Annals of Emerging Technologies in Computing (AETiC)
Vol. 9, No. 4, 2025

Research Article

Design of Enterprise Data Security Management Based on IoT and CNN

Fan Gao

Huanghe Jiaotong University, China
gaofan1272024@163.com

Received: 28 April 2025; Accepted: 27 July 2025; Published: 1 October 2025

Abstract: In the era of rapid digital transformation, enterprise data security faces increasingly complex and dynamic threats. Traditional defense mechanisms are complicated to effectively respond to real-time risks, mainly when enterprises rely extensively on Internet of Things (IoT) devices. To address this problem, this paper proposes and implements a dynamic intelligent security assessment and early warning system based on ResNet-50 architecture and IoT technology. The system builds a distributed IoT data collection platform to collect multi-source data such as network traffic, device status changes, and user behavior in real time. It uses the optimized ResNet-50 model to analyze high-dimensional heterogeneous data streams accurately. The system is deployed in a cloud computing environment and can process large-scale data with low latency. It can instantly detect abnormal activities, conduct threat assessment, and issue alerts based on contextual information. Experimental results show that the system has an accuracy rate of 98.6% for distributed denial of service (DDoS) attacks and 96.2% for malware data leaks, with an average response time of 1.03 seconds, significantly better than traditional detection methods. This study provides an efficient and scalable solution for enterprise data security protection and lays a foundation for further integrating AI-driven models with IoT infrastructure.

Keywords: ***Convolutional Neural Network; Data Security Detection; Enterprise Data Security; IoT; Residual Network***

1. Introduction

Amid the digital revolution, enterprises are experiencing unprecedented technological innovation, and the broad use of information technology has dramatically improved business efficiency and innovation capabilities [1-2]. However, with the popularization of network connections and the large-scale deployment of IoT (IoT) devices, enterprise data security faces increasingly complex and changing challenges [3-4]. Traditional defense measures, including firewalls, intrusion detection systems, and antivirus software, deal well with static threats. Still, their limitations are gradually emerging in the face of dynamically changing real-time attacks [5-7]. Such measures are usually based on predefined rules or signatures to determine known threats and lack efficient response capabilities for unknown or zero-day attacks [8]. In addition, the enterprise network environment is becoming more and more complex. Traditional methods have difficulty processing massive real-time data streams and cannot provide immediate security early warnings, which puts enterprises in a passive position when facing potential security risks [9-10].

To address these challenges, this paper develops a data security monitoring and early warning system based on ResNet-50 (Residual Network-50) and IoT technology. The system uses real-time data collected by IoT devices [11] and enhanced machine learning (ML) schemes to dynamically assess the enterprise's network security status. By applying ResNet, the gradient vanishing problem in deep networks can be effectively solved, thereby more accurately capturing subtle pattern changes in data [12-14]. As an efficient convolutional neural network (CNN), ResNet's powerful feature extraction capabilities provide new possibilities for security threat detection [15-16]. IoT technology offers a wide range of sources for data collection. From network traffic to device status changes, the system can use various types of real-time data to comprehensively monitor the enterprise's digital environment [17-19].

Fan Gao, "Design of Enterprise Data Security Management Based on IoT and CNN", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 61-78, Vol. 9, No. 4, 1 October 2025, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2025.04.005, Available: https://aetic.theiaer.org/archive/v9/v9n4/p5.html.

This study implements an innovative solution: constructing a distributed IoT data collection platform to obtain structured and unstructured real-time data streams from different IoT devices. The data is subjected to feature extraction and normalization by the preprocessing module to ensure the consistency of the input and the effectiveness of model training. The optimized ResNet-50 model is deployed on the cloud server as the core analysis engine to process high-dimensional real-time data. The scheme can adjust the architecture according to specific enterprise application scenarios, such as modifying the fully connected layer to adapt to particular security threat detection tasks. When the system is running, it continuously monitors the incoming data. Once the abnormal activity is detected, the deep analysis process is immediately started to check the content and flow of the data packet, determine whether there is a security threat based on the context information, and issue corresponding alarms according to the threat level.

This study is dedicated to improving the precision and response speed of enterprise data security threat recognition. By combining IoT technology and DL technology, risks can be quickly predicted in complex network environments to help enterprises promptly prevent possible data threats. This solution can promote the advancement of information security technology, provide innovative methods for protecting enterprise data security and personal digital assets, and promote the development of data security management in the direction of intelligence.

2. Related Works

Enterprise data security management mainly includes intrusion detection systems (IDS), rule-based detection systems, and DL schemes. Traditional IDS, Snort, and Suricata [20] can effectively identify known attacks, but are insufficient in responding to novel attacks. Rule-based detection systems, such as security information and event management (SIEM) platforms, gather and review multi-source log data and perform threat detection with preset rules [21]. Still, their response speed to unknown threats is slow. Some investigators have deployed ML technology to boost recognition precision and flexibility. Varzaneh and Rafsanjani [22] studied a rule-based classification of intrusions based on a fuzzy model and proposed an algorithm for generating rules dynamically based on improving detection rates while minimizing false alarms. To further develop rule weights for realization, a genetic algorithm (GA) was used by them, further increasing the ability of the system to handle complex and dynamic security threats. The method was tested on the commonly used KDD99 dataset, and experimentally found that there was a significant improvement in detection rate with reduced false alarm rates. However, even though its effectiveness was proven, the method was still highly reliant on a large amount of labeled training data, highlighting the requirement for more advanced and label-efficient techniques in the field of intrusion detection. DL schemes have brought new hope to enterprise data security control. The exploration of Kumar [23] analyzed the use of convolutional neural networks (CNNs) in malware detection and proposed a novel malware classification framework known as MCFT-CNN. The model utilized fine-tuned CNNs to detect sophisticated malware threats without manual feature extraction, binary analysis, or reverse engineering, thereby bypassing the difficulty of code obfuscation and tight encryption methods. In particular, the MCFT-CNN modified the ResNet50 architecture by adding a dense fully connected layer and combining its output with ImageNet-derived features that were then passed through a SoftMax layer for final classification. The model achieved a high accuracy rate of 99.18% upon training on the MalImg dataset, along with a fast prediction time of 5.14 ms. Furthermore, it featured high generalizability, maintaining a high accuracy rate of 98.63% on the larger Microsoft Malware Challenge dataset (~500 GB) with similar effectiveness. These results reflect the viability and scalability of deep learning-based methods like MCFT-CNN in modern malware detection.

Thapa and Duraipandian [24] proposed a malicious traffic classification system using a Long Short-Term Memory (LSTM) model to enhance Distributed Denial-of-Service (DDoS) attack detection. This approach aimed to mitigate the limitations of classical behavior-based and intrusion detection systems, which often process network data sequentially and have a high dependence on labeled datasets, leading to performance degradation, network loop occurrences, and bandwidth inefficiency. Through the use of the temporal modeling inherent in LSTM, their artificial intelligence-based method reduced the dependence on hand-specified data sequences and minimized the requirement for manual labeling procedures. Experimental evaluations showed that the model provided an outstanding classification accuracy of 99.5%, outperforming modern approaches by 5% in terms of accuracy and throughput, thus establishing its

applicability to real-time network security applications. In addition, Lu *et al.* [25] proposed Inception-LSTM (ICLSTM), a deep learning hybrid architecture that focused on improving the detection of malicious behavior in encrypted traffic. This approach resolved the limitations of traditional machine learning algorithms, specifically their dependence on manual feature extraction. By transforming data related to encrypted traffic into grayscale images, the ICLSTM architecture could automatically recognize important features using the combination of Inception and Long Short-Term Memory (LSTM) networks. In an effort to resolve the class imbalance problem, certain weight adjustments during training were made, improving the classification fairness across different genres of encrypted traffic. Empirical testing on the ISCX 2016 dataset showed that the model achieved an accuracy rate well over 98% in both the detection of normal encrypted services and the classification of VPN-encrypted traffic. Such findings reflected the effectiveness of the framework in examining encrypted traffic while at the same time reducing the complexity in feature extraction. Ren *et al.* [26] proposed a sophisticated intrusion detection system called ID-RDRL, which combines Recursive Feature Elimination (RFE) and Deep Reinforcement Learning (DRL) to promote the discovery of network vulnerabilities. Through the use of RFE, the system effectively eliminated over 80% of unnecessary features, thus substantially reducing the input space to enhance the efficiency of learning. Next, the significant features were examined through a neural network to determine relevant patterns, after which the classifier was trained with DRL to enable continuous interaction with the surrounding environment. This iterative process allowed the system to dynamically adjust its detection approach, thus significantly enhancing its responsiveness and adaptability to new threats. In experiments carried out using the CSE-CIC-IDS2018 dataset, which mimics real-world network traffic scenarios, the system delivered exceptional performance in complicated environments, thus supporting its effectiveness and resilience in intrusion detection. Xu *et al.* [27] proposed a multimodal deep learning framework for a holistic evaluation of network security threats based on the fusion of heterogeneous data sources, such as network traffic, system logs, and user behavior. The framework utilized Graph Convolutional Networks (GCNs) to handle the complex interdependencies between network entities, which greatly improved the predictive power of the model. The multimodal transformation allowed the efficient integration of heterogeneous data, resulting in improvements in accuracy and system robustness. In the context of edge computing, the method outperformed conventional approaches in multiple performance metrics, such as accuracy, precision, recall, F1-score, and AUC-ROC, at over 90% accuracy in proactive threat detection. The research also highlighted the need for further work in the area of developing transformation methods, interoperability enhancement, and model adaptation to large and dynamic network environments. Zhou *et al.* [28] researched the use of graph neural networks (GNNs) to enhance security in massive-scale Internet of Things (IoT) environments with special focus on detecting synchronized attack patterns by analyzing inter-node connections in communication and social networks. They proposed a new hierarchical adversarial attack (HAA) generation method to reveal the vulnerabilities inherent in GNN-based network intrusion detection systems (NIDS) by using a level-aware black-box adversarial framework with the inclusion of a shadow GNN and saliency map techniques to generate adversarial samples that require minimal adjustments to features. Additionally, they created a hierarchical node selection algorithm based on random walk with restart (RWR) to identify and target nodes of high structural significance that played a significant role in influencing model performance. Experimental results using the UNSW-SOSR2019 dataset proved that the HAA approach caused a decrease in the classification accuracy of GNN-based detectors like GCN and JK-Net by more than 30%, thus revealing severe vulnerabilities in present security solutions intended for IoT and emphasizing the need for stronger GNN defense mechanisms.

