IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 13 October 2023, accepted 11 November 2023, date of publication 14 November 2023,
date of current version 28 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333013

RESEARCH ARTICLE

Evaluation of Network Security Grade Protection Combined With Deep Learning for Intrusion Detection

SHAODAN LIN[1,2], CHEN FENG[3], TAO JIANG[2,4], AND HUASONG JING[5], (Member, IEEE)

[1]College of Mechanical and Intelligent Manufacturing, Fuzhou 350007, China
[2]Fujian Chuanzheng Communications College, Fuzhou 350007, China
[3]Department of Information Engineering, Fuzhou Polytechnic, Fuzhou 350007, China
[4]College of Information and Intelligent Transportation, Fuzhou 350007, China
[5]Fujian Zhong Xin Network Security Information Technology Company Ltd., Fuzhou 350007, China

Corresponding authors: Shaodan Lin (linshaodan66@qq.com) and Tao Jiang (361899811@qq.com)

This work was supported in part by the Fujian Science and Technology Plan Guided Project 2021H0034, and in part by the Construction of Collaborative Innovation Center for Intelligent Equipment Application Technology in Fujian Vocational Colleges Project MJK [2016] No. 7.

ABSTRACT Using deep learning models, we predict information system security indicators and obtain corresponding security evaluation scores. The scores of these predicted security evaluation are used as the input data of regression tree model, and the security grade protection evaluation system is constructed. The model training process involves four different models: VGG19, ResNet-50, XceptionNet, and EfficientNet. Based on the training results, we find that the EfficientNet model consumes fewer computational resources in single detection while achieves a detection accuracy of 99.93%. Subsequently, we apply the CART regression tree to assess the network security posture of 14 commercial systems. The test results of the model show that the mean absolute percentage error(MAPE) is 0.029 and the correlation coefficient is 0.9. These empirical results strongly support the performance of the proposed model and show its significant potential in improving security assessments. With these training results, we gain preliminary insights into the performance of each model and select the EfficientNet model with the best performance for the generation of subsequent security posture evaluation data.Ultimately, the developed security grade protection assessment system provides a reliable and efficient evaluation means for the network security.

INDEX TERMS CART regression tree, data collection and analysis, EfficientNet, security indicators, security data correlation.

I. INTRODUCTION

As our society becomes more interconnected and technologically advanced, the challenges of protecting our systems and society from evolving threats is becoming increasingly complex. Therefore, it is an ongoing topic to design more efficient and effective cybersecurity solutions [1], [2]. Cybersecurity refers to utilizing various methods to protect the systems from threats and vulnerabilities, and to efficiently provide correct services to users. Combined with the experience in grade consulting, grade evaluation and risk assessment, the analysis technology of network security grade protection conducts key interpretation, indicator reference, inspection guidance and rectification guidance, and builds a knowledge base for grade protection requirements analysis, compliance strategy baseline recommendation library, security configuration operation knowledge base, and security risk knowledge base. It also provides support and evidence for gap assessment, grade evaluation and operational security self-inspection in the construction of user units' grade protection.

The associate editor coordinating the review of this manuscript and approving it for publication was Mueen Uddin.

The Managers have an overall understanding of the security situation, so that they can quickly respond to the complex security threats [3]. Some scholars have conducted relevant research on security evaluation models. Ksiezopolski et al.proposed the approach how to introduce the adaptability to the network in 2009 [4]. As the general assessment methods were simply to calculate the risk value, Hu et al. proposed a risk assessment model based on classified security protection in 2010 [5]. Cai et al. discussed the grade-protection evaluation of network systems in extrlanet systems in 2017. He discussed the grading of grade protection, selection of evaluation objects and institutions, and safety rectification [6]. Based on the existing security technology and products, Sun et al. analyzed the relevant technical management requirements of the scheme design and studied the design method and process by combining with the information security grade protection system [7]. At present, there are some problems in traditional methods of graded protection evaluation, such as different understanding of safety indicators, ambiguity of judgments, uncertainty of judgment intervals, and unreasonable division of evaluation indicator weights. In view of the above problems, Zhi et al. proposed an evaluation method for security grade protection in data center network based on fuzzy synthesis and AHP [8]. Savva et al. considered the joint problem of protecting the confidential demands from eavesdropping attacks and protecting the network from link failures in elastic optical networks (EONs) [9]. Based on Python language and Django framework, Wang et al. completed the construction of cloud-computing network security grade protection evaluation system [10]. Based on the above research, we decide to build a new network security grade protection evaluation system from two aspects of technology and management. We match the different grade protection control items, security grade of objects, asset status, policy status, risk situation and system situation, evaluate the algorithms and self-learning functions through AI intelligent gap, comprehensively evaluate the overall security compliance and risk situation centered on the grading system based on the risk assessment models, and help users reflect the security situation of the grading objects. Quantitative analysis is performed on the grading objects to build a security evaluation model based on the international standard specifications. According to the comprehensive multiplication method, we evaluate the relative levels of assets, threats and vulnerabilities by considering the key parts of security risk assessment, threats, vulnerability severity and their related factors, such as likelihood of threat occurrence, likelihood of threat success, and vulnerability severity of assets. Then, the relative value of the risk is calculated by multiplying these values, and the risk level is determined based on the range of the risk level. In the process of network situational awareness, it is necessary to model the complex networks, analyze the network security situations, and provide the quantitative results for network situational awareness [11]. To achieve this process, it is necessary for the situational awareness model to have a powerful knowledge base that can quickly detect and match the network situations, and perform inference to provide reliable situational awareness results. In recent years, with the impressive performance of artificial intelligence algorithms in feature extraction, many researchers have studied and developed solutions for network security situational assessment based on artificial intelligence. Yang et. al. proposed a new calculating security indexes method based on CNNs [12]. Yuan et al. proposed a ML-based method for malware detection that utilizes more than 200 features extracted from both static and dynamic analysis of Android app. There are many challenges when developing a flexible and efficient NIDS for unforeseen and unpredictable attacks [13]. Javaid et al. proposed a deep-learning-based approach to develop such an efficient and flexible NIDS [14]. Tang et al. applied a deep-learning approach to conduct the flow-based anomaly detection in an SDN environment [15]. Kim et al. constructed an IDS model with deep learning approach [16]. The experimental results show that the proposed technique can respond to the attack in real time and significantly improve the detection ratio in controller area network (CAN) bus [17]. Yin et al. explored how to model an intrusion detection system based on deep learning, and proposed a deep-learning approach for intrusion detection using recurrent neural networks (RNN-IDS) [18]. Al-Qatf et al. proposed an effective deep-learning approach based on the STL framework, named self-taught learning (STL)-IDS [19]. Ferrag et al. investigated deep learning approaches for cyber security intrusion detection, the datasets used, and a comparative study [20]. These indexes can help assess the network situation. Bao et al. aimed at the background of big data and AI to optimize the design of information security situation awareness system, including optimizing system hardware configuration, standardizing the synchronous operation mechanism of AI in multiple data security perception, improving the information security situation inference algorithm, designing the system software structure, and adding comparative repair steps based on security characteristic parameters [21].

