Artificial Intelligence-Based Anomaly Detection for Large-Scale Web Data Security Monitoring

Siyao Xu*, Yan Li, Kai Zhang, Weiming Li and Jieshao Lai

Electric Power Research Institute, Guangdong Power Grid Co. Ltd., Guangzhou 510000, Guangdong, China
E-mail: xusiyao_work@yeah.net
**Corresponding Author*

Received 29 January 2026; Accepted 11 March 2026

Abstract

With the rapid development of the World Wide Web and the popularity of Internet applications, the generation and exchange of data have exploded. Large-scale data generation and transmission also bring severe security challenges. In response to the problems that existing anomaly detection methods are difficult to jointly model the semantic context and temporal dependencies in non-encrypted scenarios, and that single-modal feature information is insufficient in encrypted scenarios, resulting in limited detection accuracy, this study proposes two artificial intelligence anomaly detection methods that are adapted to different scenarios. For non-encrypted/low-encrypted scenarios, a BERT-LSTM-TextCNN parallel fusion architecture is proposed. This architecture extracts high-order semantic features, long-term dependency features, and multi-scale local features through parallel branches, and achieves complementary enhancement of multi-perspective

Journal of Web Engineering, Vol. 25_5, 915–944.
doi: 10.13052/jwe1540-9589.2557

information through feature concatenation, effectively solving the problem of difficult collaborative modeling of multiple types of features in non-encrypted scenarios. For multi-encrypted scenarios, a detection method based on improved ResNet and cross-modal feature fusion is proposed. Different from traditional methods that only rely on deep learning features, the study adaptively weights and fuses the deep semantic features extracted by ResNet with flow statistics features and temporal features and optimizes the fusion weights through a learnable random forest, breaking through the bottleneck of insufficient single-modal feature information in encrypted traffic. The precision reached 97.18%, the recall rate reached 95.26%, and the F1-score reached 96.21%. The AUC values were all greater than 0.97, the false positive rate was 8.12% lower than the traditional method, and the single-batch data detection time was only 37.25 s. In the multi-encryption scenario, the precision, recall rate and F1-score of the cross-modal feature fusion method were 98.48%, 87.30% and 92.57%, respectively. This effectively solves the detection limitations caused by feature ambiguity in encrypted environments. In summary, the artificial intelligence anomaly detection method effectively improves detection accuracy and efficiency and provides a feasible technical path for building a comprehensive World Wide Web data security monitoring system.

Keywords: Artificial intelligence, anomaly detection, web data, network security monitoring, deep learning.

1 Introduction

Driven by Internet technology and World Wide Web (Web) applications, the amount of global data is growing at an unprecedented rate. The explosive growth of Web data not only promotes information sharing and business innovation but also brings unprecedented challenges to network security [1, 2]. Abnormal behaviors hidden in Web traffic data, such as network attacks, data leaks, and illegal access, pose serious threats to personal privacy, corporate assets, and even national security [3, 4]. Therefore, how to effectively detect abnormal behaviors from large-scale Web data has become particularly important. Current traditional anomaly detection technology mainly relies on rule matching, statistical analysis and machine learning methods. However, as network attack methods continue to evolve and diversify, traditional detection methods have gradually exposed their limitations [5, 6]. Firstly, rule-based methods are difficult to adapt to new attack modes and are prone to produce

many false positives and false negatives. Secondly, shallow statistical and machine learning models often fail to capture complex patterns and high-order semantic information in the data, leading to insufficient detection accuracy [7, 8].

To optimize the detection accuracy and efficiency of large-scale Web data, for abnormal network traffic data in non-encrypted or low-encrypted scenarios, a method that fuses the Bidirectional Encoder Representation from Transformers (BERT), Short-term Memory (LSTM) and Text Convolutional Neural Network (TextCNN) is built. The study combines the improved ResNet model and cross-modal feature fusion to effectively monitor Web data in multi-encrypted environments. The innovation points of this research are as follows: (1) A parallel fusion architecture of BERT-LSTM-TextCNN is proposed. Different from the existing serial fusion or single model, this architecture extracts semantic features, temporal dependencies, and local patterns through parallel branches, and achieves complementary enhancement of multi-perspective information through feature concatenation, effectively solving the problem that multiple types of features are difficult to be jointly modeled in non-encrypted scenarios. (2) An encryption traffic detection method based on improved ResNet and cross-modal feature fusion is designed. For encrypted scenarios, deep semantic features, statistical features, and temporal features are adaptively weighted fused, and the fusion weights are optimized through a learnable random forest, breaking through the limitation of traditional methods that rely on a single modal feature, and significantly improving the detection accuracy in encrypted environments. (3) A unified detection framework covering both non-encrypted and encrypted scenarios is constructed, and the robustness of the model under different data scales, different encryption algorithms, and different attack types is systematically evaluated, providing comprehensive performance references for practical deployment.

2 Related Works

Currently, scholars have conducted research on network data security monitoring. Al-Quayed et al. built a predictive intrusion detection and defense model relying on situational awareness in view of the network security intrusion threats faced by wireless sensor networks in the context of Industry 4.0. This model could effectively optimize the accuracy and response speed of intrusion detection [9]. Piplai et al. proposed a knowledge-enhanced neural symbolic artificial intelligence framework that integrated knowledge

graphs, deep learning, and symbolic reasoning to detect and explain cyber threats. The results showed that this method significantly improved the interpretability of threat intelligence and the adaptability of the model to complex attacks while maintaining high detection accuracy and provided a more transparent and inferable defense method for network security [10]. Omer et al. designed a probabilistic neural network intrusion detection and classification framework optimized by the firefly algorithm in view of the current situation where network intrusions occur frequently due to the surge in network dependence and it was difficult for traditional detection methods to balance speed and accuracy. The framework had a comprehensive accuracy of 98.99% on the KDD-CUP 99 data set, and was superior to existing methods in precision, recall, F1-score and other indicators, which could achieve rapid and accurate identification and classification of multiple types of attacks [11]. Faced with "algorithm-key" double transparency, Pleshakova et al. proposed an active defense network security paradigm with prompt engineering as the traction and large language model self-learning password as the core. The first inference success rate of this paradigm in complex tasks was increased by more than 30% compared with the internal benchmark [12]. Aiming at the current situation where financial fraud techniques are rapidly evolving, traditional rule systems are difficult to capture new complex patterns, are sensitive to noise, and have high false positive/negative costs, Udayakumar et al. proposed a deep learning framework integrating noise reduction modules to simultaneously detect and finely classify network security threats and financial fraud. The framework had an accuracy of 93.35% and a precision of 98.85% on large real transaction data sets. It could provide low-noise, high-sensitivity, and easy-to-deploy real-time risk control support for scenarios with high security requirements such as banks, e-commerce, and online payments [13].

Although the aforementioned research has yielded favorable results, due to the huge amount of Web data and the increasingly widespread application of encryption technology, the detection of encrypted traffic has become more difficult [14, 15]. Meanwhile, as network attack methods are constantly updated, traditional detection methods are difficult to adapt to new attack modes. Based on this research, the BERT-LSTM-TextCNN and the ResNet model combined with cross-modal feature fusion are proposed for large-scale Web data security monitoring, aiming to optimize the detection accuracy and efficiency.

3 Anomaly Detection Methods for Large-Scale Web Data Based on Artificial Intelligence

3.1 Web Page Data Anomaly Detection Model Based on BERT-LSTM-TextCNN

Currently, in anomaly detection of large-scale web data, traditional detection methods rely on rule matching or shallow feature extraction, which struggle to deal with complex and changeable network attack patterns. Based on this research, a hybrid model BERT-LSTM-TextCNN is first proposed and applied to detect abnormal network traffic data in non-encrypted or low-encrypted scenarios. Before performing anomaly detection on web page data, the original web traffic data should be effectively preprocessed, as presented in Figure 1.