Xia *et al.* [29] proposed an optimized convolutional neural network architecture called SparkNet, which aimed to meet the high computational resource requirements of deep learning models on edge devices with limited resources. In their efforts to minimize memory usage and optimize inference efficiency, SparkNet achieved a decrease in model parameters and computational complexity by a factor of 150 while preserving accuracy. The architecture was tested using four traditional benchmark datasets, MNIST, CIFAR-10, CIFAR-100, and SVHN, therefore validating its applicability and practicability. For ease of deployment in edge computing environments, the authors implemented SparkNet on an Intel Arria 10 GX1150 FPGA, leading to the implementation of SparkNet on Chip (SparkNOC). This hardware implementation mapped each layer to a distinct hardware block, thus allowing pipelined execution and enhancing parallelism as well as energy efficiency. The FPGA-based system registered a processing performance rate of 337.2 GOP/s and an

energy efficiency of 44.48 GOP/s/W, outperforming the performance indicators of modern state-of-the-art solutions in terms of speed and resource utilization, thereby validating the applicability of SparkNet for real-time AI inference at the edge. In addition, Banabilah *et al.* [30] explored the promise of Federated Learning (FL) as a privacy-centric framework that enables collaborative machine learning while preserving data confidentiality. This method improves the ability of heterogeneous organizations to build models collaboratively without requiring the sharing of sensitive data, thus greatly enhancing data security. They conducted a comprehensive review that not only explored the core FL-related technologies, system architectures, and privacy-preserving mechanisms first proposed by Google but also extended the exploration to its varied applications and market relevance. The study classified existing research on FL in various technological domains, including Artificial Intelligence, the Internet of Things, blockchain technology, Natural Language Processing, autonomous driving, and resource management. In addition, the authors explored the application of FL across varied realms such as healthcare, education, and industry, and emphasized promising avenues in personalized services, battery optimization, and government programs. Through the provision of a comprehensive appraisal of both the theoretical frameworks and practical applications, their review provided critical insights into the emerging age of federated learning and its strategic relevance in secure, decentralized model training. Liang *et al.* [31] performed an in-depth evaluation of the application of adversarial machine learning (ML) in the field of network security, focusing on improvements in the attack resistance and robustness of deep learning models, especially in edge computing scenarios relevant to the development of smart cities. Their study highlighted the ability of Edge AI to facilitate the deployment of deep neural networks (DNNs) on devices with constrained resources for a wide range of applications, including facial recognition, intelligent healthcare, and autonomous driving. They, however, noted that DNNs are highly susceptible to adversarial attacks, small input changes leading to the model making incorrect predictions, thus compromising critical security infrastructures. The authors systematically categorized the defense measures into three main categories: model-based, data-based, and auxiliary network-based methods. They also elaborated on prominent limitations in current research, including high computational needs, a lack of model interpretability, and poor handling of class imbalance under dynamic, real-world enterprise data environments. The paper concluded on an examination of the current arms race between attacks and defenses, highlighting the need for more interpretable and stronger defense mechanisms against adversarial attacks in future secure AI deployments.

Sun and Bai [32] explored the field of enterprise information security in the context of global informatization by integrating Internet of Things (IoT) and Artificial Intelligence (AI) technologies with modern enterprise management approaches. They designed a management platform based on IoT, specifically engineered to improve enterprise information security, comprising four main modules: IoT data mining, equipment management, key management, and database management. The platform's architectural design was developed to meet the needs of modern management, with security aspects integrated directly into the system architecture. Multiple performance tests were conducted using various testing methods, such as concurrency, stress, large-scale data processing, and security testing. The results showed that the platform had a 100% operational success rate, with an average response time of 0.13 seconds in concurrency and stress testing, and 0.25 seconds for event entry operations. Further, CPU use during monitoring tasks always remained below 20%. These results confirmed the reliability and suitability of the platform for regular enterprise operations, thus providing a reference model for future AI-based solutions for enterprise security management. Yue [33] carried out an inquiry into the integration of Internet of Things (IoT) and deep learning technologies for enhancing intelligent decision-making in the field of enterprise management. The study adopted a mixed-methods design, which coupled telephone interviews with online questionnaires, thus obtaining information from about 100 technical experts working with 25 well-known enterprises. The study first determined the basic concepts of IoT, deep learning, and intelligent decision-making, then determined the main problems of enterprise decision systems. Based on the findings obtained, a new decision-making model was designed by combining IoT and deep learning, focusing on the optimization of human resource management, digital file management, and the dynamic monitoring of production and work flows. The simulation results and subsequent analyses revealed that the system proposed outperformed conventional decision-making methods. Specifically, it raised the utilization rate of enterprise information management to 84.2% and increased the efficiency of intelligent decision-making

by an average of 28.7%. The findings highlighted the potential advantages of enhancing enterprise competitiveness and improving decision-making accuracy through the use of IoT and deep learning.

So, from the literature examined, it is evident that though traditional IDS and rule-based IDS are structured and interpretable, they adapt poorly to evolving threats. Then again, deep learning architectures such as CNNs, LSTMs, and hybrid models (such as ICLSTM, ID-RDRL) offer high accuracy and feature extraction capability but typically demand high computational resources and training data. The robustness to adversarial attacks, model interpretability, and low-latency implementation issues are particularly important for dynamic or resource-limited settings such as edge devices. Further, even while federated learning and multimodal models provide potential solutions to issues of data heterogeneity and privacy, they also bring with them convergence and interoperability issues. Therefore, in this article, we propose a data security monitoring and early warning system that combines the real-time perception functions of IoT devices with the powerful feature learning function of an optimized ResNet-50 model. This approach aims to offer scalable, context-aware, and high-precision threat detection suitable for enterprise networks under the conditions of various and evolving threat environments.

In spite of the favorable individual outcomes noted in the studies examined, there are considerable gaps and contradictions. For instance, MCFT-CNN and SparkNet models focus on both accuracy and speed but tend to overlook adversarial susceptibility and cross-dataset generalizability. Conversely, methods such as HAA-GNN reveal these shortcomings but do not provide specific defense mechanisms. Likewise, federated learning frameworks tackle privacy issues but presume equal data quality across clients, which is seldom realized in practical enterprise environments. Besides, few efforts integrate IoT-derived contextual knowledge with deep neural models for enabling edge adaptive decision-making. Few hybrid frameworks can simultaneously optimize detection performance, model interpretability, computational efficiency, and adversarial robustness. This paucity of integration calls for an integrated, resilient framework capable of integrating multi-source data in real time while maintaining security, efficiency, and adaptability.

The suggested hybrid model, ResNet-50–IoT, aims to fill this interdisciplinarity gap by architectural optimization, context-aware data handling, and scalability.

3. Enterprise Data Security Management Design

3.1. Distributed IoT Data Collection Platform Construction

This paper builds a distributed IoT data collection platform for real-time monitoring and early warning of enterprise network security status. The platform is designed to obtain structured and unstructured real-time data streams from different types of IoT devices, including but not limited to network traffic, device status changes, etc. The data serves as the basis for subsequent DL model analysis. In addition, special attention is paid to data privacy and security in the platform design to ensure strict security standards are followed during data collection, transmission, and storage.

3.1.1. Data Source Selection

The data sources selected in this paper include network traffic, device status, and log data, which can provide security information at different levels and help enterprises identify potential data security threats more effectively. Network traffic data is generated in real time and can reflect the activities currently in the network. They are mainly captured through the mirror ports of some network sensors to ensure that all key network nodes are covered. Device status information changes when there is abnormal data. It can be obtained from various IoT devices through the application programming interfaces (API) to detect whether the device is invaded promptly. Log data records detailed operation logs of the system and application, which provides rich contextual information and helps to analyze security incidents deeply. Network traffic and device status data can be supplemented by collecting security logs from firewalls, IDS, and application servers.

3.1.2. Data Collection Process

The data collection nodes acquire data from various network nodes and IoT devices. After preliminary processing, the collected data is transferred to the central data processing center in real time for unified management and analysis. The central data processing center performs preprocessing work, such as

cleaning and denoising the data from the data collection nodes, to guarantee the accuracy and reliability of the collected data. It can provide input for subsequent DL schemes by centrally managing and processing all collected data. Fig. 1 displays the framework of data collection.

