[1]Ashalakshmi R,
[2]S Hemalatha

Hybrid LCNN for Beneficial Student Career Prediction: Leveraging the Potential of LSTM and Optimized CNN

JES Journal of Electrical Systems

Abstract: - Student career prediction has been done recently by measuring the variables that might affect career decisions with the use of questionnaires or diagnostics. However, it is challenging to accurately predict someone's professional choices because of the complexity of each person's beliefs and ambitions. This paper uses behavioural data from students to forecast career options to overcome this challenge. Rigorous pre-processing procedures are used to clean and validate initially collected data. Then the feature extraction includes central tendency measures, quantitative measures of dispersion, and Mean Absolute Deviation (MAD) to characterize the dataset's distribution and variability. After feature extraction, introduce a new Hybrid LCNN method, which combines long-term and short-term memory (LSTM) with optimized Convolutional Neural Network (CNN) architecture. This optimized CNN and LSTM get extracted features in parallel alongside finely tuned time model training. Then introducing the Barnacle Pathfinder Optimization Algorithm (BPOA) which combines the concepts of Barnacles Mating Optimizer (BMO) and Pathfinder Algorithm (PFA), hyperparameter optimization of LSTM and CNN Also, use BPOA models to solve CNN weights, and effect on the final forecast result Increase it. The predictions produced by the LSTM and optimized CNN components are combined using a weighted mean to exploit the synergistic effects of the two architectures. The proposed Hybrid LCNN model achieved the accuracy of 99.25% for learning rate 70% and 99.48% for learning rate 80%.

Keywords: Student career prediction, Mean Absolute Deviation, Long Short-Term Memory, Barnacle Pathfinder Optimization Algorithm, and Convolutional Neural Network.

INTRODUCTION

Student career is an important part of their lives as it determines their future prospects and is crucial for their personal and professional development. A student can achieve financial security, work fulfilment, and a feeling of significance in life by making a wise career decision [1]. Along with social recognition and distinction, it can also offer chances for career and personal growth. Overall, a successful career that has been well-planned can positively impact a student's confidence, self-worth, and general quality of life. Student academic achievement has always had a significant role in defining both the status of the institution and the future of the student [2]. From the very beginning of their education, pupils must be prepared and structured in order to compete and accomplish the goal. Therefore, it is critical to continuously review their performance, determine their areas of interest, gauge their progress toward their objectives, and determine whether they are on the correct route leading to their intended destination [3]. This aids in their self-improvement, encouraging them to pursue a better job route in the event that their abilities fall short of their objectives, and enabling them to assess themselves prior to reaching the career peak point [4].

Selecting a career is a difficult task these days. For the most part, this is because students lack the business development and understanding that comes with having to work from an early age [5]. Also, when choosing a job route that will yield the greatest outcomes, the students are heavily impacted. Insufficient knowledge prevents students from making independent judgments, which may lead to issues in the road [6]. Students should choose the best-paying employment for them in order to prevent this issue in the future [7]. By selecting a career path that is unsuitable for them, people run the risk of landing a job they detest or having little experience in. The students look to fortune tellers

[1] Department of Computer Science, Karpagam Academy of Higher Education, Eachanari, Coimbatore, 641021, Tamilnadu, India.
Corresponding author(s). E-mail(s): ash.lekshmi@gmail.com

in the hopes that they can help them make decisions because they are indecisive. To make the most accurate predictions about the future, avoid depending on fortune tellers [8].

As the most well-defined job option worldwide, architecture is one of the greatest career streams, with the exception of healthcare, which most students choose, sometimes out of passion and sometimes under duress from their parents [9]. Every year, so many engineering students graduate from college. After graduating, a lot of students select their career stream. With so many career possibilities and job contest available today, choosing the appropriate occupation has become a difficult process. They frequently choose careers that are not a good fit for their interests, skills, and personalities [10]. Due to familial pressure and the need for high salary, some students are even pressured to choose an engineering career path. They are dissatisfied with the former students who finished engineering school and began working for multinational corporations but who lacked enthusiasm and aptitude. Thus, the younger generation is beginning to choose the streams that appeal to them [11]. Additionally, each track offers a variety of job options, and students are presented with several employment chances in every sector, exposing them to a wide range of aspects related to career choices [12].

However, academic achievement and student sociodemographic information are the two most crucial subjects in the area of educational data mining. Every educational institution depends heavily on the performance of its students, and this performance is typically assessed using the students' final grades [13]. Students' final grades are determined by their assessment results, major test scores, course design, and participation in various extracurricular activities offered by the school. These data, which include sociodemographic information and student grades, can be retrieved to provide insightful information on the student's standing in every class or subject [14]. Making an appropriate selection of career paths for senior secondary school students is a challenge that this recommendation engine could solve using techniques and methods from statistics, data mining, and neural networks. Using neural networks, this kind of technology may also help students choose the best career path in senior high school based on their skills and abilities, which will boost their happiness and foster a beneficial connection between the school and the students [15]. The primary contributions of the paper are given as,

- A hybrid LCNN model combining LSTM and optimized CNN architecture is proposed, where the hyper parameter is optimized during training for improved prediction performance.
- A new optimization algorithm called BPOA is introduced, that combines the concepts of Barnacles Mating Optimizer (BMO) and Pathfinder Algorithm (PFA) for hyperparameter tuning of LSTM and CNN, thus improving the accuracy of the model.
- Using the BPOA model to adjust the weights of the CNN component, increase its impact on the final forecast results, and potentially improve the overall performance of the model.

The study is structured as follows: Section 2 discusses recent research that has been published; Section 3 provides a full explanation of the suggested methodology; Section 4 discusses results obtained using current methodologies; and Section 5 provides a thorough conclusion to wrap up the paper.

1. LITERATURE REVIEW

In 2022, Zeineddine, *et. al.,* [16] have suggested automated machine learning (AML) methods to increase the accuracy of student performance forecasts based on data that is already available prior to the start of the course. Many initiatives and research projects have focused on the use of academic and behavioural data from students to classify them and predict their future performance using advanced statistics and machine learning.

In 2021, Shen, *et. al.,* [17] have examined how 483 Chinese college students' primary self-evaluations, career calling, and challenges making career decisions relate to one another. The findings indicate while career calling and core self-evaluation both adversely predict professional decision-making challenges, core self-evaluation positively predicts career calling. In order to further address the challenges associated with vocational decision-making, college students' sense of calling in life may be further developed through strengthening their fundamental self-evaluation.

Bai & Hira [18] have proposed a hybrid model for predicting students' employability that combines Deep Belief Networks and softmax Regression (DBN-SR). To achieve data consistency, pre-processing is first applied to the student data to eliminate unnecessary features. Additionally, in order to improve the prediction model's performance, the best subset of original features is chosen using a feature selection model based on the crow search method. The accuracy of the suggested employability prediction model is more than 98%.

Guleria & Sood [19] have presented a method for using AI and ML in student career counseling. Employability and academic traits that are critical to student skill development and job placements are examined in the educational dataset by machine learning-based White and Black Box models. Under the proposed system, an instructional dataset provided by the study is used to train White Box and Black Box models. The predictions came true in 97% of cases.

In 2022, Quinlan & Renninger [20] have looked at how students' interests grow and how that relates to their decision about their careers and desire for fulfilling, engaging job. Results indicated that the majority of university students majoring in science had established interests that drove their program selection, and there was a correlation between topic interest and career decision-making. Through the desire to follow that interest in their profession, students' interest in their topic was found to be a strong predictor of career decidedness, according to regression studies.

Yagci, et. al., [21] have proposed a revolutionary machine learning algorithm-based model that uses midterm exam results as input data to predict undergraduate students' final test scores. A comparison of the outcomes of machine learning algorithms, including Naive Bayes, logistic regression, random forests, nearest neighbor, support vector machines, and k-nearest neighbor, was conducted in order to predict the final exam marks of the students. According to the findings, the suggested model has a 70–75% classification accuracy.

Zeng, et. al., [22] have investigated the role that professional flexibility and academic self-efficacy play as two potential mediators between future work self and life fulfilment and hope. 636 Chinese vocational high school students answered questions about career adaptability, academic self-efficacy, hope, future work self, and life happiness. According to the research, job adaptability, academic self-efficacy, hope, and life satisfaction were highly connected.

In 2022, Kehinde, et. al., [23] have created an Artificial Neutral Network (ANN) to forecast student performance based on demographic characteristics of the student. This will help the university choose applicants for admission who have a high likelihood of succeeding by analysing the past academic records of those admitted, ultimately producing high-caliber graduates. With an accuracy of more than 92.3%, it demonstrated the potential usefulness of ANN as a forecasting tool and a criterion for candidates applying to universities.

Nie, et. al., [24] have proposed the Cluster Centres Based on XGBOOST (ACCBOX) model as a method to forecast the profession choices of students. A crucial aspect of planning a college student's life is making a well-informed career choice. The experimental results of predicting students' career choices, obtained by analysing 13 M behavioural data points from over 4,000 students, unequivocally demonstrate the superiority of the proposed strategy over the state-of-the-art methods already in use.

In 2022, Wang, et. al., [25] have developed a machine learning method known as extreme gradient boosting (XGBoost) to predict the career choices of college students using real-world information collected in a specific college. To be more precise, the data contain information from 18,000 graduates' college years about their schooling and employment choices. To further understand the data and evaluate the significance of distinct aspects, SHAP (Shapley Additive explanation) was used. With an accuracy value of 89.1%, the findings demonstrate that XGBoost is capable of making reliable predictions about students' profession choices.

