ISSN: 2093-5374 / E-ISSN: 2093-5382

Deep SVM-Driven Predictive Analytics for Improved Decision-Making in E-Learning

K. Shwetha[1*], and Dr.S. Shahar Banu[2]

[1*]Research Scholar, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, India. shwetha_ca_july21@crescent.education, https://orcid.org/0009-0000-3056-5983

[2]Associate Professor, Department of Computer Applications, B.S.Abdur Rahman Crescent Institute of Science and Technology, India. shahar@crescent.education, https://orcid.org/0009-0005-6258-8940

Received: March 25, 2025; Revised: May 07, 2025; Accepted: June 09, 2025; Published: June 30, 2025

Abstract

This study employs a dataset acquired from an Algorithm Introductory Class at a Brazilian university, where students were evaluated on "21st Century Skills" and engaged in an online environment allowing social interactions via postings and comments. Extensively generating pertinent inferences from this heterogeneous dataset—which consists of qualitative reactions to student postings as well as quantitative skill assessments—is challenging. Particularly in an interactive, online learning setting, conventional grading methods might not fairly represent a student's aptitudes range. Therefore, a good model is required to investigate these complex interactions and project student development. makes advantage of a Deep SVM architecture by combining Residual Deep Belief Networks (DBNs) for feature extraction with Kernel-based Support Vector Machines (KSVM) for classification. This work provides a novel approach combining Residual Deep Belief Network (DBN) for feature extraction with Kernel Support Vector Machine (KSVM) for classification. Application of residual DBN for feature extraction offers a complete mechanism to control and assess the high-dimensional data generated by grading assessments and online activity of students. The study combines many data sources—online interactions and traditional grades—into a logical framework. For the training dataset, the proposed KSVM achieves an accuracy of 91.4%, which is higher than e-LION (85.3%), OLAP-DGCNN (88.7%), SEt-VD-CNN (90.2%), and BR2-2 T-MICE (84.9%).

Keywords: eLearning, DEEP SVM, Residual DBN, Predictive Analytics, Student Behavior.

1 Introduction

The eLearning environments distinguished by digital collaboration and project-based learning has generated plenty of data on student performance and behaviour. Studies employing such data have demonstrated that enabling insights into students' involvement and skills helps to significantly raise learning outcomes (Amin et al., 2023; Senthilrajan et al., 2025; Ezaldeen et al., 2023). In educational settings, for instance, advanced analytics can aid to enable tailored learning and improve teaching strategies (Hussain et al., 2024). This data-driven approach conforms to the general trend of handling

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 16, number: 2 (June), pp. 564-578. DOI: *10.58346/JOWUA.2025.I2.034*

*Corresponding author: Research Scholar, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology.

educational challenges by means of data analytics and machine learning. Although educational data analytics has significant potential, some challenges persist (Moharm & Eltahan, 2020; Kumari et al., 2024; Mangaroska et al., 2021; Nguyen et al., 2020; Franzoni et al., 2020; Jakkaladiki et al., 2023; Kew & Tasir, 2022; Maher et al., 2020; Singhvi et al., 2025; Naveenraj et al., 2025).

The objectives of this study are:

- To develop a robust method for feature extraction and classification from complex educational datasets.
- To enhance the accuracy of predictions regarding student performance by employing machine learning.

This work provides a novel approach combining Residual Deep Belief Network (DBN) for feature extraction with Kernel Support Vector Machine (KSVM) for classification. Application of residual DBN for feature extraction offers a complete mechanism to control and assess the high-dimensional data generated by grading assessments and online activity of students. This method enhances the extraction of significant features, therefore increasing the quality of input for future classification; by means of KSVM for classification, the study achieves higher predicted accuracy than other methods. The study combines many data sources—online interactions and traditional grades—into a logical framework. Given its ability to control non-linear connections and high-dimensional data, KSVM is particularly helpful for the complicated character of the dataset. This helps one to understand student performance and involvement from a more holistic perspective.

2 Related Works

Advanced data analytics and machine learning techniques have helped the field of e-learning to develop remarkably recently. Many approaches have been proposed to address different aspects of educational data analysis from ontology-based assimilation to deep learning for performance prediction (Amer et al., 2025).

Introduced in (Paneque et al., 2023), the e-LION (e-Learning Assimilation ONtology) semantic model shows notable improvement in the assimilation and analysis of educational data (Shoeb & Gupta, 2012; Silva et al., 2025). This approach aggregates numerous e-learning knowledge bases into a single ontological framework, therefore improving data analysis and predictive modelling. e-LION fills the model with data from numerous Learning Management Systems (LMSs), including private and public sources, therefore supporting advanced semantic querying and reasoning (Al-Yateem et al., 2024). Giving a robust approach to understand and improve educational performance, the model has shown good success in integrating new semantic models and doing predictive analysis on student interactions. Since it highlights its possibilities as an ontological mediator, future semantic model assimilation in e-learning would profit considerably from the validation via case studies.