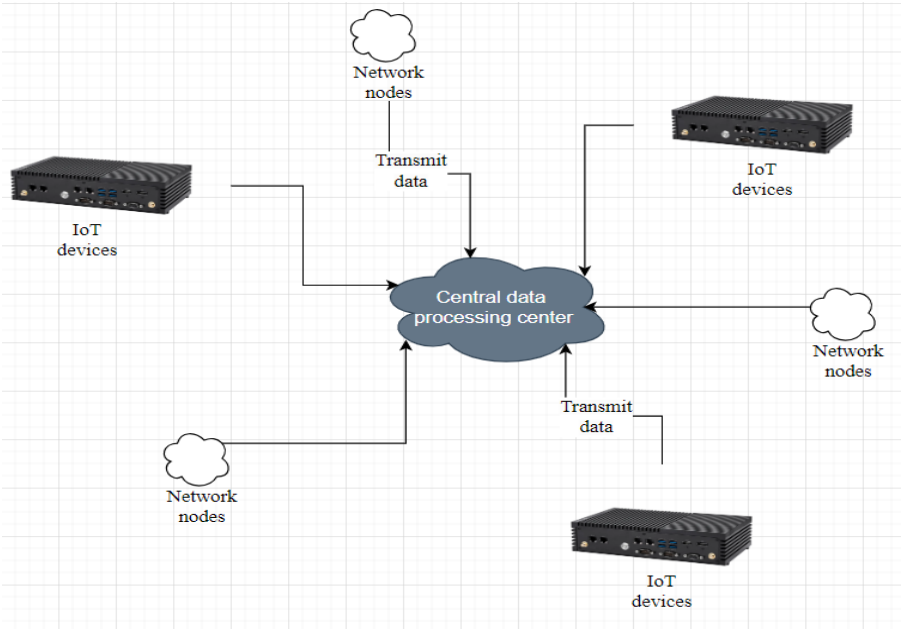


Figure 1. Framework for collecting data from IoT devices and network nodes

3.2. Data Preprocessing

3.2.1. Feature Extraction and Normalization

This paper preprocesses the collected raw data to guarantee the consistency of input data and the effectiveness of the scheme training. The preprocessing process mainly includes feature extraction, data cleaning, normalization, and attribute retrieval. This paper primarily collects three data types: network traffic, device status information, and log data. Their raw data are usually high-dimensional and contain much redundant information, which may lead to overfitting or slow convergence of subsequent model training. Therefore, extracting features from these three types of data first is necessary to optimize the following model inputs. Table 1 shows the main feature descriptions and extraction methods of these three data types.

Table 1. Summary of key data features

Category	Feature name	Description	Extraction method
Network traffic	Packet length distribution	Length distribution of each data packet	Mean, variance, and entropy
	Flow duration	Duration of the data stream	Timestamp difference
	Protocol type	The transmission protocol used	Protocol field analysis
Device status information	CPU utilization	Device CPU usage	Sliding window method
	Memory occupancy	Device memory usage	Calculating the ratio of used memory to total memory
	Temperature	Device temperature	Sensor reading
Log data	Event type	Event type in the security log	Log analysis
	Event frequency	The count of times a specific type of event occurs within a particular period	Counting method
	IP address	Involved IP address	Log analysis

3.2.2. Data Cleaning and Normalization

In enterprise data security management, data preprocessing and cleaning are key steps to ensure the quality of input data and model training results. This study combines specific application scenarios and focuses on data cleaning, mainly dealing with two types of problems: missing value processing and outlier detection. This paper uses the mean filling method to process missing values and the isolation forest algorithm for outlier detection. In addition, it uses the Min-Max normalization method to standardize the features.

Missing values are a common problem in network traffic, device status information, and log data. This article deploys mean filling to handle the missing data:

$$x_{imputed} = \begin{cases} x', & \text{if } x_i \text{ is missing} \\ x_i, & \text{otherwise} \end{cases} \quad (1)$$

Among them, x' displays the mean of the feature x, and x_i displays the feature value of the i-th sample. In this way, missing values can be effectively filled.

In enterprise network security, outlier detection is essential to identifying potential threats. Isolation forest can efficiently process complex data structures and is highly sensitive to outliers. The formula is:

$$\text{anomaly\_score}(x) = \frac{E(h(x))}{c(n)} \quad (2)$$

Among them, c(n) is a constant related to the sample size n. If anomaly_score(x) exceeds the set threshold, x is considered an outlier.

This paper chooses the Min-Max normalization method to scale the feature values to the interval [0,1]. Min-Max normalization applies to most ML schemes and can handle features of different magnitudes. The formula is:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \quad (3)$$

3.2.3. Attribute Retrieval

Reasonable attribute retrieval can improve the model training speed and efficiency, avoid overfitting, and effectively utilize the enterprise's computational resources to reduce the system's operating costs. This paper adopts various attribute retrieval methods, combined with specific business needs and data characteristics, to ensure that the selected features are representative. The filtering method and packaging method are chosen to select features.

The filtering method is a preliminary screening based on statistical tests, which is used to quickly remove those features that are irrelevant or less relevant to the attack behavior. It includes mutual information and a chi-square test. Mutual information can capture linear and nonlinear correlations. The formula is:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \quad (4)$$

Among them, X is a feature vector, representing a specific network traffic or device status feature. Y is a target variable, indicating whether it is an attack behavior (0 means normal, and 1 means attack).

This exploration deploys the chi-square test to evaluate the correlation between discrete features such as protocol type and port number in network traffic and attack labels. The formula is:

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \quad (5)$$

Among them, O_i displays the measured frequency; E_i displays the predicted frequency; k displays the count of categories.

The packaging method is adopted to train the scheme multiple times. Finally, the optimal feature combination is found. This paper chooses recursive feature elimination as the specific implementation of the packaging method. The formula is:

$$S_{best} = arg\ max\ accuracy(S) \quad (6)$$

Among them, S is a feature subset.

3.3. Model Optimization

ResNet-50 is a classic CNN widely used in image identification and other fields due to its deep residual learning characteristics [34-35]. This paper optimizes the fully connected layer of ResNet-50 to meet the requirements of enterprise data security management and ensure that it can effectively handle complex data security threats.

3.3.1. Fully Connected Layer Adjustment

In security threat detection tasks, the choice of output dimension is directly related to the scheme's classification ability. For binary classification problems (such as normal traffic and malicious traffic), the output dimension of the fully connected layer is set to 2, corresponding to two types of labels. The output

dimension is set to the count of categories for multi-classification problems (such as multiple types of attack behaviors). The formula is:

$$y = Wx + b \tag{7}$$

This paper uses the softmax function in the last layer. The formula is:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{8}$$

Among them, z is the input vector; C signifies the overall count of categories; $\sigma(z)_i$ displays the probability of the i-th category. The predicted probability of each category is obtained through the softmax function to make a classification decision.

3.3.2. Residual Block Design

One of the core innovations of ResNet is the use of residual connection (skip connection), which effectively solves the gradient vanishing problem in deep networks [36]. Through the residual connection, the scheme can transmit information to deep layers more smoothly without increasing the complexity of the network. The formula is:

$$y = F(x, W_i) + x \tag{9}$$

This way, even in an intense network, the gradient can be directly passed to the shallow layers through the residual connection, avoiding the gradient vanishing problem.

This paper adopts a bottleneck structure in each residual block to accelerate training further. The bottleneck structure consists of 3 convolutional layers, where the 1*1 convolutional layer is utilized to diminish the count of channels; the 3*3 convolutional layer is deployed to derive local features; the last 1*1 convolutional layer is deployed to restore the count of channels. Fig. 2 presents the schematic diagram of the residual block applied to the bottleneck structure:

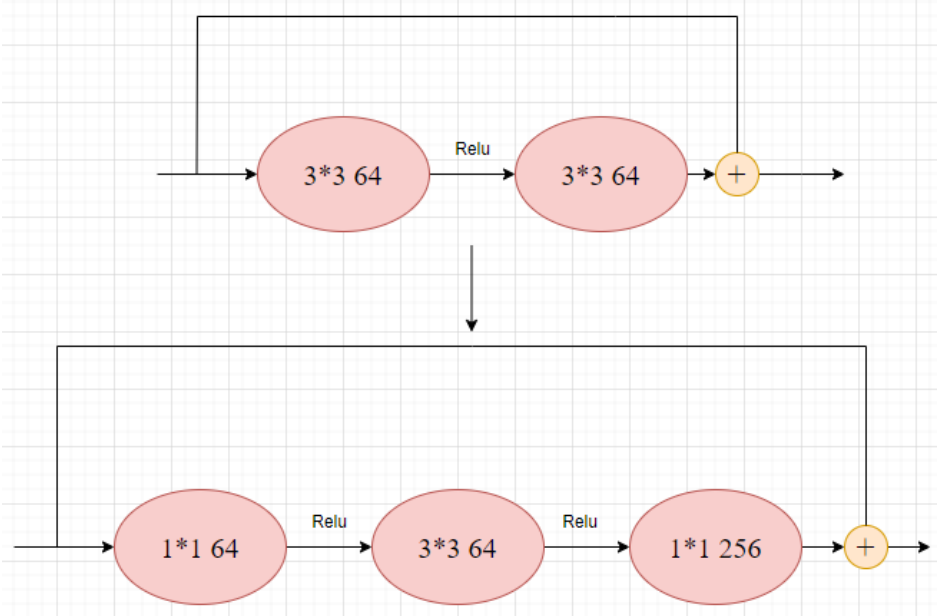


Figure 2. Schematic diagram of the residual block applied to the bottleneck structure

3.3.3. Batch Normalization

Adding a batch normalization layer after each convolutional layer can stabilize the training process and accelerate convergence. The formula is:

$$x' = \frac{x - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \tag{10}$$

Among them, x is the input feature; μ_B and σ_B^2 are the mean and variance of the current batch, respectively; ε is a small constant used to prevent zero division errors. The normalized feature x' is further adjusted by scaling and translation operations:

$$y = \gamma x' + \beta \tag{11}$$

Among them, γ and β are learnable parameters used to restore the scale and offset of the feature. Batch normalization can effectively alleviate the problem of internal covariate shift, make the scheme easier to train, and improve generalization ability.

