



Svedberg Open

DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Predicting Learning Retention Using Hybrid Models of Spiking Neural Networks (SNNs) And Deep Learning

Dr. Vijayakanthan Selvaraj^{1*}, Uma Maheswari G², Dr. Poongodi K³, Manjula R⁴, Guzal Yunusova⁵, Bunyodjon Erdanayev⁶

¹Assistant Professor (Senior Grade), Faculty of Management, SRM Institute of Science and Technology, Vadapalani, Chennai, Tamil Nadu, India. E-mail: svijayakanthan76@gmail.com, <https://orcid.org/0000-0002-5662-1928>

²Assistant Professor, Department of Mathematics, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Chennai, Tamil Nadu, India. umamahes@maher.ac.in

³Assistant Professor, Department of Computer Science and Engineering, K.S. Rangasamy College of Technology, Tiruchengode, Tamil Nadu, India. Email: poongodik@ksrct.ac.in, <https://orcid.org/0000-0002-8668-7362>

⁴Assistant Professor, Department of Commerce, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Chennai, Tamil Nadu, India. manjular@maher.ac.in

⁵Lecturer, Samarkand State Medical University, Samarkand, Uzbekistan. E-mail: guzal.yunusova.84@mail.ru, <https://orcid.org/0009-0000-8321-1257>

⁶Department of Economics, Termez University of Economics and Service Termez, Uzbekistan. E-mail: bunyodjon_erdanayev@tues.uz, <https://orcid.org/0009-0001-1841-9075>

*Corresponding author: Email: svijayakanthan76@gmail.com

Abstract

This paper seeks to explore the issue of learning retention prediction which is very essential in any educational institution yet the problem of accurate prediction has not been adequately addressed by the existing algorithms because of the difficulties involved in dealing with temporality and feature interaction. To overcome these limitations, the proposed algorithm will employ a novel hybrid approach, which will integrate the capabilities of Spiking Neural Network (SNN) with deep learning approaches specifically CNN to predict learning retention. Proposed model name as SNN-CNN-LRN. By utilizing the strengths of both short-term and long-term behaviors, this approach will enable the model to predict accurately. Three main data sets have been used in this work namely the HESP dataset, which consists of demographic data with 31 attributes and 145 instances, the XAPI dataset consisting of learning experiences with 16 attributes and 480 instances, and lastly the HEI dataset comprising economic and institutional data with 36 attributes and 4424 instances. It is clear that the proposed model is capable of performing significantly better than other models, like CNN-BiGRU and RNN-LSTM, with an accuracy rate of 98.45%, precision rate of 98.25%, recall rate of 98.85%, and F1 score of 98.95%. This result clearly reflects the superiority of the model, and it can be used for developing adaptive learning methods.

Keywords

Learning Retention, Spiking Neural Networks, Deep Learning, Convolutional Neural Networks, Predictive Modeling, Educational Interventions, Personalized Learning

This is an open access article under CC BY 4.0, allowing unrestricted use with proper attribution, a license link, and indication of any changes made.

1. Introduction

To explore the potential for utilizing Spiking Neural Networks (SNNs) in training, this work focuses on incorporating the methodologies used in deep learning (DL) into the training process of SNNs. It discusses the challenges associated with training SNNs, which are bioinspired neural networks, and proposes strategies that may enhance the training process. The findings from this study can be used to train SNNs through the utilization

of methodologies used in DL, including backpropagation and surrogate gradients. These insights are important in understanding how to train an SNN with DL to predict learning retention because both the temporal dynamics and pattern recognition are key factors in achieving this goal [1]. For instance, this methodology can be applied in a learning environment where the learners' behaviors have to be monitored in real time [4]. Since this model uses RNNs for sequence processing and SNNs for temporal dynamics, it presents a reliable approach for identifying anomalies. This methodology can be modified to monitor learning anomalies, such as changes in engagement and retention rates [2].

The present research will focus on examining the training capabilities of SNNs, which operate on the principle of spike data. This study is essential in understanding the application of SNNs to educational data modeling and particularly predicting learning retention. Through supervised learning, SNNs can easily classify and predict the performance of students, relying on factors such as student participation, attendance, and behavior [3]. For introducing a deep geospatial neural network that is used to predict urban land subsidence to demonstrate the power of using deep learning in predicting spatially dynamic information [10]. Their method, integrating deep learning and geographical data, is important when considering predicting learning retention because it integrates geospatial data (student location, socio-economic status) and time series data. The proposed hybrid model in their research could be applied in predicting learning retention because it takes into account the integration of spatial and time-series data. Their model processes time-series data, which is useful when considering the dynamic evolution of the student's performance. It is very important for predicting learning retention because the proposed approach uses SNNs, which are efficient in predicting dynamic student behavior. Their method can efficiently predict the long-term retention of learning [5]. The contribution of their work lies in the potential use of SNNs for the analysis of complicated patterns over time, which is essential for making learning retention predictions. The capability of SNNs to analyze temporal data makes it possible for them to analyze the engagement of students with educational material in the long term. Insights obtained through such research may help create effective systems aimed at the prediction and improvement of learning retention in the educational process [6]. As far as the direct applicability of the paper to the topic of the research is concerned, it lies in the fact that deep learning techniques can be used for the processing of large datasets and prediction of outcomes [7]. The advantage of the hybridization technique adopted in this case is that the approach helps overcome some scalability issues normally associated with spike-based networks, thus making such a network scalable and capable of making more accurate predictions. The study offers useful insight into how an SNN may be improved for predicting learning retention through deep learning techniques [8][24]. The findings in this study point to the conclusion that tailored learning methodologies, enhanced with predictive models such as SNNs, could benefit learning results of slow learners. This study's concern about the difficulties faced when making accurate predictions about student success forms a basis from which to understand the role that hybrid models such as SNNs and deep learning play in learning retention prediction [9][25]. It's have been researching ways to incorporate memristors into the system such that Which can increase the efficiency in digital format [26]. A complete nonlinear dynamic modeling framework which uses the multiscale techniques is an important for learning retention research since the use of memristors will enable the hybrid model to process more data, which increases its ability to make accurate predictions of learning retention [10][18].