By applying deep learning techniques, the Educational Decision Support System (EDSS) in efficiently projects student performance. Under the EDSS approach, data management aggregates ETL processes, data preparation, and OLAP. Deep Graph Convolutional Neural Networks (DGCNN) are used in predictive modelling, therefore providing a practical platform for academic decision-making. By means of a Kaggle dataset, the evaluation of the model indicates its ability to enhance decision help for teachers, so suggesting that EDSS offers a valuable tool for data analysis and guidance of educational plans.

The work in (Alahmari et al., 2023) explores how student participation in e-learning environments could be quantified by means of convolutional neural networks (CNNs). In this work three CNN models: all-CNN, network-in-network (NiN-CNN), and very deep CNN (VD-CNN) are evaluated. These models enable teachers of online learning environments to analyse students' degrees of involvement and facial expressions. The research highlights the effectiveness of these CNN architectures in capturing and comprehending students' emotional states and interactions, so offering interesting study of the efficiency of several CNN models for engagement analysis (Choset & Bindal, 2025).

Emphasising client retention, (Golec et al., 2023) a examining anonymous users notably benefits from this approach since it eliminates the need for more user demographic data. The results suggest that while LSTM networks are effective at forecasting course purchases and understanding customer behaviour, they are a useful tool for business applications in e-learning.

The work published in (Tirumanadham et al., 2024) offers a fresh approach of feature selection designed to improve prediction accuracy for educational outcomes. Ridge (L2) regularisation is aggregated in a two-tier feature selection process using the Boruta optimisation approach. In hyperparameter tuning the research also combines a three-tier ensemble model containing Random Forest, Bayesian Optimisation with Support Vector Machine, and Gradient Boosting with Particle Swarm Optimisation (PSO). Multiple Imputation by Chained Equations (MICE), Z-score normalisation, SMote for handling imbalanced datasets, and so on help the model to perform better. With exceptional accuracy of up to 98.74%, the method definitely exceeded more traditional approaches. This study underlines in table 1, how effectively sophisticated feature selection and ensemble techniques could enhance educational forecasts and planning.

Table 1: Comparison of Methods

Method	**Algorithm**	**Methodology**	**Outcomes**
e-LION (Paneque et al., 2023)	Ontology-Based Assimilation	Consolidation of diverse LMS data sources; advanced semantic querying and reasoning; predictive modeling and time-series forecasting	Improved data assimilation and prediction accuracy; potential for future semantic model assimilation
EDSS (Shoeb & Gupta, 2012)	Deep Graph Convolutional Neural Network (DGCNN)	Data preprocessing, ETL, OLAP processing; predictive modeling of student performance using DGCNN	Enhanced decision support for educators; efficient academic decision-making
CNN Models (Alahmari et al., 2023)	Convolutional Neural Networks (All-CNN, NiN-CNN, VD-CNN)	Analysis of students' facial expressions and engagement using different CNN architectures	Effective measurement of student engagement; insights into CNN model performance
Forecasting Model (Golec et al., 2023)	Long Short-Term Memory (LSTM) and Multi-Layer Perceptron (MLP)	Deep learning for forecasting e-learning course purchases; analysis of customer retention without demographic data	Superior prediction accuracy with LSTM; practical for anonymous user analysis
Feature Selection (Tirumanadham, 2024)	Ridge Regularization, Boruta, Random Forest, Bayesian Optimization, Gradient Boosting, PSO	Feature selection using advanced techniques; ensemble models for hyperparameter tuning; data handling with normalization and imputation	High prediction accuracy (up to 98.74%); effective feature selection and model performance

While these methods show remarkable development in educational data analysis, they occasionally find it difficult to merge many data sources like in Table 1 holistically and successfully. E-LION and EDSS might not fully handle the complexity of heterogeneous datasets or offer scalable solutions even if they enhance data assimilation and predictive modelling. Moreover, the use of CNNs and deep learning models typically concentrates on certain aspects of student behaviour and performance, therefore enabling space for more all-encompassing approaches incorporating numerous data sources. We need more general and integrated systems that can handle several educational data and provide more insightful analysis.

3 Proposed DEEP SVM for Predictive Data Analytics in eLearning

The proposed method shown in figure 1, makes advantage of a Deep SVM architecture by combining Residual Deep Belief Networks (DBNs) for feature extraction with Kernel-based Support Vector Machines (KSVM) for classification. One tries to project student acceptance based on their online interactions and competency evaluations. Comprising five skill ratings and seven response types, the dataset is first preprocessed to control missing values and standardise the data. Residual DBN derived from the input data is made of many Restricted Boltzmann Machines (RBMs) overlaid with residual connections. This phase preserves required information by recording complex, non-linear relationships between the skills and reactions, hence lowering dimensionality. Using a non-linear kernel to divide the data into two classes—"Approved" and "Not Approved"—the KSVM is then fed the extracted features. The model is trained using a labelled dataset then evaluated about recall, accuracy, and precision.