3.3.4. Loss Function Selection

In practical applications, security threat detection tasks often have the problem of category imbalance: normal samples are far more common than attack samples. This paper uses the weighted cross-entropy loss function to deal with this problem. A higher weight is given to the minority class. The formula is:

$$L = -\sum_{i=1}^{C} w_i y_i log(y_i') \quad (12)$$

In Formula (12), w_i displays the weight of the i-th category. Usually, the weight can be adjusted dynamically according to the number of samples of each category. It is assumed that the count of normal samples is N_{normal}, and the count of attack samples is N_{attack}. To boost the training weights of attack samples, the weights can be defined as:

$$w_{normal} = \frac{N_{total}}{N_{normal}} \quad (13)$$

$$w_{attack} = \frac{N_{total}}{N_{attack}} x \quad (14)$$

3.3.5. Regularization Techniques

This paper adds L2 regularization to the loss function to avoid overfitting. L2 regularization improves generalization ability by penalizing larger weight values. The formula is:

$$L = L_{CE} + \lambda ||\theta||^2 \quad (15)$$

Among them, L_{CE} is the cross-entropy loss; θ is the scheme's parameter; λ is the regularization coefficient. The intensity of regularization can be controlled by adjusting the value of λ. A larger value of λ can lead to stronger regularization but may reduce the expressiveness of the scheme. A smaller value of λ allows the scheme to fit the data more flexibly but may also raise the risk of overfitting.

This paper applies the Dropout technique in the fully connected layer, which haphazardly drops some neurons, forcing the scheme to rely on different combinations of neurons during training and avoid over-reliance on certain specific neurons. The formula is:

$$y = dropout(x, p) \quad (16)$$

In Formula (16), x displays the input feature, and p signifies the dropout rate, which indicates the proportion of random drops during each training. For example, if p=0.5, about half of the neurons are haphazardly dropped in each training, and the remaining neurons continue to participate in forward propagation and backpropagation. In this way, the scheme becomes more robust during training and can perform better during testing.

3.4. Model Training

3.4.1. Data Set Preparation

(1) Normal behavior data:

First, normal network traffic, device status, and user behavior data within the enterprise are collected as negative samples. These data reflect the normal activity patterns of the enterprise in daily operations and are the basis for building security threat detection schemes. This paper collects data from multiple network nodes and IoT devices in different periods, for example, network traffic captured from core switches, servers, terminal devices, and other locations, as well as status information obtained from devices such as cameras and sensors.

During the data collection process, this paper strictly abides by the principles of ethics and privacy protection to ensure that the collection, transmission and storage of all data comply with relevant laws and regulations and internal corporate security standards. The collection of normal network traffic, device status and user behavior data is anonymized to avoid direct association with personal identity information. At the same time, before obtaining data, the relevant personnel are clearly informed of the purpose of the data and obtain necessary authorization. In addition, through the design of a distributed IoT data collection platform, encryption technology and access control mechanisms are used to further ensure data security, prevent

sensitive information leakage or unauthorized access, thereby maximizing user privacy while improving the enterprise's data security management capabilities.

(2) Attack behavior data:

This paper obtains known attack behavior data from public data sets and actual attack cases as positive samples to train the scheme to identify various attack behaviors. These data sets contain various common attack types, such as DDoS attacks, SQL (Structured Query Language) injection attacks, and port scans. Mixing these attack data with normal behavior data allows the scheme to distinguish between regular and malicious traffic. In addition, these features are combined with actual data within the enterprise to construct a comprehensive attack behavior data set.

(3) Data annotation:

The collected data must be annotated to mark whether each record is an attack behavior to train the supervised learning model. This paper uses Formula (17) to annotate each sample x_i:

$$y_i = \begin{cases} 0, if\ x_i\ is\ normal \\ 1, if\ x_i\ is\ attack \end{cases} \quad (17)$$

Among them, y_i is the label of the sample x_i. 0 displays normal behavior, and 1 displays attack behavior. This paper adopts semi-supervised learning and active learning techniques to reduce the workload of manual annotation.

3.4.2. Training Strategy

This paper divides the training data into multiple batches to effectively utilize computational resources and avoid memory overflow. Only a part of the data is loaded for training each time. The training data set is assumed to contain N samples divided into B batches, each containing M samples. The formula is:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta L(\theta_t, x_b, y_b) \quad (18)$$

Among them, θ_t is the scheme parameter of the t-th step; η signifies the learning rate; L signifies the loss function; x_b and y_b are the input features and labels of the current batch, respectively. Batch training effectively manages memory usage, and each update is based on a small batch of data, avoiding the gradient explosion problem. In addition, batch training can also ensure a certain degree of randomness, which helps the scheme improve its generalization ability.

The dynamic learning rate scheduling strategy of cosine annealing is used to gradually reduce η during training and eventually approach zero. The learning rate scheduling formula of cosine annealing is:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})\left(1 + cos\left(\frac{t}{T}\pi\right)\right) \quad (19)$$

An early stopping mechanism is also set during the training process. This paper uses the performance indicators on the validation set to monitor the scheme's performance. When the performance on the validation set no longer improves, the training is terminated early to prevent the scheme from continuing to optimize the training set and ignoring the validation set. The specific implementation is:

$$if\ val\_loss_{t+1} > val\_loss_t,\ stop\ training \quad (20)$$

Among them, val_loss_t is the loss value on the validation set in step t. The early stopping mechanism stops the training when the scheme reaches the best performance to avoid overfitting the training data. In addition, the best scheme parameters during training are recorded and restored after the training to ensure that the final model has the best generalization ability.

Table 2 shows the training parameter settings of the scheme in this paper, including batch training, η scheduling, early stopping strategy, regularization technique, and weight adjustment of the loss function. In the batch training stage, this paper sets the batch size to 64 and the count of training epochs to 3,000, which applies to all training stages. Regarding learning rate scheduling, the initial η is 0.01, and the minimum η is 0.0001. The early stopping strategy sets the early stopping value to 10 and the minimum variation to 0.001 to avoid overfitting. In the regularization technique, the L2 regularization coefficient is 0.0001 and is applied to all trainable parameters. The dropout rate is set to 0.5 and applied to the fully connected layer. The loss function is set to weighted cross-entropy loss, where the weight of the standard category is 0.5 and the weight of the attack category is 2.0, to deal with the problem of category imbalance.

Table 2. Summary of training strategy parameter settings

Training strategy	Parameter name	Parameter value	Application location
Batch training	Batch size	64	All training phases
	Training epochs	3000	Each epoch
Learning rate scheduling	Initial learning rate	0.01	All training phases
	Minimum learning rate	0.0001	All training phases
Early stopping	Early stopping value	10	Validation set monitoring
	Min variation	0.001	Validation set monitoring
Regularization technique	L2 regularization coefficient	0.0001	All trainable parameters
	Dropout rate	0.5	Fully connected layer
Loss function	Weight of the normal category	0.5	Weighted cross-entropy loss
	Weight of the attack category	2.0	Weighted cross-entropy loss

4. Enterprise Data Security Evaluation

4.1. Verification of Different Abnormal Attack Behaviors

This paper uses the 5-fold cross-validation method to validate the scheme. This paper divides the training dataset into five subsets. Finally, the scheme evaluation result is the average performance on all validation sets. The formula is:

$$accuracy = \frac{1}{k}\sum_{i=1}^{k} accuracy_i \tag{21}$$

Among them, accuracy is the accuracy of the i-th verification.

Table 3 shows the identification accuracy and average response time for various abnormal attack behaviors in enterprise data security detection. A 5-fold cross-validation method is utilized. The outcomes show that the overall accuracy is high, indicating that the detection system performs well. The average accuracy of DDoS attack identification is 98.6%, and the accuracy is stable and high in each fold, with an average response time of 0.97s. The average accuracy of malware data theft, port scanning attack, SQL injection attack, and ARP (Address Resolution Protocol) spoofing attack identification is 96.2%, 96.4%, 96.2%, and 96.8%, respectively, and the average response time is 1.04s, 1.02s, 1.09s, and 1.03s, respectively. Although the performance is lower than the DDoS attacks, it also shows a high identification ability and a fast response speed. The scheme's average response time to various attacks in this paper is 1.03s, and its response speed remains relatively fast. The accuracy of each fold in Table 3 shows that each attack behavior identification in different folds fluctuates slightly. This suggests that the scheme has good stability and generalization ability and can effectively help enterprises prevent most abnormal attack behaviors.