In Figure 1, large-scale web traffic data are first collected from multiple data sources, including normal traffic and abnormal traffic. During the data cleaning process, duplicate data are mainly removed, mean filling is used to deal with missing values and invalid data are filtered. To make the data comparable, the study uses Z-score for data standardization. Finally, based on the characteristics of Web traffic data, features related to anomaly detection are extracted. For feature extraction, the study cites a two-stage

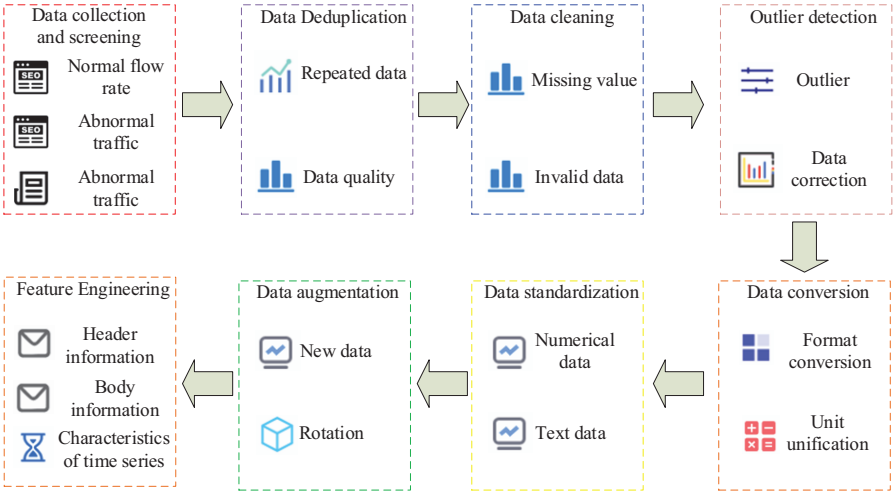


Figure 1 Data preprocessing process.

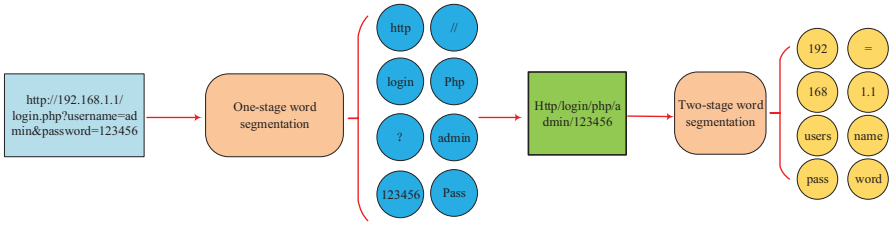


Figure 2 Schematic diagram of two-stage word segmentation.

word segmentation extraction method, and the relevant diagram is shown in Figure 2.

In the two-stage word segmentation extraction shown in Figure 2, the first-stage word segmentation will identify and separate each component of the URL. The second stage is to further process on the basis of the first stage, and combine the lexical units obtained by word segmentation in the first stage into new semantic units [16, 17]. In the simple splitting method based on delimiters, it merely mechanically splits the URL using non-alphanumeric characters. Although this method is simple to implement and has low computational cost, it has two inherent drawbacks: semantic fragmentation and loss of structural information. The first stage of the two-stage word segmentation also performs atomic unit splitting, but the second stage recombines these fragmented units into complete semantic blocks through statistical co-occurrence and semantic templates. At the same time, it retains the structured information of parameters through parameter generalization, thereby providing higher-quality input for subsequent semantic understanding. After obtaining the data features, the study uses one-hot encoding to convert the text features into word embedding vectors and imports them as input into the BERT-LSTM-TextCNN model [18]. The basis of the BERT-LSTM-TextCNN is the pre-trained BERT model and LSTM-TextCNN model. BERT is pre-trained on much text data and is able to capture rich language patterns and semantic information. Figure 3 presents the relevant structure of the BERT.

In Figure 3, its input is a word embedding vector, including high-dimensional vector representation of word embedding, position embedding and paragraph embedding. The BERT output is presented in Equation (1):

$$H_{BERT} = BERT(X_i) \tag{1}$$

In Equation (1), X_i signifies the input word embedding vector and H_{BERT} signifies the output of the BERT model. Depending on the high-order semantic features of the BERT model, the research further uses LSTM

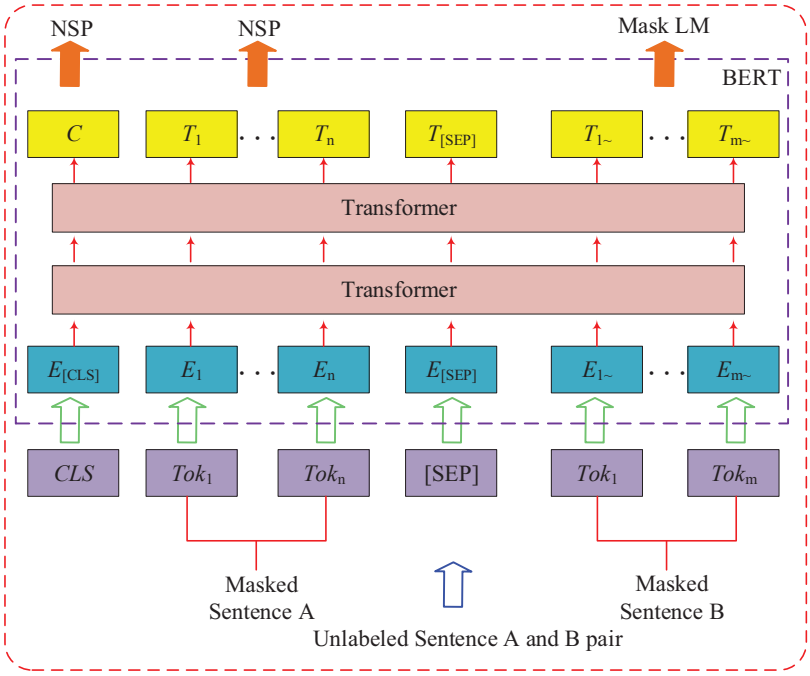


Figure 3 Schematic diagram of the BERT structure.

and TextCNN models to extract high-order semantic features and multi-scale local features in Web traffic data. Among them, high-order semantic features refer to the deep semantic information beyond the literal meaning in the text data. In Web traffic data, a simple URL string or protocol field value can only provide shallow information, while high-order semantic features can capture the contextual relationships and implicit intentions between these strings. Multi-scale local features refer to the key patterns extracted from different length segments of the text sequence. In Web requests, attack features may appear in consecutive character sequences of different lengths. The LSTM is a special recurrent neural network that can capture long-term dependencies in sequence data. Considering that not all time-steps of traditional LSTM are equally important to the final prediction in some cases, the research introduces an attention mechanism into LSTM so that the model can adaptively emphasize important steps. The attention mechanism can be expressed as a weighted summation process, where the weight reflects the importance of each time-step [19, 20]. The specific attention mechanism weight involves a score function, and the output of the score function is normalized by

SoftMax to ensure that the sum of all weights is 1. The relevant mathematical expression is shown in Equation (2):

$$\begin{cases} e_t = v^T \tanh(W_h h_t + W_s s) \\ \alpha_t = \dfrac{\exp(e_t)}{\sum_{i=1}^{T} \exp(e_i)} \\ \tilde{h} = \sum_{t=1}^{T} \alpha_t h_t \end{cases} \tag{2}$$

In Equation (2), h_t signifies the LSTM hidden state at t, s represents the learnable query vector, W_h and W_s represent learnable weight matrices, v represents the learnable weight vector, e_t represents the unnormalized attention score, α_t represents attention weight and $\tilde{h}$ represents the LSTM output after weighted summation. After the final LSTM output is obtained, it is finally spliced with the multi-scale local features extracted by TextCNN to further enhance the feature representation ability. TextCNN can extract local features of text data through convolutional layers and pooling layers. To reduce parameter and computational complexity, separable convolution is introduced to optimize. Separable convolution is divided into depth convolution and point-wise convolution. The relevant diagram is shown in Figure 4.