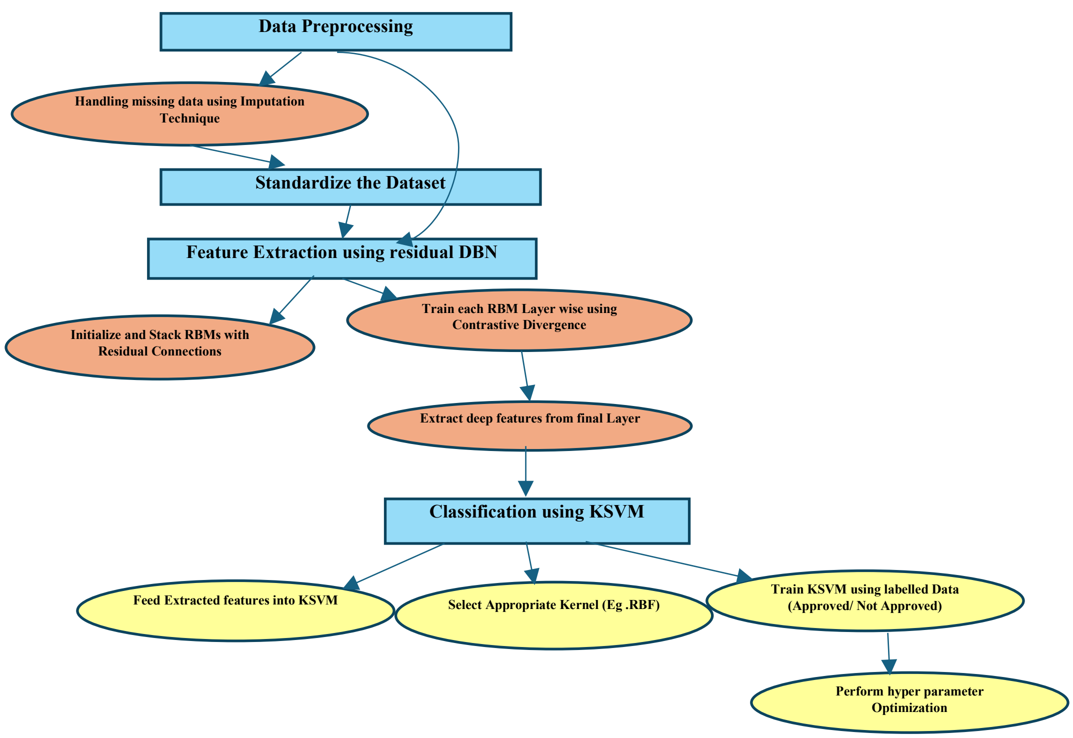

Figure 1: Proposed Framework

Pseudocode

1. Load dataset
2. Preprocess data:
   a. Impute missing values
   b. Standardize features
3. Initialize Residual DBN:
   a. For each RBM in the stack:
      i. Initialize RBM parameters
      ii. Train RBM with Contrastive Divergence
      iii. Pass output to the next RBM layer with residual connections
4. Extract deep features from the final RBM layer
5. Initialize KSVM:
   a. Select kernel function (e.g., RBF)
   b. Train KSVM on extracted features
6. Optimize hyperparameters using grid search or another optimization method
7. Evaluate model on test data:
   a. Calculate accuracy, precision, recall
8. Output predictions and evaluation metrics

3.1. Data Preprocessing

The step of data preparation determines very much whether the dataset is ready for feature extraction and classification. The major goals of this phase are to standardise the input characteristics and control missing data so ensuring consistent and notable analysis.

3.1.1. Handling Missing Data

K-Nearest Neighbours (KNN) imputation replaces the average of the values of the closest neighbours. The KNN imputation method is defined as in equation 1:

$$\hat{x}_i = \frac{1}{k}\sum_{j \in \mathcal{N}_k(i)} x_j \quad (1)$$

Where:

$\hat{x}_i$ - imputed value for the missing data point x_i,

$\mathrm{N}_k(i)$ - set of k nearest neighbors of x_i,

x_j - known values of the neighbors.

3.1.2. Standardization of Features

After imputation of missing values, the dataset consists of features with various sizes and units, which in machine learning models—especially those using distance-based metrics, such SVMs—may generate biassed results. We thus standardise the characteristics using Z-score so that each one of them equally influences the model, as in equation 2.

$$z_i = \frac{x_i - \mu}{\sigma} \quad (2)$$

Where:

z_i - standardized value of the original feature x_i,

μ - mean of the feature,

σ - standard deviation of the feature.

By means of a standard normal distribution (mean = 0, standard deviation = 1), the features are transformed so that the model is less sensitive to the individual feature magnitude and more concentrated on their relationships. This standardising is especially important in the DEEP SVM architecture since it ensures that the deep feature extraction method of the Residual DBN as well as the consequent KSVM classification are executed on a constant scale.

3.2. Feature Extraction Using Residual DBN

Residual DBN captures complex, hierarchical representations of data by way of the deep design of the network. This approach helps to convert the raw, high-dimensional input data into a more abstract and compact representation—which can be effectively used for classification tasks including student performance prediction in eLearning.