At the same time, SaaS model evaluation and wireless communication based on data security have been widely applied in software service industries such as data security and wireless communication. For example, this research implements a security evaluation framework to uncover security vulnerabilities in e-commerce operations beyond checkout/payment integration [22]. Swetnam et al. presented a strategic blueprint for creating and managing SaaS cyberinfrastructure and IaC as free and open-source software [23]. In data security area, Cai introduced the construction of wireless communication network security risk prediction model and the implementation of prediction scheme based on recurrent neural network(RNN), which provides an effective method for mobile wireless network security data

prediction [24]. The rigorous security analysis and comparative study show that the mechanism proposed by Das et al. has significantly better security and comparable communication and computational costs than the relevant schemes [25]. Aiming at the quantum algorithm which can solve the problem of large integer decomposition and discrete logarithm in polynomial time, Zhang et al. proposed an anti-quantum computing key management scheme for clustered sensor networks [26].

We proposed a model for network security grade protection detection using deep learning techniques. The model uses computing, networking and storage resources as basic building blocks, and selects and predefines a technical architecture based on system requirements. The specific implementation usually involves integrating software virtualization technology (including computing, networking, storage and security virtualization) into the same unit node, and each unit node can be aggregated through the network to achieve modular seamless expansion and build a unified resource pool [27]. Real-time adaptive decision-making response is made according to the actual situation, and the emergency response plans can be quickly generated. Security policies are proactively pushed to critical devices on the entire network, so that the security incidents can be alerted in real time. It needs to fully protect against security threats and achieve intelligent data linkage to enhance the defense capabilities [28]. The fusion of multiple security capabilities enables all the security capability modules to fully correlate the information generated during the data detection process, which completely changes the criticism of information fragmentation in traditional security devices. Users can fully grasp the threat situation without manual mining and analysis [29].

The existing network security evaluation methods have notable limitations [30]. Traditional graded protection assessment methods suffer from inconsistency, subjectivity, and poor adaptability due to human assessments, hindering their effectiveness in rapidly evolving threat scenarios [31]. Deep learning integration in security assessment is still limited, missing the opportunities to harness complex data patterns. Scalability issues hinder the incorporation of additional security control points and network expansion. Real-time situational awareness is often insufficient, impacting proactive defense. Inadequate quantitative analysis impedes prioritization and risk communication, while resource-intensive approaches deter regular assessments.In response, our research leverages deep learning to offer a precise, scalable, and real-time network security assessment solution. Our approach enhances the accuracy, adapts to evolving networks, and provides actionable insights for robust security risk management.

II. METHOD

To further optimize the security assessment system of the information systems, we adopt an assessment strategy that focuses on network intrusion security indicators. As the intrusion detection system is the main source of security elements in situation awareness, its accuracy directly affects the assessment of network security. It can detect the maliciousness without compromising the security of hosts and networks [32]. This article further combines network intrusion security with security factors, such as devices and applications, to construct a multifactor fusion network security situation assessment scheme, further improving the effectiveness and accuracy of security situation assessment.

A. INTRUSION DETECTION ALGORITHM BASED ON DEEP LEARNING

In terms of improving the network security grade of the system, the traditional network security protection system mainly focuses on the protection of the boundary, such as controlling the corresponding gate switch or defining the firewall to directly reject the attack. It also uses various physical isolation or network protection methods to prevent the attack from penetrating the network. The traditional model for handling abnormal network traffic tends to be defensive, and thus there will be hidden threats. Once the defense is breached, it is easy to be further invaded. As the new network attacks become increasingly complex and diverse, the traditional method has many problems, such as difficulty in data feature extraction, low accuracy, high false alarm rate, and high operating costs. In order to solve the above problems, we built an efficient network intrusion detection system based on deep learning [33], [34]. This system can detect various types of attacks, including PortScan, Web Attack, DOS, DDoS, Brute Force and Bot, achieving the monitoring and perception of network traffic intrusion and improving the security grade of the service platform. The detection process is shown in Fig.1.

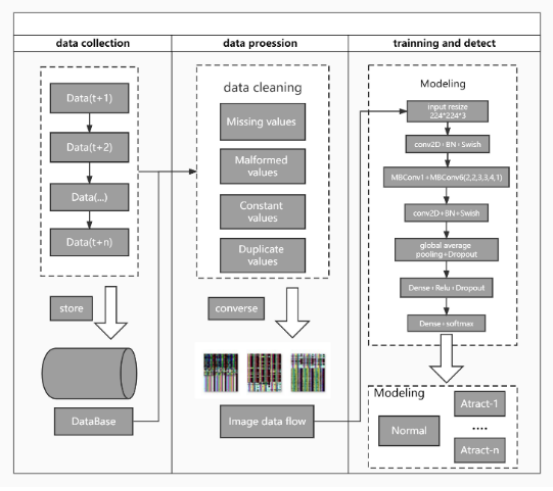

FIGURE 1. Framework diagram of intrusion detection algorithm.

1) DATA PREPROCESSING

To enhance the data availability of the model, it is necessary to perform preliminary processing on the log data read in

real time. This processing includes data cleaning, conversion and dataset enhancement. Data cleaning involves deleting all missing values, irregularly formatted values, infinite values, and features unrelated to classification. Features that are not directly related to attack behavior, such as Destination Port, Bwd PSH Flags, and Fwd URG Flags, may even interfere with model training. Therefore, they should also be deleted.

