frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 06 January 2025
DOI 10.3389/fphy.2024.1492114

OPEN ACCESS

EDITED BY
Chengyi Xia,
Tianjin Polytechnic University, China

REVIEWED BY
Yasuko Kawahata,
Rikkyo University, Japan
Keke Shang,
Nanjing University, China
Nasreddine Abdelli,
Ecole Nationale Supérieure en Sciences et Technologies de l'Informatique, Algeria

*CORRESPONDENCE
Zhongshan Chen,
ycddczs@163.com

RECEIVED 06 September 2024
ACCEPTED 12 December 2024
PUBLISHED 06 January 2025

CITATION
Zhu D, Wei Y, Cai J, Wang J and Chen Z (2025) A security data detection and management method in digital library network based on deep learning.
Front. Phys. 12:1492114.
doi: 10.3389/fphy.2024.1492114

A security data detection and management method in digital library network based on deep learning

Diyin Zhu[1], Yihang Wei[1], Jiali Cai[1], Jingwen Wang[1] and Zhongshan Chen[2]*

[1]Library, Shaoxing University Yuanpei College, Shaoxing, China, [2]School of Mathematics and Informaion Science, Nanjing Normal University of Special Education, Nanjing, China

With the rapid growth of data volume in digital library and the increasingly complex network environment, traditional network security measures are no longer able to meet their security needs. In response to the problems of low detection accuracy and long detection time in traditional network security methods, we propose a digital library network security data detection and management method based on bidirectional gated recurrent unit and temporal convolutional network detector (BiGRU-TCNDetector) using the powerful capabilities of deep learning technology. It efficiently and intelligently detects and manages security data in digital library network. The method combines the structures of temporal convolutional network (TCN) and bidirectional gated recurrent unit (BiGRU) to extract spatial and temporal features from digital library network security data. And it improves the accuracy of security data detection based on BiGRU-TCN. In addition, the importance of each security data feature is calculated through attention mechanism to reduce the loss of important digital network security data information, which is then output by the global pooling layer to the classifier for classification. Finally, comparative experiments are conducted to verify that the digital library network security data detection method based on BiGRU-TCNDetector has better detection performance compared to other methods, providing a solid technical guarantee for the stable operation and sustainable development of digital library.

KEYWORDS

security data, detection, deep learning, digital library network, performance

Highlight

- This paper proposes a digital library network security data detection method.
- The method reduces the lack of key information and further improving the accuracy.
- The method is validated compared to other baselines, demonstrating good performance.

1 Introduction

Digital library has become an important platform for knowledge dissemination and academic exchange in modern society. They not only possess a lot of digital resources, but

also achieve convenient access and efficient management of resources through advanced information technology. However, while digital library enjoys the convenience brought by information technology, it also faces increasingly severe cybersecurity challenges. These challenges include but are not limited to system vulnerabilities, hacker attacks, virus propagation and data breaches, which seriously threaten the stable operation of digital library and network data security [1].

In the early stages of modern library network construction, people emphasize more on the convenience and availability of the network, while neglecting the security of the network. When the coverage of the library network is limited to the library LAN, the security of the library network is not highlighted. With the acceleration of digital library construction, library launch comprehensive network information services. The data in the network environment refers to the information stored in network file servers and other servers, as well as the information propagated in the network. Data security in the network environment is the essence of network security. Network security is a comprehensive discipline that involves multiple disciplines such as information security technology and cryptography [2]. Its essence is data security on the network.

The digital library network system has the characteristics of wide distribution and channel commonality. Although it increases the practicality of the network, it inevitably increases the vulnerability of the system, making digital library has to face the serious challenge of network security. The proliferation of network viruses, malicious attacks by hackers and the security needs of readers' network information have raised higher requirements for the network security of digital library [3, 4]. Generally speaking, the network security of digital library refers to the mechanism by which the various components of the digital library network system are not damaged, tampered with or leaked due to accidental reasons. This ensures the continuous and normal operation of the digital library network system. Its ultimate goal is to maintain reliable confidentiality, integrity, availability and controllability during the processing and transmission of digital library network information.

The digital library is a key construction project for library work in the new era and it is also the focus of global competition in science, technology and culture. It marks the level of development of science, technology, culture and information. At the initial stage of the construction of digital library system networking and informatization, people pay more attention to its usability and convenience, but ignore the security of Internet information resources. With the continuous development of digital library systems, library has begun to provide comprehensive network information services and share technological. However, due to the openness of their system architecture, wide distribution, channel commonality and resource sharing, digital library has to face severe challenges in network information security. A series of security issues such as data loss, information leakage, website tampering, system and application software vulnerabilities, hacker attacks, malicious code and misoperation by internal personnel seriously affect the normal operation of digital library, causing unnecessary losses to the system itself and users [5–7]. Therefore, studying various network factors that pose a threat to the digital library system and improving the security system of the digital library system has become an important issue related to the construction and development of the digital library system.

Deep learning has made significant progress in various fields such as image recognition and natural language processing due to its powerful data processing and self-learning capabilities [8]. Deep learning technology has also been widely applied in scenarios such as network attack detection, malware identification and abnormal traffic analysis, demonstrating its unique advantages. Compared to traditional detection techniques, deep learning has demonstrated significant advantages in processing massive amounts of data and recognizing complex feature patterns. This technology has the ability of automatic learning, which can mine and extract hidden and highly complex feature patterns from large datasets. This process not only reduces the need for manual intervention, but also greatly improves the accuracy and execution efficiency of detection. Specifically, deep learning models use multi-layer neural network structures to abstract feature representations of data layer by layer, ultimately achieving precise identification of unknown or abnormal patterns, which is difficult to match with traditional rule-based or statistical methods.

There is a close relationship and mutual promotion between the security data detection and management method in digital library network based on deep learning and social physics. The combination of the two can provide a more intelligent, efficient, and secure network environment for digital libraries. In specific application scenarios of digital library, deep learning mainly relies on a large amount of data for model training and prediction, while social physics provides in-depth understanding and modeling capabilities for user behavior, network traffic and other data. Combining the two can achieve data fusion and complementarity, improving the accuracy and efficiency of data detection and management. And the introduction of deep learning is particularly crucial for enhancing network security protection capabilities. By integrating anomaly detection, intrusion detection and defense systems, as well as risk assessment and other technical means, a comprehensive and intelligent security protection system can be constructed. The anomaly detection system can monitor network traffic and system behavior in real time, identify and report any activities that deviate from the normal operating mode in a timely manner. And it can effectively curb the spread of potential threats. Intrusion detection and defense systems further enhance this capability, not only detecting intrusion behavior, but also automatically taking defense measures such as blocking malicious traffic, isolating infected devices, etc. In addition, the integration of deep learning makes the risk assessment process more intelligent, enabling dynamic adjustment of risk levels and response strategies based on historical data and real-time intelligence. This ensures that security measures are always synchronized with the latest threat situation. Through the synergistic effect of comprehensive technological means, the security protection capability of the system can be significantly improved, providing a solid guarantee for the stable operation of digital library and the security of data.