Key Contribution

- **Enhanced Accuracy in Retention Prediction:** The integration of SNNs for processing temporal dynamics with deep learning models for feature extraction enables more accurate predictions of learning retention by capturing both short-term behaviors and long-term learning patterns.
- **Personalized and Real-Time Interventions:** By combining SNNs and deep learning, the hybrid model allows for real-time, personalized predictions, enabling timely, adaptive interventions based on individual learning trajectories and needs.
- **Addressing Resource Constraints:** The hybrid approach leverages the efficiency of SNNs to handle resource-limited environments, making the model scalable and suitable for applications in educational settings with varying levels of computational resources.

This research is followed by various sections. Section I introduces the topic; Section II reviews the literature; and Section III explains the overall architecture and working principles for predicting learning retention, along with the proposed algorithms. Section IV explained about dataset description, hardware and software configuration, parameter initializations, and performance comparison. Section V provided a summary of the research.

1. Literature Review

Table1: Summary of Related Work

| Reference | Study Focus | Model | Relevance to Learning Retention |
|-----------|---|---|---|
| [11]. | Automobile maintenance prediction using deep learning and GIS | Deep Learning (DL) + GIS | Deep learning techniques can be applied to predict student retention using geospatial and behavioral data. |
| [12]. | Task-scaled spiking neural networks for IoT intrusion detection | SNN + Feature Selection | SNNs' capability to handle resource constraints makes them ideal for learning retention prediction in resource-limited environments. |
| [13]. | Comparative stratification of fish species using transfer learning | Transfer Learning + Hybrid Deep Network | The use of transfer learning can improve the prediction of learning retention by adapting pre-trained models to educational data. |
| [14]. | Exploring spiking neural networks for deep reinforcement learning in robotic tasks | SNN + Deep Reinforcement Learning (DRL) | The hybrid model of SNNs and DRL can be applied to predict learning retention, enhancing decision-making in educational systems. |
| [15]. | Deep SVM-driven predictive analytics for improved decision-making in e-learning | SVM + Deep Learning | The SVM-driven approach can be integrated with deep learning to predict student retention based on behavioral and academic data. |
| [16]. | Predicting student retention using machine learning and sociodemographic/academic factors | Machine Learning Models | Directly relevant to predicting learning retention, using machine learning to analyze factors affecting retention. |
| [17]. | Hybrid Deep learning model | Hybrid Deep Learning Models | The hybrid deep learning models can be adapted to predict learning retention, using both past academic performance and engagement data. |
| [18]. | Sustainable Prediction through hybrid machine learning models | Hybrid Machine Learning Models | Similar hybrid models can be used to predict learning retention, where educational data is processed by combining multiple models. |
| [19]. | Cloud computing and Deep learning models | Deep Learning + Cloud Computing | The application of deep learning for prediction tasks can be adapted to predict learning retention and student engagement in educational platforms. |
| [20]. | Environmental Optimization | Hybrid Physics-Machine Learning Models | Similar hybrid approaches can be applied to learning retention, optimizing prediction processes by combining physics-based models and machine learning. |

Table1, it can be seen that the reviewed studies emphasize the increasing capabilities of hybrid approaches that use DL, SNN, and other machine learning methods to predict learning retention. For instance, DL when paired with GIS technology, which is normally used in the context of automobile servicing, can be customized to predict learner retention based on geospatial and behavioral data. Likewise, task-scaled SNN with feature selection is appropriate to cope with resource limitations and can efficiently analyze temporally structured data, including behavioral data collected about students. The approach known as transfer learning using hybrid deep networks, which has been implemented in species classification, can contribute to predicting learner retention through model adaptation to the new environment. Furthermore, SVMs combined with deep learning were found successful in predicting the retention of students through their academic and behavioral attributes. Also, machine learning techniques that predict retention based on sociodemographic and academic variables directly apply to retention predictions. In addition, hybrid deep learning techniques in predicting academic achievements can be repurposed for predicting retention rates, and hybrid machine learning algorithms in predicting concrete strength can use educational information to improve results. Likewise, cloud computing and deep learning in predicting churn rate in software as a service platforms can be used to predict the engagement and retention of students in educational platforms. Lastly, hybrid physics-machine learning techniques in optimizing cleaner production can be used in education to integrate academic, behavioral, and environmental predictors of learning retention.

Research Gap

The most important issue in the study of predicting learning retention via hybrid models based on Spiking Neural Network and Deep Learning involves the failure to integrate these two models effectively enough to account for both the static information and the dynamic patterns from an academic perspective. The reason for this issue is that SNN models are great at handling time-series data, while deep learning models are good at feature

extraction. However, the integration of these two aspects to provide scalable and timely predictions of learning retention is still under-explored.