To effectively utilize the feature extraction capability of deep learning and obtain the accurate network attack detection results, it is necessary to convert network traffic data sets from text form to image form. This conversion requires a more efficient and effective use of deep learning techniques. To accomplish this, a non-linear transformation is first used to normalize the features of each data type. The normalized data is then scaled to a range of [0, 255]. The data transformation process is as follows:

$$X' = 255 \times \text{QuantileTransformer(X)} \tag{1}$$

where X' represents the resulting data after transformation, which is a new dataset consisting of network traffic data that has been processed and converted.X denotes a specific numerical value within the original dataset, which can be a sample or a feature. *QuantileTransformer(X)* is a non-linear transformation function used to normalize the original data x.Be transformed in this way, the data can be visualized in form of images, which can be used as the input of a deep learning model.

2) MODEL BUILDING AND OPTIMIZATION

We converted the text data of network traffic into two-dimensional image data, and utilized the feature extraction ability of deep neural networks to learn various types of network attack patterns, As shown in Fig.2. Specifically, we focused on the following attack types: PortScan, Web Attack, Brute Force, DDoS, Bot, and DoS, generating the corresponding image datas.

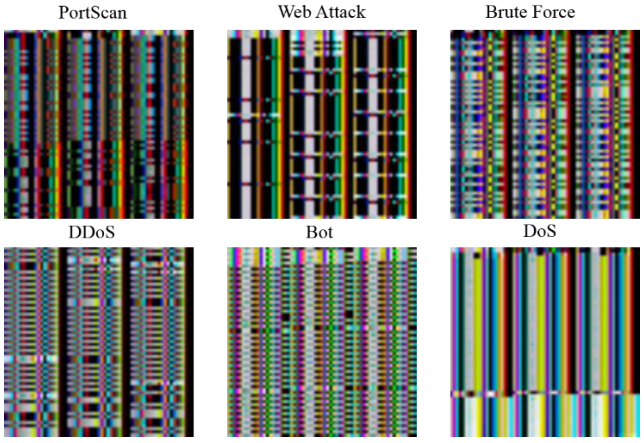

FIGURE 2. The text data of network traffic into two-dimensional image data.

(1) PortScan Image Data: These images represent the PortScan attack patterns in network traffic. PortScan indicates that a malicious user scans target hosts to discover open ports and services, conducting the subsequent attacks. PortScan image data displays the frequency of port scanning and the distribution of port numbers on different target hosts.

(2) Web Attack Image Data: These images reflect web attack patterns in network traffic. Web attacks target web applications with malicious behaviors, such as SQL injection, cross-site scripting (XSS), and others. Web Attack image data presents different attack patterns on various web applications, including malicious injection attempts on specific parameters and insertion of malicious code.

(3) Brute Force Image Data: These images reveal brute force attack patterns in network traffic. Brute force attacks try all possible password combinations to crack passwords or access protected resources. Brute Force image data attempts multiple logins for specific accounts or resources, and each attempt may involve different username and password combinations.

(4) DDoS Image Data: These images represent the Distributed Denial of Service (DDoS) attack patterns in network traffic. DDoS attacks overload or make a target system unavailable by simultaneously launching a large number of requests from multiple sources. DDoS image data shows the concentration of source IP addresses, request frequency, and other related information.

(5) Bot Image Data: These images display bot attack patterns in network traffic. Bot attacks involve malicious software automatically executing attack activities through infected computers. Bot image data presents different patterns of malicious activities on various computers, such as bulk sending of spam emails and DDoS attacks.

(6) DoS Image Data: These images demonstrate Denial of Service (DoS) attack patterns in network traffic. DoS attacks overload a target system with malicious traffic or requests until it becomes unavailable. DoS image data shows the concentration of requests and the duration of attacks in network traffic.

We utilize the feature extraction ability of deep neural networks to learn various types of network attack patterns.Then, the extracted features are classified through classifiers to achieve network intrusion detection. We select EfficientNet-B0 as the backbone network of the network intrusion detection model. We replace the last fully-connected layer of EfficientNet with two fully-connected layers and regard the corresponding Softmax activation function as the output layer [35]. This approach can effectively identify the different types of cyber-attacks. The loss function used in this model is the categorical cross-entropy loss function, which is defined as follows:

$$loss(pd, ed) = -\sum_{x} pd(x) \log(ed(x)) \tag{2}$$

where *pd(x)* is the real sample distribution probability, and *ed(x)* is the predicted sample distribution probability.

B. THE CALCULATION OF INTRUSION SECURITY INDICATORS

The evaluation results of intrusion security indicators are determined by the severity and impact of threats on network security. The severity of threats is determined by the number of occurrences of various attacks, as well as the severity factors of various attacks. Therefore, we use an efficient neural network to detect the current security status of the network, and identify the types and quantities of possible network attacks. The impact of threats is calculated by referring to the evaluation table of attack impact for the common vulnerability scoring system. The scores for confidentiality (C), integrity (I), and availability (A) are shown in Table 1.

TABLE 1. The evaluation results of attack impact.

Indicators	Impact Degree	Impact Value
C	None/Low/Hight	0/0.22/0.56
I	None/Low/Hight	0/0.22/0.56
A	None/Low/Hight	0/0.22/0.56

Using Table 1, we calculate the threat impact of each attack as follows:

$$P_i = C_i + I_i + A_i \tag{3}$$

The calculation method of intrusion security index is shown in Formula (4).

$$C_i = \frac{1}{N - M_n} \sum_{t=1}^{n-1} T_i \times P_i \tag{4}$$

where N represents the total number of network traffic data samples collected by the system platform, n represents the number of identifiable network attack types($n = 6$), M_n represents the number of occurrences of normal data, and P represents the threat impact score.

We obtained the scores for confidentiality, integrity, and availability of the information system through the deep learning model. Based on these scores, we calculated the threat impact of each attack. Then, according to these security indicators, we classify the security grade of the information system into three aspects: network intrusion security, device and computing security, and application and computing security. This comprehensive evaluation enables us to assess the overall security performance of the information system.