For preventing and resisting network security attacks and intrusion behaviors, the network security detection system, as a network security protection system that has both firewall defense capabilities and the ability to resist real-time attacks on the network. This can not only protect the inside of the network, but also strengthen the protection function of the outside of the network,

providing better protection mechanisms for network security and more comprehensive guarantees for the security of the people and the country. More and more researchers are introducing deep learning into security data detection to address issues such as low detection performance, high false alarm rate and long detection time when using traditional machine learning methods for network security detection. In deep learning, Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) have more parameters and stronger representation capabilities in feature extraction and their performance is superior to traditional machine learning methods [9]. By using different neural networks to establish different network structures and optimizing feature extraction, better classification prediction models can be obtained. By setting different neural networks with different network structures and continuously optimizing feature extraction, better training models can be provided for classification prediction. As a result, deep learning-based security data detection methods are constantly improving and have become an important guarantee for the network security of current and future digital library network.

Deep learning is played an increasingly important role in various fields [10]. Traditional deep learning methods are widely applied in the field of security data detection, but these methods are not suitable for digital library network security data detection in professional fields and the classification results obtained are not satisfactory. Therefore, deep learning is applied to the field of security data detection in digital library network for identification and management. Deep learning can extract nonlinear features from a large amount of chaotic and disordered data to form more abstract high-level data representations and then further learn to obtain better results. Therefore, applying deep learning to the field of security data detection for managing digital library network is an effective and promising method, as well as an inevitable trend. Our main contributions are summarized as follows.

(1) We propose a digital library network security data detection method and management based on BiGRU-TCNDetector to address the issues of low efficiency, low detection accuracy and inability to detect specific types of attack threats in current deep learning-based security data detection methods. Firstly, TCN is used for preliminary feature extraction of digital library network security data. Then, the output data of the convolutional layer is used as input for BiGRU to capture long-term dependencies between sequences and preserve the correlation between features of digital library network security data.
(2) The digital library network security data detection and management method based on BiGRU-TCNDetector adds attention mechanism to BiGRU-TCN, reducing the lack of key information in security data and further improving the accuracy of model detection. Finally, the data is fed to the softmax classifier for classification through a global average pooling layer, reducing the computation of network parameters and thus shortening the detection time.
(3) By conducting multiple experimental comparisons on different datasets, the digital library network security data detection method based on BiGRU-TCNDetector is validated compared to other baselines, demonstrating good performance in detecting and managing security data in digital library network scenarios.

The rest of this article consists of four parts. Section II is related literature related to the work. Section III provides a detailed introduction to the digital library network security data detection method based on BiGRU-TCNDetector driven by deep learning. Section IV analyzes the comparative effect of the digital library network security data detection method based on BiGRU-TCNDetector through experiments and metrics based on multiple datasets and baselines. Finally, Section V is the summary.

2 Literature review

Considering the large and complex amount of data present in the real digital library network environment, traditional machine learning methods for training security data detection models were insufficient to address this challenge. Therefore, deep learning technology had gradually been favored in the field of security data detection. Du et al. [11] proposed a deep long short-term memory network model (LSTM) of DeepLog, whose core lied in the application of LSTM, aiming to transform complex system logs into processable natural language sequence forms. The core advantage of this model was its ability to automatically capture and learn inherent pattern features from log data generated during normal system operation. When the system log showed behavior that deviates from these learned normal patterns, DeepLog could effectively identify and mark it as abnormal, thereby achieving early warning of potential system problems. The DeepLog had the ability of online incremental learning, which allowed the model to automatically adjust and optimize its internal parameters with the continuous influx of new log data without the need to retrain the entire dataset. This ensured that the model could keep up with the dynamic changes of the system, adapt and identify new log patterns in a timely manner. Shaikh et al. [12] proposed a deep learning framework based on autoencoders and recurrent neural networks to achieve access control of network traffic. The framework preprocessed the dataset using autoencoders and trained the classification model using LSTM. Experimental results had shown that this method could effectively reduce false alarm rates. Javaid et al. [13] proposed an intrusion detection method based on sparse autoencoder and softmax regression, which used sparse autoencoder to extract features in an unsupervised manner. Secondly, a classifier was constructed using softmax regression algorithm to detect abnormal network traffic. The performance of the proposed method was experimentally demonstrated on the network benchmark dataset NSL-KDD and compared with some previous works, including accuracy, precision, recall and f-measures values. Su et al. [14] proposed a traffic anomaly detection model BAT to address the issues of insufficient accuracy and complex feature engineering in the field of security data detection. This model integrated a bidirectional long short-term memory network (BiLSTM) with attention mechanism, aiming to optimize the extraction and utilization of network traffic features. Specifically, attention mechanism was introduced to finely screen the sequence of data packet vectors processed by BiLSTM, thereby focusing on the key information that constitutes the network flow vector. In addition, the BAT model also integrated multi-layer convolutional neural networks (CNNs), aiming to deeply explore the local features of data, which were crucial for accurately characterizing

network traffic behavior. Subsequently, the extracted features were efficiently classified using a softmax classifier, achieving accurate recognition of network traffic. As an end-to-end solution, BAT could automatically learn and abstract hierarchical key features from raw data, simplifying the model construction process and improving generalization ability. The experimental results showed that BAT achieved an accuracy of 84.25% on NSL-KDD, demonstrating significant advantages in anomaly detection accuracy, efficiency and robustness, improving the level of network security protection.

In recent years, machine learning and deep learning techniques had been widely applied in fields such as object detection, had demonstrated superior performance. Therefore, many researchers had also applied deep learning techniques to security data detection tasks. Such security data detection methods aimed to use deep learning and neural network techniques to learn features such as data flow, control flow and syntax flow in source code. This determined detection boundaries by comparing the differences in syntax and semantic features between security samples and risk samples. Huo et al. [15] proposed a novel convolutional neural network model of NP-CNN, which utilized vocabulary and program structure information from natural language or programming language source code to learn key features that could automatically locate security data, achieving fine-grained detection of security data. Li et al. [16] designed a vulnerability detection architecture using deep program convolutional neural network (DP-CNN) aimed at predicting software defects. The core of this architecture was to extract representation vectors from the abstract syntax tree of the program, and then embed these vectors into the deep learning model for model training. Yuan et al. [17] developed a distributed denial of service (DDoS) attack detection strategy based on recurrent neural networks (RNNs), which focused on extracting feature patterns from network traffic time series to achieve effective tracking of network attack activities. Wang et al. [18] proposed a hierarchical intrusion detection system (HAST-IDS) that integrates convolutional neural networks (CNN) and recurrent neural networks (RNN). The system first used CNN to capture spatial features in the data, and then uses RNN to deeply analyze the time series characteristics of the data. The performance of experimental data on ISCX2012 is particularly impressive, with HAST-IDS achieving an accuracy rate of up to 99.89% and a recall rate of 96.96%, while maintaining an extremely low false alarm rate. Javed et al. [19] combined convolutional neural networks (CNNs) with Attention Enhanced Gated Recurrent Units (GRUs) to detect vehicle intrusion attacks on the Controller Area Network (CAN) bus, demonstrating the effectiveness and potential of this combination in specific security domains. Compared with existing methods, the experimental results of this model improved accuracy, recall and F1 score by 12.69%, 10.91% and 10.79%, respectively. Compared with the most advanced models at that time, this model improved by 5.72%, 4.94% and 5.32% in these three performance indicators, respectively. Sinha et al. [20] proposed a deep learning model that combines one-dimensional CNN and BiLSTM. The model performed binary and multi classification tasks on UNSW-NB15 and achieved binary and multi classification accuracies of 93.10% and 82.08%, respectively. And it could accurately classify most of the attack class samples in UNSW-NB15, with a relatively low FPR, which was better than the most advanced models at that time. Kasongo et al. [21] proposed a method combining wrapper-based feature extraction units and feedforward neural networks (WFEU-FFDNN), which used the WFEU extra tree algorithm to generate the optimal feature vector consisting of 22 attributes on UNSW-NB15. The vector was used as input and the feedforward neural network was used for classification. The experimental results show that the binary and multi classification accuracies of this method were 87.10% and 77.16%, respectively.