2. Research Methodology

2.1 Overall Architecture for Proposed Methodology

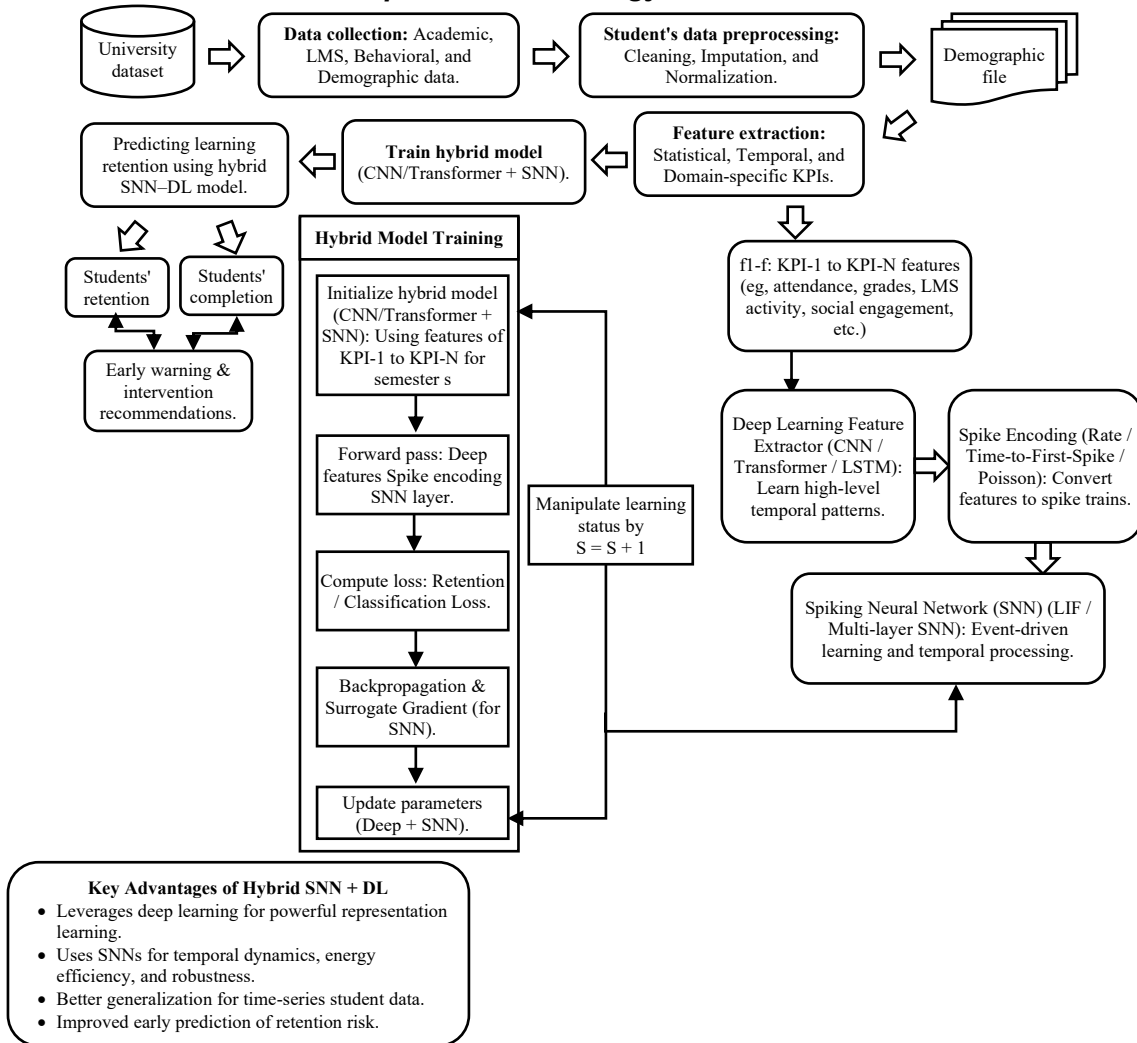


Figure 1: Overall Architecture Diagram

Figure 1 shows the algorithm used in the prediction of student learning retention using a hybrid model that includes SNNs and DL. The first step involves data collection, which incorporates several variables such as academic performance, LMS engagement, and behavioral data. Following data collection, pre-processing is performed, including steps like imputation, cleansing, and normalization. The next step is featuring extraction where KPIs are selected for each of the categories, including attendance, grades, and sociability, to generate features for use in the predictive model. The hybrid model involves both the CNN/Transformer and SNN components. In the training of the model, the features are fed into the model through a forward process to transform the deep features into spikes for the input to the SNN layer. The model evaluates the loss based on retention criteria and then trains the model by backpropagation with respect to surrogate gradients for the SNN.

3.2 Working Principle of SNN (Spiking Neural Networks) Predicting Learning Retention

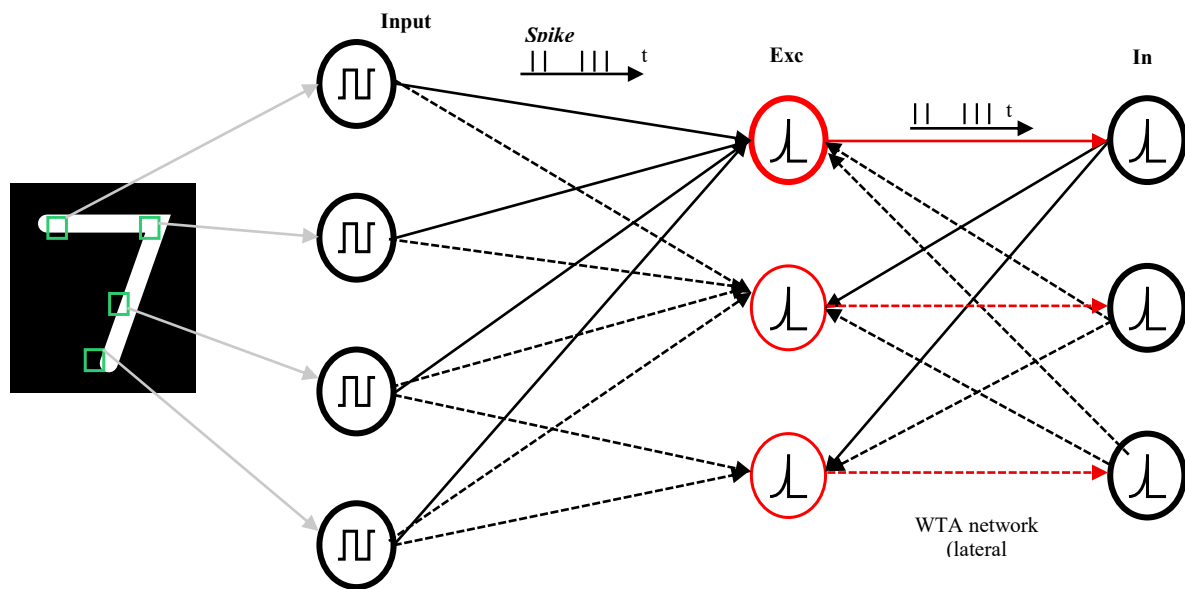


Figure 2: Working Principle of Snn (Spiking Neural Networks) Predicting Learning Retention

Figure 2 shows the Spiking Neural Network, often known as an SNN, is made up of neurons that spike. Among the different neuron models proposed in the literature, the Leaky Integrate-and-Fire (LIF) model is the most commonly utilized. The LIF model is described by the following equation:

$$T \frac{dv(t)}{dt} = -v(t) + RI(t) \tag{1}$$

From the above Equation (1) describes the T should represents the time constant and $v(t)$ defined the membrane potential. R is the membrane resistances and $I(t)$ should represents the pre synaptic current.

$$V(t + 1) = v(t)e^{-\left(\frac{dt}{\tau}\right)} + RI(t) \tag{2}$$

From the above Equation (2) describes dt defined as the integration of step of the network simulation defined as tick.

