IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 June 2024, accepted 21 July 2024, date of publication 1 August 2024, date of current version 3 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3436541

RESEARCH ARTICLE

The Detection of Abnormal Behavior by Artificial Intelligence Algorithms Under Network Security

HUI CAO
Library and Information Center, Hunan Railway Professional Technology College, Zhuzhou 412001, China
e-mail: nucoo@126.com

ABSTRACT With the continuous evolution of network attack methods, traditional rule-based and signature-based security strategies are becoming increasingly hard to deal with increasingly complex network threats. The research focuses on the problem of network traffic anomaly detection in network security, and proposes an improved Transformer and Generative Adversarial Networks network traffic anomaly detection model. The innovation lies in utilizing the Patch segmentation in the Transformer module to reduce information loss, while introducing random masked data blocks to enhance the anti-interference ability of Generative Adversarial Networks, and proposing a class balance model. Therefore, a Transformer Multi Receive Field Fusion (Trans-M) model for network traffic anomaly detection is constructed. The performance test results showed that after category balancing, the accuracy, recall, and F1-score of each model were been significantly improved. The accuracy of the Trans-M model on the balanced dataset arrived 98.12%, an improvement of 8.59% compared to before balancing. The recall rate of the Trans-M model was improved by 8.62% to 97.86%. On Balanced F Score (F1-score), the highest score of the Trans-M model was 98.46%, which was 8.18% higher than before balancing. The experiment outcomes demonstrate that the raised network traffic anomaly detection system is superior to common anomaly traffic detection models and can meet the actual network security protection needs.

INDEX TERMS AI, network security, traffic detection, GAN, Transformer.

I. INTRODUCTION

In the digital age, network security has become a core issue in the global information technology field. With the continuous evolution of network attack methods, traditional rule-based and signature-based security strategies are becoming increasingly hard to deal with increasingly complex network threats. Especially in the context of the speed growth of big data and cloud computing technology, traditional network security defense methods are no longer able to meet the growing demand for data security. Recently, the growth of Artificial Intelligence (AI) has brought breakthroughs to network security, especially in the field of abnormal behavior detection, showing great potential [1], [2], [3]. Network Traffic Anomaly Detection (NTAD) is an important branch in the field of network security, with the main purpose of identifying malicious activities in the network, such as Distributed Denial of Service (DDoS) attacks, phishing, and malware propagation. Traditional anomaly detection systems often rely on specific data patterns or behavioral rules, but these methods have limited effectiveness when facing advanced and diverse attack methods. Therefore, research explores to use AI algorithms to raise the accuracy and adaptability of detection systems. The research aims to design a novel NTAD system by combining Generative Adversarial Networks (GAN) and Transformers. GAN have shown excellent performance in simulating complex data distributions due to their powerful generation ability and self-learning characteristics [4]. As an efficient sequence processing model, Transformer has significant advantages in processing time series data [5]. By combining the merits of GAN and Transformer, the research targets to build an intelligent detection

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

system that can effectively simulate normal network traffic characteristics while accurately identifying and responding to abnormal behavior.

The research is mainly composed of four parts. The first part introduces network malicious activities and AI algorithms. The second part constructs an NTAD system based on GAN and Transformers. The third part tests and analyzes the effectiveness of the model and algorithm. The fourth part summarizes and discusses the content of the article.