C. CONSTRUCTION OF SECURITY INDICATORS

Because the evaluation conclusion of the security grade protection object is influenced by the scores of multiple security control points, we adopt three different types of security factors as evaluation indicators, including network intrusion security, device and computing security, and application and computing security indicators. The secondary indicators of equipment and computing security, as well as application and computing security indicators, have determined at least five indicator values of control points. The security grade indicators of the evaluated information system can be finally represented as $C = \{C_1, C_2, C_3\}$, where C_1 represents the network intrusion security indicators, C_2 represents the device and computing security, and C_3 represents the application and computing security indicators. The indicators of device-and-computing security and application-and-computing security are respectively $C_2 = \{C_{21}, C_{22}, C_{23}, C_{24}, C_{25}, C_{26}\}$ and $C_3 = \{C_{31}, C_{32}, C_{33}, C_{34}, C_{35}, C_{36}, C_{37}\}$. For example, in the "device and computing security indicators", "identity authentication" is C_{21} and "access control" is C_{22}, as shown in Table 2.

TABLE 2. Composition of indicators of the measured control points in the measured system.

Other Security Indicators	
Security Grade Indicators	Security Control Point Indicators
Network Intrusion Security C_1	Network Intrusion Security C_{11}
Device and Computing Security C_2	Identification C_{21}
	Access Control C_{22}
	Security Audit C_{23}
	Malicious Code Prevention C_{24}
	Resource Control C_{25}
	Remaining Information Protection C_{26}
Application and Computing Security C_3	Identification C_{31}
	Access Control C_{32}
	Security Audit C_{33}
	Communication Integrity C_{34}
	Communication Confidentiality C_{35}
	Software Fault Tolerance C_{36}
	Resource Control C_{37}

From the security classification in the statistical list of information security measurement indicators and the analysis of the basic indicator points in Table 2, it can be seen that information security measurement consists of several security grade measurement indicators and security control frequency indicators. Fuzzy set theory can play a decisive role in network assessment, which essentially uses conceptual fuzzy theory to study the undetermined things and quantifies them into information that can be displayed by a computer through a matrix model. In this research, the fuzzy comprehensive judgmental decision-making method is used for information system measurement in the grade assessment conclusions. The grade assessment conclusions are:

"Excellent", "Good", "Fair" and "Poor". We calculate the score as follows:

$$100 - \frac{\sum_{k=1}^{p}\sum_{i=1}^{m(k)} \text{non-conformity assessment item weight}}{\sum_{k=1}^{p}\sum_{i=1}^{m(k)} \text{assessment item weight(W}i\text{)}} \times 100 \quad (5)$$

where p is the total number of items measured by the number of subjects, and m(k) is the number of corresponding subjects. The determination and comprehensive calculation models are shown in Table 3.

TABLE 3. Judgment basis for rating assessment conclusion.

Measured Value	Judgment Basis
Excellent	There are security problems in the tested object, but it will not lead to medium or high grade security risks, and the system has an overall score of 95 or more (including 95 points).
Good	There are security problems in the tested object, it will not lead to high grade security risks, and the overall score of the system is 85 or more (including 85 points).
Medium	There are security problems in the tested object, but it will not lead to medium or high grade security risks, and the overall score of the system is 75 or more (including 75 points).
Difference	There are serious security vulnerabilities in the tested object, and it will lead to high grade security risks such as information system paralysis, data loss and theft. The overall score of the tested object is less than 75.

From Table 3, we can see that the assessment results break the bottom line of 60 points, and the basis of determination is more scientific.

D. SECURITY ASSESSMENT BASED ON CART REGRESSION TREE

According to the quantitative indicators in Section II-B,We utilize a deep-learning model to predict the security indicators of the information system, obtaining the corresponding security evaluation scores. These predicted security evaluation scores are then used as input data of the regression tree model, so as to construct the security grade protection evaluation model.we use the CART regression tree to calculate the network situation value in that time period, and finally evaluate the attack degree of the network according to the network security level table. Assuming that the number of application systems used as evaluation case samples is and the number of security control points for each sample is, the data matrix can be constructed as shown in the Formula (6):

$$R = (r_{ij})_{m\times n} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \quad (6)$$

in which r_{ij} represents the score of the j-th security control point in the i-th application system. The calculation method is shown in Algorithm 1, and the corresponding comprehensive score of each application system is $Y = \{y_1, y_2, \cdots, y_m\}$.

Algorithm 1 Weighting of Measurement Indicators

Inputs: the weight coefficient of the evaluation object index μ_j, the matrix security control point f_{jl}, and the eigenvalue of the security control point λ_l

Outputs: three categories of indicators r_{hi}

(1) For each security control points $l \in [0, m]$:

Calculate the weight coefficient of secondary indicators

$$\mu_j = \sum_{l=1}^{m} f_{jl}\lambda l \quad (7)$$

(2) Calculate the weight values of secondary indicators. Let w_j be the weight value of each indicator, then

$$w_j = |\mu_j| \div \sum_{j=1}^{n} |\mu_j| (j = 1, 2, \cdots, n) \quad (8)$$

$$\sum_{i=1}^{n} w_i = 1 \quad (9)$$

(3) Finally, calculate the scores of the three categories of indicators C_{hi} in the following formula:

$$r_{hi} = \sum_{i=1}^{n} W_{ij}C_{ij} (i, j = 1, 2, \cdots, n) \quad (10)$$

Based on this feature space, we construct the corresponding CART regression tree. Traverse each feature in R (i.e., each column), take the corresponding values in feature j (r) as splitting points s in turn, and then divide the original feature space into two subspaces R_1 and R_2:

$$R_1(j, s) = \left\{ r | r^{(j)} \le s \right\} \quad (11)$$

$$R_2(j, s) = \left\{ r | r^{(j)} > s \right\} \quad (12)$$

First of all, we calculate the mean value of the output values R_1 and R_2 (i.e., the overall scores of each application system):

$$\hat{c}_m = \frac{1}{N_m} \sum_{r_i \in R_m(j,s)} y_i \,,\; r \in R_m \,,\; m = 1, 2 \tag{13}$$

