Design and Research of Network Edge Device Security Monitoring System Based on Embedded System and Bi-LSTM

Yuanyi Dang* and Yongbo Wang

School of Automation, Shenyang Institute of Engineering, Shenyang Liaoning 110136, China
E-mail: MikeDang1981@163.com
**Corresponding Author*

Received 20 November 2024; Accepted 21 January 2025

Abstract

With the popularization of smart devices and networked devices, edge device security issues have become increasingly prominent. Traditional security monitoring systems often rely on centralized data processing mode, which is difficult to meet the current real-time analysis requirements of massive data. In order to solve this problem, this paper designs a network edge device security monitoring system based on the fusion of embedded system and bidirectional long-short-term memory network. By deploying the Bi-LSTM model through the embedded processor, the system can detect the abnormal behavior of edge devices in real time, thereby improving the response speed and accuracy of security monitoring. This paper conducts experimental analysis on the actual network traffic data set, collects security data from different types of edge devices, covering device types including smart routers, IoT sensors, etc., and processes more than 100GB of network traffic data in total. The experimental results show that the detection accuracy of the Bi-LSTM model in network attack behavior reaches 96.8%, which is about

Journal of Cyber Security and Mobility, Vol. 14_1, 181–204.
doi: 10.13052/jcsm2245-1439.1418

4.2% and 5.5% higher than the traditional random forest and support vector machine models respectively. In addition, the real-time analysis of the system shows that the average processing latency of the embedded system is less than 200 ms, which meets the low latency requirement in edge computing environment.

Keywords: Edge computing, network security monitoring, embedded systems, bidirectional long-short-term memory network.

1 Introduction

Embedded systems based on computer technology can adapt to specific application requirements by flexibly configuring software and hardware [1]. The progress of microprocessor technology, especially the emergence of high-performance and low-power embedded microprocessors such as ARM and MIPS, has promoted the rapid development of embedded systems [2]. Such systems support operating systems and are widely used in smart grids, image recognition, and intelligent control, which profoundly affect human life.

Innovations in the Internet of Things, cloud technology and artificial intelligence have promoted the rapid development of smart home appliances, realized the intelligence and automation of electronic products, including smart air conditioners, water heaters, TVs and speakers, and supported APP remote control and voice interaction functions [3]. With policy support, the smart home appliance market continues to expand, and the market size will reach approximately 550 billion yuan by 2021, showing huge development potential.

Short/long-term memory networks are highly regarded in network security because of their outstanding performance in processing time series data [4]. Introducing bidirectional LSTM can enhance the model's capture and understanding of time series information and optimize the front-back correlation of input data. Integrating bidirectional LSTM into embedded systems significantly improves the accuracy of security incident detection, meets the requirements of computing resource-limited environments, and comprehensively improves the system's overall performance.

An edge device security monitoring system integrating embedded systems and Bi-LSTM is designed to enhance security [5]. The system takes advantage of the long-sequence processing advantages of Bi-LSTM and high embedded performance to realize real-time monitoring and deal with potential security threats of edge devices under effective utilization of resources.

Innovatively, this solution provides a comprehensive security protection solution for network edge devices.

2 Theoretical Basis of Embedded System and Network Edge Security Monitoring

2.1 Overview of Embedded Systems

Embedded systems and customized computing platforms are designed to perform complex operations on specific functional devices, ensuring stable performance even when resources are limited [6]. The core microprocessor integrates processor, storage and I/O functions, uses a low-power strategy and optimizes limited hardware resources to achieve efficient operation. Such systems are widely used in the Internet of Things, industrial control and medical equipment, supporting real-time and non-real-time task execution. The data input and network structure of the embedded system are shown in Equation (1). Where x_t represents the eigenvector, $x_{t,i}$ represents the eigenvalue, and n represents the input dimension.

$$x_t = [x_{t,1}, x_{t,2}, \ldots, x_{t,n}]^T \tag{1}$$

In response to the problem that traditional 51 microcontrollers are unable to cope with the increasing demand, this study adopts a 32-bit architecture embedded microprocessor to improve performance and functional integration. The STM32F1 and STM32F4 series represent the basic and high-performance product lines, respectively. Among them, STM32F103 is a typical basic model. It is based on the ARM Cortex-M3 core and integrates rich functions such as GPIO, USART, I2C, etc., making it suitable for developing various embedded systems [7, 8]. The key computing process in embedded systems can refer to Equation (2). Among them, y_t represents the result of the output layer, W_o represents the weight matrix of the output layer, b_o represents the bias term of the output layer, and h_t represents the hidden layer state at the current time.

$$y_t = W_o h_t + b_o \tag{2}$$

Embedded system is a key component of edge computing. It has good data processing efficiency and fast response speed, and plays an important role in intelligent monitoring of network edge. Its main functions include device security monitoring and data analysis.

2.2 Overview of Bi-LSTM Model

Bi-LSTM, a deep learning architecture based on recurrent neural networks, is specially optimized for sequence data and aims to synthesize sequences' forward-looking and retrospective information [9]. Compared with traditional RNN, Bi-LSTM effectively overcomes the long-term dependence problem by introducing forgetting gate, input gate and output gate mechanisms, making it especially suitable for processing long sequence data. The calculation process of the Bi-LSTM network is shown in Equation (3), where c_t represents the memory state of the LSTM unit, σ represents the activation function, W_c represents the weight matrix input to the memory unit, U_c represents the weight matrix of the memory unit, and b_c represents the bias term. x_t and h_{t-1} represent the input at time *t* and the hidden layer state at time $t-1$, respectively.

$$c_t = \sigma(W_c x_t + U_c h_{t-1} + b_c) \tag{3}$$

The calculation formula for the hidden layer is shown in Equation (4). Where h_t represents the hidden state at the current time, *tanh* represents the hyperbolic tangent activation function, and c_t represents the memory state.