In Figure 4, each input channel of deep convolution applies a convolution kernel independently, that is, the convolution operation of each channel is performed separately, and there is no cross-channel information interaction [21, 22]. Pointwise convolution is a 1×1 convolution operation used to combine features from different input channels. Finally, after combining the pre-trained BERT model with the LSTM and TextCNN models, the BERT-LSTM-TextCNN model for Web page data anomaly detection can be obtained. Finally, the overall framework of Web page data anomaly detection relying on the BERT-LSTM-TextCNN is shown in Figure 5.

In the anomaly detection framework for Web traffic data in non-encrypted or low-encrypted scenarios shown in Figure 5, the input of raw data includes SQL, IP addresses, and commands [23]. Secondly, the second data processing stage includes special string recognition, hidden meaning analysis, and word segmentation. Finally, the abnormal Web page data is detected through the BERT-LSTM-TextCNN model.

In the BERT-LSTM-TextCNN model, the model adopts a parallel structure. The original Web data, after preprocessing and encoding by the BERT

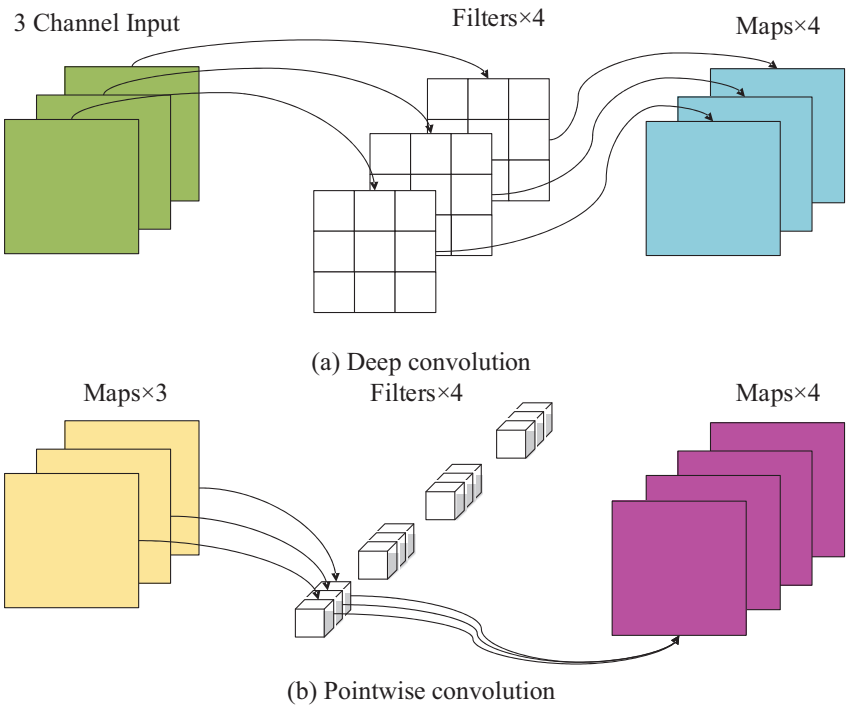


Figure 4 Schematic diagram of separable convolution.

model, generates word vector representations rich in contextual semantics. These representations are simultaneously input into the LSTM branch with attention mechanism and the TextCNN branch. The LSTM branch is responsible for capturing the long-range dependencies of the sequence, while the TextCNN branch extracts local context features through multiple convolutional kernels of different sizes, each with 128 units. Finally, the feature vectors output by the two branches are concatenated to form the final fused features, which are input into the fully connected layer for classification.

In the parameter setting of the model, the LSTM network adopts a 2-layer structure, with a hidden layer dimension of 256 to fully capture the long-distance dependencies of the Web request sequence. The TextCNN branch uses 3 parallel convolutional layers, with convolution kernel sizes of 3, 4, and 5, and each convolutional kernel has 128 units, aiming to capture n-gram features of different granularities. The introduced attention mechanism outputs weights for each time-step of LSTM through a learnable query vector, enabling the model to focus on the key parts of the sequence.

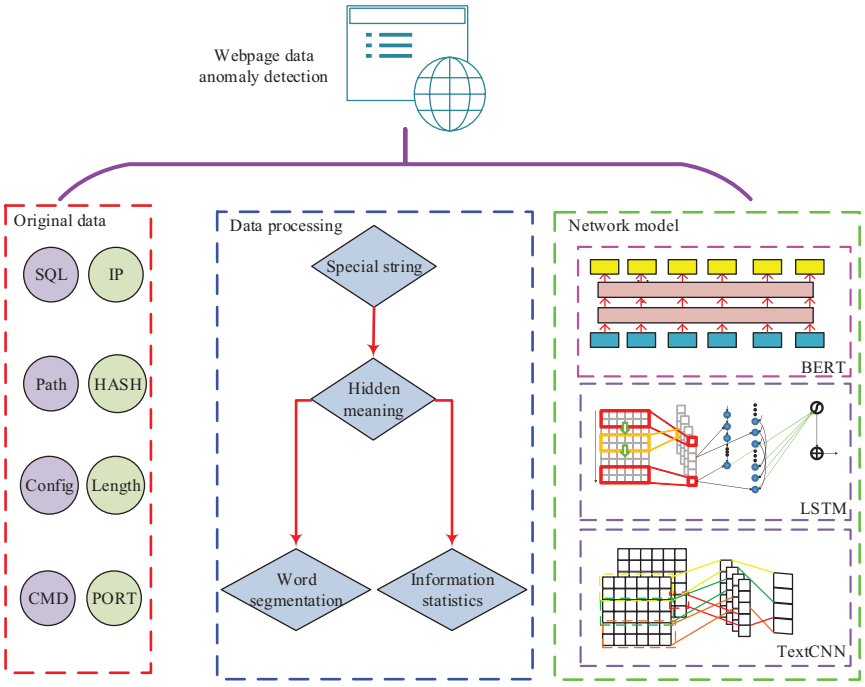


Figure 5 The overall framework for detecting abnormal Web page data.

During the training process, the weights of the BERT module are initialized using the pre-trained BERT-Base (12 layers, 768 dimensions) and are jointly fine-tuned with the LSTM and TextCNN modules. The optimizer uses Adam, with an initial learning rate of 2e-5 (for the BERT part) and 1e-3 (for the randomly initialized classification layer). To prevent overfitting, Dropout is applied after the fully connected layer with a rate of 0.5. The loss function is cross-entropy loss.

3.2 Multi-layer Encrypted Web Data Anomaly Detection Method Based on Cross-modal Feature Fusion

In low-encryption scenarios, the BERT-LSTM-TextCNN model effectively mines high-order semantic features and temporal patterns in Web traffic data, optimizing the accuracy and efficiency. However, in real-world network environments, the data encryption environment is often more complex. For network traffic data detection in multi-encryption environments, the ResNet model is used to capture the implicit structural pattern of encrypted data.