And then, we can find the optimal split feature j' and optimal split point s' through the traversal in (1), so that the sum of the mean square errors of the output values R_1 and R_2 is minimized, as shown below:

$$\min_{j',s'} \left[\min_{c_1} \sum_{r_i \in R_1(j',s')} (y_i - c_1)^2 + \min_{c_2} \sum_{r_i \in R_2(j',s')} (y_i - c_2)^2 \right] \tag{14}$$

Partitioning R_1 and R_2 obtained by dividing R using j' and s' as left and right subtrees of the root node R [36]. Repeat the above steps on the left and right subtrees, and recursively generate the regression tree. To avoid overfitting the training dataset and enhance the generalization ability of the regression tree, the pre-pruning algorithm is used to reduce the number of leaf nodes [37]. The degree of reduction in mean square error is used as the criterion for pruning. If the difference between the sum of mean square errors of the left and right subtrees obtained by splitting a node and the mean square error before splitting is less than a certain threshold, the node will not be split. At this point, the node becomes a leaf node in the regression tree, and its value is the average of the recorded comprehensive scores. The final generated CART regression tree can be represented as follows:

$$f(x) = \sum_{m=1}^{M} \hat{c}_m I(x \in R_m) \tag{15}$$

in which *f(x)* represents the output of the CART regression tree model. This output value typically serves as a prediction for security assessment, used to evaluate the degree of network attacks or other relevant security performance indicators. Here, c_m stands for the value of a leaf node, signifying the predicted score for that node, while I is an indicator function indicating whether the input data point belongs to the region defined by the leaf node.

After completing the construction of the regression tree through the above steps, we input the scores of the security control points of the tested system to obtain the corresponding leaf node value. This value is the predicted value of the protection grade conclusion of the tested system.

III. RESULTS

A. MODEL TRAINING RESULTS

In this study, we trained four different models, namely VGG19, ResNet-50, XceptionNet, and EfficientNet, and then selected the model with the best performance. We used three Tesla V100 GPUs to conduct the training, and set the batch size as 32 to obtain pre-trained weights.The dataset was divided into three sets with a ratio of 7:1.5:1.5, resulting in 11,900 samples for the training set, 2,850 samples for the validation set, and 2,850 samples for the test set. The input images were resized to 224*224 with 3 channels. The learning rate for training was set to 0.001, and the Adam optimization algorithm was used, where the weight of first-order moment estimate is 0.9 (i.e., gradient's first moment) and the second-order one is 0.999 (i.e., gradient's second moment).During the training process, the ''early stop'' technique was employed. The model's performance on the validation set was continuously monitored, and if there was no improvement within a specified number of epochs, the training process would be terminated early, and the model with the best performance on the validation set would be saved. This approach prevents overfitting and excessive training of the model on the training data, ensuring better generalization performance. As shown in Fig.3, the number of training steps of VGG19 model is 930, with an accuracy of 0.803; the number of training steps of ResNet-50 model is 558, with an accuracy of 0.987; the number of training steps of XceptionNet model is 1116, with an accuracy of 0.988; and the number of training steps of EfficientNet model is 744, with an accuracy of 0.999.

As shown in Fig.4, the umber of training steps of VGG19 model is 930, and the minimum training gradient value is 0.01806. The number of training steps of ResNet-50 model is 558, and the minimum training gradient value is 0.00875. The number of training steps of XceptionNet model is 1116 and the minimum training gradient value is 0.00003. The number of training steps of EfficientNet model is 744 and the minimum training gradient value is 0.00044.

According to the training results, ResNet-50 and XceptionNet models perform well, with the accuracy reaching

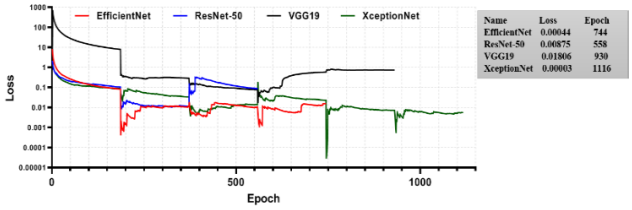

FIGURE 3. Comparison of training accuracy for four models.

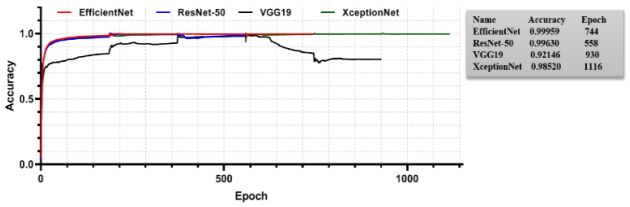

FIGURE 4. Comparison of training gradients for four models.

98.7% and 98.8% respectively. However, the performance of VGG19 model is relatively poor, the accuracy of which is only 80.3%. EfficientNet model performs best on the training set, with an accuracy of close to 100% and a small training gradient. It is the top-performing model. Through these training results, we can have a preliminary understanding of the performance of each model and select the best performing model for the generation of security situational assessment data.

B. SECURITY SITUATION ASSESSMENT RESULTS

The security grade protection evaluation model fully considers the influence of each security control point on the evaluation conclusion and controls the size of the regression tree through dimensionality reduction and pruning, ensuring the efficiency of the prediction process. Experiments show that this model can effectively predict the security evaluation conclusion scores of information systems. The results can provide quantitative indicators for the security evaluation of security control points in related business systems. The scores of security control points and comprehensive scores of 14 business systems are shown in Table 4.

TABLE 4. Sample evaluation cases.