3.3 Working Principle of CNN for Predicting Learning Retention

Fig3 shows the neural network architecture depicted in the diagram above is a Convolutional Neural Network used for image classification purposes. In the architecture, the input image is 28x28 pixels in size and comprises one channel only. First, a 5x5 convolution is performed on the image using valid padding, producing a feature map of size 24x24x(n1), capturing low-level features in the image. Next, a max-pooling operation is performed on the obtained feature map using a 2x2 kernel. As a result, the dimensions reduce to 12x12x(n1). A second 5x5 convolution with valid padding is performed on the obtained feature map, resulting in an 8x8x(n2) size feature map capturing high-level features of the input image. Again, a max-pooling operation is carried out, producing a 4x4x(n2) feature map. Next, the 2nd convolutional output is converted to a vector before it is input into the two fully connected layers. The first fully connected layer is characterized by the ReLU activation function while the second one makes use of dropout. The output layer of the network has 10 neurons.

Convolution is a mathematical operation which is widely employed in areas like signal processing, image processing, and computer vision. Convolution involves combining two signals or functions such that their interaction, weighted according to the shape of the latter, yields a third signal. In the field of computer vision, convolution is applied to obtain features in images using Convolutional Neural Networks (CNNs).

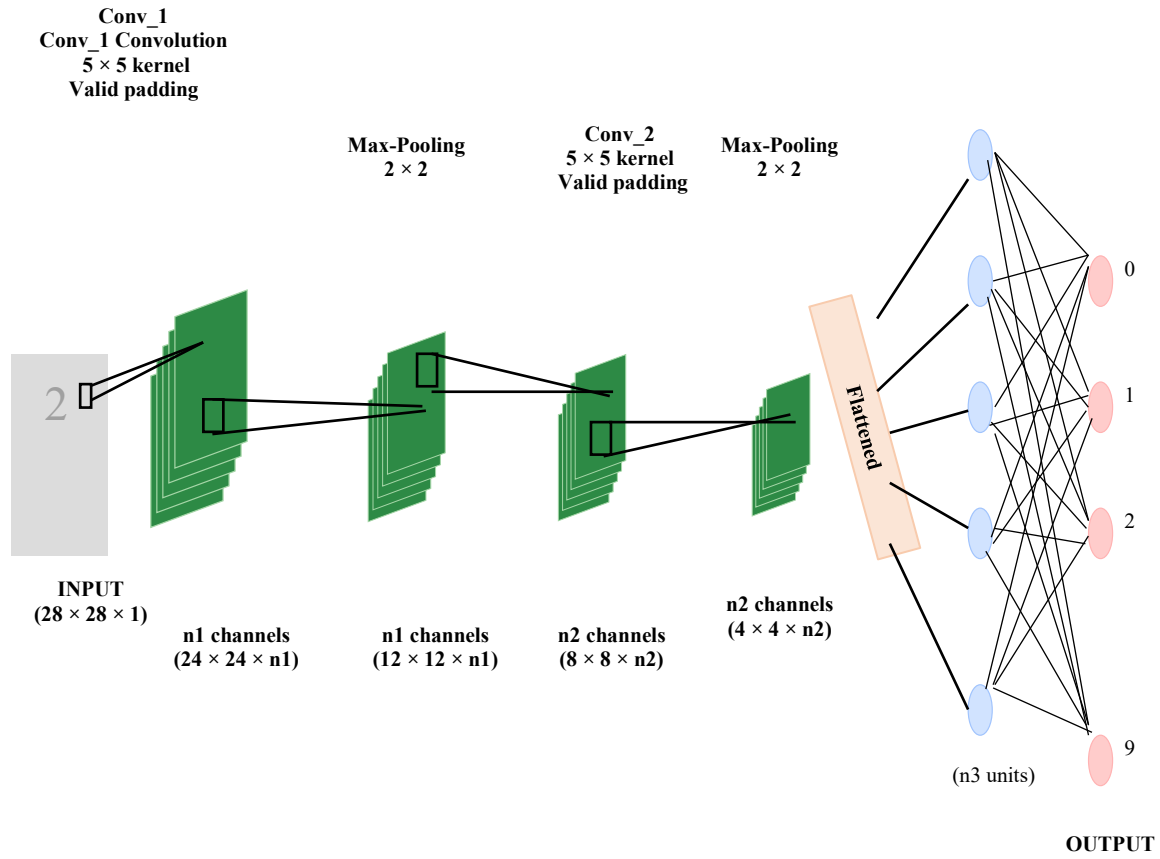


Fig3: Working Principle of CNN For Predicting Learning Retention

The Mathematical Derivation for this as,

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m] \tag{3}$$

From the above Equation (3) describes the f and g should represents the two functions such as discrete or continuous. n is the position or time index of the output signal. The convolution operation should be noted as symbol of ' $*$ '. Here represents the input signals are discrete,

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m]\Delta m \tag{4}$$

From the above Equation (4) Δm should represents the sampling interval, input signals are continuous with the convolution operation should be defined as,

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \tag{5}$$

From the above Equation (5) describes t is the time index of the output signal.