1.1. Problem statement

A wide range of approaches are currently being used in educational and career counselling research, all with the goal of using computer models for predictive analysis. These research projects, which run from 2021 to 2022, explore a variety of topics including artificial neural networks, machine learning, and Social Cognitive Career Theory. Even while each strategy has unique benefits, such as increased precision, high accuracy, and effective data management, they also have disadvantages. Across the range of approaches, problems with data collecting, high mistake rates, upfront expenses, and setup times continue are the obstacles. Navigating these complexities and determining the best approaches for predicting student performance, career choices, and counselling are the needs in educational environments is needed. To solve these challenges, this paper introduces a hyperparameter tuning based deep learning model for accurate career prediction.

2. PROPOSED METHODOLOGY

The proposed method aims to accurately predict the career prediction choices by using behavioural data and deep learning techniques. It begins with the collection of student's behaviour data, followed by rigorous pre-processing to ensure data quality. Feature extraction is characterized using central tendency measure, quantitative measure of dispersion, and MAD. The model architecture introduces a new Hybrid LCNN approach, combining optimized CNN and LSTM architecture, with parallel processing of extracted features and fine-tuning during training Hyperparameter tuning is achieved through BPOA, LSTM and optimized CNN parameters and CNN weights are optimized to increase its impact. Model forecasts are combined by weighted mean using synergistic effects. The block diagram of the detailed proposed model is shown in Figure 1.

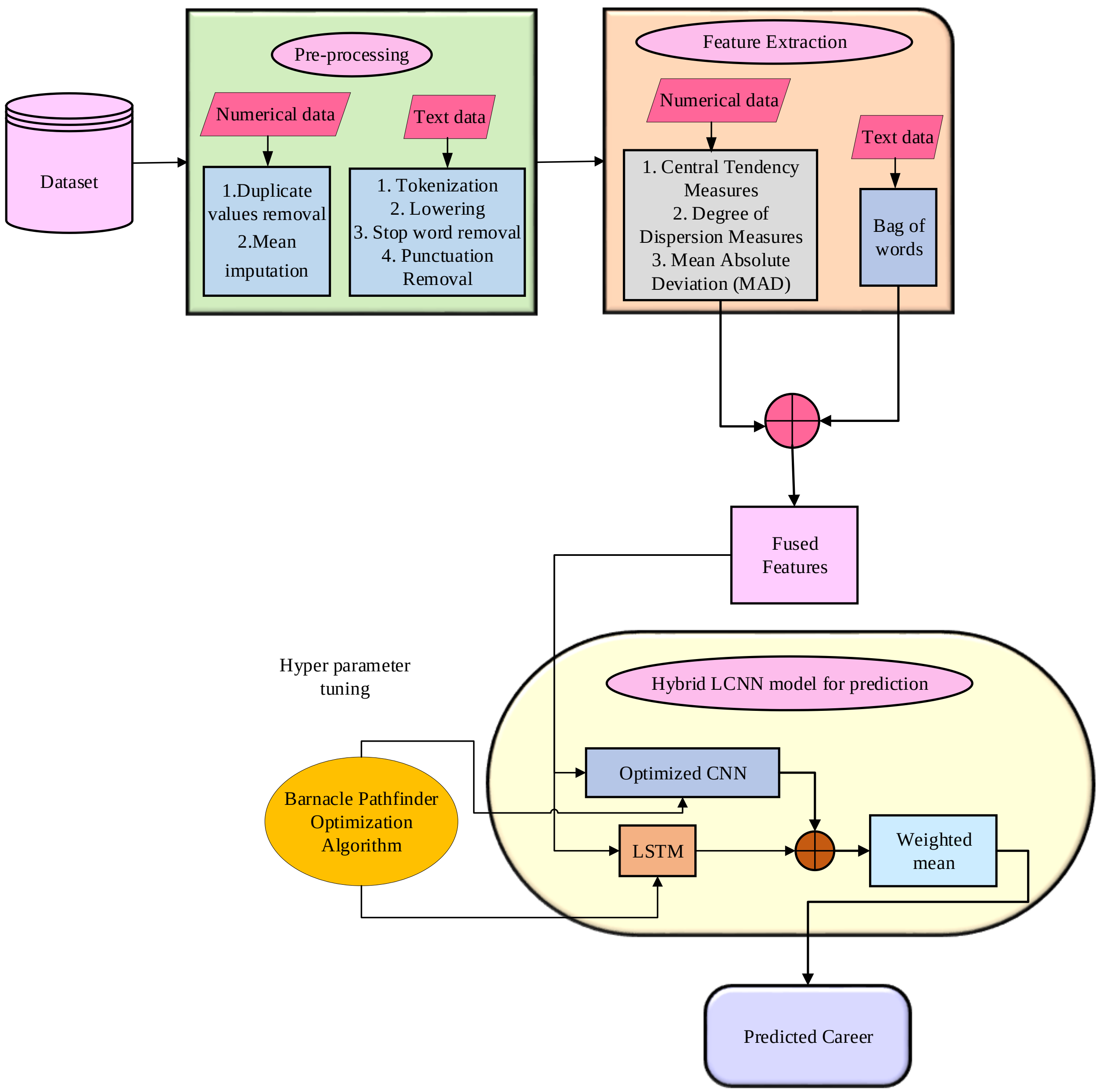

Figure 1: Workflow of the proposed student Career prediction model

3.1. Pre-processing

The dataset has both the numerical and text data. So, in the pre-processing stage the data are separately pre-processed using several techniques. After pre-processing, the pre-processed numerical data and the text data is given to the feature extraction stage.

3.1.1. Pre-processing for Numerical data

In this stage, the duplicate values must be eliminated in order to preserve data integrity while pre-processing numerical data. The mean imputation method is used to handle missing data. The available mean value can be used to replace any missing entries.

i) Duplicate values removal

The first stage of the pre-processing is to remove the duplicate values present in the dataset. It can remove the repeated numerical values from the dataset.

ii) Missing value imputation using Mean imputation

It is known as "missing data imputation" when missing values are substituted with logical values. For replacing lost data, this is the method that is most commonly utilized. Missing values are replaced with the sample mean, median, or mode, depending on how the data are distributed. This approach has drawbacks even if it is straightforward to implement. As each missing value is replaced with the mean, or the imputation value, the distribution takes on a different shape when the number of missing values is large. The standard deviation decreases when comparing the data obtained before and after imputation. If more values are missing, the standard deviation will decrease even further. This method can be slightly improved by splitting the data into smaller groups.

3.1.2. Pre-processing for Text data

Tokenization is the first step in pre-processing text data; it divides text into smaller pieces like words or phrases. Lowercasing is the next step, which guarantees consistency for precise feature extraction. Afterward, stop words are eliminated, sharpening the focus on important content by getting rid of unnecessary terms. Finally, removing punctuation reduces noise and symbols, which enhances the precision of later text normalization procedures.

i) Tokenization

The process of breaking up complex text, such as paragraphs, into smaller portions known as tokens is referred to tokenization. There are two classifications,

- To create a list of sentences out of a paragraph, use the sent tokenize () function.
- Word tokenization: To split a sentence into a list of terms, use the tokenize () function. Put all the libraries into use that are required to tokenize the input data.

a) Sentence tokenization

To do this, a text paragraph must be divided into its individual phrases. Commas and full stops, which are used in English and other languages as sentence and paragraph delimiters, make this work easier. Nevertheless, the process is not straightforward because English abbreviations also use the same sign.

b) Word tokenization

Tokenization of a text paragraph into its individual words is the procedure known as word splitting. The usage of delimiters such as word space can facilitate the process of identifying the beginning of a new word in some languages. However, the process is significantly more difficult because compound words also employ the same spaces.

ii) Lowercasing

Text written in lowercase is essential because otherwise the computer might interpret the same word twice in capital letters or another font style. For example, "Love" and "love" may seem different and have different vectors when the text is vectorized for feature extraction. This is the easiest pre-processing method that yields the most accurate results, but it is also the one that is sometimes skipped.

iii) Remove stop-words

In natural language, words that are unnecessary or have little meaning are eliminated, which is known as stop words. There are several frequently used stop words in every written composition. Common features like and, or, but, pronouns, she, and so on should be removed because they hardly affect the text mining process. Because of how frequently these terms appear, it can be difficult to understand the content of the text files that contain them. By concentrating on the most pertinent terms in a document, stop words can be removed to make the text more logically organized. This process enhances system performance by handling less text data.

iv) Punctuation Removal

Punctuation or special symbols are used in the sentences. The main goal of the text data that was gathered from the electronic websites that were used for the study is usually to reduce any unnecessary noise, tokens, and symbols that must be removed before carrying out any more normalization method activities. Text processing entails cleaning the text to improve the accuracy of classifiers. One of the most important things to do during text normalization is to eliminate unnecessary and odd characters. This could be the point when special letters or punctuation affect the results of machine learning and natural language processing when texts are analysed to extract characteristics or information.

3.2. Feature Extraction

In this stage the feature extraction is separately performed for numerical and text data. Degree of dispersion measurements like range, variance, and standard deviation are extracted. In central tendency measures, the metrics like mean, median, and mode values of the pre-processed data are extracted. The BoW model is applied to textual data.

3.2.1. For Numerical data

To determine the typical value and distribution of a dataset, central tendency measures for numerical data, such as mean, median, and mode, are calculated. Then the dispersion metrics include range, variance, and standard deviation are extracted. Furthermore, MAD based features are extracted which include the variability of data points around the median value.

i) Central Tendency Measures

Among these factors, Mean, Median, Mode are calculated to understand the specific value and distribution of the dataset.

- **Mean**

The mean, also referred to as the average, is an arithmetic mean of a set of data and a statistical indicator of central tendency. It is calculated by taking the average of all the variables in the dataset and dividing the result by the total number of variables.

$$Mean = \frac{Sum\ of\ all\ values}{Total\ number\ of\ values} \quad (1)$$

- **Median**

Another technique to quantify central tendency is to look at the median, which shows the midway value in an ordered set of data. Half of the numbers in the data are above and half are below the median. The data is divided into two equal halves. Compared to the mean, the median is less prone to outliers, or exceptionally high or low figures. To define the median, Eq. (2) is used.

$$Median = \begin{cases} \frac{l+1}{2},\ for\ odd \\ \frac{l}{2}, for\ even \end{cases} \quad (2)$$

Where l is the last number of the series.