II. RELATED WORKS

With the development of technology, network security has developed into a hot topic in recent years. Mabodi et al., aiming at the characteristic of Internet of Things (IoT) providing universal intelligent prediction for nodes, used the method of checking node information to reduce gray hole attacks. This method completed the process of identifying node trust in the IoT, examining routing, and discovering gray hole attacks. The experimental results showed a high detection rate [6]. Dlshad and Askar analysis of IoT network security and deep learning models suggested that utilizing the extensive connectivity capabilities of the IoT had a certain promoting effect on network security management and development, and improved the service quality of user experience [7]. Sun et al. proposed a content-based image retrieval (CBIR) and sustainable computing wireless network model to enhance the resistance of WSN to shortcut key attacks. They optimized CBIR was used to strengthen computational robustness and combined WSN with computational models to improve the robustness of the raised method. Finally, the model's accuracy and robustness were experimentally verified [8]. Xu et al. put forward a novel stock price trend prediction network with Reinforcement Learning (RL) and combined with Attention Mechanism (AM), using a bidirectional Gated Recurrent Unit (GRU) network to cut down the noise of news texts and learn news level representations with richer semantics. The experiment findings expressed that this model was far superior to existing models and had better performance [9].

AI algorithms have been widely applied in various fields. The construction of smart cities has been facilitated by the utilization of sensor systems and information and communication technology, which has enhanced the convenience and security of user life. The simultaneous adoption of 5G technology to enhance communication infrastructure, reduce commuting times, and reinforce public safety has the potential to influence the advancement of intelligent perception and smart cities [10]. Furthermore, the security and privacy issues associated with urban life can be mitigated through the application of IoT and blockchain technologies. The deployment and security framework of blockchain technology can be utilized to access and evaluate smart cities, thereby improving the security and reliability of smart cities and communication [11]. In the field of medical information applications, numerous studies have employed machine learning, deep learning, and medical IoT technologies to enhance the application of computer intelligence and apply AI to healthcare, thereby optimizing the functionality of healthcare systems [12]. Lee et al. classified detection schemes for different deep learning networks in the process of network intrusion detection using AI deep learning techniques, and compared evaluation metrics and datasets [13]. Nevertheless, AI algorithms were extensively utilized in the domain of communication security. However, the deployment of network security solutions was proven to be inadequate in addressing the evolving nature of network threats and attack vectors. Therefore, in terms of network traffic attacks, the utilization of AI still lacked rich development and design. Therefore, research needs to improve detection models in deep learning technology to enhance the accuracy of attack defense. For network intrusion detection system, M. Y. Aldarwbi et al. proposed to use advanced speech recognition deep learning technology to detect network traffic intrusion, and compared the performance with benchmark data set. The results proved that the deep learning intrusion detection algorithm had high accuracy and low false alarm rate [14]. The dragonfly and ant lion optimization algorithms, as well as the minimization of testing costs and time for on-chip systems, were employed to optimize the scheduling time of benchmark circuits. The enhanced ant colony optimization algorithm also reduced the testing time to a certain extent. For electronic component manufacturing faults in on-chip systems, the improved artificial bee colony algorithm was used to optimize the testing scheduling algorithm for the benchmark circuit, resulting in the least algorithm testing time and reducing the cost of chip testing [15], [16], [17]. Tian et al. believed that existing mechanical fault diagnosis methods mainly dealt with noise signals in the time or frequency domains without fully considering the noise features. Therefore, a noise resistant wavelet-based self-attention network was proposed. This method combined a frequency-oriented fusion module and a Transformer module to suppress noise in the two fields. The experiment findings denoted that this method had superior performance under different signal-to-noise ratios [18]. Tang Z et al. explored the ability of Transformers in pedestrian attribute recognition tasks and proposed a Dual Relationship Transformer (DRTormer) framework. An Attribute Relationship Module (ARM) with Transformer encoder was designed using Visual Transformer (ViT) as the feature extractor to capture the relationships between attributes. The results demonstrated the superiority of the proposed DRTormer over existing methods [19]. Li et al. proposed a novel topology optimization framework with Subset Simulation (SS) by combining SS with GAN. The topology optimization algorithm guided by SS and GAN could promote efficient topology optimization of periodic structures. The effect and efficiency of the raised method was proved through topological optimization of two-dimensional periodic structures [20]. Ezaldeen et al. applied the NPSO algorithm to find the significance of relationship types between concepts to complement a simulated recommendation system

with the highest ranking for dynamic learners. They investigated CLM and ECLM concept models. The simulation outcome revealed that ECLM surpassed other existing methods, with a Mean Reciprocity Rate (MRR) of 0.780 [21]. Oseni et al. proposed an interpretable deep learning-based intrusion detection framework for IoT systems, and utilized dataset validation to improve the transparency and resilience of the detection system, reducing network attacks [22]. Shrivastava and Kamble proposed the use of deep learning algorithms, Convolutional Neural Networks (CNN), GANs, etc. to classify and test unsupervised data in order to demonstrate the effectiveness of deep learning strategies against network attacks [23].