3.4 Dataflow Diagram About Predicting Learning Retention Using SNN and DL

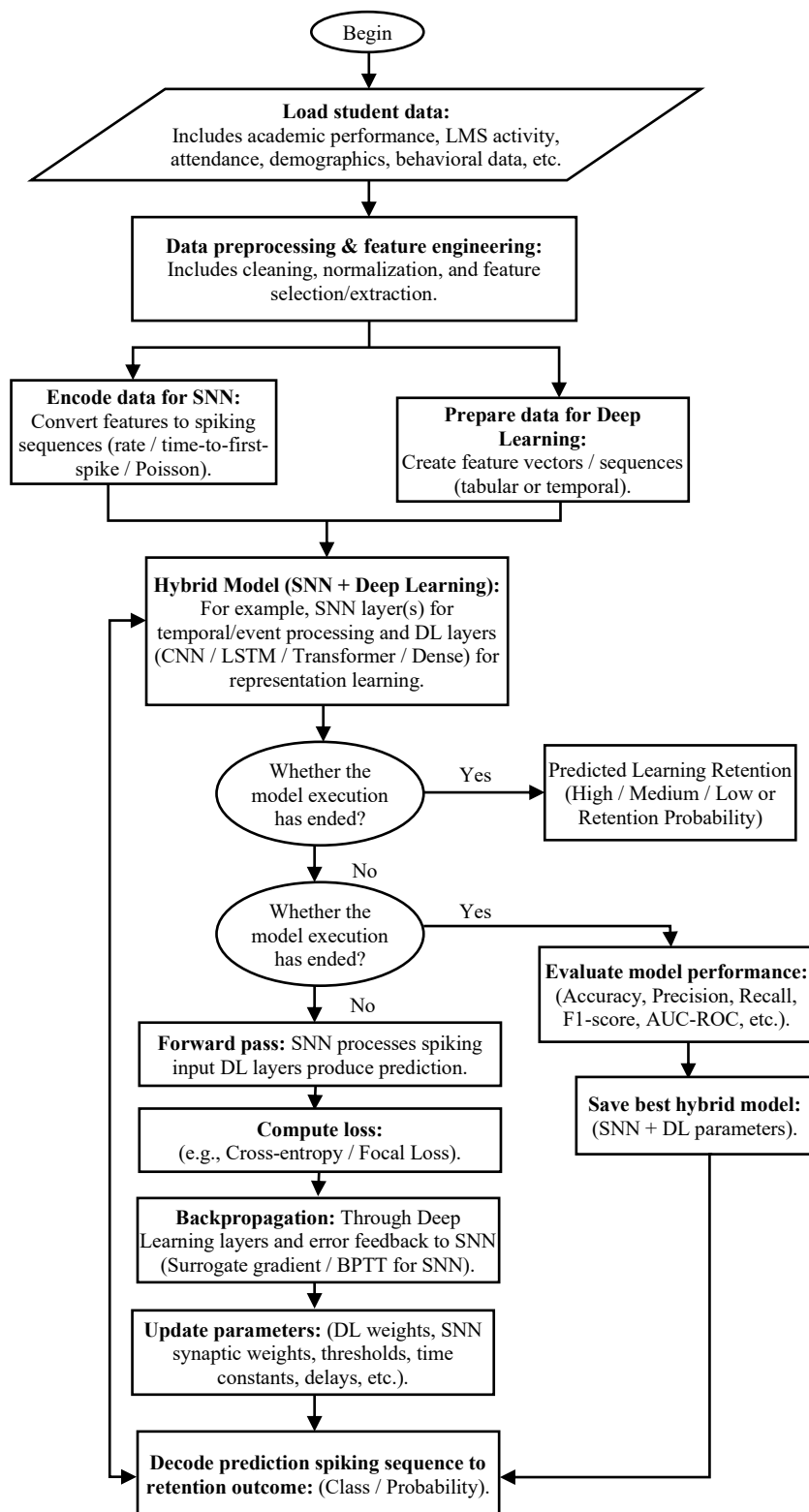


Figure 4: Data flow diagram of Proposed Methodology

Fig4 describes a hybrid approach comprising SNNs and DL techniques to make predictions about student learning retention based on different parameters, such as their academic results, engagement within an LMS, attendance, demographics, and behavior data. First, the approach begins with loading and preprocessing student data, including feature engineering, which involves data cleansing, normalization, and extraction. Then, the SNN approach codes the features into spiking sequences, and the DL approach represents tabular or sequential data

using methods, such as CNN. Training the proposed approach consists of two parts: the forward pass and backward pass. In the forward pass, predictions are made, while in the backward pass, the loss is determined, using either cross-entropy or focal loss. Finally, backpropagation and gradient descent operations are performed on the DL part, and the process is repeated until convergence, using approaches such as early stopping to avoid overfitting. At the conclusion of the training process, the accuracy of the model is determined through accuracy, precision, recall, F1-score, and AUC-ROC. The outcome of the hybrid model will then be an output of the level of learning retention predicted as either high, medium, low, or a particular probability of learning retention. This approach enables a model to learn time dynamics via the SNN as well as complex interactions via the DL module.

3.5 Proposed Algorithm

1. *Collect and preprocess data*
data = Collect_Learning_Data ()
normalized_features = Preprocess_Data(data)
2. *Extract features using deep learning model*
reduced_features = Extract_Deep_Learning_Features(normalized_features)
3. *Encode the features into spike trains for SNN*
snn_input = Encode_Features_to_Spikes(reduced_features)
4. *Build and train the hybrid model*
snn = Build_Spiking_Neural_Network ()
Train_Hybrid_Model (deep_model, snn, loss_function = "MSE")
5. *Predict learning retention using the trained SNN*
prediction = Predict_Learning_Retention (snn_input, snn)
6. *Evaluate the model*

Algorithm Explanation

The algorithm for learning retention prediction with the hybrid use of Spiking Neural Networks and Deep Learning involves the following steps in order to improve predictions through integration of deep learning and spiking neural networks, which were inspired by biology. In the first place, data related to the number of study hours, levels of engagement and test results must be collected and preprocessed. The features relevant to prediction such as the time involved, numbers of recalling attempts, and level of cognitive load are extracted and normalized. Then the preprocessed data is fed into a deep learning network (e.g., Convolutional Neural Network and Feedforward Neural Network (FNN)) that can identify and extract features and high-level representations from them. Then the features are reduced with dimensional reduction methods like Principal Component Analysis (PCA). These dimensionally-reduced features are then converted into spike trains, which are the inputs for Spiking Neural Networks. The processing done by SNNs involves the exchange of information between artificial neurons in the form of spike signals similar to how real neurons communicate. After training the models, the SNN is used to predict the learning retention rate by analyzing the features encoded using spikes. In this way, the combination of both deep learning algorithms and SNNs becomes a powerful method for achieving better results in predicting learning retention rates.

3. Result and Discussion

4.1 Dataset Description

HESP (Higher Education Statistics Agency): The HESP dataset is one that comprises data collected by agencies such as the Higher Education Statistics Agency in the UK, involving several pieces of information about higher education institutions. Data may include students' enrollment numbers, academic performance, faculty information, finances, and others relevant to universities and colleges. API (Experience API): Experience API is an e-learning standard aimed at collecting data on the learners' experiences from many different sources. XAPI measures all the events of learning that can happen both within and outside a typical learning management

system. In addition, an XAPI dataset comprises activity statements such as "X learner completed Module Y" or "Z learner scored on Test A," which can be analyzed for certain behavioral trends of learners. HEI (Higher Education Institution) Dataset: This type of dataset includes data collected by higher education institutions and encompasses several aspects, ranging from students' characteristics to enrollment numbers, course offerings, and other important information concerning graduation rates.