- **Mode**

The mode represents the most fundamental approach to quantifying central tendency. The most often occurring score in a distribution is this one. Or, to put it another way, the score that is most commonly obtained by sample members. The possibility that a particular distribution could have multiple modes is an interesting problem. "Normal" bell-shaped distributions have a single mode in the middle of the distribution that coincides with the mean and median.

ii) Degree of Dispersion Measures

The type of features includes Range, Variance, Standard Deviation to quantify data spread from the central value.

- **Range**

The range frequently gives a reasonable indication of variability when a distribution is devoid of extreme values. When paired with central tendency measurements, the range can reveal details about the breadth of the distribution. On the other hand, the range could be misleading if the data set includes outliers.

- **Variance**

Variance is a statistical measure of the variation among individual values in a set of data. To put it another way, variance is the difference between each number in the set and the mean (average), and hence the difference between each individual number in the set.

$$Variance = \frac{\sum(x_i - \mu)^2}{N} \tag{3}$$

- **Standard Deviation (SD)**

Data dispersion from the mean is measured and is referred to as SD (σ). The data appear to be clustered around the mean when the standard deviation is low, but they are more widely distributed when the standard deviation is high.

$$SD\ (\sigma) = \sqrt{\frac{\sum(x_i - \mu)^2}{N}} \tag{4}$$

Where μ is the mean value, x_i is the input variable, and N is the total number of input variables.

iii) Mean Absolute Deviation (MAD)

An extra measure of dispersion called MAD is computed. It is a trustworthy indicator of the variability of data points around the median value, giving insight into the dispersion of the data. Although there are other measures of dispersion or variability, such the standard deviation and variation, the MAD is more robust since it remains constant even when the extreme values fluctuate. The median deviation, which is computed using the median of absolute values, takes the absolute difference from the median of the given data.

$$MAD = median[|s_i - \tilde{s}_i|] \tag{5}$$

$$\tilde{s}_i = median(s_i) \tag{6}$$

Where $\tilde{s}_i$ denotes an individual observation in the provided data, and $median(s_i)$ denotes the data's median.

3.2.2. For Text data (Bag of words)

A method for feature extraction from a text that can be applied to text-based feature extraction is the Bag of Words model. To put it simply, it's a collection of words used in a word count text to define a sentence. It contains two components: a list of well-known terms and a measure to ascertain if such terms are present. One more thing about BoW is that they don't always appear in the same order. Creating a vocabulary from all of the unique terms in the dataset is the first stage. The next stage is to enumerate each of these unique terms and track their recurrence in each and every piece of data. Lastly, provide the feature fusion technique for effective prediction accuracy. The features derived from textual and numerical data are combined and given to the prediction stage for the purpose of predicting student careers.

3.3. **Student Career Prediction using Hybrid LCNN model**

This section presents the Hybrid LCNN technique, which predicts Career prediction more accurately by combining optimal CNN structures with LSTM. The BPOA algorithm is used to finetune the hyperparameters of the CNN and LSTM models during training. Additionally, the BPOA optimization model is used to optimize the CNN component's weights, which improve prediction accuracy. Using a weighted mean, predictions from the optimized CNN and LSTM components are then integrated to produce a final Career behaviour prediction.

3.3.1. Optimized CNN

A CNN is a Deep Learning system that can do rather complex tasks using text, images, audio, videos, and other types of material. It's a multi-layer neural network that can-do tasks like object detection, image classification, and segmentation based on visual inputs. By assigning them adjustable weights and biases, it can differentiate between various objects in a data. CNN uses learnable weights and biases to create neurons. The CNN model is optimized using the BPOA algorithm and the hyperparameters like momentum, weight, learning rate, epoch and batch size are also fine-tuned. All neurons receive input, do a dot product, and then may or may not follow the stimulus in a non-linear fashion. A single differentiable score function is reflected throughout the network, from the extracted features at one end to the class scores at the other. Figure 2 displays the construction of CNN.

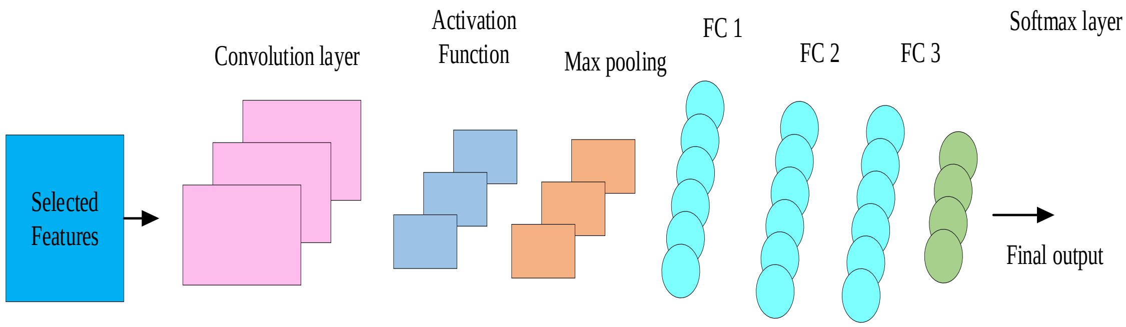

Figure 2: Architecture of the basic CNN model

i) Convolutional Layer

The input of CNNs is typically represented as a tensor, the shape of which is dictated by the amount, width, height, and depth of the data. Following its passage through a convolutional layer, the input is abstracted into a feature map, with the width, height, and number of channels serving as its defining characteristics. Each convolutional layer in a neural network is defined by certain attributes, such as convolutional kernels defined by their height and width—hyper-parameters set during network design. The layer also establishes the quantity of input and output channels, which are crucial hyper-parameters influencing the network's architecture and capacity for effective feature extraction. To guarantee compatibility and appropriate convolution operations inside the network architecture, the number of channels in the input feature map must correspond with the depth of the convolution filter, which represents the input channels. Convolutional layers forward the input result of its convolution to the subsequent layer. Layer l uses $x^{l(r^j)}$ to represent the j^{th} local area and K_i^l to indicate the weights and bias of the i^{th} filter kernel, respectively. Thus, the following Eq. (7) is a description of the convolution process:

$$y^{l(i,j)} = K_i^l * x^{l(r^j)} = \sum_{j=0}^{W} K_i^l(j') * x^{l(j+j')} \tag{7}$$

In frame I of layer $l + 1$, the j'^{th} weights are indicated by the notation $K_i^l(j')$, and the dot product of the kernel and local areas is indicated by the notation $*$ and W represents the kernel's width.

ii) Activation Layer

Every convolutional layer's output is run through an activation function that amplifies positive vectors and suppresses negative vectors from the layer before it. Leaky ReLU is the activation function that we employ in the article. ReLU can reduce the likelihood of vanishing gradient and boost feature sparsity when compared to other activations. The formula for Leaky ReLU is given in Eq. (8) as,

$$LeakyReLU(x) = \begin{cases} x, & x \geq 0 \\ ax, & x < 0 \end{cases} \tag{8}$$

Where the value of a is set to 0.2, it signifies the hyper-parameters.

iii) Max Pooling Layer

Local or global pooling layers can be added to convolutional networks to streamline the underlying computation. The dimensionality of the data is decreased using pooling layers, which aggregate the neuron cluster outputs from one layer into a single neuron in the following layer. Local pooling is the method used to merge small clusters, typically 2 × 2. Global pooling affects all of the convolutional layer neurons. Calculating a maximum or an average can also be done by pooling. Every neuronal cluster in the layer above contributes its maximum value when max pooling takes place. Due to its ability to decrease the spatial size of features and accelerate learning, a pooling layer is often used in place of a convolutional layer. Pooling operations come in different varieties, however max-pooling is the most commonly utilized one. To create location-invariant features, the max-pooling layer applies a local max function to the input features with a predetermined kernel/pool size. Here is an explanation of the max-pooling improvement:

$$P^{l(i,j)} = \max_{(j-1)W+1 \leq t \leq jW} \{a^{l(i,t)}\} \tag{9}$$

iv) Dropout Layer

The process of training many neural networks with different designs simultaneously is approximated by a regularization approach termed dropout. Over the course of training, some layer outputs are "dropped out," or indiscriminately ignored. Because of this, the layer will look and act differently from a layer that has a different amount of nodes and is not as related to the one preceding it. In basic terms, a unique "view" of the present layer is used for each layer change during training. Nodes in a layer are forced to take on a probabilistic degree of responsibility for the inputs because dropout introduces noise into the training process.

v) Fully Connected Layer

A specific number of locations in the layer above each neuron provide input to neural networks. When a layer is fully connected, all of the neurons in the layer ahead of it can receive input from each other. A convolutional layer receives input from a restricted subset of its predecessor layer's neurons. The subarea frequently has a square shape. The area within a neuron that takes the information is known as its receptive field. As a result, the receptive field of a fully connected layer is the entire layer before it. The whole receptive area of a convolutional layer is larger than that of the preceding layer. By imposing weights on the input generated by the feature analysis, a fully connected layer forecasts an accurate label.

vi) Softmax

The softmax function is an extension of a multidimensional logistic function. It is often used as the last activation function of a neural network in multinomial logistic regression to normalize the output of the network to a probability distribution across the expected output class. A multi-class classification neural network's numerical output is obtained from its last linear layer logits. Softmax uses the exponents of each output and the sum of those exponents to normalize each number, converting this numerical output into probabilities. This ensures that the total output vector and the sum of the probabilities add up to one. Typically, cross entropy loss is used as the loss function for such a multi-class classification assignment.