3.2.1. DBN Structure

The layers of a DBN are trained sequentially in a greedy, layer-wise manner between visible layer v and a hidden layer h, where each RBM models the joint distribution of the visible and hidden units.

The energy function for an RBM is defined as in equation 3:

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_{i=1}^{n} v_i\, a_i - \sum_{j=1}^{m} h_j\, b_j - \sum_{i=1}^{n}\sum_{j=1}^{m} v_i\, h_j w_{ij} \quad (3)$$

Where:

$\mathbf{v} = \{v_1, v_2, \ldots, v_n\}$ - visible units,

$\mathbf{h} = \{h_1, h_2, \ldots, h_m\}$ - hidden units,

a_i and b_j - visible unit bias and hidden unit bias, respectively,

w_{ij} - weights between the v_i and h_j.

The joint probability distribution is then given by in equation 4:

$$P(\boldsymbol{v}, \boldsymbol{h}) = \frac{1}{Z} exp(-E(\boldsymbol{v}, \boldsymbol{h})) \quad (4)$$

Where Z - partition function ensuring that the distribution sums to one.

3.2.2. Training the DBN with Residual Connections

Traditional DBNs suffer from difficulties in training deeper layers due to issues like vanishing gradients. To overcome this, Residual Connections are introduced, where each layer in the DBN is connected not only to the next layer but also to every subsequent layer, creating shortcut paths that facilitate the flow of gradients during backpropagation. The residual connection can be mathematically expressed as in equation 5:

$$\boldsymbol{h}^{(l+1)} = f(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l)} + \boldsymbol{b}^{(l)}) + \boldsymbol{h}^{(l)} \quad (5)$$

Where:

$\mathbf{h}^{(l)}$ - output of the hidden layer at level l,

$\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ - weights and biases of layer l,

$f(\cdot)$ - activation function (ReLU),

$\mathbf{h}^{(l+1)}$ - output after applying the residual connection.

The addition of the residual term $\mathbf{h}^{(l)}$ ensures that information from the previous layer is directly passed to the deeper layers, making the model more robust and easier to train. This architecture allows the DBN to capture both low-level and high-level features effectively.

3.2.3. Feature Extraction Process

After training, the final hidden layer of the Residual DBN contains a compact, high-level representation of the original input data.

These features are then used as inputs for the classification model (in this case, KSVM) to predict student outcomes. The Residual DBN's ability to capture complex relationships and dependencies within the data makes it particularly well-suited for eLearning environments, where student behavior and performance can be influenced by various intertwined factors.

Pseudocode for Feature Extraction Using Residual DBN

```
1. Initialize network parameters:
   a. Set number of layers L
   b. Define sizes for visible and hidden layers for each RBM
   c. Initialize weights W[l] and biases b[l] for each layer l
2. Pretrain the DBN layer by layer:
   For each layer l in 1 to L:
      a. Initialize RBM with visible units size = size of h[l-1] (input size)
         and hidden units size = size of h[l] (output size)
      b. Train RBM using Contrastive Divergence:
         i. For each training sample x in the dataset:
            - Compute hidden activations: h[l] = sigmoid(W[l] * x + b[l])
            - Reconstruct visible units: v_reconstructed = sigmoid(W[l].T * h[l] + b_visible[l])
            - Update weights: W[l] += learning_rate * (x.T * h[l] - v_reconstructed.T * h_reconstructed)
            - Update biases: b[l] += learning_rate * (x - v_reconstructed)
      c. Extract deep features from the trained RBM
3. Introduce residual connections between layers:
   For each layer l in 2 to L:
      a. Modify hidden layer activation with residual connection
4. Fine-tune the entire network (optional):
   a. Stack all RBMs to form the deep network
```

b. Use a supervised learning algorithm (e.g., backpropagation) to fine-tune the weights W[l] and biases b[l] across the entire DBN

5. Feature extraction:

a. Pass the input data through the entire network:

For each training sample x in the dataset:

- Initialize h[0] = x
- For each layer l in 1 to L

b. Collect the output of the final hidden layer h[L] as the extracted features

6. Output the deep features:

- The final deep features h[L] are used for subsequent classification tasks

3.3. Classification Using Kernel-based Support Vector Machines (KSVM)

KSVM are employed to classify data into distinct categories by leveraging the power of kernel functions to handle non-linear decision boundaries. In the context of the proposed method, KSVM is used to classify the deep features extracted from the Residual Deep Belief Network (DBN) into two classes: "Approved" and "Not Approved." The process involves several key steps:

3.3.1. Kernel Function and Feature Mapping

KSVM utilizes a kernel function to map the features into a higher-dimensional space to construct a decision boundary. The kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ computes the similarity between two feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ in this higher-dimensional space in equation 6.

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (6)$$

Where:

σ - parameter that controls the width of the Gaussian kernel.