System Number	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	Overall Score
1	4.95	14.09	3.64	-6.31	-6.49	-0.47	-3.32	-4.59	-2.77	1.86	70.23
2	-13.1	-3.25	-3.72	5.33	-5.67	3.32	-1.52	0.74	1.50	-0.64	93.74
3	6.60	4.51	6.61	-4.13	0.21	-3.65	-1.88	-1.62	11.72	8.02	70.07
4	4.13	1.00	2.97	-10.0	6.55	13.46	-6.40	-0.80	4.51	-3.19	75.38
5	-13.1	-2.97	-3.66	4.85	-5.64	3.54	-1.65	0.67	1.34	-0.43	93.87
6	-7.77	-4.71	-6.36	-7.57	6.35	3.390	11.18	-0.54	-2.46	2.10	88.33
7	15.94	-8.81	-11.8	1.17	-3.08	-0.23	-1.15	0.94	-1.63	-0.08	81.87
8	2.06	-1.92	-4.13	-3.69	-6.63	-10.9	7.05	-7.28	3.21	-3.69	80.89
9	-5.93	-8.30	5.09	-2.03	8.35	-6.38	-6.97	-3.46	-7.04	-2.37	81.45
10	2.02	-3.04	10.04	8.28	-4.04	2.12	4.02	7.19	-4.26	7.20	87.74
11	5.79	-10.9	12.54	3.22	1.32	0.09	1.79	-0.66	-0.32	-2.41	80.97
12	16.84	-8.33	-12.4	2.11	-1.23	0.00	-1.88	1.13	0.03	-0.49	81.26
13	1.54	17.16	-1.31	11.33	8.78	0.23	2.23	1.91	1.93	-2.62	76.37
14	-12.7	-4.30	-1.63	-2.63	1.86	-4.47	-3.57	1.94	3.93	0.54	81.89

Using the above samples as input, we construct a deep learning model based on the security control points of business systems, with a threshold of 0.5 as the mean square error reduction. Another 5 business systems are deployed in the same network domain, and the corresponding security control point scores are obtained through vulnerability analysis and penetration testing. Based on the scores, a test set is constructed to evaluate the trained decision-tree model. We use three indicators, namely mean absolute percentage error (MAPE), mean square error (MSE), and coefficient of determination (R2), to evaluate the fitting degree of the constructed model, which is calculated as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{16}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{17}$$

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{18}$$

where n is the number of samples, y_i is the true value of the comprehensive score for the i-th sample, $\hat{y}_i$ is the predicted value of the comprehensive score for the i-th sample, and $\bar{y}$ is the mean value of the true comprehensive score for the n samples.

According to Table 5, the mean absolute percentage error (MAPE) of the test result *MAPE* is 0.029, which is close to 0, indicating that the predicted values of the comprehensive score obtained through the model have a small error. The MAPE represents the accuracy of the model's predictions. In this case, a MAPE of 0.029 or 2.9% means that on average, the model's predictions of comprehensive scores for the business systems are very close to the actual scores. It indicates that the model can reliably estimate the security status of these systems, which is crucial in security assessment. A low MAPE suggests that the model's predictions are highly trustworthy and can be used safely.

TABLE 5. Evaluation results of test set.

System Number	y_i	$\hat{y}_i$	$(\hat{y}_i - y_i)^2$	$(y_i - \bar{y})^2$	$\left\|\frac{\hat{y}_i - y_i}{y_i}\right\|$	*MAPE*	*MSE*	R^2
1	88.5	88.035	0.216225	81	0.0052542			
2	71.13	75.12	15.9201	70.0569	0.0560945			
3	72.02	70.15	3.4969	55.9504	0.0259650	0.0293609	7.44151	0.9197
4	72.18	76.37	17.5561	55.5824	0.0580493			
5	93.67	93.805	0.018225	200.7889	0.0014412			

The Coefficient of Determination, often named R-squared, measures how well the independent variable (in this case, the model's predictions) explains the variance in the dependent variable (the actual comprehensive scores). An R^2 of 0.9 means that approximately 90% of the variance in the actual scores can be explained by the model's predictions. In the context of security assessment, it strongly indicates that the model is effective in capturing the underlying patterns and trends in the security control point scores. It suggests that the model is a valuable tool for evaluating the security posture of business systems.

In fact, these results mean that the proposed model is highly accurate and effective in predicting security grades. Security assessments are critical for identifying vulnerabilities and threats, and such an accurate model can assist organizations in making informed decisions about their security strategies and resource allocations. That MAPE is low but R^2 is high indicates that the model's predictions closely align with security conditions in the real world, which enhances its practical utility in security assessments and decision-making processes.

IV. DISCCUSION

As the main source of security elements in situational awareness, the accuracy of the intrusion detection system affects

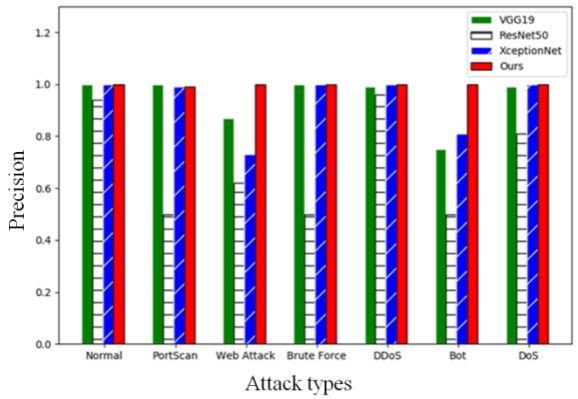

FIGURE 5. Recognition accuracy of different types of attacks.

TABLE 6. Accuracy and efficiency of four models.

Model	ModelSize(MB)	AT(ms)	ACC（%）
VGG19	114M	0.1259	99.87
Reset50	228M	0.081	92.61
Xceptionnet	220M	0.073	99.90
EfficientNet	*49M	*0.052	*99.93

* Represents the the optimal value.

the assessment of network security directly. The performance of the proposed method is evaluated based on the four commonly used performance indicators, including precision, recall, accuracy and F1-measure. Acc shows the ratio of correct predictions with respect to all samples. The precision defines the ratio of truly detected attacks to all packets that are classified as attacks. The F1 score is defined as the harmonic mean of precision and recall, mathematically represented as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad \Pr e = \frac{TP}{TP + FP}$$
$$\mathrm{Rec} = \frac{TP}{TP + FN} \quad F1 = 2 \times \frac{\Pr e \times \mathrm{Rec}}{\Pr e + \mathrm{Rec}} \tag{19}$$

where TP represents true positive, FN indicates false negative, FP represents false positive, and TN represents true negative.