3.3.2. LSTM

LSTM, a unique type of RNN with the ability to recognize past events. LSTM's memory cells enable it to handle a variety of time series issues that were outside the scope of earlier machine learning methods. LSTM is especially useful and well-suited for use in the time series domain since it may reject superfluous material and maintain significant past data in the memory cell state. Its architecture generally consists of three gates: output, forget, and input. The hyperparameters like learning rate, epoch and batch size are finetuned using the BPOA optimization. The LSTM unit controls this extensive application by changing the input values that are stored in the cell state. To decide which data from the memory cell state should be maintained or destroyed, the forget gate layer (f_t) is the first gate to operate. Below is the formula that represents the forget gate layer.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \tag{10}$$

Where f_t is the forget gate at time t, b_f is the bias of the forget gate, x_t is the input variable at time t, σ is the neuron's sigmoid function, and h_{t-1} is the output of the previous cell state at time (t – 1). To decide how much fresh information the neuron needs to retain, the input gate uses two decision-making processes. Which values are updated first is decided by using the sigmoid function. After the tanh layer $\tilde{C}_t$ gives a updated result, the cell state is modified. The definitions of the formulas are

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{11}$$

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{12}$$

Where i_t is the input gate at time t, W_i denotes the input weight, and $\tilde{C}_t$ denotes the cell state candidate at time t; b_i denotes the input gate bias and b_c denotes the cell state bias. The final outcome is determined using the output layer. The formula is expressed as:

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{13}$$

Where W_o represents the weight of neurons in the output, b_o denotes the biases in the output gate, and o_t denotes the output gate at time t. The cell state is modified by multiplying the input gate output value it with the candidate value $\tilde{C}_t$, by the multiplication of the forget gate outcomes f_t and the cell state at the preceding time step c_{t-1}. The following Eq. (14) can be used to update the cell's state:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{C}_t \tag{14}$$

The final output or hidden cell status at the given time is shown by h_t, and it may be updated using the results of the output gate and the cell state at that phase. Below is a definition of the Eq. (15):

$$h_t = o_t \odot \tanh(c_t) \tag{15}$$

Over the course of an hour, data on the workload of four distinct virtual machines is input into the LSTM model utilised in this investigation. A time frame of 48 parts is used to set up the input. The supplied data contains traces of four virtual machine workloads. The basic structure of the LSTM model is shown in Figure 3.

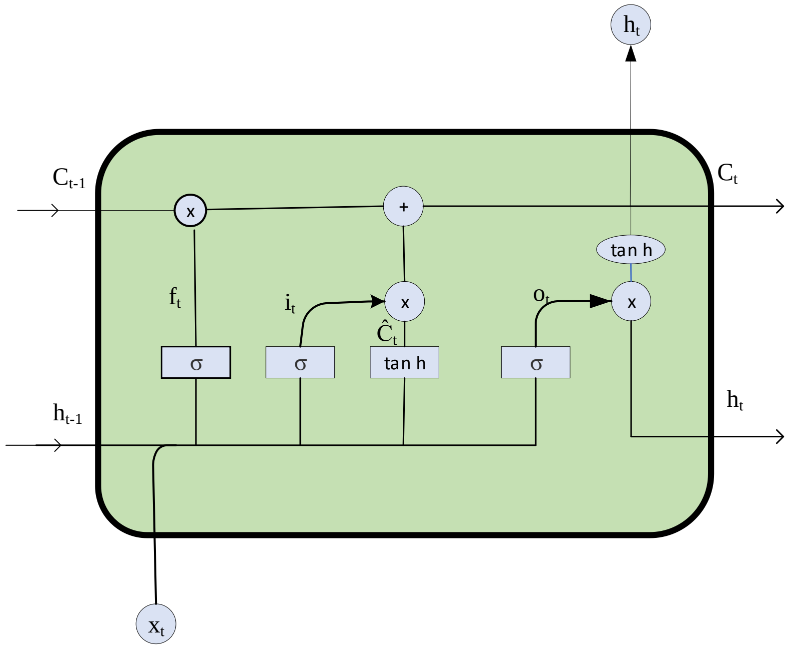

Figure 3: Architecture of the LSTM model

3.3.3. Hyperparameter tuning using BPOA

At the training stage, the CNN and LSTM components are adjusted to maximize their respective performances. BMO and PFA are two different optimization methods that are combined to provide a unique BPOA strategy that is used to accomplish this finetuning. The BPOA model increases the overall predictive performance of the model by combining these two optimization strategies to guarantee that the CNN and LSTM architectures are optimally optimized to extract the most relevant characteristics from the data. The mathematical equations for the BPOA model is given below,

i) Initialization

The population matrix in the proposed BMO can be represented as follows, and it is assumed that barnacles is the candidate solution:

$$X = \begin{bmatrix} x_1^1 & \cdots & x_1^N \\ \vdots & \ddots & \vdots \\ x_n^1 & \cdots & x_n^N \end{bmatrix} \tag{16}$$

Where n is the number of population and N is the number of control parameters. The control elements in Eq. (16) are susceptible to the problem's upper and lower bounds, which are as follows:

$$u_b = [u_b(1), \ldots., u_b(i)] \tag{17}$$

$$l_b = [l_b(1), \ldots., l_b(i)] \tag{18}$$

Where the upper and lower boundaries of the i^{th} variable are denoted by u_b and l_b. After the vector X has been evaluated, It is arranged so that the top of the vector X contains the best solution found thus far.

ii) The mode of selection

The suggested BMO picks two barnacles based on the length of their penises, which is a novel method of mate selection when compared to earlier evolutionary algorithms like GA, DE, etc. The selection process mimics the behaviour of barnacles based on the following assumptions:

- The barnacle's penis length (p_l) will be the only consideration in the random selection procedure.
- Although a female barnacle can likely be fertilized by multiple males in the real world, each barnacle has the ability to both give its own sperm and receive sperm from other barnacles. Additionally, each barnacle can only be fertilized by one barnacle at a time.
- When the same barnacle is ultimately selected via the selection process, it suggests that self-fertilization was meant to take place. Although most barnacles do not self-mate, self-mating is extremely uncommon. For this reason, self-mating will not be taken into consideration in this paper if no new offspring will be produced at this time.
- If the selection at a particular iteration is more than the preset p_l, the sperm casting process is initiated.

It is evident that, in accordance with the previously stated assumptions, this approach enforces the exploitation (points no. 1 and 2) and exploration (points no. 4) processes. Only one of the barnacles #2–#7 can mate with barnacle #1 at a given iteration if we suppose that a barnacle's maximal penis length is seven times greater than its size (p_l = 7). If barnacle #1 selects barnacle #8, the usual mating process does not occur because the choice is greater than the allowed amount. As a result, the offspring generation occurs before the sperm casting procedure, or exploration, as it will be covered later. Naturally, this is only the way the algorithm functions in terms of virtual distance; it has nothing to do with the barnacles' actual distance. The following straightforward choices are utilized, which are presented in mathematical forms:

$$B_d = rand\ per\ m(n) \tag{19}$$

$$B_m = rand\ per\ m(n) \tag{20}$$

Where n is the number of barnacle individuals and B_d and B_m are the parents to be matched. Eq. (19) and Eq. (20) demonstrate that the choice is done at random and meet condition number 1 in the earlier subsection.

iii) Updated Reproduction phase

Compared to other evolutionary algorithms, the reproduction method suggested by BMO is a little different. As there are no precise formulas or equations to determine how barnacles reproduce, the Hardy-Weinberg principle-based BMO mostly focuses on the genotype frequencies or inherited traits of the parents of barnacles in order to produce the future generation. During reproduction, a Fluctuation Rate vector might add controlled randomization. This can assist the algorithm in navigating various areas of the search space and preventing it from becoming trapped in local optima. To illustrate the simplicity of the proposed BMO, the following formulas are proposed to produce novel characteristics of offspring from the parents of barnacles:

$$x_i^{N_{new}} = px_{B_d}^N + qx_{B_m}^N + A \tag{21}$$

$$\text{Where} \quad A = u_2 . e^{\frac{-2k}{k_{max}}} \tag{22}$$

Where A is the Fluctuation rate vector, p is the normal distribution of a variable among [0, 1], $q = (1 - p)$, and Dad and Mum of the barnacles are the variables chosen in Eq. (19) and Eq. (20) respectively. In this case, the vectors u_1 and u_2 have random values between $[-1, 1]$, and k_{max} is the maximum number of iterations that can be made. The percentage of Dad and Mom's traits that are ingrained in each successive generation of kids can be represented by the numbers $p\ and\ q$. Consequently, the child receives the parenting styles of both parents according to the likelihood of a random number between 0 and 1.

The process of exploitation occurs if the barnacles chosen for mating fall within the range of Dad's barnacle's penis length (Eq. (21)). As previously indicated, in the BMO, the sperm cast is used in the exploration phase. When the number of barnacles chosen for mating surpasses the original pl value, sperm cast occurs. The vibration may cause the offspring to head toward unknown areas in the search space, which might result in the finding of superior solutions. The following is the technique for casting sperm:

$$x_i^{n_{new}} = rand\ ()x_{B_m}^n + \varepsilon \tag{23}$$

$$\varepsilon = \left(1 - \frac{K}{K_{max}}\right).u_1.D_{ij} \tag{24}$$

$$D_{ij} = \|x_i - x_j\| \tag{25}$$

Where ε denotes the Vibration vector, D_{ij} denotes the separation between two members. It should be mentioned that Eq. (23) illustrates the straightforward method by which the barnacle's progeny evolve. For the purpose of exploration, the mother's barnacle is used to develop the new offspring. This occurs as a result of Mum's barnacle receiving and using the water sperm that the additional barnacles have discharged elsewhere to create the new offspring.