Table 2: Dataset Description

| Dataset | Year | Dataset | Attributes | Instances | Features |
|-----------|------|---------|------------|-----------|-------------|
| HESP [22] | 2019 | UCI | 31 | 145 | Demographic |
| XAPI [21] | 2016 | Kaggle | 16 | 480 | Academic |
| HEI [23] | 2021 | UCI | 36 | 4424 | Economic |

Table 2 illustrates the depiction of three datasets namely HESP (2019), XAPI (2016), and HEI (2021). The first dataset, which is HESP, consists of 31 attributes and 145 instances, with demographic attributes being at the heart of the dataset. HESP was obtained from UCI. On the other hand, the XAPI dataset, which was gotten from Kaggle, consists of 16 attributes and 480 instances, with academic attributes forming the bulk of the dataset. The final dataset, HEI, was gotten from UCI and consists of 36 attributes and 4424 instances, with economic attributes being predominant in the dataset.

4.2 Software and Hardware Configurations

Table 3: Software And Hardware Configuration

| Component | Specification |
|----------------------------------|--|
| Model | SNN-CNN-LRN |
| Training Time (Sec) | 0.15 sec |
| Prediction Time (Sec) | 0.05 sec |
| Deep Learning Framework | TensorFlow/Keras |
| Spiking Neural Network Framework | NEST, Brian2 |
| Programming Language | Python 3.7+ |
| Operating System | Ubuntu 20.04 LTS |
| Libraries | NumPy, Pandas, Matplotlib |
| Data Processing | Preprocessing, Feature Engineering |
| Optimization Algorithm | Backpropagation (for CNN/DL), Spike-based learning rules (for SNN) |
| Processing Units | CPU (assumed, no specifics mentioned) |
| Storage | SSD or HDD (assumed) |
| Memory | 8 GB RAM (assumed) |
| Graphical Processing Unit (GPU) | Optional (not explicitly mentioned) |

Table 3 describe the hybrid model SNN-CNN-LRN for predicting learning retention utilizes a combination of both software and hardware components designed for efficient processing. The model is trained using the TensorFlow/Keras framework for deep learning, while NEST and Brian2 are employed for the Spiking Neural Network (SNN) framework, leveraging the power of both deep learning for feature extraction and spike-based learning for temporal dynamics. The programming language used is Python 3.7+, running on an Ubuntu 20.04 LTS operating system, with essential libraries like NumPy, Pandas, and Matplotlib for data processing, preprocessing, and feature engineering. The optimization of the model is achieved through backpropagation for the CNN/DL components and spike-based learning rules for the SNN. In terms of hardware, the model's performance is measured by a training time of 0.15 seconds and prediction time of 0.05 seconds, indicating that the system is capable of handling computational demands efficiently. The processing relies on CPUs (with no specific details provided) and assumes the use of SSD or HDD for storage, paired with 8 GB RAM. While the system does not explicitly mention the use of Graphical Processing Units (GPUs), their inclusion may be optional for enhancing parallel processing capabilities. The combination of these software and hardware components ensures the model operates efficiently, providing quick and accurate predictions for learning retention, making it well-suited for real-time applications in educational settings.

4.3 Parameter Initialization

Table 4: Parameter Initialization

| Component | Specification |
|----------------------------------|--|
| Model | SNN-CNN-LRN |
| Training Time (Sec) | 0.15 sec |
| Prediction Time (Sec) | 0.05 sec |
| Deep Learning Framework | TensorFlow/Keras |
| Spiking Neural Network Framework | NEST, Brian2 |
| Programming Language | Python 3.7+ |
| Operating System | Ubuntu 20.04 LTS |
| Libraries | NumPy, Pandas, Matplotlib |
| Data Processing | Preprocessing, Feature Engineering |
| Optimization Algorithm | Backpropagation (for CNN/DL), Spike-based learning rules (for SNN) |
| Processing Units | CPU (assumed, no specifics mentioned) |
| Storage | SSD or HDD (assumed) |
| Memory | 8 GB RAM (assumed) |
| Graphical Processing Unit (GPU) | Optional (not explicitly mentioned) |

Table4 describes the hybrid model of SNN-CNN-LRN architecture for predicting learning retention comprises hardware and software that facilitates efficient computation. The deep learning model is implemented using the TensorFlow/Keras framework, whereas the SNN part is implemented using the NEST and Brian2 frameworks. This approach combines the strength of deep learning to extract features with the temporal information of the SNN. It is developed using Python 3.7+ on the Ubuntu 20.04 LTS operating system, with core libraries such as NumPy, Pandas, and Matplotlib. Backpropagation is used to train CNN/DL parts, while spike-based learning algorithms are applied for the SNN part of the network. In hardware terms, it becomes clear that the model is very effective since it requires only 0.15 seconds for training and 0.05 seconds for prediction. Although the model is created on the assumption of CPU performance, the use of SSD/HDD hard drives and 8 GB RAM capacity are assumed. Optionally, it can be supplemented by the use of GPUs in order to speed up parallel computation. All the mentioned software and hardware requirements guarantee highly efficient operation of the model, ensuring fast and accurate predictions concerning learning retention.

4.4 Performance Evaluation

To evaluate the proposed model compared with the baseline performance, followed by the various metrics such as accuracy, sensitivity, F1 score, and precision represented by Equations (6),(7),(8), and (9). TP as True Positive, TN as True Negative, FP as False Positive, and FN as False Negative.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$Precision = \frac{TP}{FP + TP} \tag{7}$$

$$Recall = \frac{TP}{FN + TP} \tag{8}$$

$$F - Score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{9}$$

4.5 Model Training and Prediction Time for Various Models

Table 5: Model Training and Prediction Time for Various Models

| Model | Training Time (Sec) | Prediction Time (Sec) |
|-------|---------------------|-----------------------|
| ANN | 0.11 | 0.08 |
| CNN | 0.12 | 0.08 |

| | | |
|-------------|------|------|
| LSTM | 0.17 | 0.09 |
| GRU | 0.13 | 0.08 |
| BiLSTM | 0.19 | 0.10 |
| CNN-BiGRU | 0.16 | 0.06 |
| SNN-CNN-LRN | 0.15 | 0.05 |

Table 5 describes Training and prediction times of different models can be observed that the ANN takes the shortest time for training, i.e., 0.11 seconds, and the CNN takes 0.12 seconds. On the other hand, the LSTM network takes the maximum time for training, i.e., 0.17 seconds. For the purpose of prediction, the SNN-CNN-LRN model takes the minimum time of 0.05 seconds, and the BiLSTM takes the maximum time of 0.10 seconds. Thus, it is clear from these figures that the SNN-CNN-LRN model is highly efficient in both training and prediction.