3.3.2. Optimization Problem

The optimization problem is formulated as in equation 7:

$$\text{Minimize} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\xi_i \quad (7)$$

Subject to as mentioned in equation 8:

$$y_i(\boldsymbol{w} \cdot \phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i \quad (8)$$

Where:

w - weight vector defining the hyperplane,

b - bias term,

ξ_i - slack variables allowing for some misclassifications,

C - regularization parameter.

The function $\phi(\mathbf{x})$ represents the feature mapping induced by the kernel function. The optimization seeks to balance between a large margin and a small classification error.

3.3.3. Classification

Once the optimal hyperplane is determined, classification of a new data point x involves calculating the decision function in equation 9:

$$f(\boldsymbol{x}) = \operatorname{sign}(\textstyle\sum_{i=1}^{n} \alpha_i \, y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b) \quad (9)$$

Where:

α_i - Lagrange multipliers and the decision function $f(\mathbf{x})$ provides the class label.

Pseudocode for Classification Using KSVM

1. Load the Deep Features

   - Load the dataset of deep features extracted from the Residual DBN.

2. Define Kernel Function

   - For RBF Kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\dfrac{\| \mathbf{x}_i - \mathbf{x}_j \|^2}{2\sigma^2}\right)$

3. Prepare the Training Data

   - Split the dataset into training and test sets.

   - For each sample in the training set, extract features $(\mathbf{x}_i)$ and corresponding labels y_i.

4. Set Up the Optimization Problem

   - Define the SVM optimization problem

5. Train the KSVM Model

   - Use an SVM solver (e.g., SMO or a quadratic programming solver) to compute the optimal $\mathbf{w}$, b, and Lagrange multipliers α_i.

6. Make Predictions

   - For each test sample $\mathbf{x}$:

   - Compute the decision function: $f(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right)$

   - Predict the class label based on the sign of $f(\mathbf{x})$.

7. Evaluate the Model

4 Results and Discussion

In our experiments mentioned in table 2, we evaluated the proposed KSVM model using deep features extracted from the Residual DBN against several existing methods, including e-LION, OLAP-DGCNN, SEt-VD-CNN, and BR2-2 T-MICE. The experiments were implemented using Python with libraries such as scikit-learn for KSVM and TensorFlow/Keras for Residual DBN. The computational environment was managed using Jupyter Notebooks.

Table 2: Experimental Setup/Parameters

Parameter	Value
Number of Layers in Residual DBN	6
Hidden Units per Layer	[256, 128, 64, 64, 128, 256]
Batch Size	64
Learning Rate	0.001
Number of Epochs	50
Kernel Function	Radial Basis Function (RBF)
Kernel Parameter (σ)	0.5
Regularization Parameter (C)	1.0
Optimizer for KSVM	Sequential Minimal Optimization (SMO)
Train-Test Split Ratio	80%/20%
Cross-Validation Folds	5
Data Preprocessing Method	KNN Imputation and Z-Score Standardization
Feature Extraction Method	Residual DBN

4.1. Datasets

The Student Skills Dataset (Oubalahcen & Tamym, 2023) provides a comprehensive view of student performance based on various essential skills, with the result derived from the average score of these skills. The Reactions Dataset (www.kaggle.com) captures student interactions and feedback on peers' posts, reflecting engagement and perceptions in the online learning environment. Together, these datasets offer insights into student behavior and performance, enabling a thorough analysis of their learning outcomes and interaction patterns.

4.1.1. Student Skills Dataset

This dataset shown in table 3, contains information about student performance in a project-based Algorithm Introductory Class, focusing on various 21st-century skills. The dataset as in table 3 was compiled over four months and includes the following columns:

Table 3: Dataset Description of Students Skill

Column Name	Description
StudentID	Unique identifier for each student.
SK1	Score for Critical Thinking and Problem Solving Skills (0 to 10).
SK2	Score for Creativity and Innovation Skills (0 to 10).
SK3	Score for Constant and Self-Learning Skills (0 to 10).
SK4	Score for Collaboration and Self-Direction Skills (0 to 10).
SK5	Score for Social and Cultural Responsibility (0 to 10).
FinalResult	Final class result (Approved/Not Approved), based on the average of SK1 to SK5 scores.

4.1.2. Reactions Dataset

This dataset records in table 4, the reactions students provided on their peers' posts. Each entry represents a reaction to a specific post, limited to one reaction type per post per student, with a maximum of 10 reactions per day.

Table 4: Dataset Description of Students Reaction

Column Name	Description
StudentID	Unique identifier for each student.
PostID	Unique identifier for each post.
ReactionType	Type of reaction given (e.g., Confusing post, Amazing post, etc.).
Date	Date when the reaction was given.