Table 3. Detection accuracy and average response time of different abnormal attack behaviors

Abnormal attack behavior	1	2	3	4	5	Average accuracy	Average response time (s)
DDoS attack	98.2%	98.8%	98.5%	98.6%	98.9%	98.6%	0.97
Malware data theft	95.8%	96.5%	96.0%	96.2%	96.3%	96.2%	1.04
Port scanning attack	96.5%	96.0%	96.8%	96.2%	96.3%	96.4%	1.02
SQL injection attack	96.0%	96.5%	96.2%	95.8%	96.3%	96.2%	1.09
ARP spoofing attack	96.5%	96.3%	96.8%	97.3%	97.0%	96.8%	1.03

Table 4 shows the F1 score results of different attack types in 5-fold cross validation, reflecting the comprehensive performance of the model under class imbalance.

Table 4. F1-score of different abnormal attack behaviors

Abnormal attack behavior	1	2	3	4	5	Average F1 score
DDoS attack	97.60%	98.00%	97.80%	97.70%	98.10%	97.80%
Malware data theft	96.10%	96.60%	96.30%	96.40%	96.50%	96.30%
Port scanning attack	96.40%	96.20%	96.70%	96.30%	96.60%	96.50%
SQL injection attack	96.30%	96.70%	96.40%	96.20%	96.50%	96.40%
ARP spoofing attack	96.70%	96.50%	96.90%	97.10%	97.00%	96.90%

Table 4 shows the F1 score results of different attack types in 5-fold cross-validation, reflecting the comprehensive performance of the model under class imbalance. As can be seen from the table, the improved ResNet-50 model performs particularly well in identifying DDoS attacks, with an F1 score of 97.8%, showing the model's high-precision detection capability for large-scale traffic anomalies. At the same time, for malware data leakage, port scanning attacks, SQL injection attacks, and ARP spoofing attacks, the model's F1 scores are 96.3%, 96.5%, 96.4%, and 96.9%, respectively, indicating that it has strong robustness

and balance in dealing with a variety of complex attack types. Overall, the model's stable performance on various types of attacks verifies its efficiency and reliability in enterprise data security management.

4.2. Comparison of the Effectiveness of Diverse Schemes in Identifying Diverse Types of Attacks

In enterprise network intrusion detection research, in addition to the ResNet model used in this paper, there are other commonly used schemes, such as LSTM, random forest, XGBoost (Extreme Gradient Boosting), and support vector machine (SVM), to protect enterprise data security. This paper verifies the efficacy of the improved ResNet-50 scheme in determining abnormal attack behaviors by comparing the recall (R) and precision (P):

$$P = \frac{TP}{TP+FP} \tag{22}$$

$$R = \frac{TP}{TP+FN} \tag{23}$$

A 5-fold cross-validation method is used for each model. During model training, the key parameters of each model are adjusted. Figs. 3 and 4 present the precision and recall results of different enterprise data security detection schemes.

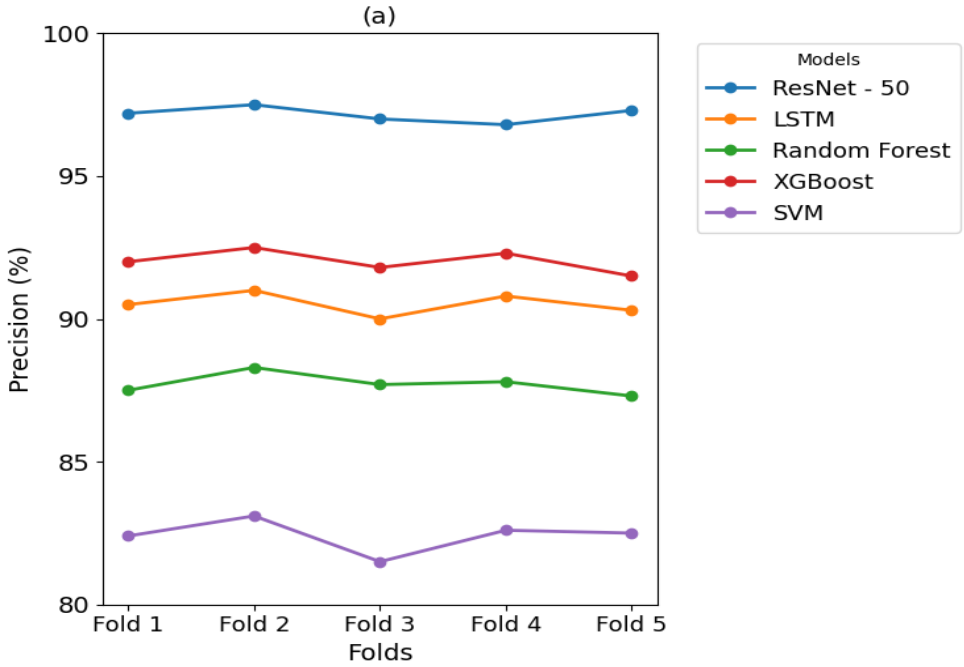


Figure 3. Precision of diverse schemes in enterprise data security detection

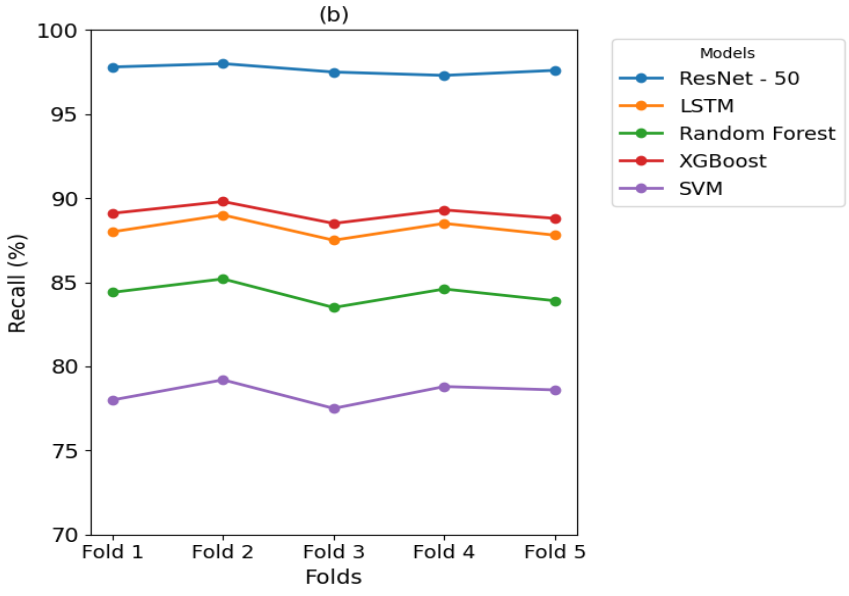


Figure 4. Recall of diverse schemes in enterprise data security detection

Figs. 3 and 4 show that diverse schemes perform differently in enterprise data security detection. The improved ResNet-50 model in this paper maintains high performance regarding precision and recall. The precision of the ResNet-50 model in the five folds is 97.2%, 97.5%, 97.0%, 96.8%, and 97.3%, respectively, with an average precision of 97.2% and an average recall of 97.6%. It performs well in dealing with large-scale traffic anomalies such as DDoS attacks. It can accurately identify features such as the source IP address distribution and traffic intensity changes of attack traffic. The average precision of LSTM in the five folds is 90.5%, and the average recall is 88.2%. It has advantages in processing time series data, but its ability to extract complex spatial features is weak, and it is easy to misjudge and miss port scanning attacks. Random forest's average precision and recall are 87.7% and 84.3%, respectively. The construction of multiple decision trees enables it to tolerate data noise. Still, its ability to handle high-dimensional complex relationships is limited, and it is easy to misjudge the distinction between normal and disguised traffic. XGBoost's average precision reaches 92.0%, and the average recall is 89.1%. It handles structured data well, but its feature learning ability is still not as good as the improved ResNet-50 in this paper, and it may miss the detection of novel encrypted malware data theft. The average precision of SVM is 82.4%, and the average recall is only 78.4%. It can still handle linearly separable data, but in the enterprise data security detection scenario, the computational cost of nonlinear complex data is high, and it is greatly affected by the kernel function parameters, making it challenging to classify complex traffic and variable behaviors accurately.

4.3. AUC of Diverse Schemes

This paper calculates the false positive rate (FPR) and true positive rate (TPR) of different methods at different thresholds. The 5-fold cross-validation is still used. The receiver operating characteristic (ROC) curve is plotted, and the area under the curve (AUC) is computed:

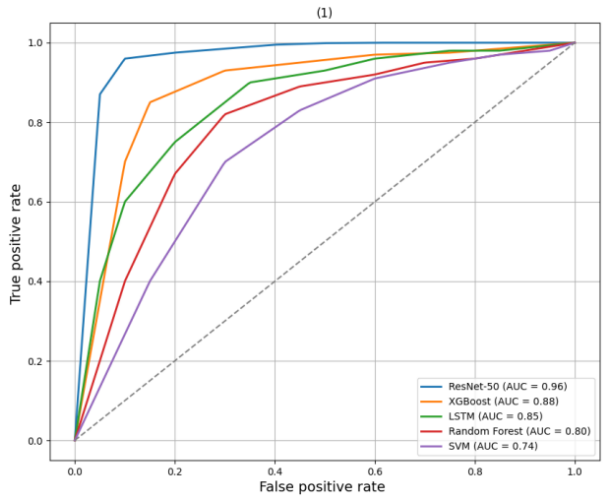
$$AUC = \int_0^1 TPR(f)df \tag{24}$$

TPR(f) is the true positive rate under different threshold f. The closer the value of AUC is to 1, the better the scheme's classification performance. Fig. 5 presents the ROC curve of different methods at different folds.