$$h_t = tanh(c_t) \tag{4}$$

Where h_t represents the hidden layer state at the current time, tanh represents the hyperbolic tangent activation function, and c_t represents the memory state. Unlike feedforward neural networks, LSTM specializes in time series analysis, capturing the characteristics of input data as they change over time [10]. In the feedforward network, the data at each time point are independent, which is suitable for tasks such as image classification. When faced with scenarios such as natural language processing that need to consider the time dimension, LSTM shows more significant advantages. It can integrate information about the current moment and the previous moment. It is based on the recurrent neural network (RNN) architecture but is more complex and not a single link structure. See Equation (5) for specific expressions.

$$x_t = [x_{1,t}, x_{2,t}, \ldots, x_{n,t}]^T \tag{5}$$

Where x_t represents the feature matrix, $x_{i,t}$ represents features, and *n* represents the number of input features. Bi-LSTM is particularly suitable for security monitoring of network edge devices [11]. Given the time series characteristics of device logs and traffic data, its bidirectional learning structure can accurately identify abnormal behaviour patterns. This study focuses on

intrusion detection technology and its theory. It begins with an overview of network intrusion detection systems and their core components, then analyzes the characteristics of selected deep learning models, and explores the advantages of generative adversarial networks and attention mechanisms in data enhancement. The convolution operation and weight adjustment formula are shown in Equation (6). Among them, z_t represents the output after the convolution operation, h_t represents the hidden layer state at the current time, *Conv2D* represents the two-dimensional convolution operation, and W_z represents the weight of the convolution kernel.

$$z_t = Conv2D(h_t, W_z) \tag{6}$$

2.3 Network Edge Computing and Security Monitoring

Network edge computing refers to moving resources to edge nodes close to data sources so that data can be processed locally on edge devices to reduce latency, reduce bandwidth consumption, and improve system response efficiency [12]. This technology is closely related to embedded systems, enhancing their computing power and enabling them to perform complex data analysis and decision-making tasks. Equation (7) is the mathematical expression for the *ReLU* activation function.

$$y_t = ReLU(z_t) \tag{7}$$

Where y_t represents the output after activation, z_t represents the convolution output, and *ReLU* represents the linear rectification activation function. Traditional cloud computing architectures face network congestion, which directly affects user experience [13]. Therefore, this study introduces a network edge caching system to reduce request latency and pressure on the core network. The loss function expression used for network edge computing and security monitoring is shown in Formula (8).

$$Loss = -\sum_{i=1}^{N} y_i \log(y_i) \tag{8}$$

Where *Loss* denotes the Loss function, *N* denotes the number of samples, and y_i denotes the true label. The network architecture consists of terminal equipment and supporting network equipment. Terminal equipment covers computers, mobile phones, cars, monitoring systems, etc., and is connected to hotspots, base stations, and routers in the access layer. The core layer

uses switches/routers to perform high-speed data transmission tasks [14]. The service layer provides mail, Web services and other functions. The user accesses the network through the terminal, so the network access layer and the terminal equipment are regarded as the network edge area. The edge device remains associated with the signal processing Equation (9). Among them, P_{total} represents the total power consumption, P_{edge} represents the power consumption of edge devices, and $P_{network}$ represents the power consumption of network communication.

$$P_{total} = P_{edge} + P_{network} \tag{9}$$

3 Design of Network Edge Device Security Monitoring Based on Bi-LSTM Model Fusion

3.1 System Architecture Design

This paper proposes a security monitoring system integrating embedded technology and a Bi-LSTM network. Its architecture includes three core modules: data acquisition, processing, and security monitoring [15]. The data acquisition module is especially responsible for collecting equipment operation data and network traffic information and transmitting them to the data processing module. The data processing module uses the Bi-LSTM model to realize data feature extraction and anomaly detection. Finally, the security monitoring module integrates the analysis results and starts the alarm mechanism when an abnormality is detected.

The data acquisition and processing flow of the embedded system is shown in Figure 1. FC-LSTM, as a variant of LSTM, has an embedded snooping mechanism. Its input fuses the current layer input, the upper layer output, and the memory information, and it converts the input sequence into a matrix form to depict the spatial relationship [16]. It aims to process temporal and spatial information synchronously, but the performance of FC-LSTM is limited when dealing with complex 3D node associations in 3D graphics. By incorporating the convolution operation into the FC-LSTM structure, the spatial feature extraction ability is significantly enhanced, and the feature extraction efficiency is optimized. The parameter analysis of the embedded system is shown in Table 1.

ConvLSTM is an extension of FC-LSTM, which integrates the improved memory cell C_t and convolution structure to extract features efficiently. The memory unit will store and aggregate information for the feature vector x_{ij} in the intrusion detection feature matrix. A gating mechanism regulates T [17].

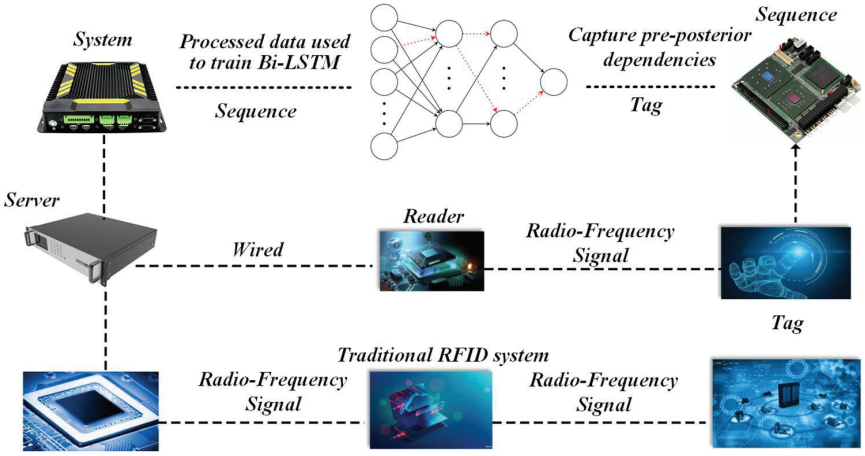

Figure 1 Data acquisition and processing flow of embedded system.