3 Deep learning driven security data detection method and management in digital library network

3.1 Overview of BiGRU-TCNDetector architecture

BiGRU-TCNDetector is a digital library network security data detection method based on deep learning fusion model. Due to the advantages of the BiGRU, such as simple structure, fast computation speed and the ability to remember long-term dependency information, TCN can extract key features from serialized information and use pooling operations to reduce the dimensionality of code vector representations, resulting in a fine-grained feature matrix. Therefore, we use BiGRU and TCN to respectively learn the semantic and syntactic information of security data types in digital library network. Secondly, the trained local models are fused to construct a detectable multi type digital library network security data fusion model, which detects whether there are risks and determines the specific types of security data [22].

The deep learning fusion model designed mainly consists of local models, model fusion and global model, as shown in Figure 1. Local models can be divided into left model and right model. The left model is composed of BiGRU, which mainly learns and remembers longer context dependent information in the coarse-grained security data of digital library. The right model is composed of TCN, whose main function is to extract important syntactic features in the fine-grained security data of digital library to assist in determining the specific types of network security.

The function of the model fusion layer is to use the Merge layer to fuse the trained left and right models into a global model. Its principle is to fuse the model parameters from the training outputs of the two local models. The global model consists of an attention mechanism layer and a softmax layer. The purpose of adding attention mechanism is to help the neural network focus its attention on the code elements and their dependent features that can distinguish the types of digital library network security data. By assigning higher weights to key features, it effectively distinguishes different security data with subtle differences, thereby guiding the neural network to train more efficient multi type digital library network security data detection models. The softmax layer is the activation layer, which outputs the classification results of the detection model to achieve secure management of digital library network data.

For solving the problem of unclear and difficult to accurately classify security data features of different types of digital library network, we draw on the idea of attention mechanism to solve the problems of complex text semantic features and unclear differences

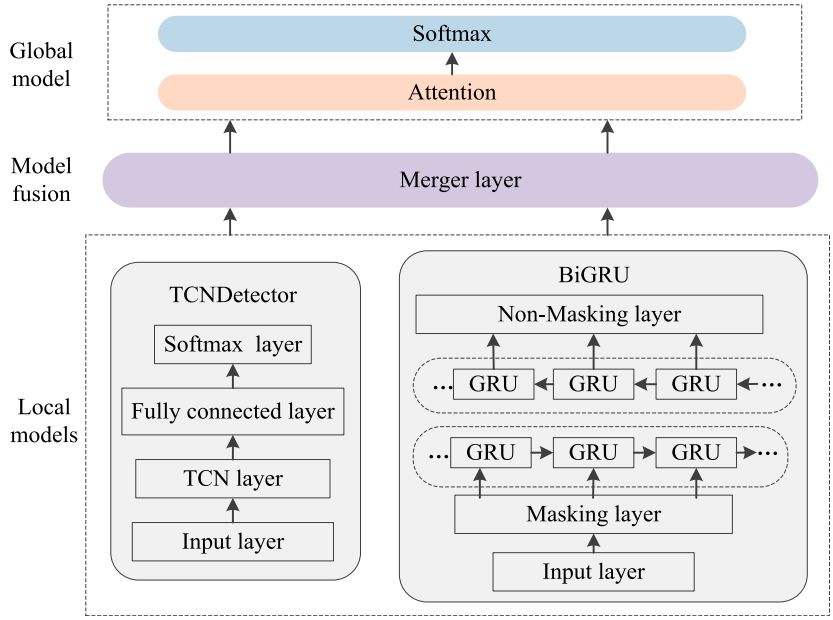

FIGURE 1
BiGRU-TCNDetector architecture.

in various features. In the global model training process, attention mechanism is introduced to enable the neural network to focus more attention on key code features and their dependent variables.

The security data detection of digital library network is essentially a multi classification problem, aimed at detecting whether there are network security attacks in the input and which specific type of security attack they are. Therefore, we choose the Cross Entropy Loss Function as the loss function [23], which is defined as shown in Equation 1.

$$Loss = \frac{1}{N}\sum_{i} L_i = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log p_{ic} \tag{1}$$

where N represents the number of samples, c is the number of categories, y_{ic} represents the sample label, when sample i belongs to category c, $y_{ic} = 1$, otherwise $y_{ic} = 0$. The p_{ic} indicates the probability that the sample i belongs to category c. In order to prevent model overfitting, we add L2 regularization weight decay term after Equation 1 to ensure that the weight vector of the model is at a small value and avoid gradient explosion, the improved loss function is shown in Equation 2.

$$Loss = \frac{1}{N}\sum_{i} L_i = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic} log p_{ic} + \frac{\lambda}{2}||W||^2 \tag{2}$$

where λ is the penalty factor and $||W||^2$ is the weight norm. Learning rate decay (lrDecay) is a factual technique for training modern neural networks. The working principle of lrDecay is to maintain the learning rate at a relatively large value during the initial training phase, which helps accelerate model training or helps the network escape pseudo local minima. During the training process, as the number of iterations accumulates, the learning rate gradually decreases. This is aimed at guiding the network model to approach a local optimal solution more smoothly, effectively suppressing oscillation phenomena during the training process and promoting stable convergence of the model. Therefore, we update the learning rate in an exponential decay manner and its update strategy is shown in Equation 3.

$$lrDecay = lr * \gamma^{\frac{N}{n}} \tag{3}$$

where lr represents the current learning rate, γ is the decay factor, N represents the total number of iterations and n means that for every n iteration, the learning rate decreases once by multiplying the decay factor.