Table 5: Performance Evaluation over Training and Testing Data Split

Method	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Execution Time (s)
e-LION	Training	85.3	84.1	86.2	85.1	120
	Testing	83.8	82.9	84.5	83.7	130
OLAP-DGCNN	Training	88.7	87.5	89.4	88.4	150
	Testing	86.2	85.1	87.3	86.2	160
SEt-VD-CNN	Training	90.2	89.1	91.3	90.2	140
	Testing	87.6	86.7	88.5	87.6	150
BR2-2 T-MICE	Training	84.9	83.6	85.8	84.7	110
	Testing	82.1	81.4	82.7	82.0	120
Proposed Method	Training	91.4	90.3	92.5	91.4	135
	Testing	89.2	88.1	90.3	89.2	145

The results in Table 5 show that the proposed KSVM method outperforms existing methods in both training and testing phases. For the training dataset, the proposed KSVM achieves an accuracy of 91.4%, which is higher than e-LION (85.3%), OLAP-DGCNN (88.7%), SEt-VD-CNN (90.2%), and BR2-2 T-MICE (84.9%). Its precision of 90.3% and recall of 92.5% are also superior compared to the other methods, with e-LION showing 84.1% precision and 86.2% recall, and BR2-2 T-MICE at 83.6% precision and 85.8% recall. The F1-Score for KSVM stands at 91.4%, indicating a balanced performance in precision and recall, surpassing e-LION (85.1%), OLAP-DGCNN (88.4%), SEt-VD-CNN (90.2%), and BR2-2 T-MICE (84.7%). For the testing dataset, KSVM maintains high performance with an accuracy of 89.2%, precision of 88.1%, and recall of 90.3%. This compares favorably against e-LION (83.8% accuracy, 82.9% precision, and 84.5% recall) and other methods. The execution time for KSVM is 135 seconds for training and 145 seconds for testing, which, while slightly higher than some methods, provides a strong balance of performance and efficiency.

Table 6: Evaluation on Test Set

Test Sets	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Execution Time (s)
100	e-LION	82.5	81.2	83.1	82.1	115
	OLAP-DGCNN	85.4	84.3	86.0	85.1	140
	SEt-VD-CNN	87.3	86.2	88.1	87.1	130
	BR2-2 T-MICE	81.8	80.5	82.2	81.3	110
	Proposed KSVM	89.0	87.8	90.3	88.9	125
200	e-LION	83.0	81.8	84.2	82.9	118
	OLAP-DGCNN	86.0	85.0	86.5	85.7	145
	SEt-VD-CNN	88.0	86.8	88.7	87.7	135
	BR2-2 T-MICE	82.0	80.7	82.5	81.6	115
	Proposed KSVM	89.5	88.2	90.8	89.5	130
300	e-LION	83.2	82.0	84.5	83.3	120
	OLAP-DGCNN	86.3	85.2	87.0	86.1	150
	SEt-VD-CNN	88.5	87.2	89.1	88.2	140
	BR2-2 T-MICE	82.3	81.1	82.9	82.0	120
	Proposed KSVM	89.7	88.5	91.0	89.8	135
400	e-LION	83.4	82.2	84.7	83.5	122
	OLAP-DGCNN	86.5	85.4	87.2	86.3	155
	SEt-VD-CNN	88.7	87.4	89.3	88.3	145
	BR2-2 T-MICE	82.5	81.4	83.1	82.3	125
	Proposed KSVM	89.8	88.7	91.2	89.9	140
500	e-LION	83.6	82.4	84.8	83.7	125
	OLAP-DGCNN	86.7	85.6	87.4	86.5	160
	SEt-VD-CNN	88.8	87.5	89.4	88.4	150
	BR2-2 T-MICE	82.6	81.5	83.3	82.4	130
	Proposed KSVM	89.9	88.8	91.3	90.0	145

The proposed KSVM method consistently shows superior performance as in Table 6 across all test sets compared to existing methods. The accuracy of KSVM ranges from 89.0% to 89.9%, surpassing e-LION (82.5% to 83.6%), OLAP-DGCNN (85.4% to 86.7%), SEt-VD-CNN (87.3% to 88.8%), and BR2-2 T-MICE (81.8% to 82.6%). Precision, recall, and F1-score also favor KSVM, with precision from 87.8% to 88.8%, recall from 90.3% to 91.3%, and F1-score from 88.9% to 90.0%. This performance is notably higher than e-LION (81.2% to 82.4% precision, 83.1% to 84.8% recall), OLAP-DGCNN (84.3% to 85.6% precision, 86.0% to 87.4% recall), SEt-VD-CNN (86.2% to 87.5% precision, 88.1% to 89.4% recall), and BR2-2 T-MICE (80.5% to 81.5% precision, 82.2% to 83.3% recall). Although the execution time for KSVM (125s to 145s) is higher than some methods, the trade-off is justified by its superior accuracy and balanced performance metrics.

5 Conclusion

The proposed KSVM method significantly outperforms existing techniques—e-LION, OLAP-DGCNN, SEt-VD-CNN, and BR2-2 T-MICE—across various performance metrics. The KSVM achieves higher accuracy, precision, recall, and F1-score in comparison to the other methods, demonstrating its effectiveness in classifying and analyzing student behavior and performance in the eLearning dataset. Specifically, KSVM's accuracy ranges from 89.0% to 89.9%, while precision and recall reach up to 88.8% and 91.3%, respectively, consistently surpassing the benchmarks set by existing methods. The method's F1-score also reflects its balanced performance, ranging from 88.9% to 90.0%, which is superior to the scores of other methods. Although the execution time for KSVM is slightly higher, ranging from 125 to 145 seconds, this trade-off is justified by its superior predictive performance. Overall, the KSVM method offers a robust solution for analyzing complex student behavior and performance data, providing actionable insights and enhancing the effectiveness of eLearning systems. The results affirm the effectiveness of KSVM in handling large-scale datasets and its potential for broader applications in educational data analytics.