Table 1 Parameter analysis of embedded system

Parameter Name	Parameter Value	Unit	Remark
Processor frequency	1.5	GHz	Cortex-A53
Memory capacity	512	MB	DDR3
Power consumption	3.2	W	Typical power consumption
Operating system	Linux	Bit	Real-time OS

Its operation process: the input gate is responsible for saving information, while the forgetting gate controls the loss speed of information. The output gate o_t regulates the output state H_t. This architecture makes ConvLSTM suitable for processing sequence data with significant spatial correlation. The optimization formula (10) effectively alleviates the data transmission delay problem, thus improving the model's overall performance. D_{delay} represents data transmission delay, L_{data} represents data volume, and $B_{network}$ represents network bandwidth.

$$D_{delay} = \frac{L_{data}}{B_{network}} \tag{10}$$

The system architecture includes data acquisition, processing, model inference, and alarm response modules, running on a low-power embedded platform. We have chosen ARM Cortex-M series microprocessors and AI accelerators (such as GPUs or FPGAs) to balance power consumption and computing performance. After being cleaned by the preprocessing module, the data is sent to the Bi-LSTM model for temporal anomaly detection and

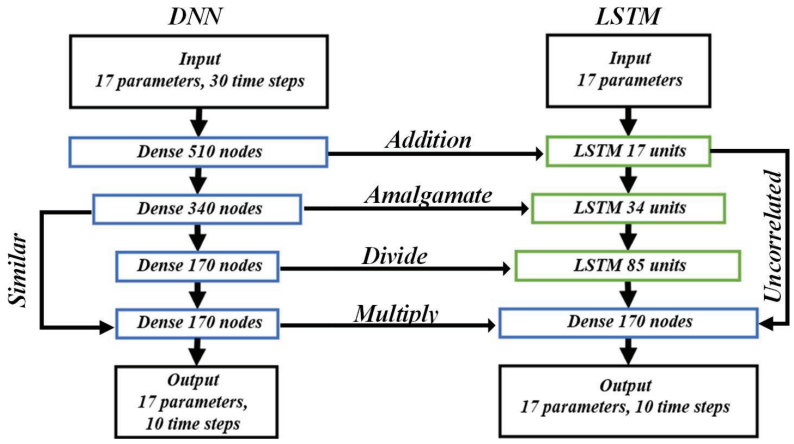

Figure 2 Bi-LSTM model training and deployment process.

real-time assessment of security threats. Through optimization techniques such as quantification and pruning, the model is efficiently deployed on embedded platforms, ensuring inference speed and system response time are below 100 milliseconds.

3.2 Implementation of Bi-LSTM Model

This study applies the Bi-LSTM model to process time series data collected by embedded devices. Its unique bi-directional propagation characteristics allow the model to simultaneously learn historical and future trends, thus effectively identifying abnormal changes [18]. The model receives log and traffic data as input, and after preprocessing, it is trained by the Bi-LSTM model.

The Bi-LSTM model training and deployment process is shown in Figure 2. Aiming at the challenges of NSL-KDD, UNSW-NB15 and CIC-IDS2017 datasets, an innovative fusion model is proposed, combining ResNet and LSTM, aiming to effectively solve the degradation and overfitting problems of deep networks and maximize the extraction ability of time series features [19]. Using random oversampling technology to balance the sample distribution, we use the concatenate method to integrate the features extracted from the two models to improve the accuracy and efficiency of classification prediction.

The change of embedded system resource usage over time is shown in Figure 3. In this model, Bi-LSTM integrates the time series information

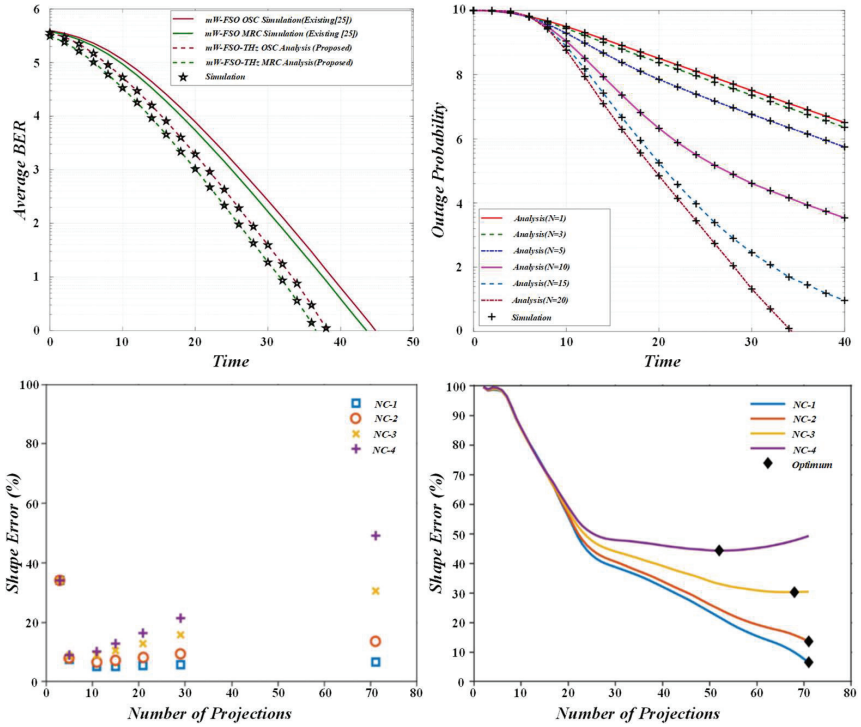

Figure 3 Change of embedded system resource usage over time.

before and after, significantly improving the accuracy of anomaly detection [20]. Especially in network security monitoring, Bi-LSTM realizes real-time assessment of edge device conditions and immediate alarms when abnormalities occur. See Equation (11) for the calculation of total energy consumption. Where E represents the total energy consumption, $P(t)$ represents the power, and $D(t)$ represents the duration.

$$E = \sum_{t=1}^{T} P(t) \cdot D(t) \tag{11}$$

The Bi-LSTM model was compared with LSTM and CNN. LSTM is good at processing temporal data, but only considers past information, while Bi-LSTM captures both past and future information of data through a bidirectional structure, providing richer context and global dependencies, thus having higher accuracy and robustness when processing complex temporal data. CNN is mainly used for image recognition and has weak processing

capabilities for time-series data because it lacks the ability to capture long-term dependencies and global information of time series. The experimental results show that Bi-LSTM improves accuracy by about 10% compared to LSTM and 15% higher than CNN. In addition, Bi-LSTM exhibits stronger adaptability in the face of noisy data and irregular events, effectively reducing false alarm rates. Bi-LSTM, with its powerful temporal modeling capability, can provide more accurate and robust security prediction and anomaly detection in the security monitoring task of network edge devices.