3.2 Local model based on BiGRU

Although traditional RNNs can remember contextual information of text sequences, their memory length is short and their ability is weak. The original LSTM cannot fully learn the security data features of digital library networks, resulting in low model detection accuracy. Gated recurrent unit (GRU) [24] not only has the ability to extract time series, but also can preserve memory for a long time. Compared to LSTM, GRU has a simpler structure, fewer parameters and better detection performance. In response to the above issues, we adopt BiGRU to learn the long-term dependency relationships between security data in digital library network. Compared with LSTM, it has a simpler structure, fewer parameters, faster computation speed and can remember long-term dependency information between codes. BiGRU is also similar to BiLSTM, including GRU in both forward and reverse directions and GRU in reverse direction, which can better demonstrate the influence of each attribute in the sequence data on its front and back attributes, thereby improving the model's time series prediction ability.

BiGRU effectively captures the complex time series features and contextual dependencies in digital library network security data through its unique bidirectional processing mechanism. Its network processes data from front to back and captures historical information. The other network processes from back to front and predicts future trends. This bidirectional structure enables BiGRU to have a more comprehensive understanding of the temporal characteristics in data, thus performing well in tasks such as anomaly detection and security recognition. The forward GRU and reverse GRU in BiGRU are two independent hidden layers that read information in chronological order and perform forward and reverse GRU calculations respectively. Finally, the determined values of GRU are jointly output. The forward and backward derivation formulas are shown in Equations 4, 5.

$$\begin{cases} \vec{r}_t = \sigma\left(\vec{W}_r \cdot \left[\vec{h}_{t-1}, \vec{x}_t\right]\right) \\ \vec{z}_t = \sigma\left(\vec{W}_z \cdot \left[\vec{h}_{t-1}, \vec{x}_t\right]\right) \\ \tilde{h}_t = tanh\left(\vec{W}_{\tilde{h}} \cdot \left[\vec{r}_t * \vec{h}_{t-1}, \vec{x}_t\right]\right) \\ \vec{h}_t = \left(1 - \vec{z}_t\right) * \vec{h}_{t-1} + \vec{z}_t * \vec{x}_t\right) \end{cases} \tag{4}$$

$$\begin{cases} \overleftarrow{r}_t = \sigma\left(\overleftarrow{W}_r \cdot \left[\overleftarrow{h}_{t-1}, \overleftarrow{x}_t\right]\right) \\ \overleftarrow{z}_t = \sigma\left(\overleftarrow{W}_z \cdot \left[\overleftarrow{h}_{t-1}, \overleftarrow{x}_t\right]\right) \\ \tilde{h}_t = tanh\left(\overleftarrow{W}_{\tilde{h}} \cdot \left[\overleftarrow{r}_t * \overleftarrow{h}_{t-1}, \overleftarrow{x}_t\right]\right) \\ \overleftarrow{h}_t = \left(1 - \overleftarrow{z}_t\right) * \overleftarrow{h}_{t-1} + \overleftarrow{z}_t * \overleftarrow{x}_t\right) \end{cases} \tag{5}$$

The final output is shown in Equation 6.

$$H_t = \left[\vec{h}_t, \overleftarrow{h}_t\right] \tag{6}$$

where $\vec{h}_t$ and $\overleftarrow{h}_t$ are the output results of forward GRU and backward GRU respectively and H is the hidden state of BiGRU at time t. In addition, BiGRU also achieves adaptive control over the flow of security data information in digital library network through built-in update and reset gate mechanisms. These gating mechanisms can dynamically adjust network parameters based on current inputs and historical states, ensuring that the model remains efficient and stable when dealing with complex and changing security data. This ability is particularly important for network environments such as digital library that require continuous monitoring and rapid response, helping to detect and respond to potential security threats in a timely manner.

Although BiGRU can simultaneously obtain data information from both directions before and after the same time, making the detection results closer to the true values, BiGRU has many parameters and high computational complexity. Therefore, a single BiGRU is prone to the problem of long model training time when conducting digital library network security data detection. Therefore, we use BiGRU, Masking layer and Non-Masking layer to build the network structure. BiGRU is mainly used to learn data features, while the Masking layer is mainly used to solve sequence problems of different lengths. In order to control the input data format in the same dimension, it uses 0 padding data. However, filling a large amount of meaningless 0 data will have a certain impact on the learning effect of the neural network. Therefore, in this chapter, the Masking layer is used to mask the invalid 0 padding data. After the model training is completed, the output results are subjected to Non-Masking operation for easy fusion of the subsequent model output results.

3.3 Local model based on TCN

Due to the need for RNN to process information sequentially, which means waiting for the previous data to be processed before continuing to process the next data, the time overhead is relatively high [25]. The mechanism of TCN is designed to not rely on the accumulation of historical information, and each operation remains independent. This feature enables it to process data in parallel, significantly reducing processing time and improving computational efficiency. In addition, due to the backpropagation of TCN and the different time directions of sequences, this avoids the problems of gradient vanishing or exploding that often occur in RNN. Compared to BiGRU, TCN may be more computationally efficient as it avoids loop operations, thereby reducing the complexity of state updates. Therefore, in response to the advantages of TCN, we propose using TCN to detect network security in digital library and design a digital library network security data detection model TCNDetector based on time convolutional networks.

The architecture of TCNDetector is carefully constructed, integrating TCN layer, fully connected layer, and softmax layer. Among them, the fully connected layer is responsible for deep analysis of the preceding features through nonlinear transformation, mining complex correlations between features and effectively curbing the risk of model overfitting by introducing dropout mechanism. The softmax layer serves as a classifier to perform fine classification processing on the vectors passed by the fully connected layer. The TCN layer serves as the input interface, receiving data in vector form and is equipped with 64 filters. This setting ensures that each layer of the neural network has sufficient neurons in the convolution operation to fully capture the temporal characteristics of the data. The *kernelsize* is set to 2, which represents the size of the kernel used in the convolutional layer. The *kernelsize* = 2 means that the input of each layer corresponds to the output of the previous layer at 2 -time steps. The *dilations* parameter is set in the form of a list, where *dilations* = [1,2,4]. The parameter *dilations* represents the interval between the input of this layer and the output of the previous layer. For example, the *dilations* = 1 indicates that the value at time t in this layer is related to the values at time t and t-1 in the previous layer. The *dilations* = 2 indicates that the input at time *t* in this layer depends on the output at time *t* and *t* – 2 in the previous layer and so on. It is worth noting that this parameter is usually set to a multiple of 2. The default activation function for TCN is relu. The fully connected layer randomly activates the neurons input to the TCN layer based on the dropout rate setting to prevent overfitting of the neural network model. In the TCN layer, there is also a dropout rate parameter inside.

We do not use dropout in the TCN layer because we compare two methods and found that setting dropout in the fully connected layer resulted in better detection performance of the neural network model than using dropout in the TCN layer. In the fully connected layer, we set the dropout rate size to 0.5. The softmax layer is used to classify the input vectors of the fully connected layer and its function is to normalize the output results of each category. The detection of security data in digital library network is a binary classification problem. Therefore, the number of categories is set to 2. We set the batch size to 64, the epoch size to 10, the optimizer to Adam,

using its default learning rate of 0.001 and the loss function to binary cross entropy.