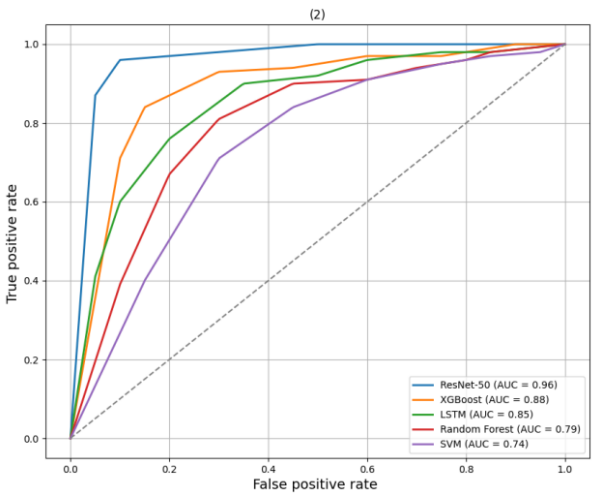
Figure 5 shows the ROC curves of five different classification schemes tested on five different folds. The general trend indicates a high level of classification performance; however, some differences can be noted. In Fold 1 (Fig. 5a), Scheme A has the highest true positive rate at almost all threshold values, leading to a significantly larger AUC compared to the others. In Fold 2 (Fig. 5b), Scheme B shows somewhat better performance with a steeper curve closer to the upper-left corner of the plot, indicating higher sensitivity. Folds 3 and 4 (Figs. 5c and 5d) show similar levels of performance among the three schemes, with some differences in the curvature of the ROC that indicate similar model responses to various data splits. In Fold 5 (Fig. 5e), the differences among the curves become even more pronounced, with Scheme D performing worse than the others, indicating a lack of generalization capacity. Overall, while some schemes rank higher consistently across the different folds, the variations noted in AUC values and the ROC curve shapes highlight the impact of data partitioning on classifier performance and ranking.

The ROC curve of the improved ResNet-50 model is closer to the upper left corner than that of the other four schemes, indicating that it can achieve a higher TPR at a lower FPR, effectively improving classification accuracy. In addition, the average AUC of the ResNet-50 model reaches 0.96, which is much higher than other schemes. In comparison, the average AUC of the XGBoost model is 0.88, and the average AUC of the LSTM model is 0.85. Although they perform well, they are still not as good as ResNet-50. The performance of random forest and SVM is relatively inferior, with average AUCs of 0.80 and 0.74, respectively, which are significantly lower than that of ResNet-50, showing that they are limited in effectiveness in dealing with complex network attack scenarios.

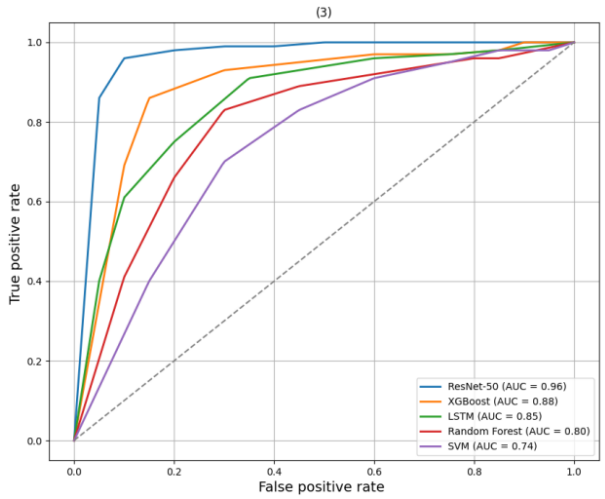
For future work, a merged or averaged ROC curve may be presented to provide a consolidated performance comparison.



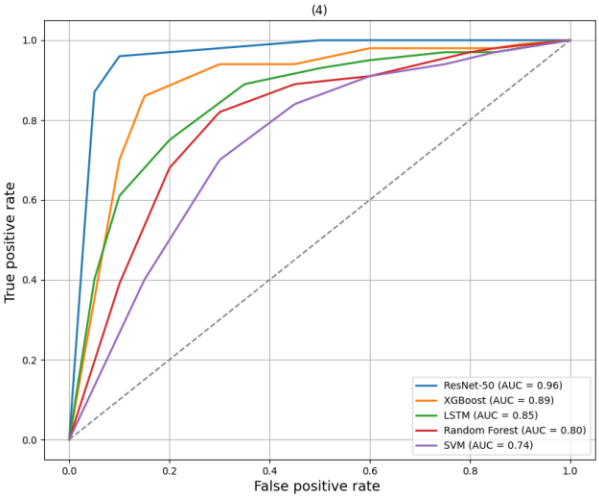
(a) ROC curves of five diverse schemes at Fold (1)



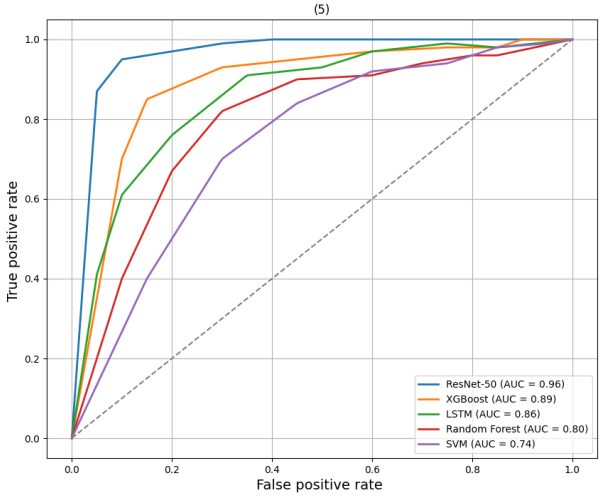
(b) ROC curves of five diverse schemes at Fold (2)



(c) ROC curves of five diverse schemes at Fold (3)



(d) ROC curves of five diverse schemes at Fold (4)



(e) ROC curves of five diverse schemes at Fold (5)

Figure 5. ROC curves of five diverse schemes at five folds

5. Discussion

The enterprise data security management system based on ResNet-50 and IoT technology proposed in this paper performs well in detecting multiple attack behaviors, especially the recognition accuracy of DDoS attacks reaches 98.6%, the recognition accuracy of malware data leakage reaches 96.2%, and the average response time is 1.03 seconds. These results show the system has significant advantages in dealing with

real-time threats in complex network environments. By combining the optimized ResNet-50 model and the distributed IoT data collection platform, the system can dynamically evaluate the enterprise network security status and quickly generate alerts based on contextual information. In addition, compared with traditional methods, the AUC value of this system reaches 0.96, which is significantly better than other models, further verifying its superiority in high-dimensional heterogeneous data processing.

However, despite the encouraging experimental results, the system's limitations cannot be ignored. First, deep learning models have high requirements for computing resources and may put pressure on enterprise infrastructure when processing large-scale real-time data streams. This increases hardware costs and may cause latency issues, especially when network bandwidth is limited or device performance is insufficient. Secondly, although the weighted cross-entropy loss function is used in this paper to alleviate the class imbalance problem, the detection accuracy of the model for the minority class still has room for improvement in the case of extreme class imbalance. This phenomenon is particularly critical when facing low-frequency but high-risk attacks, because even a single missed report can have serious consequences.

To solve the above problems, federated learning and edge computing provide potential solutions. Federated learning allows multiple companies to train models locally and share only model parameters, not raw data, thereby improving model performance while protecting privacy. This approach is particularly suitable for inter-enterprise cooperation scenarios, such as sharing threat intelligence in supply chain networks. However, federated learning also faces challenges, such as ensuring the data quality and consistency of the participants and dealing with the model convergence problems caused by heterogeneous data distribution. These issues need to be addressed in future research.

Edge computing is another direction worth exploring, as it can significantly reduce the system's computing resource requirements and improve response speed. By deploying part of the model on edge devices close to the data source, the need to transmit data to the cloud can be reduced, reducing communication delays and bandwidth consumption. At the same time, edge computing can also enhance the robustness of the system, and edge devices can still perform threat detection tasks independently even when network connections are interrupted. However, edge devices usually have limited computing power and storage capacity, so it is key to develop lightweight deep learning architectures to achieve efficient reasoning in resource-constrained environments.

Finally, the long-term stability and adaptability of the system are also issues that need attention. With the continuous emergence of new attack methods, the model may face the risk of obsolescence. Therefore, dynamic updates and online learning capabilities will become an important direction for future system improvements. Combining continuous learning technology allows the system to gradually adapt to new threats without retraining the entire model, thereby maintaining high detection accuracy and response speed. In summary, the method proposed in this paper provides a strong starting point for enterprise data security management, but its limitations also point out the direction for future research.