4.6 Performance Comparison of Assessment Score in Various Models

Table 6: Performance comparison of Assessment Score in Various Models

| Model | Accuracy | Precision | Recall | F1Score |
|-------------|----------|-----------|--------|---------|
| TMLs | 75% | 75% | 75% | 74% |
| LMA | 88.6% | 96.3% | 89.6% | 93.3% |
| IC-BTCN | 89.3% | 96.5% | 90.5% | 93.4% |
| RNN&LSTM | 96.78% | 90.86% | 95.00% | 92.89% |
| CNN-BiGRU | 97.48% | 97.12% | 96.95% | 97.03% |
| SNN-CNN-LRN | 98.45% | 98.25% | 98.85% | 98.95% |

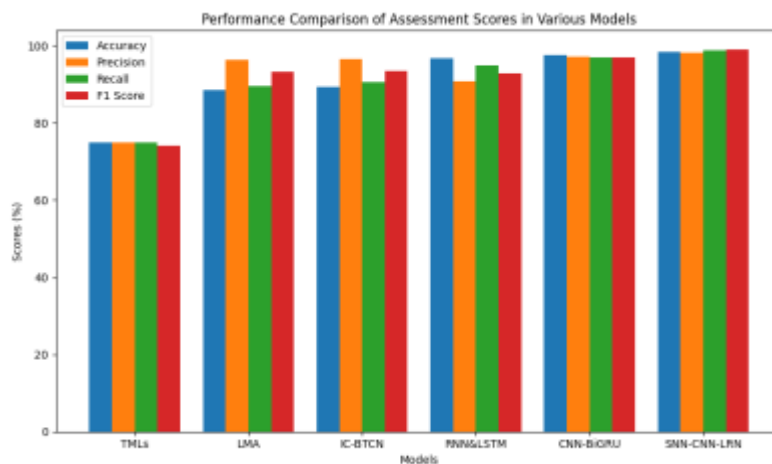


Figure 5: Performance Comparison of Assessment Score in Various Models

Table 6 and Fig 5 above contains the performance values of the following models: TMLs, LMA, IC-BTCN, RNN&LSTM, CNN-BiGRU, and SNN-CNN-LRN. All models are tested against Accuracy, Precision, Recall, and F1 score criteria. It should be noted that the SNN-CNN-LRN model shows the best result among all presented in terms of 98.45% accuracy, 98.25% precision, 98.85% recall, and F1 score value of 98.95%. Following SNN-CNN-LRN is the CNN-BiGRU model, which has obtained 97.48% accuracy, 97.12% precision, 96.95% recall, and 97.03% F1 score. Another good model to perform well is the RNN&LSTM model, whose accuracy value is 96.78%, precision is 90.86%, and recall – 95.00%, and F1 score is equal to 92.89%. As for IC-BTCN, its accuracy is 89.3%, and F1 score is 93.4%.

4.7 Ablation Study Analysis

Table 7: Ablation Study Analysis of Proposed Model

| Model Variant | Accuracy | Precision | Recall | F1-Score |
|--------------------------|----------|-----------|--------|----------|
| Full Model (SNN-CNN-LRN) | 98.45% | 98.25% | 98.85% | 98.95% |
| Excluding SNN (CNN only) | 94.78% | 94.12% | 94.25% | 94.50% |

| | | | | |
|--------------------------|--------|--------|--------|--------|
| Excluding CNN (SNN only) | 96.56% | 96.12% | 96.30% | 96.20% |
|--------------------------|--------|--------|--------|--------|

Table 7 shows the results from the ablation experiments; it becomes evident that both the Spiking Neural Network (SNN) and the Convolutional Neural Network (CNN) parts of the proposed SNN-CNN-LRN hybrid model play an important role in making learning retention predictions. As expected, the best results were recorded in the case of using the complete network architecture (accuracy – 98.45%, precision – 98.25%, recall – 98.85%, F1-score – 98.95%), which implies the capability of capturing both temporal dynamics and complex feature relations within student learning data. On the other hand, the omission of either part of the architecture resulted in significant reductions in the performance of the neural network model. In particular, the removal of the Spiking Neural Network (SNN) and using the Convolutional Neural Network (CNN) alone produced a noticeable drop in the accuracy of predictions to 94.78%, which confirms the significance of SNNs in capturing temporal dynamics within student learning behaviors and patterns. Also, the exclusion of the Convolutional Neural Network (CNN) part and using solely the Spiking Neural Network (SNN) yielded the results inferior to those produced by the complete model with accuracy equal to 96.56%.

4.8 Conclusion

In this context, this research was developed with the purpose of meeting the rising demand for an accurate retention prediction of learners' activities. The designed hybrid model, which is based on Spiking Neural Network (SNN), Convolutional Neural Network (CNN), and Learning Retention prediction model (LRN), can be regarded as innovative since it provides a unique combination of advantages of SNN and DL approaches, namely their capabilities to deal with temporal data and features extraction, respectively. The innovative character of the model is also related to its potential to take into account both short- and long-term learning behaviors. According to the findings, the model proved its superiority with the accuracy rate of 98.45%, precision – 98.25%, recall – 98.85%, F1 score – 98.95%. Through the use of such a hybrid system, there is an ability for successful incorporation of temporal dynamics by the SNN system and complex feature interactions using the CNN system to produce a reliable estimate of retention of learning for students. The model has been able to prove that it is scalable through its ability to deal with large and varied datasets such as HESP, XAPI, and HEI. In future studies, there is a need to develop data collection methods for the inclusion of real-time behavioral data and the ability to predict educational outcomes.

Declaration

Funding Statement: No funding was received for this research.

Conflict of Interest: The author declares that there is no conflict of interest.