In addition, if the ratio between positive and negative samples in the given digital library network security training dataset is different, it makes it difficult for the neural network to learn the main security data features, resulting in a high false alarm rate in detection. Therefore, we use undersampling [26] to process the data. This enables the neural network to learn more data detection rules during the training process.

3.4 Attention mechanism

Attention mechanism is widely used in defect prediction, malware classification, etc. [27]. The computational process of the attention mechanism can be divided into three steps, obtaining the query and key vectors according to the input vector, calculating the correlation α between each query vector and the key vector and extracting important information according to the attention score.

Given n input vectors $I = [x_1, ..., x_n]$, where each vector $x_i (i \in [1, n])$ represents an input information. Firstly, each input vector x_i in I multiply on W_q and W_k to obtain q_i and k_i, respectively. Secondly, the correlation $\alpha_{i,j}$ between x_i and x_j is calculated separately, where the correlation between every two-input information is calculated using the scaled point product, as shown in Equation 7. Each input vector x_i also calculates its own correlation.

$$\alpha_{i,j} = \frac{q_i \cdot k_j}{\sqrt{d_k}} \tag{7}$$

Each attention score calculated in Equation 7 needs to be normalized by softmax to obtain α', as shown in Equation 8.

$$\alpha'_{i,j} = \frac{exp(\alpha_{i,j})}{\sum_{k=1}^{n} \exp(\alpha_{i,k})} \tag{8}$$

By $\alpha'_{i,j}$, we can know the degree of association between the current input vector x_i and each input vector x_j. The value vector is obtained by multiplying each input vector x_i in I separately by W_v, and successively multiplied with the corresponding attention score and then accumulated to obtain the output vector b_i, as shown in Equation 9. The attention mechanism can accelerate the computation using matrix multiplication, where the computation of each output vector b_i is performed in parallel. $Q = W_q \cdot I, K = W_k \cdot I, V = W_v \cdot I, A'$ is calculated using Q and K and multiplied by V to obtain the output vector, as shown in Equation 10.

$$b_i = \sum_{j=1}^{n} \alpha'_{i,j} v_j \tag{9}$$

$$Attention(Q, K, V) = softmax\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)V \tag{10}$$

The attention mechanism is integrated into the BiGRU-TCNDetector, enabling the model to learn and focus on important parts of digital library network security data. In the security data of digital library network, there are often subtle but crucial abnormal signals or patterns hidden. The attention mechanism dynamically adjusts the weights of different data segments, allowing the model to prioritize processing these key informations. This ability is crucial for timely detection of potential security threats such as network attacks and data breaches. And the attention mechanism also endows the BiGRU-TCNDetector with a certain degree of interpretability. By visualizing the distribution of attention weights, it is possible to intuitively understand the key data points that the model relies on when making decisions. This not only helps to validate the rationality and effectiveness of the model, but also further optimizes the model.

4 Analysis of experimental results

4.1 Performance testing indicators

The experimental environment configuration is Core i7-12700KF, NVIDIA RTX3090 and 64 GB RAM, built on the Windows 10 operating system, with 32.00 GB of memory. We use Python as the main programming language and are based on the TensorFlow framework. TensorFlow is a deep learning framework based on Python language, which can transform abstract data information into artificial neural networks for learning, training and processing. The experimental evaluation utilizes four standard digital network security detection datasets, namely VulDeePecker [28], CIC-IDS2017 [29], UNSW-NB15 [30] and DARPA2000 [31].

For evaluating the effectiveness of the digital library network security data detection method based on BiGRU-TCNDetector, we conduct experiments on public datasets. VulDeePecker can restore the experimental results to the maximum extent possible to avoid errors caused by differences in datasets. This dataset contains 43,913 code widgets without vulnerabilities and 17,725 code widgets with vulnerabilities. CIC-IDS2017 covers some existing attack types or network security feature sets and data sources, it is a new publicly available dataset that meets real-world standards. This dataset spans across 8 files recording information for 5 days, containing 80 features and 2830743 instances, as well as richer types of attacks. CIC-IDS2017 includes various types of attacks mixed with conventional network data, making it a valuable resource for evaluating and training security data detection and management models. The original network dataset of UNSW-NB15 is created by the IXIA PerfectStorm tool. The UNSW-NB15 contains a total of 2540044 traffic data, which can be simply divided into two categories, they are secure and non-secure. It includes nine types of attacks and one type of normal traffic. Using the ArgusBro-IDS tool, 12 algorithms are developed to generate a total of 49 features using class labels. DARPA2000 is an offline testing and evaluation dataset for intrusion detection systems released by MIT Lincoln Lab. The selected data is LLDOS 2.0.2-SScenario Two and the dump file is parsed using the Winpcap tool. Specific packet features are extracted from UDP, TCP and ICMP network data types and 500 TCP, UDP and ICMP packets with a total of 27 types of features are selected as sample sets.

Due to the complex and ever-changing network environment, data differences can be tens or even hundreds of times, which is very detrimental to the training of neural networks. Therefore, the raw data needs to be normalized before neural network training to accelerate convergence speed during program execution. The main purpose of normalizing the extracted data items is to ensure that the

TABLE 1 Test results.

Dataset	Model	Accuracy (%)	Recall (%)	Precision (%)	F1 score (%)
VulDeePecker	TCN	0.814	0.832	0.823	0.856
	BiGRU	0.839	0.867	0.842	0.823
	CNN-LSTM	0.901	0.902	0.899	0.919
	GCNN-LSTM	0.887	0.902	0.884	0.891
	BiGRU-TCN	0.904	0.928	0.919	0.923
	BiGRU-TCNDetector	0.968	0.970	0.959	0.967
CIC-IDS2017	TCN	0.742	0.763	0.715	0.751
	BiGRU	0.877	0.860	0.874	0.869
	CNN-LSTM	0.781	0.817	0.812	0.806
	GCNN-LSTM	0.826	0.852	0.855	0.839
	BiGRU-TCN	0.924	0.955	0.942	0.943
	BiGRU-TCNDetector	0.971	0.982	0.969	0.974
UNSW-NB15	TCN	0.824	0.862	0.881	0.869
	BiGRU	0.828	0.877	0.898	0.882
	CNN-LSTM	0.918	0.939	0.884	0.892
	GCNN-LSTM	0.9405	0.931	0.923	0.955
	BiGRU-TCN	0.958	0.960	0.959	0.945
	BiGRU-TCNDetector	0.973	0.964	0.972	0.970
DARPA2000	TCN	0.825	0.842	0.864	0.849
	BiGRU	0.872	0.892	0.883	0.862
	CNN-LSTM	0.902	0.926	0.879	0.902
	GCNN-LSTM	0.935	0.952	0.950	0.955
	BiGRU-TCN	0.948	0.949	0.952	0.945
	BiGRU-TCNDetector	0.969	0.973	0.977	0.976

differences in changes between the data are not too large and it is best to ensure that each data item takes a value within the range of [0,1]. We adopt the normalization method mentioned earlier to preprocess the normalized data of the dataset. In addition, the experiment is evaluated based on accuracy, precision, recall and F1 score [32]. We evaluate and analyze the detection performance of models based on the detection accuracy, recall rate, F1 score, and overall accuracy of various categories of security data. F1 score is an evaluation index that comprehensively judges precision and recall. When there is a discrepancy between the accuracy and recall of the detection model, F1 score is used to weight and average the values of the two evaluation indicators to obtain a fair result.