6. Conclusions

This paper implements an enterprise data security management system based on ResNet-50 and IoT technology, aiming to address the data security challenges faced by enterprises in digital transformation. The system acquires network traffic, device status and log data in real time by building a distributed IoT data collection platform, and uses the optimized ResNet-50 model to analyze high-dimensional data streams, thereby dynamically evaluating the network security status of the enterprise. Experimental results show that the system performs well in identifying various attack behaviors, with an accuracy of 98.6% for DDoS attack identification and 96.2% for malware data leakage identification, and an average response time of 1.03 seconds, significantly better than traditional methods. However, this study still has shortcomings. The deep learning model has high requirements for computing resources, which may put pressure on the enterprise infrastructure when processing large-scale real-time data, and there is still room for performance improvement in extreme class imbalance. Future research can explore lightweight deep learning architectures to adapt to the application scenarios of edge devices, and combine federated learning and edge computing technologies to further improve privacy protection and real-time processing capabilities, provide enterprises with more intelligent and comprehensive data security management solutions, and promote the development of information security technology.

CRediT Author Contribution Statement

Fan Gao: Writing – original draft, Conceptualization.

Acknowledgement

This exploration was backed in part by the Soft Science Research Program of Henan Province: Research on the construction of a new business model of small and medium-sized enterprises in Henan province under the trend of digital economy (Project No: 242400410352).

References

[1] Loso Judijanto, Djarot Hindarto and Sentot I. Wahjono, "Edge of enterprise architecture in addressing cyber security threats and business risks", *International Journal Software Engineering and Computer Science (IJSECS)*, Print ISSN: 2776-4869, Online ISSN: 2776-3242, Vol. 3, No. 3, pp. 386–396, 20 December 2023, Published by Lembaga KITA, DOI: 10.35870/ijsecs.v3i3.1816, Available: https://journal.lembagakita.org/ijsecs/article/view/1816.

[2] Minzhao Lyu, Hassan Habibi Gharakheili and Vijay Sivaraman, "A survey on enterprise network security: Asset behavioral monitoring and distributed attack detection", *IEEE Access*, Online ISSN: 2169-3536, Vol. 12, 26 June 2024, Published by IEEE, DOI: 10.1109/ACCESS.2024.3419068, Available: https://ieeexplore.ieee.org/abstract/document/10571950.

[3] Hazheen S. Mahmood, Dildar M. Abdulqader, Rozin M. Abdullah, Halbast Rasheed, Zryan N. R. Ismael *et al.*, "Conducting In-Depth Analysis of AI, IoT, Web Technology, Cloud Computing, and Enterprise Systems Integration for Enhancing Data Security and Governance to Promote Sustainable Business Practices", *Journal of Information Technology and Informatics*, Print ISSN: 0268-3962, Online ISSN: 1466-4437, Vol. 3, No. 2, August 2024, Published by SAGE Publications, DOI: 10.14569/ijacsa.2024.0141231, Available: https://thesai.org/Publications/ViewPaper?Code=IJACSA&Issue=2&SerialNo=31&Volume=3.

[4] Amogh Deshmukh and Kiran Ravulakollu, "An efficient CNN-based intrusion detection system for IoT: Use case towards cybersecurity", *Technologies*, Online ISSN: 2227-7080, Vol. 12, No. 10, p. 203, 17 October 2024, Published by MDPI, DOI: 10.3390/technologies12100203, Available: https://www.mdpi.com/2227-7080/12/10/203.

[5] Geeta Kocher and Gulshan Kumar, "Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges", *Soft Computing*, Print ISSN: 1432-7643, Online ISSN: 1433-7479, Vol. 25, No. 15, pp. 9731–9763, 24 June 2021, Published by Springer Nature, DOI: 10.1007/s00500-021-05893-0, Available: https://link.springer.com/article/10.1007/s00500-021-05893-0.

[6] Yulin Wang, Jinheng Wang and Honglin Jin, "Network intrusion detection method based on improved CNN in internet of things environment", *Mobile Information Systems*, Print ISSN: 1574-017X, Online ISSN: 1875-905X, Vol. 2022, No. 1, p. 3850582, 8 June 2022, Published by WILEY Online Library, DOI: 10.1155/2022/3850582, Available: https://onlinelibrary.wiley.com/doi/full/10.1155/2022/3850582.

[7] Ming Wan, Jiawei Li, Ying Liu, Jianming Zhao and Jiushuang Wang, "Characteristic insights on industrial cyber security and popular defense mechanisms", *China Communications*, Print ISSN: 1673-5447, Vol. 18, No. 1, 28 pp. 130–150, January 2021, Published by IEEE, DOI: 10.23919/JCC.2021.01.012, Available: https://ieeexplore.ieee.org/abstract/document/9339836.

[8] Rasheed Ahmad, Izzat Alsmadi, Wasim Alhamdani and Lo'ai Tawalbeh, "Zero-day attack detection: a systematic literature review", *Artificial Intelligence Review*, Print ISSN: 0269-2821, Online ISSN: 1573-7462, Vol. 56, No. 10, pp. 10733–10811, 27 February 2023, Published by Springer Nature, DOI: 10.1007/s10462-023-10437-z, Available: https://link.springer.com/article/10.1007/s10462-023-10437-z.

[9] Muritala Aminu, Ayokunle Akinsanya, Dickson A. Dako and Oyewale Oyedokun, "Enhancing cyber threat detection through real-time threat intelligence and adaptive defense mechanisms", *International Journal of Computer Applications Technology and Research*, Online ISSN: 2319–8656, Vol. 13, No. 8, pp. 11–27, July 2024, Published by IJCAT, DOI: 10.7753/IJCATR1308.1002, Available: https://ijcat.com/archieve/volume13/issue8/ijcatr13081002.

[10] Ahmed Nassar and Mostafa Kamal, "Machine Learning and Big Data analytics for Cybersecurity Threat Detection: A Holistic review of techniques and case studies", *Journal of Artificial Intelligence and Machine Learning in Management*, Online ISSN: 2960-2068, Vol. 5, No. 1, pp. 51–63, 6 February 2021, Published by SageScience, Available: https://journals.sagescience.org/index.php/jamm/article/view/97.

[11] Xuan-Ha Nguyen, Xuan-Duong Nguyen, Hoang-Hai Huynh and Kim-Hung Le, "Realguard: A lightweight network intrusion detection system for IoT gateways", *Sensors*, Online ISSN: 1424-8220, Vol. 22, No. 2, p. 432, 7 January 2022, Published by MDPI, DOI: 10.3390/s22020432, Available: https://www.mdpi.com/1424-8220/22/2/432.

[12] Mohammed Subhi, Omar F. Rashid, Safa A. Abdulsahib, Mohammed K. Hussein and Saleh M. Mohammed, "Anomaly Intrusion Detection Method based on RNA Encoding and ResNet50 Model", *Mesopotamian Journal of CyberSecurity*, Online ISSN: 2958-6542, Vol. 4, No. 2, pp. 120–128, 28 August 2024, Published by Mesopotamian Academic Press, DOI: 10.58496/MJCS/2024/011, Available: https://journals.mesopotamian.press/index.php/CyberSecurity/article/view/403.

[13] Mary A. EA, "A Novel Paradigm for IoT Security: ResNet-GRU Model Revolutionizes Botnet Attack Detection", *International Journal of Advanced Computer Science & Applications*, Online ISSN: 2158-107X, Vol. 14, No. 12, 13 December 2023, Published by IJACSA, DOI: 10.14569/ijacsa.2023.0141231, Available: A Novel Paradigm for IoT Security: ResNet-GRU Model Revolutionizes Botnet Attack Detection.

[14] Swathy Akshaya and Padmavathi Ganapathi, "Augmenting Cyber Defense Counter to Zero-Day Attacks Through Predictive Analysis-A Fusion Methodology Assimilating Game Theory and RESNet Inspired Optimization Techniques", *International Journal of Communication Networks and Information Security*, Online ISSN: 2073-607X, 2076-0930, Vol. 16, No. 3, pp. 91–104, September 2024, Published by IJCNIS, DOI: 10.17762/ijcnis.v16i3.6712, Available: https://www.ijcnis.org/index.php/ijcnis/article/view/6712.

[15] Asma Shaikh and Preeti Gupta, "Real-time intrusion detection based on residual learning through ResNet algorithm", *International Journal of System Assurance Engineering and Management*, Print ISSN: 0975-6809, Online ISSN: 0976-4348, pp. 1–15, Vol. 13, 4 January 2022, Published by Springer Nature, DOI: 10.1007/s13198-021-01558-1, Available: https://link.springer.com/article/10.1007/s13198-021-01558-1.

[16] Benhui Xia, Dezhi Han, Ximing Yin and Gao Na, "RICNN: a ResNet&Inception convolutional neural network for intrusion detection of abnormal traffic", *Computer Science and Information Systems*, Print ISSN: 1820-0214, Online ISSN: 2406-1018, Vol. 19, No. 1, January 2022, pp. 309–326, Published by ComSIS, DOI: 10.2298/CSIS210617055X, Available: https://doiserbia.nb.rs/Article.aspx?id=1820-02142100055X.