In summary, many researchers have conducted extensive design and research on attack detection and protection in the field of network security. Combining different computer networks with sensor systems, conduct noise testing on specific application areas to determine network security faults. However, the applicability and detection accuracy of these methods and strategies still need to be improved. Therefore, the study proposes an NTAD based on GAN and Transformers, which provides complex technical references for subsequent detection systems targeting network attacks.

The research innovation mainly includes three points. Firstly, based on the problem of extracting local features from network traffic data, the study used Transformer module and dilated convolution module to simultaneously extract global and local feature information, thereby expanding the receptive field area of network traffic anomalies. Secondly, in the case of minority class samples and duplicate data, the use of Random Masked Data Blocks (RMD) in class balance design is studied to enhance the model's adaptability to unknown data inputs and increase the detection model's anti-interference ability. Finally, a two-way adversarial discrimination (TAD) network is combined to distinguish between real data and generated data, thereby improving the recognition performance and robustness of the detection model.

The research contributions are mainly divided into three aspects. Firstly, the proposes an NTAD based on GAN and Transformers solves the local optimization problem of traditional methods in traffic data processing, and changes the situation of data interference and overfitting in data feature extraction. Secondly, based on the deep learning model, the research combines GANs to expand the adaptability of the model to complex network environments. By utilizing the Transformer module and class balance design, the feature information of the data is effectively processed to cope with various complex abnormal behaviors, thereby enhancing the accuracy and robustness of NTAD. Finally, through the investment and design of computer network security technology level, improve the security defense level and attack detection measures, to a certain extent, ensure a safe and healthy Internet environment, and reduce the loss of economic and digital information.

III. CONSTRUCTION OF AN NTAD SYSTEM BASED ON GAN AND TRANSFORMERS

The research focuses on the problem of NTAD in network security, and proposes an NTAD model that integrates improved Transformer and GAN. Firstly, to overcome the challenges of traditional network traffic detection techniques in dealing with unknown attacks and enhance the adaptability of the model to complex network environments, a Transformer-Multi Receive Field Fusion (Trans-M) is proposed to conduct in-depth analysis and modeling of network traffic data. Furthermore, to address the issue of category imbalance, an RM-DDCG model with an improved Deep Convolutional Generative Adversarial Networks (DCGAN) is put forward.

A. DESIGN OF NTAD MODEL INTEGRATING PATCH SEGMENTATION AND TRANSFORMER

The rapid development of network technology has led to a corresponding evolution in the techniques employed by attackers, with traditional NTAD techniques facing significant challenges in keeping pace. Network traffic data can be utilized to represent the data transmission and information flow of network devices. This data is captured and analyzed by network monitoring tools, processed, and extracted in order to determine the security of the data. At present, network traffic comprises both normal and abnormal traffic. Abnormal traffic is primarily constituted by data flows originating from network failures and network attacks. The research focuses on the type of NTAD, which is network attack. In response to common DDoS attacks, an NTAD model is designed to improve traffic monitoring performance and address more complex and ever-changing network security threats. The introduction of attention technology into learning models serves to enhance the connectivity of information technology, improve the quality and robustness of feature extraction. Transformer is based on Self-attention Mechanism (SAM), which has significant advantages over traditional sequence processing models such as Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM). Unlike RNN and LSTM, which rely on the temporal progression processing information of sequences, Transformer can parallelly process the entire sequence through SAM, significantly improving computational efficiency [24], [25], [26]. This enables Transformers to effectively reduce time delays and information loss when processing long sequence data. However, the disadvantage of Transformers is that they require relatively high computational resources, especially when dealing with very long sequences. In addition, due to its parallel processing characteristics, Transformer may not be as effective in capturing temporal dynamic dependencies in sequences as RNN or LSTM in some cases [27], [28]. To overcome the limitations of Transformer in processing local feature information, the study combined Convolutional Block Attention Module (CBAM) to raise the model [29], [30]. By using the Patch