4.2 Evaluation of security data detection results in digital network based on deep learning

Based on the digital network security detection dataset, we evaluate and selecte multiple baselines for traffic security detection comparison experiments, including TCN [33], BiGRU [34], CNN-LSTM [35], GCNN-LSTM [36] and BiGRU-TCN [37]. The overall experimental results of all methods are detailed in Table 1.

According to Table 1, the digital library network security data detection method based on BiGRU-TCNDetector performs the best in accuracy, precision, recall and F1 score. And compared with

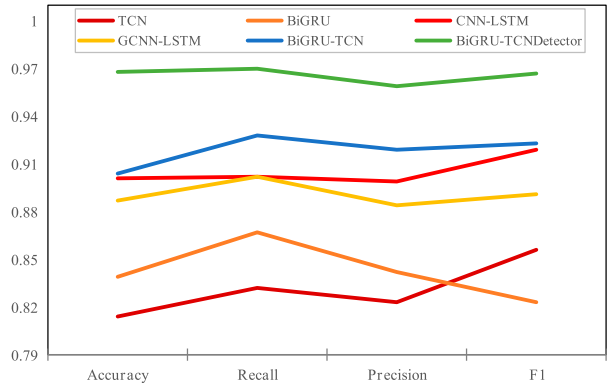

FIGURE 2
Experimental comparison based on VulDeePecker.

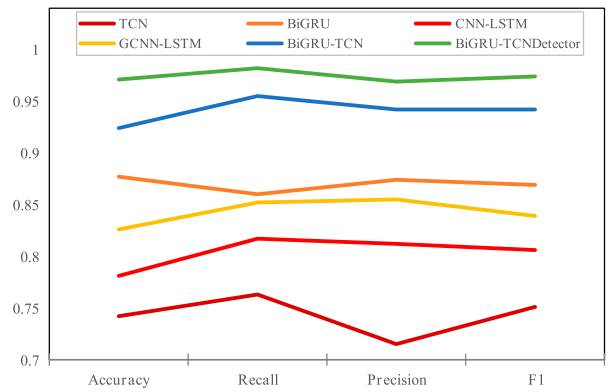

FIGURE 3
Experimental comparison based on CIC-IDS2017.

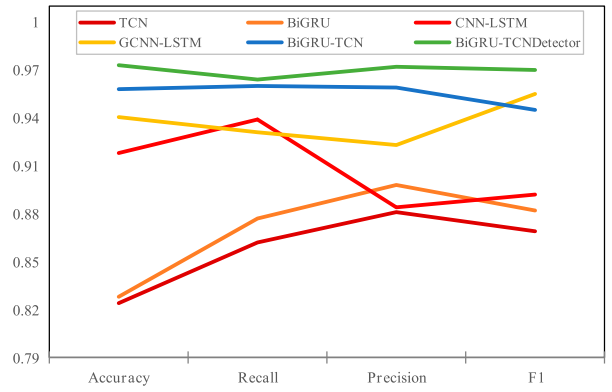

FIGURE 4
Experimental comparison based on the UNSW-NB15.

the classic TCN and BiGRU, the digital library network security data detection method based on BiGRU-TCNDetector achieves significant improvements in accuracy, precision and recall. This is because the gate control unit inside BiGRU can effectively learn

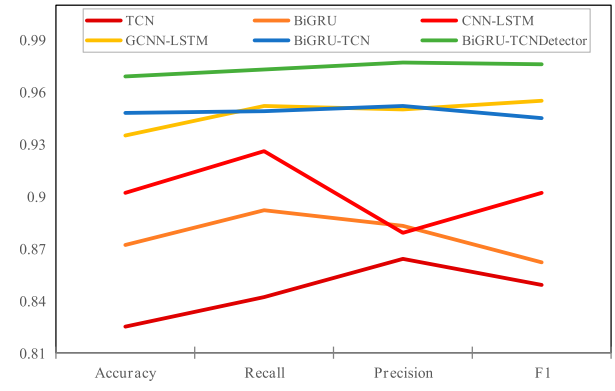

FIGURE 5
Experimental comparison based on DARPA2000.

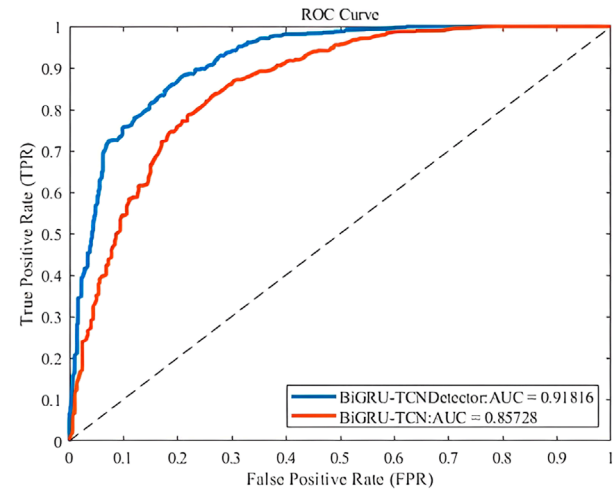

FIGURE 6
Comparison of AUC between BiGRU-TCN and BiGRU-TCNDetector.

long context dependent information between codes, proving that BiGRU-TCNDetector improves network security data detection performance compared to a single network model. It can be seen that TCN has the worst performance in network security data detection, which is due to the increased time overhead caused by too many neural network layers and the possibility of unsatisfactory detection results due to overfitting of the neural network model. Increasing LSTM has better detection performance compared to single BiGRU and TCN. The BiGRU-TCN shows significantly better detection performance than other neural networks. This result indicates that using bidirectional neural network architecture in network security data detection can more effectively capture data features compared to unidirectional neural network, thereby improving detection efficiency. On CIC-IDS2017, the digital library network security data detection method based on BiGRU-TCNDetector achieves the highest recall rate of 0.982, with an F1 score of 0.974, outperforming all baseline models, including BiGRU-TCN and GCNN-LSTM. This confirms the effectiveness of the digital library network security data detection method based on BiGRU-TCNDetector in accurately

TABLE 2 Inference time results.

Model	VulDeePecker	CIC-IDS2017	UNSW-NB15	DARPA2000	Average
TCN	0.95	12.45	7.83	6.823	7.26
BiGRU	0.83	11.83	6.86	5.842	6.24
BiGRU-TCNDetector	0.56	8.96	4.97	3.959	4.61

capturing and identifying digital library network security threats, providing strong technical support for enhancing the network security protection capabilities of digital library. Figures 2–5 are comparison charts of experimental results on different datasets, where the X-axis represents experimental metrics and the Y-axis represents the corresponding accuracy, precision, recall and F1 score of different neural network models.

