSION AND FUTURE WORK

In this paper, the SADDL model was developed for students’ learning achievement prediction. This model considered the students’ physiological attributes extracted from their posts or interactions in social media using the Latent Dirichlet Allocation (LDA). Such attributes along with the student’s academic and demographic attributes were then learned by the SADDL model to efficiently predict the academic performance grades. At last, the experimental results proved that the SADDL on the students’ dataset has 91.71% accuracy.
while using 80% training and 20% test dataset, whereas it has 89.3% accuracy when using 70% training and 30% test dataset having all attributes rather than individual attributes compared to the LightGBM, GBRT, RF, SVM and DNN models.

Thus, this model can be helpful for teachers to early predict at-risk students in specific courses and provide proper guidance to reduce academic failure rates. On the other hand, the dataset was limited. Also, the dataset acquisition was time-consuming, labor-intensive and expensive. To improve the performance of SADDL, a large-scale dataset is essential. So, future work will focus on developing a deep generative network model to augment the number of students’ data without complexity and increase the academic performance prediction accuracy.

REFERENCES


regression and multi-label classification. IEEE Access, 8: 203827-203844. https://doi.org/10.1109/ACCESS.2020.3036572