In this experiment, we tested the four networks of VGG19, ResNet-50, XceptionNet, and EfficientNet respectively, and compared the model parameters, accuracy (ACC) and average time consuming (Average Time, AT) of a single detection. The experimental results are shown in Table 5. The model parameter size reflects the calculation complexity of the model. The more complex the mode is, the larger the parameter amount and the longer the calculation time will be. The optimal backbone network structure is determined by comparing the parameters and time consumption of different backbone networks. From the comparison of accuracy and efficiency in Table 6, it can be seen that the parameter scale of EfficientNet is significantly lower than those of VGG19, ResNet-50, and XceptionNet. The model shows high detection efficiency, with the detection accuracy of 99.93%. The results show that the EfficientNet-based anomaly detection model can effectively identify potential network intrusion attacks.

To test the effectiveness of the model, multiple data samples are randomly selected from different types of attack data in the test set. It can be seen from Fig.5 that ours scheme has significant effects on identifying different types of network attacks. Especially for Web Attack and Bot attack, other models have significant errors, while the experimental scheme shows high stability. Therefore, the experimental results show that the improved intrusion detection algorithm can provide effective data support for security situation assessment.

V. CONCLUSION

In this study, we have embarked on a journey of utilizing the power of deep learning models and data analysis techniques to enhance the assessment of network security grades. We constructed a unified data analysis system that integrates advanced models like VGG19, ResNet-50, XceptionNet, and the highly efficient EfficientNet. Our findings have demonstrated that due to its remarkable detection accuracy of 99.93% and low computational resource consumption, the EfficientNet model emerges as a promising choice for single detection tasks. Afterwards, we employed the CART regression tree model to perform network security situational assessments on 14 commercial systems. The test results of our model show that the Mean Absolute Percentage Error (MAPE) is 0.029 and the correlation coefficient R2 is 0.9. These metrics indicate that our proposed model has strong predictive power and emphasize its potential to revolutionize security assessments. The significance of this research is not just about model performance. We have also explored the application of deep learning models, particularly EfficientNet. These models can provide high accuracy but exhibit small training gradients, paving the way for more efficient training and deployment in real-world scenarios. Our novel approach involves integrating these predictive security scores derived from deep learning models into a regression tree model. This holistic security grade protection assessment system provides a comprehensive solution. It includes proactive defense mechanisms, swift event responses, and post-event audits, promising a multi-faceted approach to security.

Furthermore, our research may address several challenges in traditional security assessments, such as inconsistency, subjectivity, fuzziness, and poor adaptability. By introducing deep learning techniques and analyzing advanced data, we can dynamically link network security with fields, such as automatic production lines, machinery manufacturing, electrical and electronic engineering, and control science. Looking ahead, our research will lay the foundation for further advancements. Future studies should focus on refining deep learning models, exploring new data analysis

techniques, and expanding the scope of this assessment system to cover a broader spectrum of network security scenarios. Our work offers a promising trajectory towards smarter, more accurate, and more adaptable security evaluations, which is crucial in an era of constantly evolving complexity.

In conclusion, this research plays an important role in modernizing and improving network security assessments. Through deep learning, data analysis, and artificial intelligence, we have created a robust and efficient security grade protection assessment system that can cater to the diverse security assessment needs of various industries. The insights gained from this study have the potential to reshape the landscape of security assessments and empower the organizations to stay ahead in an ever-changing digital environment.

REFERENCES

[1] Z. Zhang, H. Ning, F. Shi, F. Farha, Y. Xu, J. Xu, F. Zhang, and K.-K.-R. Choo, "Artificial intelligence in cyber security: Research advances, challenges, and opportunities," *Artif. Intell. Rev.*, vol. 55, no. 2, pp. 1029–1053, Feb. 2022.

[2] W. Duo, M. Zhou, and A. Abusorrah, "A survey of cyber attacks on cyber physical systems: Recent advances and challenges," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 5, pp. 784–800, May 2022.

[3] R. Mai and M. Wu, "Research on the quantitative assessment and security measures of hierarchical network security threat situation," *Proc. IOP Conf. Series, Mater. Sci. Eng.*, vol. 750, no. 1, 2020, Art. no. 012171.

[4] B. Ksiezopolski, Z. Kotulski, and P. Szalachowski, "Adaptive approach to network security," in *Computer Networks. CN 2009. Communications in Computer and Information Science*, A. Kwiecień, P. Gaj, and P. Stera, Eds. vol. 39. Berlin, Germany: Springer, 2009, pp. 83–93.

[5] C. Hu and C. Lv, "Method of risk assessment based on classified security protection and fuzzy neural network," in *Proc. Asia–Pacific Conf. Wearable Comput. Syst.*, Shenzhen, China, Apr. 2010, pp. 379–382.

[6] C. Cai, "Study on the protection of e-government external network level protection," in *Proc. 7th Int. Conf. Mechatronics, Comput. Educ. Informationization (MCEI)*. Amsterdam, The Netherlands: Atlantis Press, 2017, pp. 415–417.

[7] D. Sun and B. Wang, "Research on the design of the implementation plan of network security level protection of information security," in *Proc. 7th Int. Symp. Mechatronics Ind. Informat. (ISMII)*, Zhuhai, China, Jan. 2021, pp. 227–231.

[8] W. W. Zhi, X. X. Zhou, and L. Yang, "Application of fuzzy comprehensive method and analytic hierarchy process in the evaluation of network security level protection research," in *Proc. Int. Conf. Mech. Eng., Intell. Manuf. Automat. Technol. (MEMAT)*, Guilin, China, 2021, Art. no. 012187.

[9] G. Savva, K. Manousakis, and G. Ellinas, "Confidentiality meets protection in elastic optical networks," *Opt. Switching Netw.*, vol. 42, Nov. 2021, Art. no. 100620.

[10] B. Wang and Y. Xu, "Design and implementation of cloud computing network security level protection evaluation system based on Python," in *Proc. 5th Int. Conf. Comput. Inf. Sci. Appl. Technol. (CISAT)*, Oct. 2022, Art. no. 1245141.

[11] Y. Jin, Y. Shen, G. Zhang, and H. Zhi, "The model of network security situation assessment based on random forest," in *Proc. 7th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Aug. 2016, pp. 977–980.