In Figure 2, compared to the baseline, the digital library network security data detection method based on BiGRU-TCNDetector ranks first with the highest recall rate of 97%. Among other deep models, BiGRU-TCN has the highest recall rate at 92.8%, followed by CNN-LSTM with an F1 score of 91.9%. Recall rate is specifically mentioned as one of the key indicators for evaluating model performance. A high recall rate means that the model can identify and capture more real threats or abnormal behaviors. Therefore, the digital library network security data detection method based on BiGRU-TCNDetector has extremely high performance in identifying or detecting network security events, which can effectively ensure the network security of key information systems such as digital library.

Based on the same conditions mentioned above, the model is further tested and the detailed comparison of the experimental results on CICIDS2017 is shown in Figure 3. CNN-LSTM and GCNN-LSTM perform worse than BiGRU in terms of accuracy, precision, recall and F1 score. Therefore, adding an attention layer to the model may not necessarily improve its overall detection performance and is also influenced by the training dataset itself. The attention mechanism itself reallocates the weights of the original features to select key features that affect the classification results. For datasets with uniform feature importance, adding attention mechanism may focus the model's attention on local features, resulting in the loss of some key features and a decrease in the model's detection performance. On CIC-IDS2017, the test results of the digital library network security data detection method based on BiGRU-TCNDetector are higher than those of the peer model. Overall, the experiment verifies that the digital library network security data detection method based on BiGRU-TCNDetector can more effectively extract advanced features for anomaly detection from network security data, thereby improving the detection effect.

According to Figure 4, compared with TCN, BiGRU, CNN-LSTM, GCNN-LSTM and BiGRU-TCN, the digital library network security data detection method based on BiGRU-TCNDetector has a higher F1 score index of 11.62%, 9.98%, 8.74%, 1.57% and 2.65%, respectively. The experimental results verify that the proposed method has strong detection performance on various network security datasets. The digital library network security data detection method based on BiGRU-TCNDetector introduces attention mechanism and uses BiGRU and TCN to train a deep learning fusion model, which can fully explore the differential features of security data detection.

From Figure 5, it can be seen that the digital library network security data detection method based on BiGRU-TCNDetector performs the best in all indicators on DARPA2000, indicating that the proposed method exhibits high detection accuracy in multi type digital network security detection. In addition, we also introduce AUC to evaluate the value of the model, which is defined as the area under ROC. ROC is a curve plotted with false positive rate (FPR) as the x-axis and true rate (TPR) as the y-axis. The closer the AUC distance is to 1.0, the better the performance of the detection algorithm. The experimental comparison results based on VulDeePecker, BiGRU-TCN and BiGRU-TCNDetector are shown in Figure 6.

From Figure 6, BiGRU-TCNDetector occupies a higher position on the ROC curve, especially in the low FPR region, indicating that the model can more effectively detect real network security threats while maintaining a low false alarm rate. The digital library network security data detection method based on BiGRU-TCNDetector first learns rich features through local sub models. Secondly, adding an attention mechanism layer during model fusion preserves more critical data network security features, resulting in better detection performance of the trained detection model.

In addition, to further evaluate the real-time performance of models in security data detection in digital library network, we introduce inference time as a measure to test whether the model can process this data quickly and efficiently, as shown in Table 2. Experiments are conducted on four datasets, comparing TCN, BiGRU and BiGRU-TCNDetector, calculating the inference time required for prediction. By calculating the inference time of each model on different datasets, we obtain the average inference time, with TCN, BiGRU and BiGRU-TCNDetector being 7.26 s, 6.24 s and 4.61 s, respectively. This indicates that by introducing attention mechanism, BiGRU-TCNDetector effectively reduces computational complexity, demonstrating excellent value and efficiency in resource consumption, strengthening its applicability in the field of digital library network security data detection.

5 Conclusion

Digital library has increasingly highlighted their network security issues. Therefore, we propose the digital library network security data detection method based on BiGRU-TCNDetector, which utilizes the powerful capabilities of deep learning algorithms in complex data security recognition, anomaly detection and

intelligent analysis. It combines TCN and BiGRU to form BiGRU-TCN, inheriting the advantages of time convolutional networks and effectively reducing the time overhead required for training neural network models. Attention mechanism is also added to BiGRU-TCN, which connects the dependency relationships between sequences through weights, reducing the loss of critical information for secure data in the digital library network and further improving the accuracy of model detection. At the end, global average pooling is used instead of the traditional fully connected layer to deliver data to the softmax classifier. This reduces the risk of overfitting caused by the large number of parameters in the fully connected layer, further reducing computational complexity and shortening model training time. Through the analysis of experimental and comparative results, it is concluded that the digital library network security data detection method based on BiGRU-TCNDetector has high accuracy, precision, recall and F1 score. It outperforms other baseline models in network security detection and its overall testing performance is superior to the comparative method. However, due to the gap between open-source datasets and real digital library network security data environments, future work needs to study the use of more real-world data for model validation and testing to improve its security in real-time security data detection systems.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

DZ: Conceptualization, Data curation, Formal Analysis, Investigation, Methodology, Project administration, Software, Supervision, Writing–original draft, Writing–review and editing. YW: Formal Analysis, Investigation, Methodology, Software, Supervision, Validation, Writing–review and editing. JC: Data curation, Project administration, Software, Supervision, Validation, Writing–review and editing. JW: Data curation, Formal Analysis, Investigation, Project administration, Software, Validation, Visualization, Writing–review and editing. ZC: Conceptualization, Data curation, Formal Analysis, Investigation, Methodology, Project administration, Supervision, Visualization, Writing–original draft, Writing–review and editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Aslan Ö, Aktuğ SS, Ozkan-Okay M, Yilmaz AA, Akin E. A comprehensive review of cyber security vulnerabilities, threats, attacks and solutions. *Electronics* (2023) 12(6):1333. doi:10.3390/electronics12061333

2. Dunn Cavelty M, Wenger A. Cyber security meets security politics: complex technology, fragmented politics and networked science. *Contemp Security Pol* (2020) 41(1):5–32. doi:10.1080/13523260.2019.1678855

3. Miao J, Wang Z, Ning X, Xiao N, Cai W, Liu R. Practical and secure multifactor authentication protocol for autonomous vehicles in 5G. *Softw Pract Experience* (2022) 54:1852–69. doi:10.1002/spe.3087

4. Choi KS, Lee CS, Merizalde J. *Spreading viruses and malicious codes[M]//Handbook on Crime and Technology*. Edward Elgar Publishing (2023) 232–50.