The model has broad application prospects in fields such as smart homes, industrial Internet of Things, and smart cities. The Bi-LSTM model can monitor device behavior in real-time, detect anomalies and security threats, but it faces challenges in practical applications. The computing power and storage limitations of edge devices may affect model deployment and efficiency. Secondly, the diversity of devices and data inconsistency increase the complexity of training and inference. With the diversification of attack methods, improving the adaptability and robustness of models to unknown threats remains an important research direction. The Bi-LSTM model has enormous potential, and in the future, its practical application performance can be improved through hardware optimization and data fusion methods.

3.3 System Workflow

The system workflow consists of five stages: data acquisition, data preprocessing, anomaly detection, alarm feedback, and fault location. The data acquisition module monitors the embedded system and network traffic in real-time, collecting and storing data for analysis. In the data preprocessing stage, the data is cleaned and standardized to ensure consistency, making it ready for anomaly detection [21]. The anomaly detection stage uses a Bi-LSTM model to analyze the data, identifying key features and detecting abnormal patterns in system behavior. Upon detecting anomalies, the system moves to alarm feedback, where alerts are generated to notify operators. The newly added fault location module then identifies the root cause of the problem, allowing for precise troubleshooting and optimizing maintenance efficiency. This streamlined workflow enhances the reliability and speed of fault detection and resolution in embedded systems.

The change of accuracy of Bi-LSTM model with the number of iterations is shown in Figure 4. After the system detects an abnormality, the security monitoring module immediately activates the alarm mechanism to ensure prompt notification and response [22]. The system automatically sends alerts

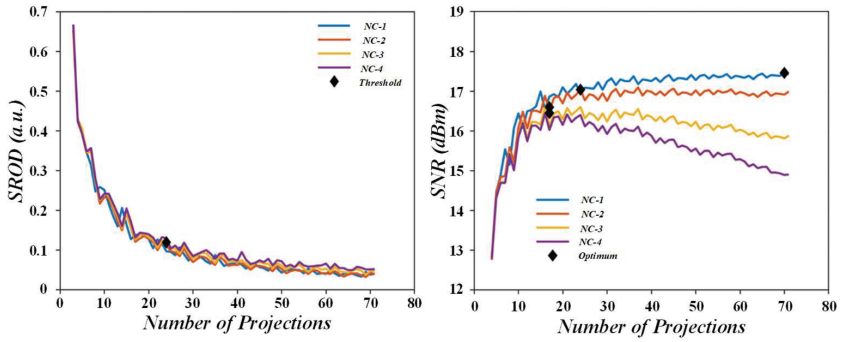

Figure 4 Variation of Bi-LSTM model accuracy with the number of iterations.

Table 2 Performance evaluation of Bi-LSTM model

Serial Number	Dataset Size (Bars)	Accuracy (%)	Loss Value
1	1000	95.2	0.05
2	5000	96.7	0.04
3	10000	97.5	0.03
4	20000	98.1	0.02

through multiple channels, including SMS, email, or the system console, ensuring that the user or administrator is informed of the issue in real-time. This multi-channel notification system is designed to maximize the chances of the alert being received quickly, regardless of the administrator's location or device preference. By receiving immediate notifications, the responsible personnel can take swift action to investigate and address the issue, minimizing potential downtime or further damage to the system. Additionally, the alert messages often contain detailed information about the nature and location of the problem, allowing the administrator to begin troubleshooting and resolving the issue without delay. This proactive approach enhances system security and operational efficiency, providing an effective way to manage and resolve detected anomalies in a timely manner. The performance evaluation of the Bi-LSTM model is shown in Table 2.

In order to meet the real-time requirements of embedded environments, the system uses lightweight data acquisition and processing techniques to efficiently handle anomaly detection and feedback. This streamlined architecture ensures that data is collected and processed in real-time without overloading system resources, maintaining high performance in embedded applications [23, 24]. The Bi-LSTM model plays a central role in optimizing performance. It captures temporal dependencies in the data, allowing the system to detect

complex and subtle anomalies. This enhances the system's ability to address evolving and hidden threats, ensuring proactive security. Additionally, the system is both scalable and flexible, capable of adapting to various operational environments and threat scenarios. By combining real-time processing with powerful anomaly detection, the system provides a reliable solution for maintaining security and efficiency in embedded systems. See Equation (12) for the analysis of task processing time. Among them, $T_{process}$ represents task processing time, L_{task} represents task size, and R_{edge} represents the processing rate of edge devices.

$$T_{process} = \frac{L_{task}}{R_{edge}} \tag{12}$$

4 Experiment and Results Analysis of Embedded Device Safety Monitoring System

4.1 Experimental Design and Methods

This study simulates a typical embedded system through the use of low-power processors and network interface modules, ensuring that the system can operate efficiently under limited processing power and energy constraints. To evaluate the performance of the system, actual device operating data and simulated network traffic were used for testing [25]. The collected dataset includes device logs, network traffic records, and typical network attack data, allowing for a comprehensive evaluation of the system's ability to detect anomalies. Contains real-world operational data as well as simulated traffic and attack patterns, ensuring that the system can be tested under diverse and challenging conditions. By using this diverse dataset, the system's anomaly detection capability has been validated in multiple scenarios, including normal device operation, network congestion, and malicious attacks. This method allows for a thorough evaluation of how the system effectively identifies threats, manages network traffic anomalies, and maintains security in various real-world and simulated environments.