3.3.4. Weighted Mean prediction

The weighted combined predictions from the CNN and LSTM models are used to generate the final prediction of the pupil Career. The weights assigned to the CNN forecasts are represented by variable weights, while those assigned to the LSTM forecasts are weighted (1 - weighted), to ensure that the sum of the weights is 1. The effect of each model's forecast on the final forecast can be relatively controlled by adjusting the values of the weights. This weighted-average approach allows the combination of the two strengths of CNNs that are excellent in capturing spatial features, and LSTMs that are proficient in capturing sequential patterns, which have finally been established with a view to improving the accuracy of all forecasts The final result is achieved weighted forecasts Divide the sum by 2 and normalize the result.

$$Prediction = \frac{W*CNN_P+(1-W)*LSTM_P}{2} \tag{26}$$

Where P denotes the prediction value, W is the weight function, CNN_P represents the predicted output of CNN and $LSTM_P$ denotes the predicted output of LSTM.

4. RESULT AND DISCUSSION

The proposed framework has been implemented using Python. 70% of the information gathered has been used for training, while the remaining 30% has been used for testing then 20% used for testing, and 80% used for training. A range of metrics, such as Root Mean Squared Error (RMSE), Mean Squared Error (MSE), Normalized Mean Squared Error (NMSE), Mean Absolute Error (MAE), Accuracy, F1-Score, and Recall, were utilized to evaluate the effectiveness of the recommended technique. The developed framework was compared with some other contemporary approaches, including CNN, DNN, SVM, RNN, GRU, LSTM, Bi-LSTM, and XGBoost, to validate the enhancement in execution. (Bi-LSTM), and Extreme Gradient Boosting (XGBoost), to validate the enhancement in execution.

4.1 Performance Metrics

Several matrices, such as MSE, MAE, NMSE, RMSE, Accuracy, F1-Score, and Recall, are utilized for evaluating its performance.

i) MSE: The average of the squared differences between the data set's original and anticipated values is called the mean square error. It calculates the residuals' variance as presented in Eq. (27).

$$MSE = \frac{1}{P}\sum_{k=1}^{P}(a_k - \hat{a})^2 \tag{27}$$

Where P is the number of data points, a_k represents the original value of k^{th} data point and $\hat{a}$ is the predicted value of k^{th} data point.

ii) RMSE: It is one of two often used, closely related assessments of the differences between observed or estimated values and real or predicted values and its mathematical equation is illustrated in Eq. (28).

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(x_i - \hat{x}_i)^2} \tag{28}$$

Where n is the number of observations, x_i denotes the observed value of i^{th} data point and $\hat{x}_i$ represents the estimated value of i^{th} data point.

iii) NMSE: An estimator of the overall differences between expected and measured values is the Normalized Mean Square Error or NMSE. Its definition is provided by Eq. (29).

$$NMSE = \frac{1}{M}\sum_h \frac{(Q_h - N_h)^2}{\bar{Q}\bar{N}} \tag{29}$$

ie) $\bar{Q} = \frac{1}{M}\sum_h Q_h,\ \bar{N} = \frac{1}{M}\sum_h N_h$

Where M represents the total number of observations, Q_h is the expected value of the h^{th} observation, N_h is the measured value of h^{th} observation, and $\bar{Q}$ and $\bar{N}$ denotes the mean of the expected and measured values.

iv) MAE: MAE is the average variance (average difference) between the significant values in the dataset and the anticipated values in the same dataset. The MAE is specified as Eq. (30).

$$MAE = \frac{1}{p}\sum_{v=1}^{p} |z_k - \hat{z}| \tag{30}$$

Where $\hat{z}$ is the predicted value of z and $\bar{z}$ denotes the Mean value of z.

v) Accuracy: Accuracy measures how near measurements of a quantity are to that number's real (true) value. Eq. (31) shows the mathematical equation of the accuracy.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{31}$$

Where TP is True positive, TN is the True negative, FP denotes False positive and FN represents False negative.

vi) F1- Score: The harmonic mean of accuracy and recall rate is known as the F-score definition. The F1-score expression is shown in Eq. (32).

$$\mathrm{F1}_{\mathrm{Score}} = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{32}$$

vii) Recall: Recall is a performance metric that is used to evaluate that how well a classification model identifies all relevant occurrences within a dataset. The ratio of True Positives to the sum of False Negatives and True Positives is how it is defined. Eq. (33) provides the formula for Recall (R):

$$Recall = \frac{TP}{TP+FN} \tag{33}$$

4.2 Overall Performance Analysis

The overall performance analysis, which compares the suggested model with existing ones in terms of MSE, MAE, NMSE, RMSE, Accuracy, F1-Score, and Recall, is given in this subsection and is based on learning rates of 70% and 80%. The performance analysis based on a 70% learning rate is shown in Table 1.

Table 1: Performance analysis based on a learning rate of 70%

	MSE	**MAE**	**NMSE**	**RMSE**	**Accuracy**	**F1-Score**	**Recall**
CNN	0.033889	0.01436	0.00468	0.184088	0.992533	0.992474	0.99242
DNN	0.072372	0.02757	0.012103	0.269021	0.987364	0.98735	0.987421
SVM	0.160827	0.066628	0.024576	0.401033	0.96726	0.967505	0.967617
RNN	0.351522	0.139001	0.0546	0.592893	0.929925	0.929731	0.92992

GRU	0.194141	0.075818	0.03457	0.440615	0.964388	0.964021	0.96405
LSTM	0.033314	0.016083	0.005587	0.182522	0.99081	0.990832	0.990819
Bi-LSTM	0.126939	0.049971	0.018802	0.356284	0.975876	0.975857	0.975921
XGBoost	0.299253	0.12579	0.04864	0.54704	0.933946	0.934037	0.934441
PROPOSED	0.026422	0.010339	0.003619	0.162547	0.994831	0.994896	0.994868

The suggested model shows impressive supremacy over competing models in several performance metrics. Concerning forecast precision, the suggested model outperforms its nearest competitor, CNN, with an MSE of 0.026422, resulting in the lowest MSE. The suggested model, on the other hand, has the lowest MAE of 0.010339, outperforming the DNN, which comes in second place with an MAE of 0.02757. The suggested model is further supported by the NMSE, which is 0.003619, far lower than the values of 0.00468 and 0.005587 for the CNN and LSTM. Besides this, the projected model is further reinforced by the RMSE value, which comes out to be 0.162547. From among the closest competition, LSTM, the recommended framework achieves better than the nearest rival, 0.182522. In the case of classification performance, the recommended framework outperforms the CNN (0.992533) and LSTM (0.99081) with the uppermost accuracy of 0.994831. Besides, it has an improved aptitude to manage precision and recall than other models like CNN, DNN, and LSTM, as shown by its F1-Score (0.994896) and recall (0.994868). therefore, compared to existing procedures, the recommended model not only achieves better than existing methods in terms of classification performance, error rates, and predicted accuracy than current models but also marks a significant development in predictive modelling. The performance analysis based on an 80% learning rate is characterized in Table 2.

Table 2: Performance analysis based on learning rate 80%

	MSE	**MAE**	**NMSE**	**RMSE**	**Accuracy**	**F1-Score**	**Recall**
CNN	0.043066	0.013781	0.006799	0.207524	0.994832	0.99478	0.99472
DNN	0.050818	0.021533	0.00791	0.225429	0.987941	0.987862	0.987876
SVM	0.203273	0.074074	0.032449	0.450858	0.966408	0.966427	0.966637
RNN	0.246339	0.094746	0.042195	0.496326	0.953488	0.953638	0.954286
GRU	0.155039	0.063738	0.023235	0.39375	0.968131	0.968255	0.968531
LSTM	0.019811	0.011197	0.003845	0.14075	0.992248	0.99253	0.992534
Bi-LSTM	0.065461	0.02584	0.012181	0.255853	0.985357	0.985469	0.985722
XGBoost	0.328165	0.119724	0.053911	0.572857	0.945736	0.945131	0.945874
PROPOSED	0.005168	0.003445	0.00078	0.071889	0.997416	0.997376	0.997472

According to Table 2, the proposed framework outperforms all current models across all listed performance metrics. First of all, the recommended standard outperforms all other models, regardless of the metric in question, with the lowermost MSE of 0.005168. For example, the next second, MSE is sophisticated for the LSTM. The same embraces true with the second last, LSTM, which has maximum values of 0.019811 than the first one, MAE. The recommended model is again exposed to be greater in terms of accuracy, given its lowermost value of 0.003445 in MAE. This shows that it can forecast events with minimal inaccuracy than the LSTM, which comes in second with an MAE of 0.011197. The dominance of the recommended model is further reinforced by the NMSE, which demonstrates that it captures variance better than the LSTM (0.003845), with a value of 0.00078. The proposed model also obtains a higher accuracy score of 0.997416, which displays its effectiveness in precisely recognizing cases. After that, LSTM comes in with an accuracy of 0.992248. Moreover, the suggested model outperforms LSTM and other present models in terms of F1-score (0.997376) and recall (0.997472), showing its ability to balance precision and recall. therefore, the recommended framework shows superior performance across all performance metrics of comparison, showing improved performance on cataloguing accuracy, error rate lessening, and heightened extrapolative accuracy. Thus, this enhancement is possibly much superior than the present models can attain.