5. Miao J, Wang Z, Wang M, Garg S, Hossain MS, Rodrigues JJ. Secure and efficient communication approaches for Industry 5.0 in edge computing. *Computer Networks* (2024) 242:110244. doi:10.1016/j.comnet.2024.110244

6. Zhang Y, Feng M, Shang K, Ran Y, Wang CJ. Peeking strategy for online news diffusion prediction via machine learning. *Physica A: Stat Mech its Appl* (2022) 598:127357. doi:10.1016/j.physa.2022.127357

7. Xiao N, Wang Z, Sun X, Miao J. A novel blockchain-based digital forensics framework for preserving evidence and enabling investigation in industrial Internet of Things. *Alexandria Eng J* (2024) 86:631–43. doi:10.1016/j.aej.2023.12.021

8. Torfi T, Shirvani RA, Keneshloo Y, Tavaf N, Fox EA. Natural language processing advancements by deep learning: a survey. arxiv preprint arxiv:2003.01200, (2020).

9. Banerjee I, Ling Y, Chen MC, Hasan SA, Langlotz CP, Moradzadeh N, et al. Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification. *Artif intelligence Med* (2019) 97:79–88. doi:10.1016/j.artmed.2018.11.004

10. Pramod A, Naicker HS, Tyagi AK. Machine learning and deep learning: open issues and future research directions for the next 10 years. In: *Computational analysis and deep learning for medical care: principles, methods and applications* (2021) p. 463–90.

11. Du M, Li F, Zheng G, Srikumar VL. Deeplog: anomaly detection and diagnosis from system logs through deep learning. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security (2017) p. 1285–98.

12. Shaikh RA, Shashikala SV. An Autoencoder and LSTM based Intrusion Detection approach against Denial of service attacks. In: 2019 1st international conference on advances in information technology (ICAIT). IEEE (2019) p. 406–10.

13. Javaid A, Niyaz Q, Sun W, Alam M. A deep learning approach for network intrusion detection system. In: Proceedings of the 9th EAI international conference on bio-inspired information and communications Technologies (formerly BIONETICS) (2016) p. 21–6.

14. Su T, Sun H, Zhu J, Wang S, Li Y. BAT: deep learning methods on network intrusion detection using NSL-KDD dataset. *IEEE Access* (2020) 8:29575–85. doi:10.1109/access.2020.2972627

15. Huo X, Li M, Zhou ZH. Learning unified features from natural and programming languages for locating buggy source code. *IJCAI* (2016) 16:1606–12.

16. Li J, He P, Zhu J, Lyu MR. Software defect prediction via convolutional neural network[C]. In: 2017 IEEE international conference on software quality, reliability and security (QRS). IEEE (2017) p. 318–28.

17. Yuan X, Li C, Li X. DeepDefense: identifying DDoS attack via deep learning. In: 2017 IEEE international conference on smart computing (SMARTCOMP). IEEE (2017) p. 1–8.

18. Wang W, Sheng Y, Wang J, Zeng X, Ye X, Huang Y, et al. HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection. *IEEE access* (2017) 6:1792–806. doi:10.1109/access.2017.2780250

19. Javed AR, Ur Rehman S, Khan MU, Alazab M. CANintelliIDS: detecting in-vehicle intrusion attacks on a controller area network using CNN and attention-based GRU. *IEEE Trans Netw Sci Eng* (2021) 8(2):1456–66. doi:10.1109/tnse.2021.3059881

20. Sinha J, Manollas M. Efficient deep CNN-BiLSTM model for network intrusion detection. In: Proceedings of the 2020 3rd international conference on artificial intelligence and pattern recognition (2020) p. 223–31.

21. Kasongo SM, Sun Y. A deep learning method with wrapper based feature extraction for wireless intrusion detection system. *Comput and Security* (2020) 92:101752. doi:10.1016/j.cose.2020.101752

22. Yu D, Zhou Y, Zhang S, Li W, Small M, Shang K. Information cascade prediction of complex networks based on physics-informed graph convolutional network. *New J Phys* (2024) 26(1):013031. doi:10.1088/1367-2630/ad1b29

23. Mao A, Mohri M, Zhong Y. Cross-entropy loss functions: theoretical analysis and applications. In: International conference on Machine learning. PMLR (2023) p. 23803–28.

24. Dey R, Salem FM. Gate-variants of gated recurrent unit (GRU) neural networks. In: 2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS). IEEE (2017) p. 1597–600.

25. Sherstinsky A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena* (2020) 404:132306. doi:10.1016/j.physd.2019.132306

26. Shelke MS, Deshmukh PR, Shandilya VK. A review on imbalanced data handling using undersampling and oversampling technique. *Int J Recent Trends Eng Res* (2017) 3(4):444–9.

27. Shen G, Chen Z, Wang H, Chen H, Wang S. Feature fusion-based malicious code detection with dual attention mechanism and BiLSTM. *Comput and Security* (2022) 119:102761. doi:10.1016/j.cose.2022.102761

28. Zou D, Wang S, Xu S, Li Z, Jin H. μ μ VulDeePecker: a deep learning-based system for multiclass vulnerability detection. *IEEE Trans Dependable Secure Comput* (2019) 18(5):2224–36. doi:10.1109/TDSC.2019.2942930

29. Rosay A, Cheval E, Carlier F, Leroux P. Network intrusion detection: a comprehensive analysis of CIC-IDS2017. In: 8th international conference on information systems security and privacy. SCITEPRESS-Science and Technology Publications (2022) p. 25–36.

30. Meftah S, Rachidi T, Assem N. Network based intrusion detection using the UNSW-NB15 dataset. *Int J Comput Digital Syst* (2019) 8(5):478–87.

31. Nwamuo O, de Faria Quinan PM, Traore I, Woungang I, Aldribi A. Arguments against using the 1998 DARPA dataset for cloud IDS design and evaluation and some alternative. In: Machine learning for networking: second IFIP TC 6 international conference, MLN 2019. Paris, France. Springer International Publishing (2020) p. 315–32.

32. Reddy BH, Karthikeyan PR. Classification of fire and smoke images using decision tree algorithm in comparison with logistic regression to measure accuracy, precision, recall, F-score. In: 2022 14th International Conference on Mathematics, Actuarial Science Computer science and statistics (MACS). IEEE (2022) p. 1–5.

33. Hewage P, Behera A, Trovati M, Pereira E, Ghahremani M, Palmieri F, et al. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Comput* (2020) 24:16453–82. doi:10.1007/s00500-020-04954-0

34. Lin X, Quan Z, Wang ZJ, Huang H, Zeng X. A novel molecular representation with BiGRU neural networks for learning atom. *Brief Bioinformatics* (2020) 21(6):2099–111. doi:10.1093/bib/bbz125

35. Wang J, Yu LC, Lai KR, Zhang X. Dimensional sentiment analysis using a regional CNN-LSTM model. In: Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: short papers) (2016) p. 225–30.

36. Wu Y, Zhu X, Huang Q, Zhang Y, Evans J, He S. Predicting the quality of tangerines using the GCNN-LSTM-AT network based on vis–NIR spectroscopy. *Appl Sci* (2023) 13(14):8221. doi:10.3390/app13148221

37. Li B, Li J, Wu X, Tang H. *A tcn-bigru network with soft thresholding and attention mechanism for the tool wear prediction* (2024).