Segmentation (PSE) algorithm to reduce the risk of information loss, and introducing the Multi Receiver Field Fusion (MRFF) algorithm, the information loss between patches is reduced, and more useful information is retained to improve the quality and robustness of features [31], [32]. In addition, to simplify the model structure, the decoder part is removed from the Transformer, making the model more efficient. The diagram of the optimized Trans-M model is denoted in Figure 1.

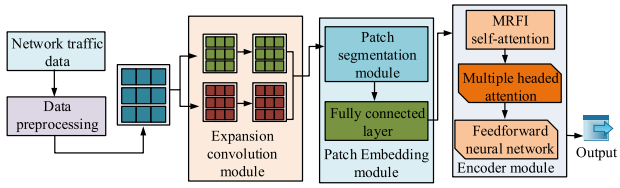

FIGURE 1. Trans-M model structure.

The model contains four key components, namely data pre-processing and Sequence to Image (Seq2Img) module, dilated convolution module, patch embedding module, and encoder module. The data pre-processing part first denoises the input network traffic data, including handling null and outliers, and standardizes the data. The Seq2Img module converts the processed data sequence into a matrix shape to adapt to subsequent convolution operations. The dilated convolution module extracts local features of input data through dilated convolution operations, where the expansion rate r controls the size of the receptive field. The patch embedding module further divides the data into small blocks and extends the feature dimension through a fully connected layer. The number of stacked encoders n indicates the depth of the model.

Firstly, due to the serious impact of outliers and null values in network traffic on data quality and detection accuracy, the Gaussian distribution of probability density function is studied in the data preprocessing stage to handle outliers in network traffic data. Concurrently, the mean filling method is employed to address the null value issue, thus guaranteeing the optimal functioning of the Trans-M model. The study also utilized the min max normalization method to standardize data and optimize model training. Secondly, the structure of Seq2Img and the dilated convolution module is shown in Figure 2.

As shown in Figure 2 (a), the input network traffic data sequence $\{x_1, x_2, \ldots, x_t\}$ undergoes matrix transformation of $H \times W$. When $H = [t/W]$, $n = (H-1)W + 1$, and t are integer multiples of W, they can be transformed into matrix form. Otherwise, matrix transformation can be achieved by adding 0 after the data sequence. In Figure 2 (b), the structure of the dilated convolution module is shown. After data pre-processing, the CxHxW dimensional data stream is transformed through Seq2Img. The dilated convolution module uses dilated convolutions with expansion rates of 1 and 2, and the data is processed in multiple stages using 1×1 and 3×3 convolution kernels. The output of stage one is X_1^1, where

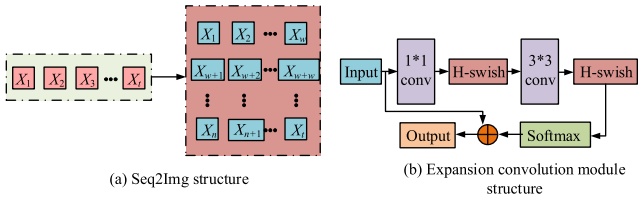

FIGURE 2. The structure of Seq2Img and the dilated convolution module.