4.3 Performance Analysis of the Projected Model: Graphical Representation

The subsection offers the outcomes of a performance scrutiny of a graphical evaluation with the other frameworks presently utilized.

i. MSE

The perceptions on various frameworks are defined by comparing the standards of the Mean Squared Error (MSE) of the learning rate of 70% and 80%, as displayed in Figure 4. As the learning rate rises, there is typically a noticeable trend of MSE declining, suggesting improved performance with a higher percentage of training data. For example, CNN's MSE increases somewhat from 0.03388857 at 70% to 0.043066322 at 80%, indicating a tiny decline in performance. On the other hand, models with lower MSE values at higher learning rates, such as DNN, RNN, GRU, LSTM, Bi-LSTM, and XGBoost, show better performance. Interestingly, the suggested model exhibits a noteworthy MSE reduction from 0.026421597 to 0.005167959, indicating markedly improved performance at the increased learning rate. Though there are a few outliers, like SVM, which exhibits a rise in MSE, the general pattern highlights how crucial larger training datasets are for enhancing model performance. These results demonstrate that how well the suggested model can adjust to a greater percentage of training data.

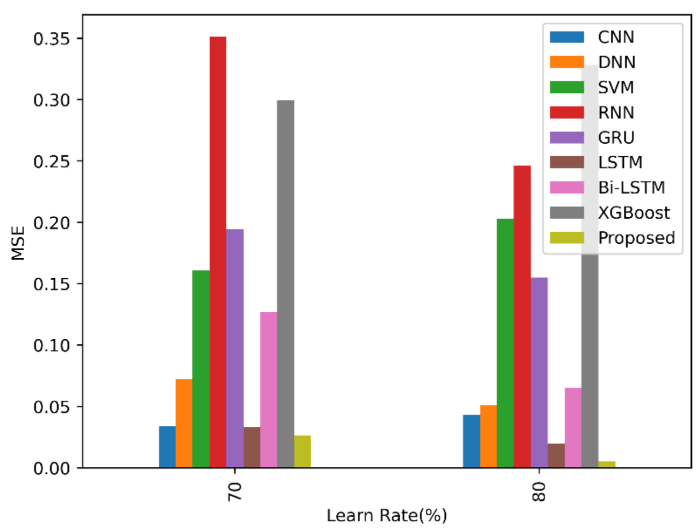

Figure 4: Analysis of MSE of the project model

ii. MAE

The improved efficacy of the suggested model is highlighted by a comparison of its MAE values with other metrics at learning rates of 70% and 80%. The suggested model regularly obtains the lowest MAE values at both learning rates, demonstrating its capacity to provide predictions with the least amount of error in comparison to other models.

The proposed model, for instance, has an MAE of 0.010338886, far lower than the next closest model, LSTM, which has an MAE of 0.016082711, even at a 70% learning rate. Comparably, the suggested model's MAE decreases even further to 0.003445306 at a learning rate of 80%, highlighting its exceptional accuracy in comparison to other models. This trend of persistent superior performance across both learning rates emphasizes the potential of the proposed model as a highly reliable predictive model and shows how effective it is at minimizing prediction mistakes. The predicted model's MAE analysis is displayed in Figure 5.

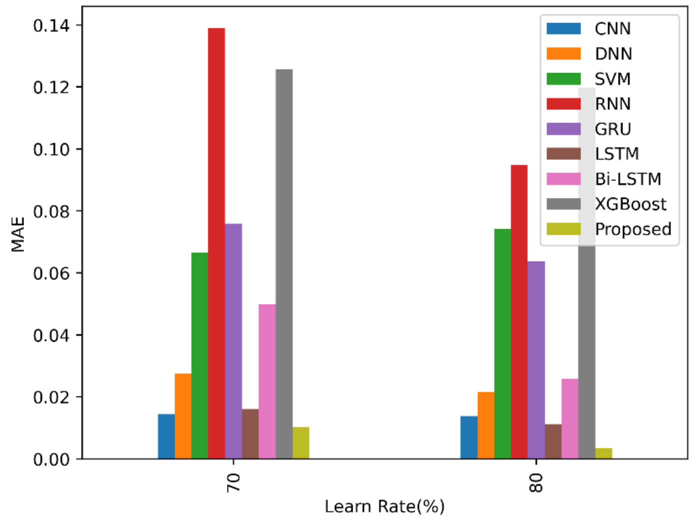

Figure 5: Analysis of MAE of the project model

iii. NMSE

The suggested model's NMSE values are compared to other metrics at learning rates of 70% and 80%, which show that different models perform at different levels. Additionally, at both learning rates, the suggested model has larger NMSE values than the majority of other models, indicating a substantially lesser ability to capture the variation of the data. The suggested model, for example, has an NMSE of 0.0486402 at a learning rate of 70%, which is greater than the NMSE values of other models such as CNN, DNN, SVM, RNN, LSTM, Bi-LSTM, and XGBoost. Similarly, the NMSE of the suggested model, at 0.053910746, is still relatively high at a learning rate of 80%. The performance analysis based on the NMSE matrix is displayed in Figure 6.

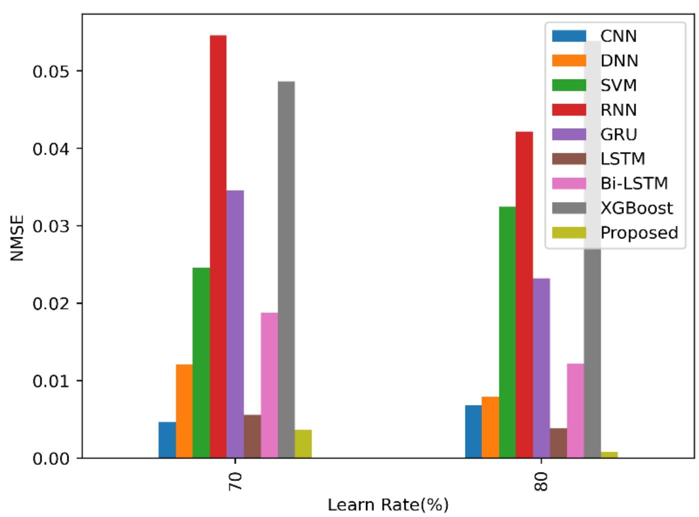

Figure 6: Analysis of NMSE of the projected model

iv. RMSE

The enhanced prediction accuracy of the suggested model is demonstrated by contrasting its RMSE values with other metrics at learning rates of 70% and 80%. The suggested model regularly obtains the lowest RMSE values at both learning rates, demonstrating its capacity to produce predictions that are more accurate than those of other models. For example, the suggested model has an RMSE of 0.162547214 at a learning rate of 70%, which is much less than the RMSE of 0.182521745 of the next closest model, LSTM. Similarly, the proposed model's RMSE reduces from 26.0905468 to 0.07188515 at a learning rate of 80%, underscoring its excellent prediction accuracy. This shows the excellence of the output data under the form of a framework, which means that the recommended model is going to make a variance when it comes to the estimate errors since it is constantly enhancing other models in terms of RMSE. The RMSE metric utilized for performance analysis is revealed in Figure 7.

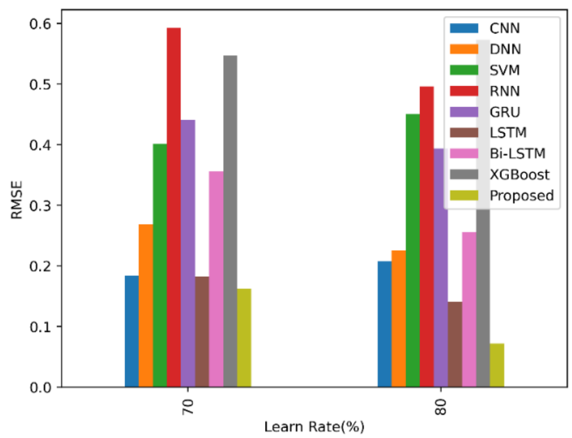

Figure 7: Analysis of RMSE of the projected model

v. Accuracy

The proposed model continually beats other measures in classifying the instances appropriately, with the learning rate values of 70% and 80% in comparison. In terms of forecast accuracy, the recommended model outperforms all the other frameworks in both learning rate values, hence displaying its competence in producing precise predictions. For instance, for the identical model, the suggested framework outperforms all other models, including CNN, DNN, SVM, RNN, GRU, LSTM, Bi-LSTM, and XGBoost, having an Accuracy of 0.994830578 at the learning rate of 70%. Likewise, the accuracy of the proposed framework will keep going up to 0.997416021 at a learning rate of 80%, showing how much better it is than all the other models in terms of accuracy. The constancy and dependability of classification for the recommended framework are stated by its determined achievement in getting the finest among the numerous models for accuracy. This predicted model's accuracy is depicted in Figure 8.

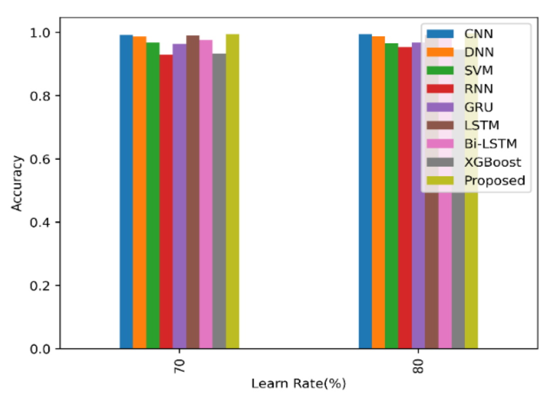

Figure 8: Analysis based on the Accuracy of project model

vi. F1-Score

It has been dependably established to achieve better than existing measures in attaining the balance between precision and recall, as appeared by the comparison of F1-Score values at the learning rate of 70% and 80%. The projected model is the finest among them in terms of F1-Score values at both learning rates and validates its capacity to precisely classify instances by minimizing false positives and false negatives. For example, the recommended framework comes out in primary among all other systems, including CNN (0.992474281), DNN (0.987350457), SVM (0.967505472), RNN (0.929730611), GRU (0.96402132), LSTM (0.99083248), Bi-LSTM (0.97585704), and XGBoost (0.934036996), with an F1-Score of 0.99489567 at a learning rate of 70%. Similarly, at a learning rate of 80%, the F1-Score of the proposed framework raises to 0.997376417, also confirming its superiority in striking the compromise between precision and recall. Therefore, the given standard shows a great deal of competence and performance in categorizing structures based on its tendency of supremacy among other models based on the F1-Score. Figure 9 displays the analysis of the F1 score.