References

[1] Alahmari, F., Naim, A., & Alqahtani, H. (2023). E-learning modeling technique and convolution neural networks in online education. In *IoT-enabled convolutional neural networks: Techniques and applications* (pp. 261-295). River Publishers.

[2] Al-Yateem, N., Ismail, L., & Ahmad, M. (2024). *A comprehensive analysis on semiconductor devices and circuits. Progress in Electronics and Communication Engineering, 2 (1), 1–15*.

[3] Amer, K. Z., Azawi, A., Jebur, H. A., Al-shammary, A. A. G., & Al-sharifi, S. K. A. (2025). The Impact and Analysis of Mechanical Factors of the Mechanized Unit on the Production of" Vigna radiata L." Crop. *Natural and Engineering Sciences*, *10*(1), 418-424. https://doi.org/10.28978/nesciences.1651195

[4] Amin, S., Uddin, M. I., Alarood, A. A., Mashwani, W. K., Alzahrani, A., & Alzahrani, A. O. (2023). Smart E-learning framework for personalized adaptive learning and sequential path recommendations using reinforcement learning. *IEEe Access*, *11*, 89769-89790. https://doi.org/10.1109/ACCESS.2023.3305584

[5] Choset, K., & Bindal, J. (2025). Using FPGA-based embedded systems for accelerated data processing analysis. *SCCTS Journal of Embedded Systems Design and Applications*, *2*(1), 79-85.

[6] Ezaldeen, H., Bisoy, S. K., Misra, R., & Alatrash, R. (2023). Semantics aware intelligent framework for content-based e-learning recommendation. *Natural Language Processing Journal*, *3*, 100008. https://doi.org/10.1016/j.nlp.2023.100008

[7] Franzoni, V., Milani, A., Mengoni, P., & Piccinato, F. (2020). Artificial intelligence visual metaphors in e-learning interfaces for learning analytics. *Applied Sciences*, *10*(20), 7195. https://doi.org/10.3390/app10207195

[8] Golec, P., Hernes, M., Sajewski, T., Walaszczyk, E., Rot, A., Fojcik, M., ... & Dziembek, D. (2023, September). Forecasting e-learning Course Purchases Using Deep Learning Based on Customer Retention. In *European Conference on Artificial Intelligence* (pp. 142-155). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-66761-9_13

[9] Hussain, T., Yu, L., Asim, M., Ahmed, A., & Wani, M. A. (2024). Enhancing e-learning adaptability with automated learning style identification and sentiment analysis: a hybrid deep learning approach for smart education. *Information*, *15*(5), 277. https://doi.org/10.3390/info15050277

[10] Jakkaladiki, S. P., Janečková, M., Krunčík, J. A. N., Malý, F., & Otčenášková, T. (2023). Deep learning-based education decision support system for student E-learning performance prediction. *Scalable Computing: Practice and Experience*, *24*(3), 327-338. https://doi.org/10.12694/scpe.v24i3.2188

[11] Kew, S. N., & Tasir, Z. (2022). Developing a learning analytics intervention in e-learning to enhance students' learning performance: A case study. *Education and Information Technologies*, *27*(5), 7099-7134. https://doi.org/10.1007/s10639-022-10904-0

[12] Kumar, T. M. S. (2024). Integrative approaches in bioinformatics: Enhancing data analysis and interpretation. *Innovative Reviews in Engineering and Science*, *1*(1), 30-33. https://doi.org/10.31838/INES/01.01.07

[13] Kumari, P. S., Chandru, A. A., Diyaneshvar, P., Kumar, K. S., Akalya, A., & Gokila, K. (2024). Enhancing Character Recognition Accuracy Through Quantum Networking With Data Augmentation and Convolutional Neural Network. In *Quantum Networks and Their Applications in AI* (pp. 30-44). IGI Global. https://doi.org/10.4018/979-8-3693-5832-0.ch003

[14] Maher, Y., Moussa, S. M., & Khalifa, M. E. (2020). Learners on focus: Visualizing analytics through an integrated model for learning analytics in adaptive gamified e-learning. *IEEE Access*, *8*, 197597-197616. https://doi.org/10.1109/ACCESS.2020.3034284

[15] Mangaroska, K., Vesin, B., Kostakos, V., Brusilovsky, P., & Giannakos, M. N. (2021). Architecting analytics across multiple e-learning systems to enhance learning design. *IEEE Transactions on Learning Technologies*, *14*(2), 173-188. https://doi.org/10.1109/TLT.2021.3072159

[16] Moharm, K., & Eltahan, M. (2020, July). The role of big data in improving e-learning transition. In *IOP Conference Series: Materials Science and Engineering* (Vol. 885, No. 1, p. 012003). IOP Publishing. https://doi.org/ 10.1088/1757-899X/885/1/012003

[17] Naveenraj, X., Pandey, N., Asha, V., Mohan, V. C. J., Sethi, G., & Dhanraj, J. A. (2025). Transforming HR Analytics With AI and Quantum Network Integration. In *AI and Quantum Network Applications in Business and Medicine* (pp. 413-430). IGI Global Scientific Publishing. https://doi.org/10.4018/979-8-3693-8135-9.ch025.