each element belongs to a $R^{C\times H\times W}$ dimensional space. The output of stage two is X_2^1, and each element belongs to the $R^{2C\times H\times W}$ dimensional space. The dilated convolution module enhances nonlinear feature representation through H-swish activation function. Moreover, it introduces Softmax activation function for normalization after convolution operation to enhance the classification effect of the model. Then, to address the issue of gradient vanishing in deep networks, residual network design is incorporated into this structure. In the PSE stage, the data is divided into multiple patches using the SoftPool algorithm, and the vector representation of each patch is calculated. Firstly, the input data is divided into $N = \frac{H}{S} \times \frac{W}{S}$ non overlapping patches, and then the SoftPool algorithm is used to calculate the vector representation of each patch in different channels. In the fully connected stage, these vectors are input into a fully connected layer network, expanding the number of channels and obtaining the final patch embedding to extract local features while preserving key channel information. Finally, to strengthen the model's ability to process network traffic data, the Encoder-based MRFF algorithm is used to improve the Transformer structure. The SAM structure of MRFF is shown in Figure 3.

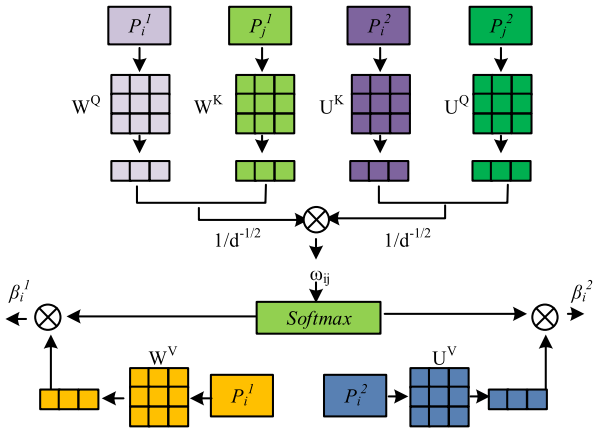

FIGURE 3. Structure of MRFF self-attention mechanism.

As shown in Figure 3, MRFF self-attention extracts long-range dependencies in features through a multi-head AM, where each head is responsible for a different information flow. The MRFF structure contains multiple self-attention units that can process feature information in parallel, and each self-attention unit can learn the correlation between different parts of the sequence. The Encoder module consists of a multi head AM and a feed-forward network, which

integrates information through the feed-forward network, while introducing residual connections and layer normalization to stabilize the training process and maintain the coherence of feature information. Information from different receptive field positions is combined to obtain the overall correlation between two patches, the calculation method for this process is shown in formula (1).

$$\omega_{ij} = \frac{1}{\sqrt{d_{\bmod el}}}\left(\left(P_i^1 W^Q\right)\left(P_j^1 W^K\right)^T\right) + \left(\left(P_i^2 U^Q\right)\left(P_j^2 U^K\right)^T\right) \tag{1}$$

In formula (1), i and j mean the ith and jth patches, respectively. P_i^1 and P_j^1 represent the embedding vectors of the ith and jth patches when the expansion rate is 1. P_i^2. P_j^2 represent the embedding vectors of the ith and jth patches when the expansion rate is 2. $d_{\bmod el}$ represents the dimension of the pattern hidden layer. W^Q and U^Q mean the matrix of the query. W^K and U^K represent the matrix of the key, and the P_i^1 dot product output formula is shown in formula (2).

$$\beta_i^1 = \sum_{j=1}^{n1} \frac{\exp\left(\omega_{ij}\right)}{\sum_{j'=1}^{n1}\exp\left(\omega_{ij'}\right)}\left(P_i^1 W^V\right) \tag{2}$$

In formula (2), β_i^1 represents the dot product output of the patch vector P_i^1. $n1$ represents the number of patches in the receptive field. The P_i^2 dot product output formula is shown in formula (3).

$$\beta_i^2 = \sum_{j=1}^{n2} \frac{\exp\left(\omega_{ij}\right)}{\sum_{j'=1}^{n2}\exp\left(\omega_{ij'}\right)}\left(P_i^2 U^V\right) \tag{3}$$