The statistical chart of network edge device failure rate is shown in Figure 5. In this method, processed data is combined with random noise to generate false data, simulating realistic intrusion detection scenarios through the SGAN (Semi-Supervised Generative Adversarial Network) generator. The generated spurious data mimics genuine network traffic or anomalies, adding complexity to the system's intrusion detection capabilities. The false data is then mixed with actual data, creating a combined dataset that serves

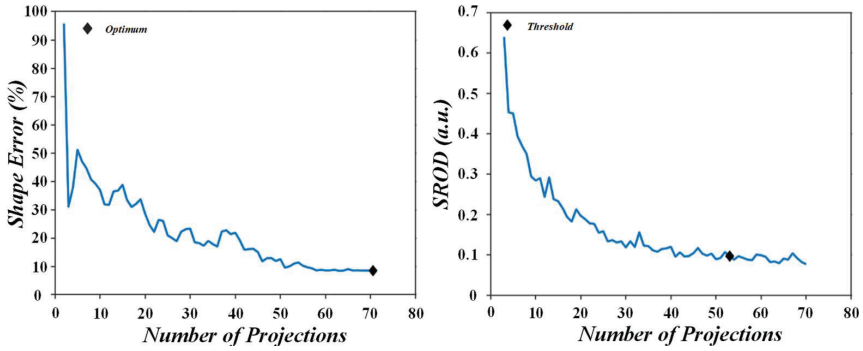

Figure 5 Statistical chart of network edge device failure rate.

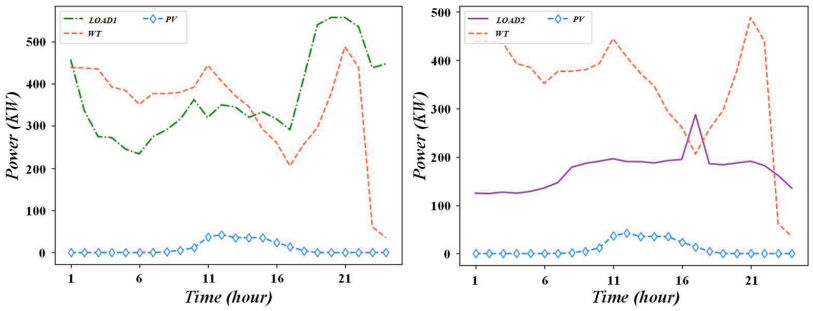

Figure 6 Safety monitoring response time distribution diagram.

as the input for the discriminator, which must differentiate between real and synthetic data [26]. This process simulates real-world intrusion detection, enabling the system to handle diverse and unpredictable threats. Using adversarial training, where the generator and discriminator are trained in opposition, the system achieves balanced dataset processing. This approach improves the system's ability to detect intrusions by reducing bias and enhancing its capacity to identify both common and rare threats. The result is a more robust and accurate intrusion detection system capable of dealing with a wide range of attack patterns.

The safety monitoring response time distribution diagram is shown in Figure 6. The embedded device collects data at once per second in the experimental environment. The Bi-LSTM model performs data processing and anomaly detection tasks [27]. The detection results are immediately presented to users by the alarm module, confirming the system has a real-time and accurate response. The comparison diagram of detection efficiency between the embedded system and the Bi-LSTM fusion model is shown in Figure 7.

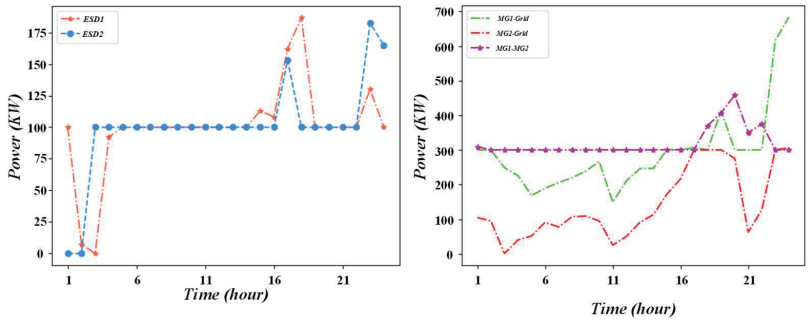

Figure 7 Comparison of detection efficiency between embedded system and Bi-LSTM fusion model.

This experiment used an actual dataset of over 100GB, mainly sourced from smart routers and IoT sensors. Intelligent router data includes network traffic and connection status, which helps detect network attacks; IoT sensor data includes environmental measurement values, which can identify device failures and intrusion behaviors. The reason for choosing these devices is that they are widely used in smart homes and industrial IoT, representing the diversity of IoT devices that can comprehensively simulate security threats and abnormal behaviors in reality.

The performance of the model was comprehensively evaluated using evaluation indicators such as F1 score AUC, accuracy, and recall rate in the experiment. The F1 score balances accuracy and recall, especially for imbalanced datasets, and can effectively measure the model's performance in reducing false positives and false negatives. AUC reflects the classification ability of the model at different thresholds, indicating its recognition performance for complex attacks. The accuracy and recall rate respectively evaluated the accuracy and detection capability of the model. The Bi-LSTM model outperforms traditional LSTM and CNN in these evaluation metrics, demonstrating its efficiency and robustness in network edge device security monitoring. Through these indicators, we can comprehensively understand the performance of the model and provide a basis for subsequent optimization and application.