[18] Nguyen, A., Gardner, L., & Sheridan, D. (2020). Data analytics in higher education: An integrated view. *Journal of Information Systems Education*, *31*(1), 61-71.

[19] Oubalahcen, H., & Tamym, L. (2023). The use of AI in E-learning recommender systems: A comprehensive survey. *Procedia Computer Science*, *224*, 437-442. https://doi.org/10.1016/j.procs.2023.09.061

[20] Paneque, M., del Mar Roldán-García, M., & García-Nieto, J. (2023). e-LION: Data integration semantic model to enhance predictive analytics in e-Learning. *Expert Systems with Applications*, *213*, 118892. https://doi.org/10.1016/j.eswa.2022.118892

[21] Senthilrajan, N., Siddhesh, O. S., Varun, A., & Vidhya, R. (2025). ProtectoLink Ease DeFi and Investing With Quantum AI and Its Applications in Blockchain Technology. In *Quantum AI and its Applications in Blockchain Technology* (pp. 57-74). IGI Global Scientific Publishing. https://doi.org/10.4018/979-8-3373-1657-4.ch005

[22] Shoeb, M., & Gupta, V. K. (2012). A crypt analysis of the Tiny Encryption Algorithm in key generation. *International Journal of Communication and Computer Technologies, 1*(1), 15-20. https://doi.org/10.31838/IJCCTS/01.01.01

[23] Silva, J. C. da, Souza, M. L. de O., & Almeida, A. de. (2025). Comparative analysis of programming models for reconfigurable hardware systems. *SCCTS Transactions on Reconfigurable Computing, 2*(1), 10–15.

[24] Singhvi, D., Nikhade, P., & Nimala, K. (2025). Enhancing Grammar Correction With Error Type Classification and Insights of Quantum NLP and Networking. In *Multidisciplinary Applications of AI and Quantum Networking* (pp. 167-194). IGI Global. https://doi.org/10.4018/979-8-3693-9336-9.ch012

[25] Tirumanadham, N. K. M. K., S, T., & M, S. (2024). Improving predictive performance in e-learning through hybrid 2-tier feature selection and hyper parameter-optimized 3-tier ensemble modeling. *International Journal of Information Technology*, *16*(8), 5429-5456. https://doi.org/10.1007/s41870-024-02038-y

Authors Biography

K. Shwetha is a dedicated academic and accomplished professional in the field of Computer Science. She currently serves as an Assistant Professor in the Department of Information Technology at Women's Christian College, Chennai. Her academic journey reflects a commitment to excellence, having earned her undergraduate and master's degrees in Computer Applications, which laid a strong foundation for her career in technology and education. Shwetha's professional trajectory is marked by her diverse experiences and contributions. She began her career as an intern at the prestigious National Informatics Centre (NIC), Chennai, where she worked in the Cloud Computing domain, gaining hands-on experience with cutting-edge technologies. Her passion for teaching and knowledge dissemination led her to start as a Computer Science teacher for secondary and higher secondary students at Bharathidasanar Matriculation Higher Secondary School. After a year of enriching experience, she transitioned into higher education, serving as an Assistant Professor at Guru Shree Shantivijay Jain College for Women, where she mentored and guided undergraduate students. Currently, Shwetha is pursuing her doctoral research at B.S. Abdur Rahman Crescent Institute of Science and Technology, specializing in Artificial Intelligence. Her research focuses on Machine Learning, Deep Learning, and Data Analytics, aiming to advance the understanding and applications of these transformative technologies. K. Shwetha's blend of academic expertise, professional experience, and ongoing research positions her as a remarkable educator and innovator in the field of Computer Science. Her dedication to fostering knowledge and inspiring future generations continues to make a significant impact in the academic community.

Dr.S. Shahar Banu is an accomplished educator and Associate Professor at B.S. Abdur Rahman Crescent Institute of Science and Technology, boasting an impressive 22 years of teaching experience. With a remarkable academic journey, she has presented and published papers in 11 international journals, showcasing her dedication to scholarly research. Her outstanding contributions have been acknowledged through awards, including the "Certificate of Appreciation" for outstanding academic results in the years 2007-2008 and 2008-2009. She has also received recognition and honorarium for her successful management of Spoken tutorial courses in 2021 and 2022, as well as her contribution to NAAC. Dr. Shahar Banu's professional journey is marked by a tireless commitment to education and academic excellence.