In formula (3), β_i^2 represents the dot product output of the patch vector P_i^2. $n2$ represents the number of patches in the receptive field, which is equivalent to $n1$. exp represents an exponential function with a constant e as the base. The self-attention calculation is indicated in formula (4).

$$A(Q, K, V) = soft\max\left(\frac{Q_1 K_1^T + Q_2 K_2^T}{\sqrt{d_{\bmod el}}}\right)$$
$$V = \left[\beta_1^1, \ldots, \beta_{n1}^1; \beta_1^2, \ldots, \beta_{n2}^2\right] \tag{4}$$

In formula (4), $Q = [Q_1; Q_2]$, $K = [K_1; K_2]$, and $V = [V_1; V_2]$ represent the concatenation of query, key, and value matrices under two receptive fields, respectively. The feature correlation is obtained by multiplying Q and K. Moreover, a large value indicates a high correlation. The result is divided by the square root of the hidden layer dimension and Softmax is calculated to determine the weight. Finally, the Softmax result is multiplied by V, emphasizing key features. The calculation formula for head SAM is shown in formula (5).

$$head_j = A\left(QW_j^Q, KW_j^K, VW_J^V\right) \tag{5}$$

In formula (5), W_j^Q, W_j^K, and W_J^V are linear transformation matrices for query, key, and value, respectively. The multi head AM calculation formula is shown in formula (6).

$$Multihead = Concat(head_1, head_2, \ldots, head_n)W^o \tag{6}$$

In formula (6), $Concat()$ represents the matrix concatenation method, and W^o represents the additional weight matrix. By concatenating n SAM and multiplying them with W^o, subspace attention is compressed to extract duplicate features. The output result of the multi head AM is utilized as the input of the feed-forward neural network, and the calculation is shown in formula (7).

$$FeedForward(x) = f(xW_1 + b_1)W_2 + b_2 \tag{7}$$

In formula (7), f is the activation function. W_1 and W_2 are weight parameters. b_1 and b_2 are bias parameters, respectively.

B. DESIGN OF CATEGORY BALANCE MODEL BASED ON GAN OPTIMIZATION

NTAD plays a pivotal role in ensuring the security of urban physical networks. Given the imbalanced distribution of real network traffic data, it is of paramount importance to construct a high-precision NTAD model under conditions of imbalanced data. To further improve the accuracy of network traffic detection, a category balancing model based on DCGAN is proposed to address the issue of category imbalance. Compared with generative models such as Autoencoders (AE) or Variational Autoencoders (VAE), DCGAN performs better in generating novelty and diversity because it can capture deeper levels of data distribution features [33], [34]. However, the training of DCGAN is relatively complex, susceptible to mode collapse, and sensitive to hyper-parameter selection in the early stages of training, resulting in insufficient stability. The category balance model proposed in the study enhances anti-interference ability and enhances the diversity of generated data by introducing RMD [35], [36]. In addition, to effectively reduce discrimination errors, a TAD model is designed to improve data discrimination accuracy, enhance model robustness and reliability through two discriminators. The structure of the RM-DDCG model is indicated in Figure 4. As shown in Figure 4, the structure diagram of the RM-DDCG model contains three main parts: generator, RMD, and TAD model. The generator module uses DCGAN to generate noise through random sampling of Gaussian distribution, which is then converted into a data matrix through deconvolution layers. The RMD module is located after the last convolutional layer to generate images of the same size as the real data. The function of the RMD module is to divide the input feature tensor into multiple sub regions. It applies random masks to each sub region, generate partially covered features through dot multiplication, and then enhance the robustness of the model through recombination, convolution processing, and spatial dropout techniques. The TAD module contains two discriminators that distinguish between the real and generated data processed by the RMD module, and calculate