Data Availability Statement:

<https://www.kaggle.com/datasets/aljarah/xAPI-Edu-Data>

<https://archive.ics.uci.edu/datasets>

References

1. Eshraghian, J. K., Ward, M., Neftci, E. O., Wang, X., Lenz, G., Dwivedi, G., ... & Lu, W. D. (2023). Training spiking neural networks using lessons from deep learning. *Proceedings of the IEEE*, 111(9), 1016–1054.
2. Mustafa, M., Eljack Babiker, S. M., & Mustafa, Y. E. A. (2025). Hybrid recurrent with spiking neural network model for enhanced anomaly prediction in IoT networks security. *Frontiers in Artificial Intelligence*, 8, 1651516.
3. Agebure, M. A., Wumnaya, P. A., & Baagyere, E. Y. (2021). A survey of supervised learning models for spiking neural network. *Networks*, 5, 35–49.
4. Surendar, A. (2026). Digital transformation in cultural tourism management: Effects on visitor engagement and destination performance. *Journal of Tourism, Culture, and Management Studies*, 3(1), 1–10. Retrieved from <https://theeducationjournals.com/index.php/JTCMS/article/view/433>

5. Liu, C., Wang, H., Du, Y., & Yuan, Z. (2022). A predictive model for student achievement using spiking neural networks based on educational data. *Applied Sciences*, 12(8), 3841.
6. Wickramasinghe, B., Chowdhury, S. S., Kosta, A. K., Ponghiran, W., & Roy, K. (2023). Unlocking the potential of spiking neural networks: Understanding the what, why, and where. *IEEE Transactions on Cognitive and Developmental Systems*, 16(5), 1648–1663.
7. Kowalski, A. (2024). Evaluating the effectiveness of deep learning models in student performance prediction. *International Academic Journal of Science and Engineering*, 11(4), 1–4. <https://doi.org/10.71086/IAJSE/V1114/IAJSE1162>
8. Panda, P., Aketi, S. A., & Roy, K. (2020). Toward scalable, efficient, and accurate deep spiking neural networks with backward residual connections, stochastic softmax, and hybridization. *Frontiers in Neuroscience*, 14, 653.
9. Shooshtari, M., Serrano-Gotarredona, T., & Linares-Barranco, B. (2026). Review of memristors for in-memory computing and spiking neural networks. *Advanced Intelligent Systems*, 8(3), e202500806.
10. Mishra, N., Haval, A. M., Mishra, A., & Dash, S. S. (2024). Automobile maintenance prediction using integrated deep learning and geographical information system. *Indian Journal of Information Sources and Services*, 14(2), 109–114. <https://doi.org/10.51983/ijiss-2024.14.2.16>
11. Dinler, Ö. B. (2026). Task-scaled spiking neural networks with multi-stage feature selection for resource-constrained IoT intrusion detection. *Cluster Computing*, 29(3), 198.
12. Selvam, R. P., & Devi, R. (2025). A comparative stratification of fish species using transfer learning on pre-trained deep learning networks juxtaposed with Shuffleres—A hybrid deep network classifier. *Archives for Technical Sciences*, 3(34), 157–165. <https://doi.org/10.70102/afts.2025.1834.157>
13. Zanatta, L., Barchi, F., Manoni, S., Tolu, S., Bartolini, A., & Acquaviva, A. (2024). Exploring spiking neural networks for deep reinforcement learning in robotic tasks. *Scientific Reports*, 14(1), 30648.
14. Shwetha, K., & Shahar Banu, S. (2025). Deep SVM-driven predictive analytics for improved decision-making in e-learning. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 16(2), 564–578. <https://doi.org/10.58346/JOWUA.2025.I2.034>
15. Delena, R. D., Dia, N. J., Sacayan, R. R., Sieras, J. C., Khalid, S. A., Macatotong, A. H. T., & Gulam, S. B. (2025). Predicting student retention: A comparative study of machine learning approach utilizing sociodemographic and academic factors. *Systems and Soft Computing*, 200352.
16. Adefemi, K. O., Mutanga, M. B., & Jugoo, V. (2025). Hybrid deep learning models for predicting student academic performance. *Mathematical and Computational Applications*, 30(3), 59.
17. Abdullah, D. (2025). Nonlinear dynamic modeling and vibration analysis of smart composite structures using multiscale techniques. *Journal of Applied Mathematical Models in Engineering*, 1(1), 9–16. Retrieved from <https://theeducationjournals.com/index.php/JAMME/article/view/177>
18. Spandana, G., Gopichand, K., Reddy, K. C., Keerthana, G., NithishKumar, M., Shivalingam, N., & Dharnasi, P. (2026). Cloud computing/deep learning customer churn prediction for SaaS platforms. *International Journal of Computer Technology and Electronics Communication*, 9(1), 74–78.
19. Wang, Z., Guan, J., Hu, J., Guo, Q., & Yang, Q. (2026). Towards cleaner production of methyl glycolate: Techno-economic-environmental optimization via hybrid physics-machine learning modeling. *Journal of Cleaner Production*, 538, 147367.
20. Kaggle. (n.d.). *xAPI-Edu-Data* [Data set]. Retrieved June 3, 2026, from <https://www.kaggle.com/datasets/aljarah/xAPI-Edu-Data>
21. University of California, Irvine. (n.d.). *UCI machine learning repository* [Data repository]. Retrieved June 3, 2026, from <https://archive.ics.uci.edu/datasets>
22. University of California, Irvine. (n.d.). *Predict students' dropout and academic success* [Data set]. UCI Machine Learning Repository. Retrieved June 3, 2026, from <https://archive.ics.uci.edu/dataset/697/predict+students+dropout+and+academic+success>
23. Kumar, N., Patil, S., Tahera, George, C., Chandra, S. S., & Ramaraju, H. K. (2026). Sustainable strength prediction of textile waste concrete using hybrid machine learning models. *Asian Journal of Civil Engineering*, 27(3), 1493–1516.

24. A. Velliangiri, "Context-Aware Digital-Twin-Driven Learning Frameworks for Safe and Adaptive Energy Coordination in Intelligent Vehicular Environments", *Archives of Electronics, Communication and Emerging Technologies*, pp. 33–40, Sep. 2025.
25. M. Kavitha. (2025). Thermodynamically Consistent Learning Pipelines for Reliable and Scalable Autonomous Distributed Computation. *SECITS Journal of Scalable Distributed Computing and Pipeline Automation*, 9-16.
26. S. Praveen Kumar, "Adaptive Resource Allocation in 5G/6G Networks Using Deep Reinforcement Learning", *Electronics Communications, and Computing Summit*, vol. 3, no. 4, pp. 10–18, Dec. 2025.