[17] Nour Moustafa, Nickolaos Koroniotis, Marwa Keshk, Albert Y. Zomaya and Zahir Tari, "Explainable intrusion detection for cyber defences in the internet of things: Opportunities and solutions", *IEEE Communications Surveys & Tutorials*, Online ISSN: 1553-877X, Vol. 25, No. 3, pp. 1775–1807, 26 May 2023, Published by IEEE, DOI: 10.1109/COMST.2023.3280465, Available: https://ieeexplore.ieee.org/abstract/document/10136827.

[18] Ines Martins, Joao S. Resende, Patricia R. Sousa, Simao Silva, Luis Antunes *et al.*, "Host-based IDS: A review and open issues of an anomaly detection system in IoT", *Future Generation Computer Systems*, Print ISSN: 0167-739X, Online ISSN: 1872-7115, Vol. 133, pp. 95–113, August 2022, Published by Elsevier, DOI: 10.1016/j.future.2022.03.001, Available: https://www.sciencedirect.com/science/article/abs/pii/S0167739X22000760.

[19] Wuchao Liang, Wenning Li and Lili Feng, "Information security monitoring and management method based on big data in the internet of things environment", *IEEE Access*, Online ISSN: 2169-3536, Vol. 9, pp. 39798–39812, 8 March 2021, Published by IEEE, DOI: 10.1109/ACCESS.2021.3064350, Available: https://ieeexplore.ieee.org/abstract/document/9371683.

[20] Dhuha S. Ghazi, Hamood S. Hamid, Mhammed J. Zaiter and Ahmed S. G. Behadili, "Snort Versus Suricata in Intrusion Detection", *Iraqi Journal of Information and Communication Technology*, Print ISSN: 2222-758X, Online ISSN: 2789-7362, Vol. 7, No. 2, pp. 73–88, 30 August 2024, Published by College of Information Engineering, DOI: 10.31987/ijict.7.2.290, Available: https://ijict.edu.iq/index.php/ijict/article/view/290.

[21] Fatima R. Alzaabi and Abid Mehmood, "A review of recent advances, challenges, and opportunities in malicious insider threat detection using machine learning methods", *IEEE Access*, Online ISSN: 2169-3536, Vol. 12, pp. 30907–30927, 26 February 2024, Published by IEEE, DOI: 10.1109/ACCESS.2024.3369906, Available: https://ieeexplore.ieee.org/abstract/document/10445123.

[22] Zahra A. Varzaneh and Marjan K. Rafsanjani, "Intrusion detection system using a new fuzzy rule-based classification system based on genetic algorithm", *Intelligent Decision Technologies*, Online ISSN: 1875-8843, Vol. 15, pp. 231–237,No. 2, 1 May 2021, Published by Sage Journals, DOI: 10.3233/IDT-200036, Available: https://journals.sagepub.com/doi/abs/10.3233/IDT-200036.

[23] Sushil Kumar, "MCFT-CNN: Malware classification with fine-tune convolution neural networks using traditional and transfer learning in Internet of Things", *Future Generation Computer Systems*, Print ISSN: 0167-739X, Online ISSN: 1872-7115, Vol. 125, pp. 334–351, December 2021, Published by Elsevier, DOI: 10.1016/j.future.2021.06.029, Available: https://www.sciencedirect.com/science/article/abs/pii/S0167739X21002247.

[24] K. Naresh K. Thapa and N. Duraipandian, "Malicious traffic classification using long short-term memory (LSTM) model", *Wireless Personal Communications*, Print ISSN: 0929-6212, Online ISSN 1572-834X, Vol. 119, No. 3, pp. 2707–2724, 13 March 2021, Published by Springer Nature, DOI: 10.1007/s11277-021-08359-6, Available: https://link.springer.com/article/10.1007/s11277-021-08359-6.

[25] Bei Lu, Nurbol Luktarhan, Chao Ding and Wenhui Zhang, "ICLSTM: encrypted traffic service identification based on inception-LSTM neural network", *Symmetry (Basel)*, Online ISSN: 2073-8994, Vol. 13, No. 6, p. 1080, 17 June 2021, Published by MDPI, DOI: 10.3390/sym13061080, Available: https://www.mdpi.com/2073-8994/13/6/1080.

[26] Kezhou Ren, Yifan Zeng, Zhiqin Cao and Yingchao Zhang, "ID-RDRL: a deep reinforcement learning-based feature selection intrusion detection model", *Scientific Reports*, Online ISSN: 2045-2322, Vol. 12, No. 1, p. 15370, 13 September 2022, Published by Springer Nature, DOI: 10.1038/s41598-022-19366-3, Available: https://www.nature.com/articles/s41598-022-19366-3.

[27] Meng Xu, Shenghan Liu and Xuewu Li, "Network security situation assessment and prediction method based on multimodal transformation in edge computing", *Computer Communications*, Print ISSN: 0140-3664, Online ISSN: 1873-703X, Vol. 215, pp. 103–111, 1 February 2024, Published by Elsevier, DOI: 10.1016/j.comcom.2023.12.014, Available: https://www.sciencedirect.com/science/article/abs/pii/S0140366423004498.

[28] Xiaokang Zhou, Wei Liang, Weimin Li, Ke Yan, Shohei Shimizu *et al.*, "Hierarchical adversarial attacks against graph-neural-network-based IoT network intrusion detection system", *IEEE Internet of Things Journal*, Online ISSN: 2327-4662, Vol. 9, No. 12, pp. 9310–9319, 15 June 2021, Published by IEEE, DOI: 10.1109/JIOT.2021.3130434, Available: https://ieeexplore.ieee.org/abstract/document/9626144.

[29] Ming Xia, Zunkai Huang, Li Tian, Hui Wang, Victor Chang *et al.*, "SparkNoC: An energy-efficiency FPGA-based accelerator using optimized lightweight CNN for edge computing", *Journal of Systems Architecture*, Print ISSN: 1383-7621, Online ISSN: 1873-6165, Vol. 115, p. 101991, May 2021, DOI: 10.1016/j.sysarc.2021.101991, Available: https://www.sciencedirect.com/science/article/abs/pii/S1383762121000138.

[30] Syreen Banabilah, Moayad Aloqaily, Eitaa Alsayed, Nida Malik and Yaser Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications", *Information Processing & Management*, Print ISSN: 0306-4573, Online ISSN: 1873-5371, Vol. 59, No. 6, p. 103061, November 2022, Published by Elsevier, DOI: 10.1016/j.ipm.2022.103061, Available: https://www.sciencedirect.com/science/article/abs/pii/S0306457322001649.

[31] Hongshuo Liang, Erlu He, Yangyang Zhao, Zhe Jia and Hao Li, "Adversarial attack and defense: A survey", *Electronics*, Online ISSN: 2079-9292, Vol. 11, No. 8, p. 1283, 18 April 2022, Published by MDPI, DOI: 10.3390/electronics11081283, Available: https://www.mdpi.com/2079-9292/11/8/1283.

[32] Hongbin Sun and Shizhen Bai, "Enterprise information security management using internet of things combined with artificial intelligence technology", *Computational Intelligence and Neuroscience*, Print ISSN:1687-5265, Online ISSN:1687-5273, Vol. 2022, No. 1, p. 7138515, 14 June 2022, Published by Wiley Online Library, DOI: 10.1155/2022/7138515, Available: https://onlinelibrary.wiley.com/doi/full/10.1155/2022/7138515.

[33] Junping Yue, "A deep learning method for intelligent decision-making in enterprise management based on the Internet of Things", *Journal of Computational Methods in Science and Engineering*, Online ISSN: 1875-8983, Vol. 23, No. 2, pp. 617–627, 1 March 2023, Published by Sage Publications, DOI: 10.3233/JCM-226613, Available: https://journals.sagepub.com/doi/abs/10.3233/JCM-226613.

[34] Sk Tanzir Mehedi, Adnan Anwar, Ziaur Rahman, Kawsar Ahmed and Rafiqul Islam, "Dependable intrusion detection system for IoT: A deep transfer learning based approach", *IEEE Transactions on Industrial Informatics*, Print ISSN: 1551-3203, Online ISSN: 1941-0050, Vol. 19, No. 1, pp. 1006–1017, 11 April 2022, Published by IEEE, DOI: 10.1109/TII.2022.3164770, Available: https://ieeexplore.ieee.org/abstract/document/9749858.

[35] V. Gowdhaman and R. Dhanapal, "Hybrid deep learning-based intrusion detection system for wireless sensor network", *International Journal of Vehicle Information and Communication Systems*, Print ISSN: 1471-0242, Online ISSN: 1741-8208, Vol. 9, No. 3, pp. 239–255, 5 July 2024, Published by INDERSCIENCE, DOI: 10.1504/IJVICS.2024.139627, Available: https://www.inderscienceonline.com/doi/abs/10.1504/IJVICS.2024.139627.

[36] Hamza Kheddar, Yassine Himeur and Ali I. Awad, "Deep transfer learning for intrusion detection in industrial control networks: A comprehensive review", *Journal of Network and Computer Applications*, Print ISSN: 1084-8045, Online ISSN: 1095-8592, Vol. 220, p. 103760, November 2023, Published by Elsevier, DOI: 10.1016/j.jnca.2023.103760, Available: https://www.sciencedirect.com/science/article/abs/pii/S1084804523001790.