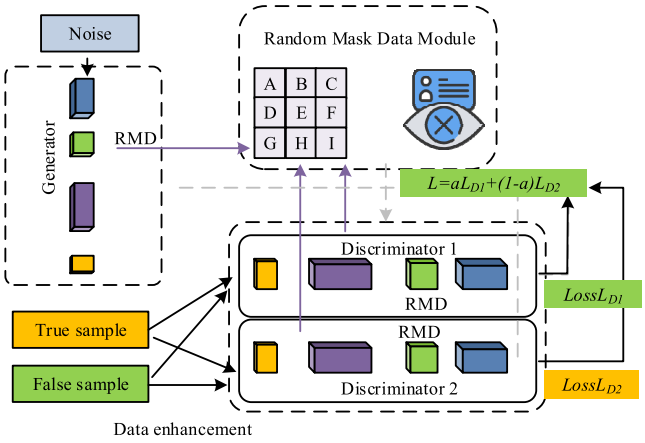

FIGURE 4. RM-DDCG model structure.

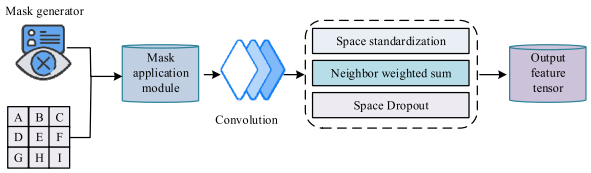

FIGURE 5. RMD model structure.

two losses $LossL_{D1}$ and $LossL_{D2}$. The discriminator distinguishes authenticity through feature extraction, reducing the target dimension during the process. The generator synthesizes targets is based on noise, gradually increasing their dimensions. DCGAN combines CNN and GAN to optimize feature extraction, reduce fully connected layers, and use deconvolution and batch normalization techniques to improve detail preservation and training stability. GAN is a model comprising two networks: generator G and discriminator D. During training, generator G selects synthetic targets based on noisy data and increases the dimensionality of the latter. The discriminator D then extracts target features to determine whether the target is true or false. During the judgment process, the dimension of the target continuously decreases. The difficulty of training GANs and the challenge of synthesizing discrete samples represent significant obstacles to overcome. Therefore, the generator combines DCGAN, utilizes convolutional structures to generate high-dimensional data from noise, and integrates RMD modules to improve data quality. Furthermore, the generator comprises a series of deconvolution layers and deconvolution layers, which are employed to map low-dimensional noise to high-dimensional space. In order to ensure that the experimental data is consistent with the real data size, the RMD module and deconvolution layer are added to the convolutional layer at the end of the DCGAN generator. This enhances the generator's generalization ability and robustness, thereby improving the quality and diversity of the generated data. The structure of the RMD model is denoted in Figure 5. As shown in Figure 5, the RMD model includes four stages: data input, random mask generation, recombination, and subsequent processing. In the data input stage, the original feature matrix is received. In the random mask generation stage, a mask is created through Bernoulli distribution and multiplied with the input data to achieve random masking of features. In the recombination stage, the masked features are integrated. Finally, after ReLU activation function and convolution processing, Dropout is used to randomly discard some connections and output the final feature representation. Due to the fact that the feature weighting and masking modules of the RMD module can affect the feature representation ability of the discriminator, an additional RMD loss function needs to be added to train the RMD module. The RMD module calculates the loss function of the discriminator using the formula (8).

$$RMD_{lossD} = s \times (rmd\_f - in)^2 \tag{8}$$

In formula (8), $lossD$ represents the loss of the discriminator. s represents the weighting coefficient. rmd_f represents the features processed by the RMD feature weighting module. in is the input. The RMD module's loss function for the generator is shown in formula (9).

$$RMD_{lossG} = s \times (rmd\_f - in)^2 \tag{9}$$

The TAD module enhances its ability to distinguish data authenticity through a dual discriminator network, where each discriminator is embedded with an RMD module to enhance feature expression. Discriminator 1 directly processes input samples, while discriminator 2 first performs transpose operations to enhance the data. This design enables two discriminators to capture data features from different perspectives and provide accurate discrimination scores. Through this game mechanism, the generator learns to generate data that is close to the true distribution, ultimately promoting the model's ability to generate high-quality data. The calculation for the Loss function L_{D1} of the discriminator is denoted in formula (10).