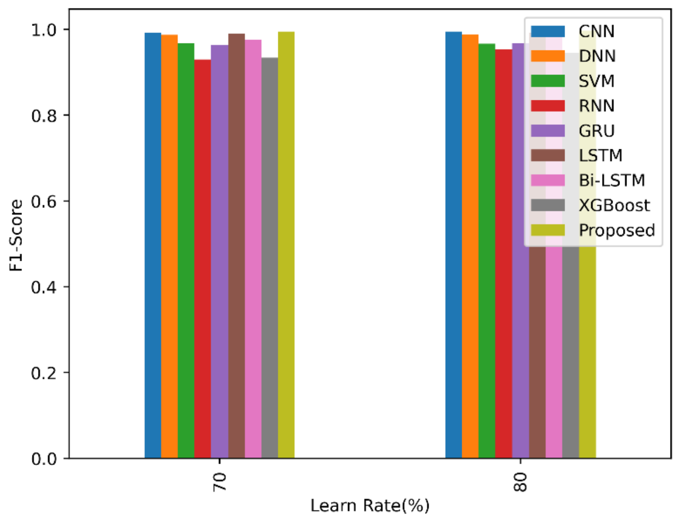

Figure 9: Analysis of F1-Score of the projected model

vii. Recall

Recall values for learning rates of 70% and 80% are compared, and the results show some interesting patterns in the models' performance. Recall often improves as the learning rate rises, suggesting improved model performance with a higher percentage of training data. With a higher learning rate, for example, CNN displays an improvement in Recall from 0.99241988 at 70% to 0.994720184 at 80%, indicating improved performance. Similar to this, models with greater recall values at higher learning rates, such as DNN, GRU, LSTM, Bi-LSTM, and the suggested model, also show enhanced performance. Nevertheless, several models show inconsistent outcomes. The influence of learning rate on recall can vary based on the model architecture and dataset features, as evidenced by the modest decline in Recall that SVM displays from 0.967616584 at 70% to 0.966637265 at 80% and the considerable improvement that RNN shows from 0.929920239 to 0.954285687. The comparative analysis underscores the need to take learning rate modifications into account during model training to maximize recall efficiency. The results highlight the possibility of enhanced model performance with bigger training datasets, as exemplified by the constant rise in Recall values among different models with increasing learning rates. A recall comparison between the proposed model and existing models is presented in Figure 10.

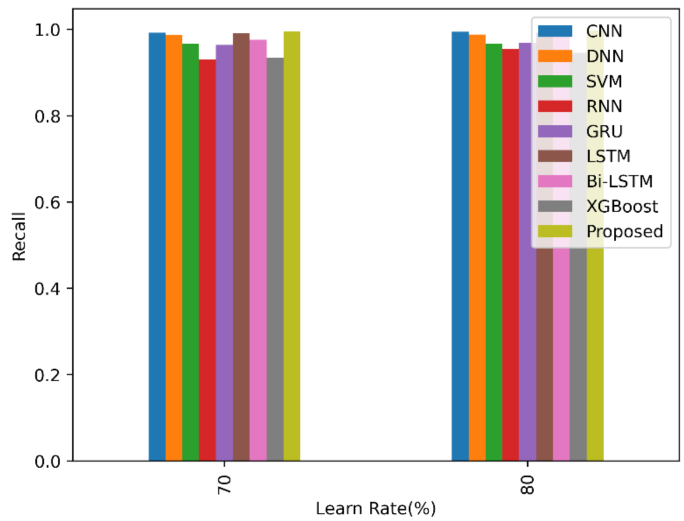

Figure 10: Analysis of Recall of the projected model

4.4 Density plot

A density plot is a type of visual representation that represents the distribution of continuous variables for an understanding of the spread of data and its form. This subsection shows the density chart for estimating the predicted and actual values for learning rates of 70% and 80%. These density charts help in framework assessment and classifying problematic regions that require enhancement in performance through graphical representation, in which the accuracy of the framework and the difference between actual and predicted values are exhibited. The density map between the actual and projected values for the learning rate of 70% is shown in Figure 11. The density map between the actual and predicted values for the learning rate of 80% is shown in Figure 12.

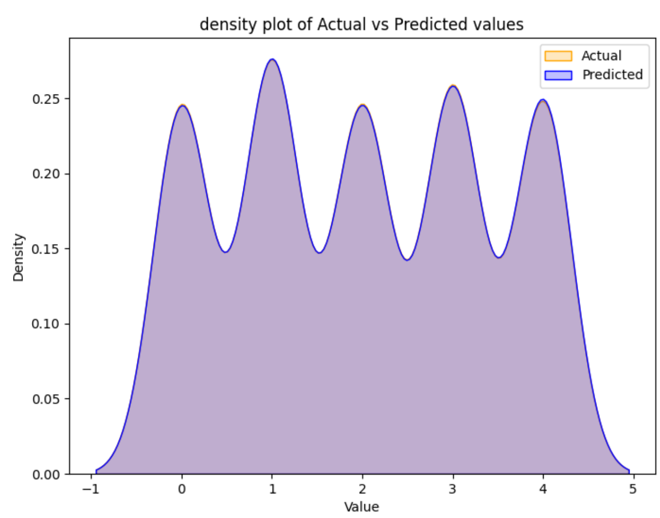

Figure 11: Density plot between actual and predicted values for learning rate 70%

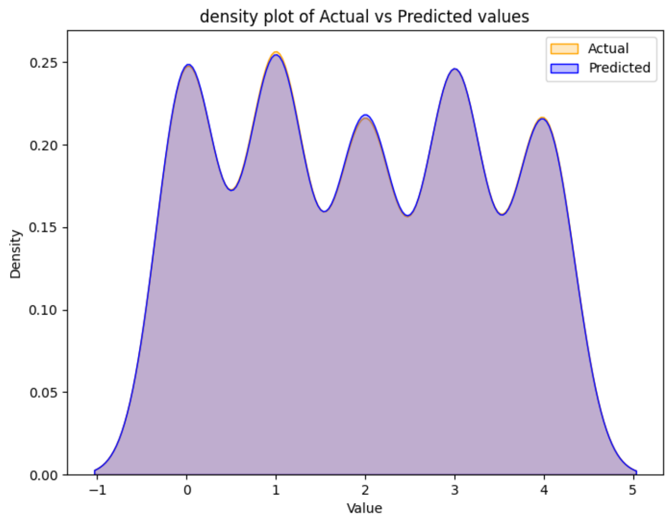

Figure 12: Density plot between actual and predicted values for learning rate 80%

4.5 Data Visualization

A software application familiarity-based data visualization is shown in Figure 10, which divides software application familiarity into three categories: adept, novice, and expert. This visualization provides information about the expected model's performance across a range of disciplines and software application competence levels. Figure 13 makes it easier to comprehend how people with varying degrees of competence use software programs across many areas by showing familiarity levels within each sector.

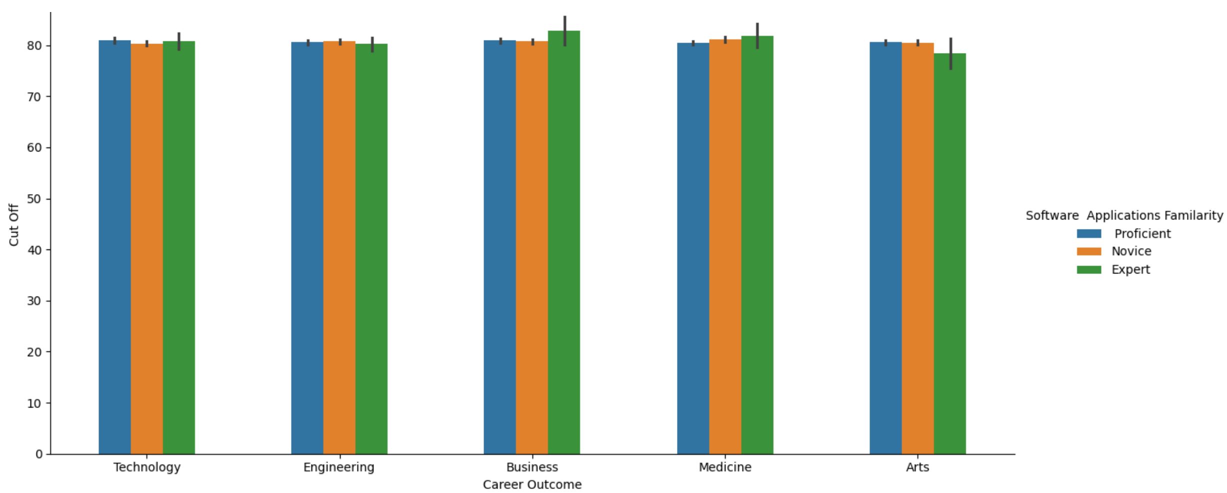

Figure 13: Data visualization of the projected model

4.6 Confusion Matrix

In the discipline of deep learning, particularly in the statistical classification problem, a particular table arrangement known as a confusion matrix also called an error matrix is utilized to visualize algorithm performance. Numerous metrics are considered, including False positive (FP), False negative (FN), True positive (TP), and True negative (TN). Figure 14 shows the confusion matrix for the suggested model below.