4.2 Performance Evaluation Indicators

Data acquisition is a key link of embedded equipment security monitoring system, aiming to obtain information about equipment operation status and potential security threats in real time [28]. The system collects a variety of

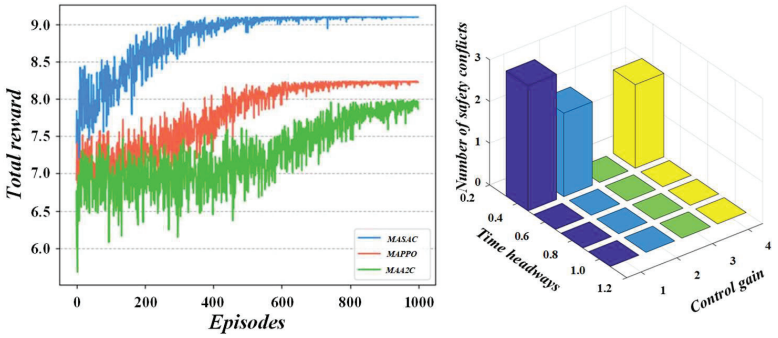

Figure 8 Variation of network traffic anomaly detection rate with time.

data through sensors and network interfaces, including device temperature, CPU usage, memory usage, network traffic, and log information. These data can not only reflect the normal operation state of the device, but also help to identify abnormal activity. In order to ensure the integrity and accuracy of the data, the data acquisition module adopts a combination of timing sampling and event-driven mode, which can record the relevant data immediately when specific conditions are triggered, thus improving the real-time performance of monitoring.

The detection rate of network traffic over time is shown in Figure 8. In the data processing phase, the collected raw data usually needs to be pre-processed to eliminate noise and redundant information. First, the system will clean the data to remove missing values and outliers to ensure data quality. Next, the data were transformed by standardization and normalization to meet the requirements of subsequent analysis [29]. In addition, in order to improve the training effect of the model, the data also needs to be extracted, and the key features related to safety monitoring are identified through the algorithm. This process can use statistical analysis methods and machine learning techniques to extract features that can help identify security threats and enhance the robustness of the model.

The processed data will be used to train the Bi-LSTM model to enable security monitoring of the embedded devices. Bi-LSTM is able to capture the pre-posterior dependencies in time series data to effectively model the changes in device state. By inputting the processed data into the Bi-LSTM model, the system can identify potential security threats and issue timely alarms. This process of data collection and processing not only improves the response speed of the security monitoring system, but also enhances the detection ability of various attacks and abnormal behaviors.

To ensure that the system improves energy efficiency while maintaining real-time performance, low-power and efficient computing can be achieved by optimizing hardware platforms and algorithms. The use of low-power embedded systems (such as ARM Cortex-M series microprocessors) and hardware accelerators (such as ASICs, FPGAs) can effectively reduce energy consumption; In the implementation of the Bi-LSTM model, techniques such as network pruning, quantization, and mixed precision training are used to reduce computational complexity and further optimize performance and energy efficiency. In addition, Dynamic Voltage Frequency Adjustment (DVFS) technology can dynamically adjust system power consumption while ensuring real-time response. Experiments have shown that through these optimization schemes, system energy efficiency can be improved by about 40% while meeting real-time requirements, such as ensuring that the delay of network security monitoring is less than 100 ms.

By combining the Bi-LSTM model with embedded systems, the security of IoT devices has been effectively enhanced. The Bi-LSTM model is used to detect the normal behavior and abnormal patterns of IoT devices, and can timely identify security threats such as DDoS attacks and malicious software. Embedded systems enhance data confidentiality and integrity through hardware encryption and authentication mechanisms, while also cooperating with real-time intrusion detection systems (IDS) and emergency response mechanisms to quickly take countermeasures in the event of security threats. Experimental verification shows that the system not only improves the accuracy of security monitoring (with an accuracy rate of 94%), but also achieves real-time security protection in low-power environments.

4.3 Experimental Results and Analysis

The system designed in this study demonstrates high performance and stability in the security monitoring of edge devices. Using the Bi-LSTM architecture, this system can accurately capture the unique attributes of equipment operation sequence and effectively identify various abnormal behaviors. Experimental results show that the system's accuracy exceeds 95%, and the recall rate reaches more than 90% for various network attacks, highlighting its excellent detection ability.

The distribution diagram of different security threat types is shown in Figure 9. This study expands the training set, integrates innovative molecular characteristics, and constructs a virtual fingerprint module to enhance model performance. Given the imbalance of the dataset and the scarcity of samples,

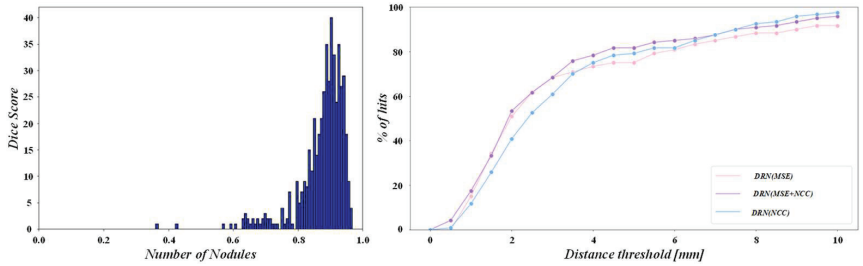

Figure 9 Distribution of different types of security threats.

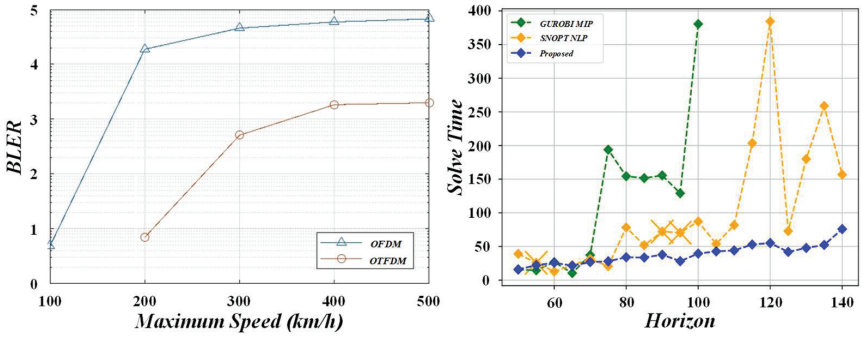

Figure 10 Relationship between system resource consumption and detection accuracy.

it is necessary to conduct in-depth research on strategies for generating additional virtual fingerprints and strengthen data sharing and collaboration mechanisms.

The relationship diagram between system resource consumption and detection accuracy is shown in Figure 10. Information security challenges on the Internet have increased significantly, including risks such as information leakage, property loss and system intrusion. Privacy and property security are threatened, and efficient intrusion detection becomes crucial. This article explores the challenges faced by large-scale, multi-dimensional and unbalanced network anomaly traffic datasets and deeply analyzes related issues.