Traditional ResNet is a deep Convolutional Neural Network (CNN) that solves the degradation problem by introducing residual learning. To adapt the ResNet model to the characteristics of encrypted data, the research has improved the residual block in the ResNet model, including dynamic weight adjustment and nonlinear activation functions. Introducing a dynamic weight adjustment mechanism into the residual connection allows the model to adaptively adjust the transmission of residual information according to the importance of input features [24]. The dynamically adjusted weight can be calculated based on Equation (3):

$$\omega = \sigma(W_f x + b_f) \tag{3}$$

In Equation (3), σ signifies the Sigmoid activation function and b_f represents the bias of the fully connected network. In the output of the residual block, the nonlinear expression ability of the model can be enhanced by adding a nonlinear activation function, which helps the ResNet model better capture complex patterns in encrypted data. The improved residual block is shown in Equation (4):

$$y = \mathrm{ReLU}(x + \omega \cdot C(x)) \tag{4}$$

In Equation (4), ReLU signifies the activation function and C signifies the convolution operation. After extracting the deep learning features of the encrypted data, the study conducts cross-modal fusion of these features with statistical features and timing features extracted from communication protocol analysis. This fusion method can maximize the complementary information between different modalities. In the cross-modal fusion process, the deep learning feature F_d is first extracted from the improved ResNet model, the statistical feature F_s is extracted by analyzing the statistical characteristics of network traffic, and the time series feature F_t is extracted from the time series analysis. For the extracted features, the study uses auto-encoders to align the features to ensure that the features are consistent in dimension. The aligned features are fused through weighted summation, where the weights are manually set based on prior knowledge. The weighted fusion is shown in Equation (5):

$$F_{fusion} = \omega_d F_d + \omega_s F_s + \omega_t F_t \tag{5}$$

In Equation (5), ω_d, ω_s and ω_t represent the weights of deep learning features, statistical features and time series features, respectively. To optimize the effect of feature fusion, the study uses a basic Random Forest (RF) to learn

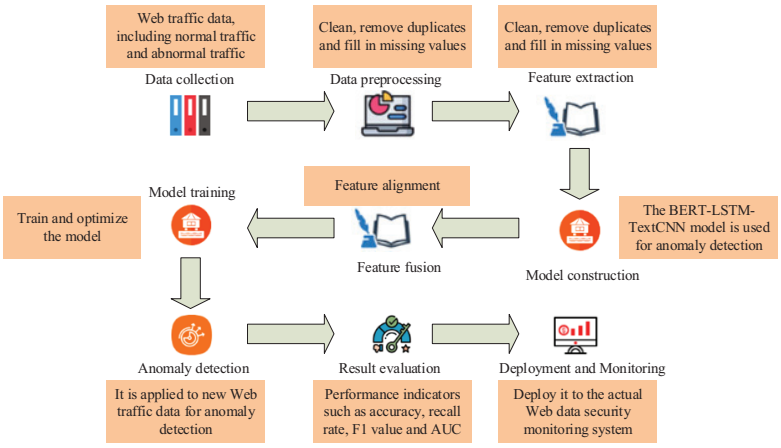


Figure 6 Large-scale Web data security monitoring process.

the optimal weights, which can be achieved by minimizing a loss function. The relevant mathematical expression is shown in Equation (6):

$$L_{\min} = \min_{\omega_d, \omega_s, \omega_t} L(F_{fusion}, Y) \tag{6}$$

In Equation (6), L signifies the loss function and Y signifies the actual label. Finally, the overall process based on large-scale Web data security monitoring is shown in Figure 6.

In Figure 6, Web traffic data are first collected from multiple data sources, including normal traffic and abnormal traffic. The collected data are cleaned and deduplicated. Missing values are filled, invalid data are filtered, and data are normalized. A two-stage word segmentation method is used to segment text data. One-hot encoding converts text features into word embedding vectors and extracts statistical features and temporal features related to anomaly detection. For non-encrypted or low-encrypted data, the BERT-LSTM-TextCNN model is used for anomaly detection. For multi-layer encrypted data, the improved ResNet model is used to extract deep learning features and perform cross-modal feature fusion. After aligning deep learning features, statistical features and time series features, cross-modal feature fusion is performed through weighted summation. We use the fused features to train a classification model and optimize the model parameters by minimizing the loss function. The trained model is applied to new web traffic data for anomaly detection. Indicators like precision, recall, and AUC are evaluated to verify the effectiveness. Finally, the model is deployed into

the actual Web data security monitoring system, and the performance of the system is continuously monitored, and adjustments and optimizations are made as needed.

4 Large-Scale Web Data Anomaly Detection and Verification Based on Artificial Intelligence

4.1 Experimental Environment Settings

To evaluate the model's monitoring effect on large-scale Web data, the study first set up a high-performance experimental environment. The experimental environment and model parameter settings are presented in Table 1.

Based on the parameter settings in Table 1, the KDD Cup 1999 and the CSE-CIC-IDS2018 are selected as the data sets. The KDD Cup 1999 contains a large number of normal and abnormal (attack) traffic records and is an important resource in network security research. It contains four types of normal traffic and 14 types of attack traffic. The CSE-CIC-IDS2018 data set contains network traffic data in a variety of attack scenarios, covering various attack types, like DoS, DDoS, and port scanning. The data set is split into training, validation and test sets according to 7:2:1. Given the extreme imbalance between normal traffic and attack traffic in network traffic data (as shown in the KDD Cup 1999 where normal traffic accounted for approximately 20% and attack traffic accounted for 80%), the research adopted stratified sampling to ensure that the proportion of each category in the sub-sets was consistent with that of the original dataset, avoiding model evaluation distortion caused by sampling bias. Considering the obvious time series characteristics of the CSE-CIC-IDS2018 dataset, the research employed a time-aware partitioning strategy, dividing the data in chronological order, ensuring that the training set data precedes the validation set and test set, simulating the model's predictive ability for future data in real scenarios, and avoiding data leakage. To ensure the reproducibility of the experiments, all random operations were set to fixed random seeds, and five independent repeated experiments were conducted, taking the average value as the final result, to reduce the influence of randomness on the results. The study also selected Auto-encoder (AE), LSTM and CNN for comparative experiments.

The research constructed a test environment covering different encryption types based on public datasets through traffic rewriting and simulation generation. The non-encrypted scenario directly adopted the original HTTP plaintext traffic from the CSE-CIC-IDS2018 dataset; the low-encryption

Table 1 Experimental settings

Component	Specification	Parameter (BERT-LSTM-TextCNN)	Value	Parameter (ResNet)	Value
CPU	Intel Xeon E5-2620 v4 @ 2.10GHz	BERT Model	BERT-Base	ResNet Version	ResNet-50
GPU	NVIDIA GeForce GTX 1080 Ti	Number of BERT Layers	12	Number of Layers	50
RAM	64 GB DDR4	Hidden Size of BERT	768	Hidden Size of ResNet	2048
Operating System	Ubuntu 18.04 LTS	Number of Attention Heads	12	Kernel Sizes of ResNet	[7, 3, 3]
Python Version	Python 3.7	LSTM Layers	2	Dropout Rate	0.3
CUDA Version	10	Hidden Size of LSTM	256	Learning Rate	1.00E-04
cuDNN Version	7.4.2	TextCNN Layers	3	Batch Size	64
TensorFlow	1.14.0	Kernel Sizes of TextCNN	[3, 4, 5]	Epochs	20
PyTorch	1.3.1	Dropout Rate	0.2	Dynamic Weight Adjustment	Enabled
Scikit-learn	0.21.3	Learning Rate	2.00E-05	Non-linear Activation	ELU
NumPy	1.16.4	Batch Size	32	/	/
Pandas	0.25.1	Epochs	10	/	/

scenario was generated by rewriting the original traffic, by downgrading the encryption protocol to SSL 3.0 and TLS 1.0 versions, and using the RC4 cipher suite and a 512-bit RSA short key to simulate an encrypted environment with security vulnerabilities; the multi-encryption scenario construction covered five typical environments: (1) The HTTPS encryption scenario directly adopted the encrypted attack traffic enabled with TLS 1.2/1.3 from the CSE-CIC-IDS2018 dataset. (2) The VPN encryption scenario was based on the public non-malicious VPN traffic dataset, using the OpenVPN TLS mode and encrypting the data with AES-256-GCM. (3) The Tor network scenario used the Tor traffic public dataset, with the traffic undergoing three-layer AES nested encryption and obfuscation processing. (4) The encrypted email scenario was generated by simulating the S/MIME and PGP protocols, encrypting the email content and attachments with AES-256 content encryption and RSA signature. (5) The multi-protocol mixed scenario generated multiple encrypted protocols such as HTTPS API calls, DNS over TLS, and DoH in the simulation environment simultaneously to simulate encrypted communication in complex network environments.