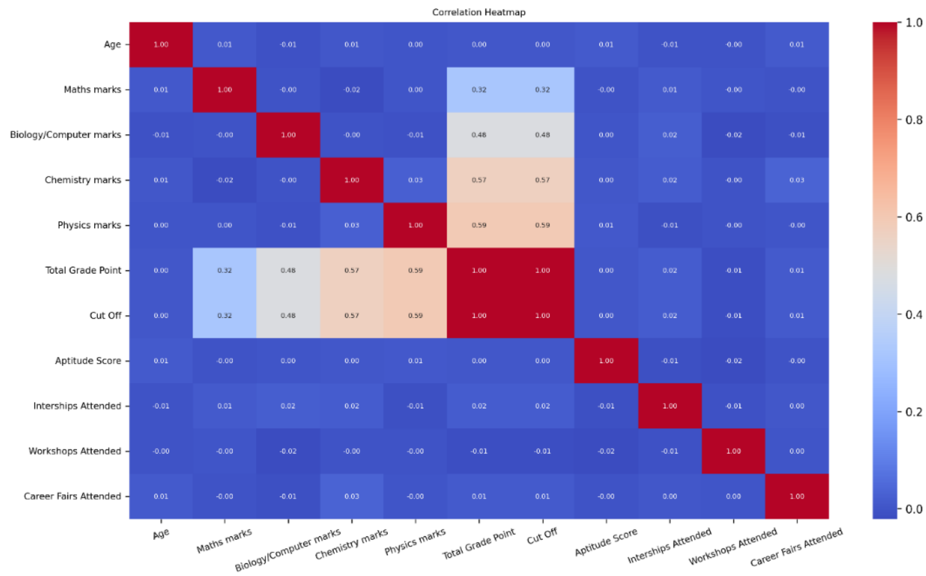

Figure 14: Confusion matrix of the suggested framework

In figure 15 a specific instance where the input is provided and the corresponding output is given.

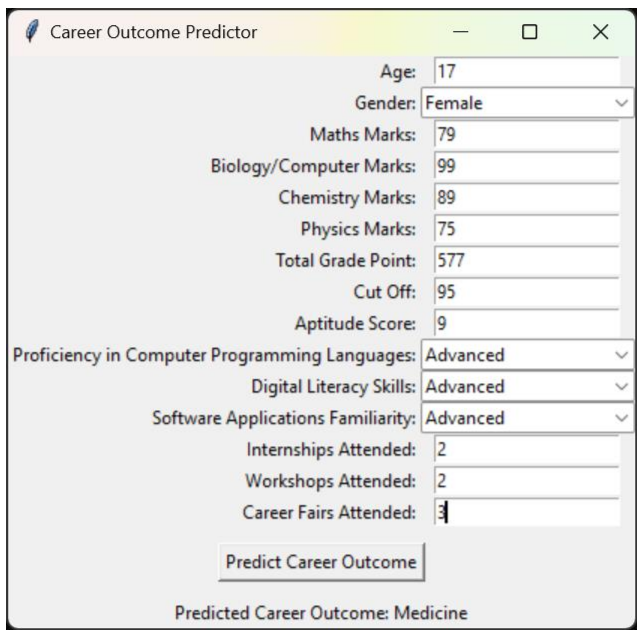

Figure 15: An instance from the Proposed Course Recommendation System

5. CONCLUSION

In conclusion, the proposed LCNN approach represents a significant breakthrough in educational data analysis and student mobility prediction combining the capabilities of LSTM with an optimized CNN, this new fusion model provides a system; it is difficult to accurately predict student behaviours and approaches environment. Through an optimized workflow including data pre-processing, feature extraction, and a new Hybrid LCNN-based Career prediction system, which ensures the accuracy and quality of the predictive model. By using advanced deep learning techniques and optimizing hyperparameters using BPOA, which increases the predictive power of the model, and provides realistic and applicable insights to student's behavioural findings empower educational stakeholders, enabling proactive intervention and personalized support strategies to improve student success and learning outcomes they. By enabling students to make more informed decisions, this advanced predictive capability drives improvements in overall efficiency, service quality, and educational effectiveness. As the educational landscape continues to evolve, LCNN the hybrid approach leads innovation, paving the way for a personalized and efficient educational experience for all stakeholders.

REFERENCE

[1]. Nie, M., Xiong, Z., Zhong, R., Deng, W. and Yang, G., 2020. Career choice prediction based on campus big data—mining the potential behavior of college students. *Applied Sciences*, *10*(8), p.2841.

[2]. Sengupta, S., Banerjee, A. and Chakrabarti, S., 2021. Prediction of future career path using different machine learning models. In *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2020, Volume 3* (pp. 241-252). Springer Singapore.

[3]. Kleine, A.K., Schmitt, A. and Wisse, B., 2021. Students' career exploration: A meta-analysis. *Journal of Vocational Behavior*, *131*, p.103645.

[4]. Çetinkaya, A. and Baykan, Ö.K., 2020. Prediction of middle school students' programming talent using artificial neural networks. *Engineering Science and Technology, an International Journal*, *23*(6), pp.1301-1307.

[5]. Bargmann, C., Thiele, L. and Kauffeld, S., 2022. Motivation matters: Predicting students' career decidedness and intention to drop out after the first year in higher education. *Higher Education*, pp.1-17.

[6]. Zhang, H., Lee, I., Ali, S., DiPaola, D., Cheng, Y. and Breazeal, C., 2023. Integrating ethics and career futures with technical learning to promote AI literacy for middle school students: An exploratory study. *International Journal of Artificial Intelligence in Education*, *33*(2), pp.290-324.

[7]. Chuang, N.K., Lee, P.C. and Kwok, L., 2020. Assisting students with career decision-making difficulties: Can career decision-making self-efficacy and career decision-making profile help. *Journal of Hospitality, Leisure, Sport & Tourism Education*, *26*, p.100235.

[8]. Luo, T., So, W.W.M., Wan, Z.H. and Li, W.C., 2021. STEM stereotypes predict students' STEM career interest via self-efficacy and outcome expectations. *International Journal of STEM Education*, *8*, pp.1-13.

[9]. Jackson, D. and Tomlinson, M., 2020. Investigating the relationship between career planning, proactivity and employability perceptions among higher education students in uncertain labour market conditions. *Higher education*, *80*(3), pp.435-455.

[10]. Lindahl, J., Colliander, C. and Danell, R., 2020. Early career performance and its correlation with gender and publication output during doctoral education. *Scientometrics*, *122*(1), pp.309-330.

[11]. Belchior, R.F. and Lyons, R., 2021. Explaining entrepreneurial intentions, nascent entrepreneurial behavior and new business creation with social cognitive career theory–a 5-year longitudinal analysis. *International Entrepreneurship and Management Journal*, *17*(4), pp.1945-1972.

[12]. Boo, S., Wang, C. and Kim, M., 2021. Career adaptability, future time perspective, and career anxiety among undergraduate students: A cross-national comparison. *Journal of Hospitality, Leisure, Sport & Tourism Education*, *29*, p.100328.

[13]. Rivera, M., Shapoval, V. and Medeiros, M., 2021. The relationship between career adaptability, hope, resilience, and life satisfaction for hospitality students in times of Covid-19. *Journal of hospitality, leisure, sport & tourism education*, *29*, p.100344.

[14]. Wei, Y., Lv, H., Chen, M., Wang, M., Heidari, A.A., Chen, H. and Li, C., 2020. Predicting entrepreneurial intention of students: An extreme learning machine with Gaussian barebone Harris hawks optimizer. *Ieee Access*, *8*, pp.76841-76855.

[15]. Baluku, M.M., Mugabi, E.N., Nansamba, J., Matagi, L., Onderi, P. and Otto, K., 2021. Psychological capital and career outcomes among final year university students: The mediating role of career engagement and perceived employability. *International Journal of Applied Positive Psychology*, *6*, pp.55-80.

[16]. Zeineddine, H., Braendle, U. and Farah, A., 2021. Enhancing prediction of student success: Automated machine learning approach. *Computers & Electrical Engineering*, *89*, p.106903.

[17]. Shen, X., Gu, X., Chen, H. and Wen, Y., 2021. For the future sustainable career development of college students: exploring the impact of core self-evaluation and career calling on career decision-making difficulty. *Sustainability*, *13*(12), p.6817.

[18]. Bai, A. and Hira, S., 2021. An intelligent hybrid deep belief network model for predicting students employability. *Soft Computing*, *25*(14), pp.9241-9254.

[19]. Guleria, P. and Sood, M., 2023. Explainable AI and machine learning: performance evaluation and explainability of classifiers on educational data mining inspired career counseling. *Education and Information Technologies*, *28*(1), pp.1081-1116.

[20]. Quinlan, K.M. and Renninger, K.A., 2022. Rethinking employability: how students build on interest in a subject to plan a career. *Higher Education*, *84*(4), pp.863-883.

[21]. Yağcı, M., 2022. Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments*, *9*(1), p.11.

[22]. Zeng, Q., He, Y., Li, J., Liang, Z., Zhang, M., Yi, D. and Quan, J., 2022. Hope, future work self and life satisfaction among vocational high school students in China: The roles of career adaptability and academic self-efficacy, personality and Individual Differences, 199, p.111822.

[23]. Kehinde, A.J., Adeniyi, A.E., Ogundokun, R.O., Gupta, H. and Misra, S., 2022. Prediction of students' performance with artificial neural network using demographic traits. In *Recent Innovations in Computing: Proceedings of ICRIC 2021, Volume 2* (pp. 613-624). Singapore: Springer Singapore.

[24]. **N**ie, M., Xiong, Z., Zhong, R., Deng, W. and Yang, G., 2020. Career choice prediction based on campus big data—mining the potential behavior of college students. Applied Sciences, 10(8), p.2841.

[25]. Wang, Y., Yang, L., Wu, J., Song, Z. and Shi, L., 2022. Mining campus big data: Prediction of career choice using interpretable machine learning method. *Mathematics*, *10*(8), p.1289.