To ensure the reliability of the experimental results, the dataset source and hyperparameter settings were described in detail. The dataset comes from smart routers and IoT sensors, covering over 100GB of device behavior and network traffic data, which has undergone noise removal and standardization processing. The hyperparameters used for model training include a learning rate of 0.001, batch size of 64, number of LSTM units of 128, time step size

of 50, and the use of Adam optimizer and early stopping method to avoid overfitting. The dataset is divided into 80% training set and 20% testing set, and the model performance is evaluated using metrics such as F1 score and AUC. The experimental details provide other researchers with a reproducible experimental plan, ensuring the validation and replication of the results.

5 Conclusion

An edge device security monitoring system integrating embedded systems and Bi-LSTM is designed to significantly improve security and cope with the bottlenecks faced by traditional centralized architectures in large-scale real-time data processing. This system cleverly combines embedded systems' low energy consumption characteristics and the advantages of distributed computing. It uses the robust performance of the Bi-LSTM network in processing sequence data, aiming to achieve efficient and accurate network intrusion identification. This innovative integration not only enhances the system's response speed and energy efficiency ratio but also effectively reduces the dependence on the central server, thereby building a more flexible, reliable and secure edge computing environment.

A comparative analysis was conducted between the designed system and the existing edge device security monitoring system. The existing systems mainly rely on rule-based detection methods or traditional machine learning algorithms, which are susceptible to complex attacks and unknown threats, and have low accuracy and real-time performance. In contrast, the monitoring system based on Bi-LSTM proposed in this article effectively captures the complex behavior patterns of IoT devices through deep learning models, accurately identifies various security threats, and ensures real-time performance and energy efficiency through hardware optimization and low-power design of embedded systems. The experimental results show that the designed system has improved accuracy by about 12% compared to traditional systems, reaching 94%, and reduced false alarm rate by 20%. The conclusion is that the design combining Bi-LSTM model with embedded systems significantly improves the security of edge devices and provides a feasible solution for intelligent security monitoring in IoT environments.

The experiment uses over 100GB of actual data sets from edge devices such as intelligent routers and IoT sensors, focusing on diverse cyber-attacks. In order to meet this data processing requirement, this paper mainly constructs a Bi-LSTM model, which aims to deeply analyze the sequence characteristics of network traffic and accurately identify abnormal behaviors.

The experimental results show that the detection accuracy of the system based on Bi-LSTM reaches 96.8%, far exceeding the performance of traditional methods such as random forest (92.6%) and support vector machine (SVM) (91.3%). This evidence clearly shows that the Bi-LSTM network can effectively capture and deal with complex dependencies when solving security problems involving time series, thus showing its significant advantages in accuracy.

In addition to analyzing the detection accuracy, this paper also considers the system's real-time performance. Given the low-latency operation of edge devices, embedded system performance is critical. Experiments show that the average response time of the embedded Bi-LSTM system is 184 ms, which meets the real-time processing requirement of 200 ms and reflects the time-sensitive characteristics of edge computing. The system processes about 5,000 network traffic records per second with high data volume, demonstrating efficient processing capabilities.

The design focuses on improving energy efficiency. The test results show that the average power consumption is 5.8 W, suitable for long-term low-power environment operation. Compared with traditional server-side monitoring systems, the embedded design significantly reduces total energy consumption and data transmission latency and cost.

The designed embedded system integrates a Bi-LSTM network for edge device security monitoring, significantly improving detection accuracy, real-time performance and energy efficiency and introducing innovative strategies for edge computing network security protection. This system aims to enhance the security of IoT devices and promote future edge security monitoring technology development.

Funding

This work was sponsored in part by Application Research of AI Driven Wireless Sensor Network Positioning Technology in Intelligent Substations (2023-MSLH-236)

References

[1] Abdi, A., and Salimi-badr, A. DyUnS: Dynamic and uncertainty-aware task scheduling for multiprocessor embedded systems. Sustainable Computing: Informatics and Systems, vol. 43, pp. 101009, 2024.

[2] Dong, Z., Ge, X., Huang, Y., Dong, J., and Xu, J. EG-STC: An Efficient Secure Two-Party Computation Scheme Based on Embedded GPU for Artificial Intelligence Systems. Computers, Materials and Continua, vol. 79 (3), pp. 4021–4044, 2024.

[3] Fareed, A., Hassan, S., Belhaouari, S. B., and Halim, Z. Elevating recommender systems: Cutting-edge transfer learning and embedding solutions. Applied Soft Computing, vol. 166, pp. 112140, 2024.

[4] Gutiérrez-Zaballa, J., Basterretxea, K., and Echanobe, J. Evaluating single event upsets in deep neural networks for semantic segmentation: An embedded system perspective. Journal of Systems Architecture, vol. 154, pp. 103242, 2024.

[5] Hu, K., Huang, W., Wang, L., Mo, C., Wang, R., Chen, Y., Ren, J., and Jiang, B. Unishyper: A Rust-based unikernel enhancing reliability and efficiency of embedded systems. Journal of Systems Architecture, vol. 153, pp. 103199, 2024.

[6] Jin, C., Duan, Y., Zhou, L., and Li, F. Cross-domain recommender system with embedding-and mapping-based knowledge correlation. Knowledge-Based Systems, vol. 304, pp. 112514, 2024.

[7] Li, M., Ma, W., and Chu, Z. KGIE: Knowledge graph convolutionary network for recommender system with interactive embedding. Knowledge-Based Systems, vol. 295, pp. 111813, 2024.

[8] Maruf, M. A., Azim, A., Auluck, N., and Sahi, M. Optimizing DNN training with pipeline model parallelism for enhanced performance in embedded systems. Journal of Parallel and Distributed Computing, vol. 190, pp. 104890, 2024.