4.2 BERT-LSTM-TextCNN Detection Model Verification

To verify the performance of the BERT-LSTM-TextCNN, the study verified its precision, recall rate and F1-score, as presented in Figure 7.

In Figure 7, the BERT-LSTM-TextCNN was better than the control model in precision, recall rate and F1-score. The BERT-LSTM-TextCNN converged when the iteration was 80. After convergence, its precision, recall and F1-score tended to 0.9718, 0.9526 and 0.9621. The performance index of the CNN model after convergence was poor, and its convergence speed was the slowest. It converged after 120 iterations. After convergence, the precision, recall and F1-score tended to 0.8092, 0.8210 and 0.8312. The study further compared the AUC values, false positive rates and single-batch data detection time, as presented in Table 2.

In Table 2, Compared with the single LSTM and CNN models, the AUC values of BERT-LSTM-TextCNN are 2.4%/3.1% and 3.4%/4.1% higher on the two datasets respectively. This proves the effectiveness of integrating the pre-trained language model with the multi-branch network structure. Particularly, although the CNN has the fastest detection speed (30.8 seconds), its false alarm rate is over 10.90%, while the BERT-LSTM-TextCNN model reduces the false alarm rate by 8.12% with only a small sacrifice in detection time (36.5 seconds), achieving a good balance between accuracy

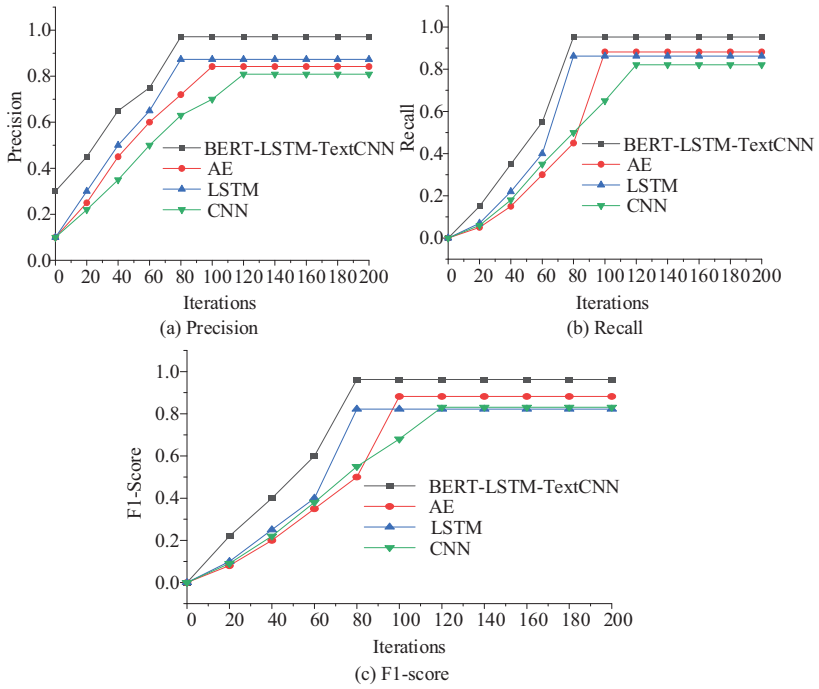


Figure 7 Comparison of (a) precision, (b) recall, and (c) F1-score curves.

Table 2 Comparison of AUC value, false positive rate and single-batch data detection time

Data Set	Metric	AUC Value	False Positive Rate	Detection Time (s)	p-value
KDD Cup 1999	BERT-LSTM-TextCNN	0.974	2.80%	36.5	/
	AE	0.92	4.10%	50.2	< 0.001
	LSTM	0.95	3.50%	42.1	< 0.001
	CNN	0.94	10.90%	30.8	< 0.05
CSE-CIC-IDS2018	BERT-LSTM-TextCNN	0.971	3.00%	38.0	/
	AE	0.91	4.50%	48.9	< 0.001
	LSTM	0.94	3.20%	43.5	< 0.001
	CNN	0.93	11.14%	33.7	< 0.05

and efficiency. Compared with AE, the BERT-LSTM-TextCNN model has significant improvements in AUC and false alarm rate, indicating that with sufficient labeled data, supervised learning methods can more accurately delineate the abnormal boundaries.

4.3 Detection Method Verification Based on Cross-modal Feature Fusion

After verifying the BERT-LSTM-TextCNN, the research further verified the performance of the improved ResNet model. Figure 8 presents the precision, recall rate and F1-score.

In Figure 8, the average precision of the improved ResNet was 98.48%, the average recall was 87.30%, and the average F1-score was 92.57%, which were higher than those of the AE, LSTM and CNN models. In contrast, the performance differences of the remaining three control models were not significant. The average values of the AE model in precision, recall and F1-scores were 84.13%, 85.63% and 84.85%, respectively, the average values of the LSTM model were 88.39%, 84.86% and 86.56%, respectively, and the average values of the CNN model were 86.08%, 82.14% and 84.05%, respectively. The improved ResNet has high precision in correctly identifying normal traffic and abnormal traffic in multi-encryption scenarios. The study finally compared the running time and inference time of the four models, as shown in Figure 9.

In Figure 9, the improved ResNet model showed the advantages on the KDD Cup 1999 data set and CSE-CIC-IDS2018. Although its running time and inference time were not the shortest, considering its excellent performance in precision, recall and F1-score, the improved ResNet provided a good balance of efficiency and performance. Specifically, the improved ResNet model had a running time of 45.67 seconds and an inference time of 36.54 seconds on the KDD Cup 1999, while the running time on the

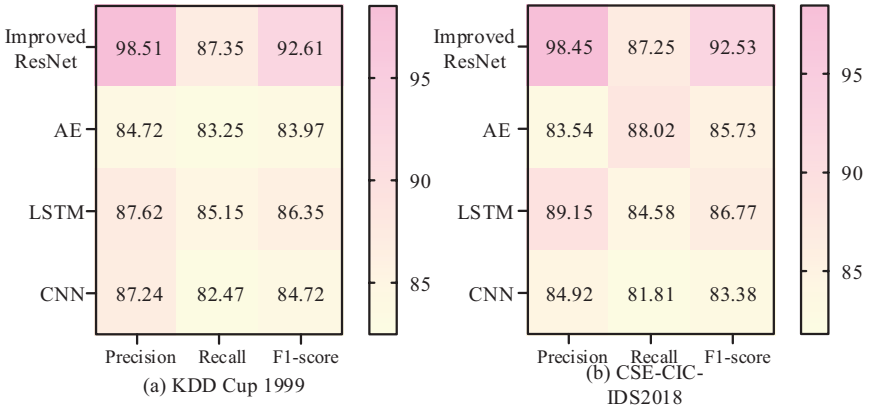


Figure 8 Comparison of precision, recall, and F1-score.