[12] H. Yang, Y. Jia, and W. H. Han, "Calculation of network security index based on convolution neural networks," in *Proc. Int. Conf. Artif. Intell. Secur.* Cham, Switzerland: Springer, 2019, pp. 530–540.

[13] Z. Yuan, Y. Lu, Z. Wang, and Y. Xue, "Droid-sec: Deep learning in Android malware detection," in *Proc. ACM Conf. SIGCOMM*. New York, NY, USA: Association for Computing Machinery, Aug. 2014, pp. 371–372.

[14] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," *SESA*, vol. 16, no. 9, p. e2, 2016.

[15] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *Proc. Int. Conf. Wireless Netw. Mobile Commun. (WINCOM)*, Fez, Morocco, Oct. 2016, pp. 258–263.

[16] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," in *Proc. Int. Conf. Platform Technol. Service (PlatCon)*, Jeju, South Korea, Feb. 2016, pp. 1–5.

[17] M.-J. Kang and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS ONE*, vol. 11, no. 6, Jun. 2016, Art. no. e0155781.

[18] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[19] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection," *IEEE Access*, vol. 6, pp. 52843–52856, 2018.

[20] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *J. Inf. Secur. Appl.*, vol. 50, Feb. 2020, Art. no. 102419.

[21] H. Bao, H. He, Z. Liu, and Z. Liu, "Research on information security situation awareness system based on big data and artificial intelligence technology," in *Proc. Int. Conf. Robots Intell. Syst. (ICRIS)*, Jun. 2019, pp. 318–322.

[22] R. Pagey, M. Mannan, and A. Youssef, "All your shops are belong to US: Security weaknesses in e-commerce platforms," in *Proc. ACM Web Conf.*, Apr. 2023, pp. 1–14.

[23] L. T. Swetnam, "CyVerse: Cyberinfrastructure for open science," *bioRxiv*, p. 2023-06, 2023.

[24] J. Cai, "Recurrent neural network technology in wireless communication network security risk prediction," in *Proc. 6th Int. Conf. Mechatronics Intell. Robot. (ICMIR)*, vol. 12301, Nov. 2022, pp. 1–9.

[25] A. K. Das, S. Roy, E. Bandara, and S. Shetty, "Securing age-of-information (AoI)-enabled 5G smart warehouse using access control scheme," *IEEE Internet Things J.*, vol. 10, no. 2, pp. 1358–1375, Jan. 2023.

[26] J. Zhang and Q. Liu, "New key management scheme lattice-based for clustered wireless sensor networks," *PLoS ONE*, vol. 18, no. 8, Aug. 2023, Art. no. e0290323.

[27] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," *Peer Peer Netw. Appl.*, vol. 12, no. 2, pp. 493–501, Mar. 2019.

[28] A. D. Jasim, "A survey of intrusion detection using deep learning in Internet of Things," *Iraqi J. Comput. Sci. Math.*, vol. 3, no. 1, pp. 83–93, 2022.

[29] W. Lee, W. Fan, M. Miller, S. J. Stolfo, and E. Zadok, "Toward cost-sensitive modeling for intrusion detection and response," *J. Comput. Secur.*, vol. 10, nos. 1–2, pp. 5–22, Jan. 2002.

[30] A. Demirpolat, "Software-defined network security," *Enabling Technologies and Architectures for Next-Generation Networking Capabilities*. Hershey, PA, USA: IGI Global, 2019, pp. 232–253.

[31] T. Naeem, "Common security issues and challenges in wireless sensor networks and IEEE 802.11 wireless mesh networks," *Int. J. Digit. Content Technol. Appl.*, vol. 3, no. 1, pp. 88–93, Mar. 2009.

[32] O. Belarbi, A. Khan, and P. Carnelli, "An intrusion detection system based on deep belief networks," in *Proc. Int. Conf. Sci. Cyber Secur.* Cham, Switzerland: Springer, 2022, pp. 377–392.

[33] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[34] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2017, pp. 1222–1228.

[35] S. H. D. Lin, C. Feng, Z. H. D. Chen, and K. X. Zhu, "An efficient segmentation algorithm for vehicle body surface damage detection," *Data Acquisition Process.*, vol. 36, no. 2, pp. 260–269, 2021.

[36] S. Sathyadevan and R. R. Nair, "Comparative analysis of decision tree algorithms: ID3, C4.5 and random forest," in *Computational Intelligence in Data Mining*, vol. 1. India: Springer, 2015, doi: 10.1007/978-81-322-2205-7_51.
[37] F. Esposito, D. Malerba, G. Semeraro, and J. Kay, "A comparative analysis of methods for pruning decision trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 476–493, May 1997.

SHAODAN LIN was born in Fuzhou, Fujian, China, in 1979. He received the master's degree in computer technology from Fuzhou University, China, in 2010, and the Ph.D. degree from Fujian Agriculture and Forestry University, China, in 2021. He was a Professor with the Fujian Chuanzheng Communication College, China, from 2021 to 2023. He is the author of two books, has published over 30 papers, and holds two invention patents. His research interests include network security, deep learning, and computer vision.

CHEN FENG was born in Hubei, China, in 1996. He received the B.Sc. degree in mathematics and in applied mathematics from Yangtze University, China, in 2018, and the M.S. degree from Fujian Normal University, China, in 2021. Since 2021, he has been a Teacher with the Information Engineering Department, Fuzhou Polytechnic. His current research interests include deep learning and computer vision.

TAO JIANG was born in Fuzhou, Fujian, China, in 1973. He received the master's degree in software engineering from Fuzhou University, in 2011. From 2020 to 2023, he was a Senior Engineer with the Fujian Chuanzheng Communication College, responsible for the school's information planning and management. His research interests include software engineering and educational informatization.

HUASONG JING (Member, IEEE) was born in Fuzhou, Fujian, China, in October 1971. He received the bachelor's degree from Fujian Normal University and the master's degree from Huaqiao University. He has a postgraduate education. He is currently a Senior Engineer. He is also the Chairperson of Fujian Zhong Xin Network Security Information Technology Company Ltd. He has published over ten academic papers and has been involved in more than 20 national, provincial, and municipal research projects. His independently developed products have been granted over 160 software copyrights and eight invention patents and 25 utility model patents. His research interests include network and data security.