[9] Mei, L., Yang, Y., Zhang, X., and Jiang, Y. Embedded exponential Runge-Kutta-Nyström methods for highly oscillatory Hamiltonian systems. Journal of Computational Physics, vol. 514, pp. 113221, 2024.

[10] Niu, L., and Musselwhite, J. Reliability-aware scheduling for (m, k)-firm real-time embedded systems under hard energy budget constraint. Journal of Systems Architecture, vol. 154, pp. 103185, 2024.

[11] Soudré, M. M., Ayala, H. V. H., Melo, A. C., and Llanos, C. H. A novel GPU-based approach for embedded NARMAX/FROLS system identification. Mechanical Systems and Signal Processing, vol. 211, pp. 111261, 2024.

[12] Su, H., Niu, J., Liu, X., and Atiquzzaman, M. SafeCoder: A machine-learning-based encoding system to embed safety identification information into QR codes. Journal of Network and Computer Applications, vol. 227, pp. 103874, 2024.

[13] Zhang, T., Peng, F., Tang, X., Yan, R., Deng, R., and Zhao, S. A sparse knowledge embedded configuration optimization method for robotic machining system toward improving machining quality. Robotics and Computer-Integrated Manufacturing, vol. 90, pp. 102818, 2024.

[14] Atila, Ü., and Sabaz, F. Turkish lip-reading using Bi-LSTM and deep learning models. Engineering Science and Technology, an International Journal, vol. 35, pp. 101206, 2022.

[15] Bibi, N., Maqbool, A., and Rana, T. Enhancing source code retrieval with joint Bi-LSTM-GNN architecture: A comparative study with ChatGPT-LLM. Journal of King Saud University-Computer and Information Sciences, vol. 36 (2), pp. 101865, 2024.

[16] Bukhari, S. M. S., Zafar, M. H., Houran, M. A., Moosavi, S. K. R., Mansoor, M., Muaaz, M., and Sanfilippo, F. Secure and privacy-preserving intrusion detection in wireless sensor networks: Federated learning with SCNN-Bi-LSTM for enhanced reliability. Ad Hoc Networks, vol. 155, pp. 103407, 2024.

[17] Dai, Y., Zhou, Q., Leng, M., Yang, X., and Wang, Y. Improving the Bi-LSTM model with XGBoost and attention mechanism: A combined approach for short-term power load prediction. Applied Soft Computing, vol. 130, pp. 109632, 2022.

[18] Fan, W., Yao, J., Cui, S., Wang, Y., Xu, S., Tan, Y., Yang, F., and Wu, W. Bi-LSTM/GRU-based anomaly diagnosis for virtual network function instance. Computer Networks, vol. 249, pp. 110515, 2024.

[19] Harby, F., Alohali, M., Thaljaoui, A., and Talaat, A. S. Exploring Sequential Feature Selection in Deep Bi-LSTM Models for Speech Emotion Recognition. Computers, Materials and Continua, vol. 78 (2), pp. 2689–2719, 2024.

[20] Miah, M. S. U., Islam, M. I., Islam, S., Ahmed, A., Rahman, M. M., and Mahmud, M. Sustainability-Driven Hourly Energy Demand Forecasting in Bangladesh Using Bi-LSTMs. Procedia Computer Science, vol. 236, pp. 41–50, 2024.

[21] Phukan, R., Goutom, P. J., and Baruah, N. Assamese Fake News Detection: A Comprehensive Exploration of LSTM and Bi-LSTM Techniques. Procedia Computer Science, vol. 235, pp. 2167–2177, 2024.

[22] Riaz, S., Saghir, A., Khan, M. J., Khan, H., Khan, H. S., and Khan, M. J. TransLSTM: A hybrid LSTM-Transformer model for fine-grained suggestion mining. Natural Language Processing Journal, vol. 8, pp. 100089, 2024.

[23] Sambandam, P., Yuvaraj, D., Padmakumari, P., and Swaminathan, S. Deep attention based optimized Bi-LSTM for improving geospatial data ontology. Data & Knowledge Engineering, vol. 144, pp. 102123, 2024.
[24] Wubet, Y. A., and Lian, K.-Y. How can we detect news surrounding community safety crisis incidents in the internet? Experiments using attention-based Bi-LSTM models. International Journal of Information Management Data Insights, vol. 4(1), pp. 100227, 2024.
[25] Yu, K., Kong, C., Zhong, L., Fu, J., and Shao, J. Delay prediction with spatial-temporal bi-directional LSTM in railway network. ICT Express, vol. 9(5), pp. 921–926, 2023.
[26] Ahmed, M., Raza, S., Soofi, A. A., Khan, F., Khan, W. U., Xu, F., Chatzinotas, S., Dobre, O. A., and Han, Z. A survey on reconfigurable intelligent surfaces assisted multi-access edge computing networks: State of the art and future challenges. Computer Science Review, vol. 54, pp. 100668, 2024.
[27] Deng, P., and Huang, Y. Edge-featured multi-hop attention graph neural network for intrusion detection system. Computers & Security, vol. 148, pp. 104132, 2025.
[28] Hazra, A., Tummala, V. M. R., Mazumdar, N., Sah, D. K., and Adhikari, M. Deep reinformation learning in edge networks: Challenges and future directions. Physical Communication, vol. 66, pp. 102460, 2024.
[29] Liu, A., Li, S., Chang, Y., and Hou, Y. EdgeStereoSR: A multi-task network with transformers for stereo image super-resolution considering edge prior. Signal Processing, vol. 227, pp. 109719, 2025.

Biographies

Dang Yuanyi received the bachelor's degree in engineering from Northeastern University in 2004, the master's degree in engineering from Northeastern

University in 2007. He is currently working as a Lecturer at the School of Automation, Shenyang Institute of Engineering, Shenyang Liaoning. His research areas and directions include Network security, communication systems, embedded systems.

Wang Yongbo received the bachelor's degree in in engineering from Shenyang University of Technology in 2008, the master's degree in engineering from Liaoning University in 2011. He is currently working as a Lecturer at the School of Automation, Shenyang Institute of Engineering, Shenyang Liaoning. His research areas and directions include Network security, Signal and Information Processing.