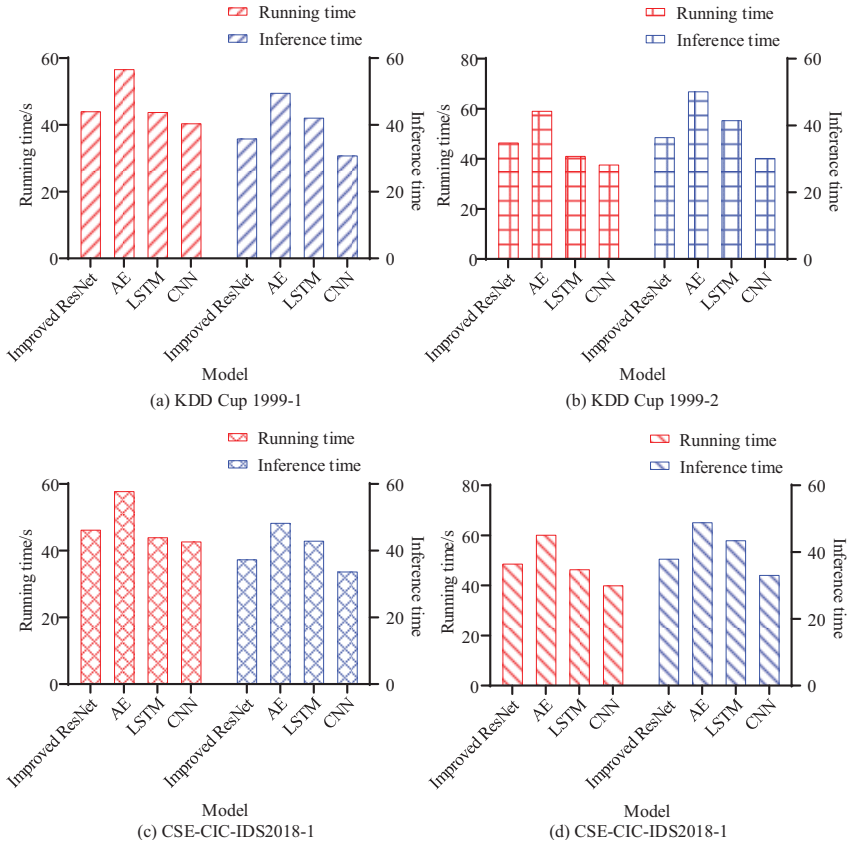


Figure 9 Comparison of running time and inference time.

CSE-CIC-IDS2018 data set was 47.89 seconds and the inference time was 38.02 seconds. In contrast, although the CNN has the shortest running time and inference time, its performance in precision, recall, and F1-score is not as good as the improved ResNet model. Although the AE model performs well in some respects, its long running time and inference time may limit its application in scenarios that require fast response. The performance of the LSTM is similar to the improved ResNet but is slightly inferior in running time and inference time. In summary, the improved ResNet model can provide faster response speed while ensuring detection accuracy, which is an obvious advantage for real-time security monitoring systems. To comprehensively evaluate the robustness of the improved ResNet model under different encryption algorithms, three common encryption scenarios were

Table 3 Performance comparison (AUC) of different models under various encryption algorithms

Encryption Algorithm	Improved ResNet	TabNet	FT-Transformer	SAINT	p-value (vs. Improved ResNet)
AES	0.983 ± 0.003	0.941 ± 0.005	0.949 ± 0.004	0.957 ± 0.004	< 0.001
RSA	0.976 ± 0.004	0.934 ± 0.006	0.942 ± 0.005	0.948 ± 0.005	< 0.001
TLS/SSL (Mixed)	0.979 ± 0.003	0.938 ± 0.005	0.945 ± 0.004	0.953 ± 0.004	< 0.001

simulated and proposed: AES encrypted traffic, RSA encrypted traffic, and mixed encrypted traffic (TLS/SSL). At the same time, three deep learning models that have performed well in the fields of tabular data and network security anomaly detection in recent years were selected as controls, namely TabNet, FT-Transformer, and SAINT. Among them, TabNet is a tabular data deep learning model based on the Transformer architecture, which achieves instance-level feature selection through the sequential attention mechanism and has both high accuracy and interpretability; FT-Transformer models feature interaction modeling through the standard Transformer encoder and has achieved the most advanced results on multiple tabular data sets; SAINT introduces sample-to-sample attention mechanisms, which can capture the correlation information between different samples. The results are shown in Table 3.

In Table 3, the AUC values of the improved ResNet model under the three encryption algorithms were significantly higher than those of the three control models ($p < 0.001$). Especially in the AES encryption scenario, the AUC of the improved ResNet model reached 0.983, which was 2.6% higher than the best SAINT model. Even in the most challenging RSA encryption scenario, the improved ResNet still maintained a high AUC value of 0.976. The results indicate that the cross-modal feature fusion method proposed in this study can effectively extract the implicit patterns in encrypted traffic and has good adaptability to different encryption algorithms.

4.4 Practical Application Verification

After verifying the performance of the ResNet model in multiple encryption scenarios, the study finally verified the detection accuracy and efficiency in practical applications. First, for non-encrypted or low-encryption scenarios, the study selected five scenarios: 1 (regular Web traffic monitoring), 2 (file download monitoring), 3 (login attempt monitoring), 4 (API call monitoring)

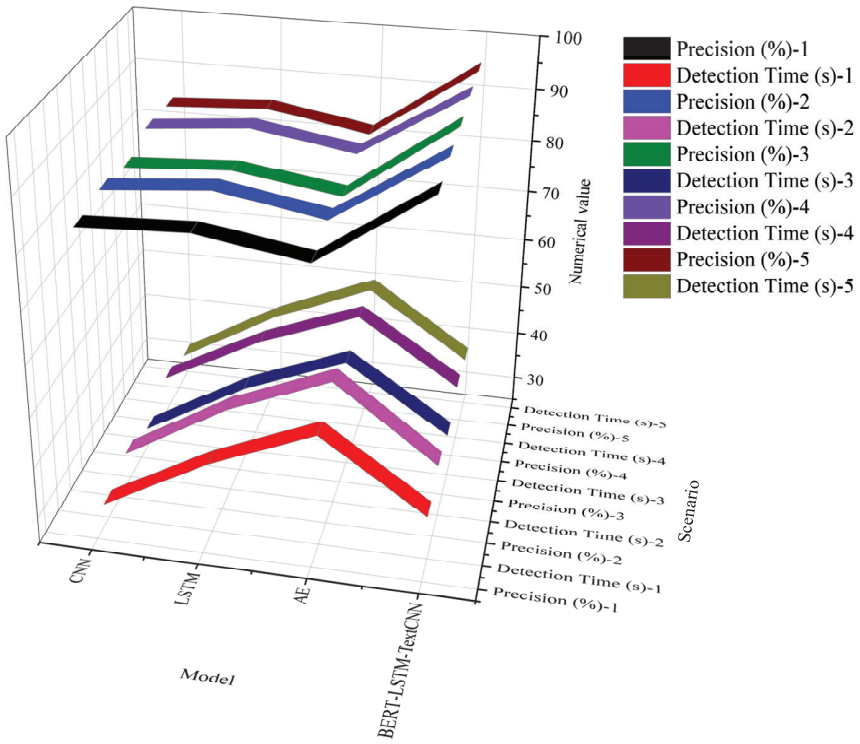


Figure 10 Comparison of detection precision and detection time.

and 5 (payment transaction monitoring). The results of detection precision and detection time are shown in Figure 10.

In Figure 10, under five different non-encryption or low encryption scenarios, the BERT-LSTM-TextCNN model showed its advantages in most cases. Especially in the API call monitoring scenario, the model achieved the highest detection precision, at 97.20%, which was better than that of the AE (85.10%), the LSTM model (88.20%), and the CNN model (86.50%). In the conventional web traffic monitoring scenario, the BERT-LSTM-TextCNN model had the shortest detection time of 35.10 seconds. Compared with the 48.50 seconds of the AE model, 40.20 seconds of the LSTM model and 29.30 seconds of the CNN model, the detection speed was relatively faster while maintaining high detection accuracy. Although the CNN model was slightly better in detection time, its precision was lower. In other scenarios, the BERT-LSTM-TextCNN model also showed good overall performance. Whether in file download monitoring, login attempt monitoring or payment

Table 4 Detection precision and detection time results in multi-encryption environments

Scenario	Model	Precision (%)	Detection Time (s)
HTTPS traffic analysis	Improved ResNet	98.45	47.89
	AE	83.54	59.43
	LSTM	89.15	45.63
	CNN	84.92	41.78
VPN traffic monitoring	Improved ResNet	98.37	46.54
	AE	82.87	57.21
	LSTM	88.46	44.37
	CNN	85.63	40.12
Tor network traffic detection	Improved ResNet	98.62	48.34
	AE	84.12	61.05
	LSTM	87.89	46.89
	CNN	83.45	39.56
Encrypted email communication monitoring	Improved ResNet	98.29	45.78
	AE	81.45	55.67
	LSTM	90.02	43.45
	CNN	86.78	42.09
Analysis of multi-protocol encrypted communication	Improved ResNet	98.58	49.22
	AE	85.09	58.84
	LSTM	86.74	47.23
	CNN	82.31	38.47

transaction monitoring scenarios, its detection accuracy exceeded 96%. In summary, the BERT-LSTM-TextCNN model has demonstrated its obvious advantages in precision in anomaly detection tasks in non-encrypted or low-encryption scenarios, especially in API call monitoring scenarios, and it also performs well in detection time. The study finally compared the detection precision and detection time in five multi-encryption scenarios, as presented in Table 4.

In Table 4, the improved ResNet model showed significant advantages in multi-encryption scenarios, especially in the multi-protocol encrypted communication analysis. Detection accuracy reached 98.58%, which was higher than the other three models. The improved ResNet can more effectively identify abnormal behaviors when dealing with complex network communications containing multiple encryption protocols. In terms of detection time, the improved ResNet model also showed good performance in multi-protocol encrypted communication analysis scenarios, with a detection time of 49.22 seconds. Although slightly inferior to the 38.47 seconds of the CNN model, considering its significant advantage in accuracy, this detection time

was completely acceptable. Compared with 58.84 seconds of the AE model and 47.23 seconds of the LSTM model, the improved ResNet model also provides a relatively fast detection speed while maintaining high accuracy. In summary, in multi-encryption scenarios, the improved ResNet model is not only far ahead in detection accuracy but also shows a good balance in detection time. It can be suitable for building an efficient and reliable Web data security monitoring system.

5 Conclusion

To optimize the accuracy and efficiency of anomaly detection in large-scale Web data security monitoring, the research combined the BERT with LSTM and TextCNN and optimized the ResNet model through cross-modal feature fusion to detect anomalies in large-scale Web data. In the research results, the accuracy of BERT-LSTM-TextCNN reached 97.18%, the recall rate reached 95.26%, and the average false positive rate was 2.90%. In the multi-encryption scenario, the improved ResNet model showed its advantages. Its precision, recall rate and F1-score were as high as 98.48%, 87.30% and 92.57%, which were higher than other control models. In practical applications, the BERT-LSTM-TextCNN model achieved the highest detection accuracy in the API call monitoring scenario, with a value of 97.20%. In multiple scenarios, the improved ResNet model achieved a detection accuracy of 98.58% in the multi-protocol encrypted communication analysis scenario, and the value was significantly better than the control model. The artificial intelligence anomaly detection method effectively improves detection accuracy and efficiency.

Muralitharan and Arumugam built an improved BERT-LSTM to address the insufficient detection accuracy of sensitive information in text [25]. Similar to the research, this algorithm also uses BERT to generate contextual embeddings and enhances the capturing ability of LSTM through the attention mechanism. The model achieved an accuracy of 92.50%, an F1-score of 85.02%, and a precision of 89.36% on the SMS Spam Collection data set. Pandey and Singh R designed a BERT-LSTM that fuses BERT and LSTM to solve the decreased accuracy of sarcasm detection in social media posts [26]. The model achieved 87.3% F1-score on the self-built code-mixed data set, which is 4.1% higher than pure BERT. It is significantly better than traditional CNN, Bi-LSTM and other baseline models, and provides a feasible solution for low-resource multi-language sarcasm recognition. Slightly different from the research results is that this report only performs sarcasm

recognition on low-resource multi-languages, which corresponds to the non-encrypted or low-encrypted scenarios in the study. Since the multi-resource and multi-language situation is not included, the generalization ability of the BERT-LSTM model has not been effectively verified. In addition, since no cross-language experiments are conducted and the degree of sharing of sarcastic features between different languages is not analyzed, it is difficult to prove that this method can reproduce the 4% F1-score improvement result when deployed in multiple languages. In contrast, the study explores the performance of the model under low encryption conditions and multi-encryption conditions based on the BERT-LSTM-TextCNN model and improved ResNet and combines statistical features with time features to perform cross-modal feature fusion, making the detection accuracy and detection time results in the study more reliable.

In summary, the artificial intelligence anomaly detection method shows obvious advantages in large-scale Web data security monitoring. It optimizes the detection accuracy and efficiency and also provides a feasible technical path for building a comprehensive Web data security monitoring system. Although the research has achieved experimental results, there are still certain shortcomings. Firstly, the performance of the model in extreme encryption environments needs further verification. Although the research covers various encryption scenarios, there are still more complex extreme encryption situations in real network environments; moreover, the robustness of the model under data sparsity conditions still has room for improvement. Future research can explore lightweight technologies, that is, using the complex integrated model as the teacher network and training a lightweight student network to maintain detection accuracy while significantly reducing computational complexity; at the same time, introducing online incremental learning mechanisms so that the model can be continuously updated and can quickly respond to sudden new attacks, narrowing the time window from the occurrence of the attack to the update of the detection model. In addition, exploring domain adaptation and domain generalization technologies can enable the model to learn domain-invariant feature representations, thereby achieving knowledge transfer in the absence of labeled data in the target domain.

Funding

This research was supported by the Guangdong Electric Power Science Academe [Project No.030000KC23110047].

References

[1] Bonikela H R. Deep Learning for Cybersecurity: AI-Based Detection of Phishing and Fraudulent Web Pages. International Research Journal of Modernization in Engineering Technology and Science, 7(4): 2533–2559, 2025. DOI: 10.56726/IRJMETS72460.

[2] Mehmood A, Shafique A, Alawida M, et al. Advances and vulnerabilities in modern cryptographic techniques: A comprehensive survey on cybersecurity in the domain of machine/deep learning and quantum techniques. IEEE access, 12(2): 27530–27555, 2024. DOI: 10.1109/ACCESS.2024.3367232.

[3] Lin H, Deng X, Yu F, et al. Grid multibutterfly memristive neural network with three memristive systems: Modeling, dynamic analysis, and application in police IoT. IEEE Internet of Things Journal, 11(18): 29878–29889, 2024. DOI: 10.1109/JIOT.2024.3409373.

[4] Dhanushkodi K, Thejas S. AI enabled threat detection: Leveraging artificial intelligence for advanced security and cyber threat mitigation. IEEE access, 2024, 12: 173127–173136. DOI: 10.1109/ACCESS.2024.3493957.

[5] Safi A, Singh S. A systematic literature review on phishing website detection techniques. Journal of King Saud University-Computer and Information Sciences, 35(2): 590–611, 2023. DOI: 10.1016/j.jksuci.2023.01.004.

[6] Feng F, Zhou Q, Shen Z, et al. The application of a novel neural network in the detection of phishing websites. Journal of Ambient Intelligence and Humanized Computing, 15(3): 1865–1879, 2024. DOI: 10.1007/s12652-018-0786-3.

[7] Kumar A, Mallik A, Kumar S. HumourHindiNet: Humour detection in Hindi web series using word embedding and convolutional neural network. ACM Transactions on Asian and Low-Resource Language Information Processing, 23(7): 1–21, 2024. DOI: 10.1145/3661306.

[8] Kaur J, Garg U, Bathla G. Detection of cross-site scripting (XSS) attacks using machine learning techniques: a review. Artificial Intelligence Review, 56(11): 12725–12769, 2023. DOI: 10.1007/s10462-023-10433-3.

[9] Al-Quayed F, Ahmad Z, Humayun M. A situation based predictive approach for cybersecurity intrusion detection and prevention using machine learning and deep learning algorithms in wireless sensor networks of industry 4.0. IEEE Access, 12(2): 34800–34819, 2024. DOI: 10.1109/ACCESS.2024.3372187.

[10] Piplai A, Kotal A, Mohseni S, et al. Knowledge-enhanced neurosymbolic artificial intelligence for cybersecurity and privacy. IEEE Internet Computing, 27(5): 43–48, 2023. DOI: 10.1109/MIC.2023.3299435.
[11] Omer N, Samak A H, Taloba A I, et al. A novel optimized probabilistic neural network approach for intrusion detection and categorization. Alexandria Engineering Journal, 72: 351–361, 2023. DOI: 10.1016/j.aej.2023.03.093.
[12] Pleshakova E, Osipov A, Gataullin S, et al. Next gen cybersecurity paradigm towards artificial general intelligence: Russian market challenges and future global technological trends. Journal of Computer Virology and Hacking Techniques, 20(3): 429–440, 2024. DOI: 10.1007/s11416-024-00529-x.
[13] Udayakumar R, Joshi A, Boomiga S S, et al. Deep fraud Net: A deep learning approach for cyber security and financial fraud detection and classification. Journal of Internet Services and Information Security, 13(3): 138–157, 2023. DOI: 10.58346/JISIS.2023.I4.010.
[14] Rajak A, Tripathi R. DL-SkLSTM approach for cyber security threats detection in 5G enabled IIoT. International Journal of Information Technology, 16(1): 13–20, 2024. DOI: 10.1007/s41870-023-01651-7.
[15] Saravanan V, Madiajagan M, Rafee S M, et al. IoT-based blockchain intrusion detection using optimized recurrent neural network. Multimedia Tools and Applications, 83(11): 31505–31526, 2024. DOI: 10.1007/s11042-023-16662-6.
[16] Sahingoz O K, Bube E, Kugu E. Dephides: Deep learning-based phishing detection system. IEEE Access, 12(2): 8052–8070, 2024. DOI: 10.1109/ACCESS.2024.3352629.
[17] Liu M, Ma Z, Li J, et al. Deep-learning-based pre-training and refined tuning for web summarization software. IEEE Access, 12: 92120–92129, 2024. DOI: 10.1109/ACCESS.2024.3423662.
[18] Imrana Y, Xiang Y, Ali L, et al. CNN-GRU-FF: a double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units. Complex & Intelligent Systems, 10(3): 3353–3370, 2024. DOI: 10.1007/s40747-023-01313-y.
[19] Alsubaei F S, Almazroi A A, Ayub N. Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics. IEEE Access, 12(2): 8373–8389, 2024. DOI: 10.1109/ACCESS.2024.3351946.
[20] Li P, Yu H, Luo X, et al. LGM-GNN: A local and global aware memory-based graph neural network for fraud detection. IEEE Transactions

on Big Data, 9(4): 1116–1127, 2023. DOI: 10.1109/TBDATA.2023.3234529.

[21] Bacanin N, Zivkovic M, Antonijevic M, et al. Addressing feature selection and extreme learning machine tuning by diversity-oriented social network search: an application for phishing websites detection. Complex & Intelligent Systems, 9(6): 7269–7304, 2023. DOI: 10.1007/s40747-023-01118-z.

[22] Yan F, Zhang G, Zhang D, et al. TL-CNN-IDS: transfer learning-based intrusion detection system using convolutional neural network. The Journal of Supercomputing, 79(15): 17562–17584, 2023. DOI: 10.1007/s11227-023-05347-4.

[23] Vatambeti R, Krishna E S P, Karthik M G, et al. Securing the medical data using enhanced privacy preserving based blockchain technology in Internet of Things. Cluster Computing, 27(2): 1625–1637, 2024. DOI: 10.1007/s10586-023-04056-0.

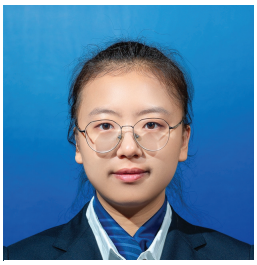
[24] Rao K V, Prasad M H M K. Deep neural network empowered bi-directional cross GAN in context of classifying DDoS over flash crowd event on web server. Multimedia Tools and Applications, 82(24): 37303–37326, 2023. DOI: 10.1007/s11042-023-15030-8.

[25] Muralitharan J, Arumugam C. Privacy BERT-LSTM: a novel NLP algorithm for sensitive information detection in textual documents. Neural Computing and Applications, 36(25): 15439–15454, 2024. DOI: 10.1007/s00521-024-09707-w.

[26] Pandey R, Singh J P. BERT-LSTM model for sarcasm detection in code-mixed social media post. Journal of Intelligent Information Systems, 60(1): 235–254, 2023. DOI: 10.1007/s10844-022-00755-z.

Biographies

Siyao Xu (February 1991–) male, graduated from South China University of Technology with a master's degree in Power Systems and Automation. After graduation, he worked as an engineer at the Electric Power Research Institute of Guangdong Power Grid Co. Ltd. His current research direction is power system network security.



Yan Li (December 1992–) female, graduated from North China Electric Power University with a major in Computer Science, and obtained a master's degree. After graduation, she worked as an engineer at the Electric Power Research Institute of Guangdong Power Grid Co. Ltd. Her current research direction is power system network security.

Kai Zhang (July 1995–) male, graduated from Nanjing University with a major in Computer Science and Technology, and obtained a master's degree. After graduation, he worked as an engineer at the Electric Power Science Research Institute of Guangdong Power Grid Co. Ltd. His current research direction is power digitization.

Weiming Li (September 1999–) male, graduated from Renmin University of China with a major in Information Technology and obtained a master's degree. After graduation, he worked as an engineer at the Electric Power Science Research Institute of Guangdong Power Grid Co. Ltd. His current research direction is power digitization.

Jieshao Lai (July 1998–) male, graduated from the Big Data program at the University of Science and Technology of China with a master's degree. After graduation, he worked as an engineer at the Electric Power Science Research Institute of Guangdong Power Grid Co. Ltd. His current research direction is power digitalization.