$$\begin{aligned} D1_{loss} = {} & BCE_{loss}(D1(x), r) + BCE_{loss}(D1(G(s)), f) \\ & + \lambda_{rmd} * RMD_{lossD1} \end{aligned} \tag{10}$$

In formula (10), BCE_{loss} represents the binary cross entropy Loss function. r and f represent the labels of real data and production data. $D1(x)$ is the discrimination result of discriminator 1. $D1(G(s))$ is the discrimination result of discriminator 1 on output $G(s)$. λ_{rmd} is the RMD weight, and RMD_{lossD1} is the loss function of RMD on discriminator 1. The calculation for L_{D2} is shown in formula (11).

$$\begin{aligned} D2_{loss} = {} & BCE_{loss}(D2(x), r) + BCE_{loss}(D2(G(s)), f) \\ & + \lambda_{rmd} * RMD_{lossD2} \end{aligned} \tag{11}$$

In formula (11), $D2(x)$ is the discrimination result of discriminator 2. $D2(G(s))$ is the discrimination result of discriminator 2 on output $G(s)$. λ_{rmd} is the RMD weight, and RMD_{lossD2} is the loss function of RMD on discriminator 2. The final loss L is shown in formula (12).

$$L = \alpha * D1_{loss} + (1 - \alpha) * D2_{loss} \tag{12}$$

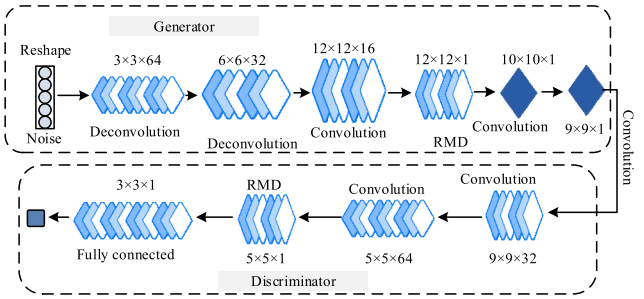

FIGURE 6. Convolution process of generative network and discriminative network.

In formula (12), α is the fusion coefficient. In RM-DDCG, both the generative module and TAD use CNN to generate data and extract features [37]. The generator up-samples the input noise layer by layer into the target data through a multi-layer deconvolution structure, and the dimensionality continuously increases during the process. The CNN of the TAD module gradually reduces the dimensionality and extracts features through convolutional layers to distinguish between true and false data. The convolutional process of generative network and discriminative network is shown in Figure 6.

As shown in Figure 6, the generative network starts with $3 \times 3 \times 1$ noise, expands its dimensions to $12 \times 12 \times 16$ through multi-layer convolution and up-sampling, then reduces it to $12 \times 12 \times 1$ through RMD processing. Finally, $9 \times 9 \times 1$ output is obtained through further convolution. The discriminative network reduces the dimensionality of $9 \times 9 \times 1$ input features to $3 \times 3 \times 1$ through convolution and performs final discrimination. In this process, the discriminator of the TAD module assesses the authenticity of the data generated by the RMD module, enhances its assessment of data authenticity while extracting data features, and thus generates high-quality network data. Afterwards, the RMD module increases data diversity and model robustness in the generation and discrimination networks, while continuous convolution and up-sampling/down-sampling can optimize feature extraction, enabling the model to find a balance between generation and discrimination.

C. DESIGN OF NTAD SYSTEM

To monitor and identify abnormal traffic and respond to potential network threats, a complete NTAD system based on optimized GAN and Transformer is proposed. The architecture of the proposed NTAD system is expressed in Figure 7.

As shown in Figure 7, the architecture of an NTAD system based on DCGAN and Transformer adopts a three-layer web development architecture pattern. The presentation layer is responsible for interacting with users and communicating with the business logic layer through the Http protocol. The core of the business logic layer is the anomaly detection module, which is responsible for processing network traffic data, achieving anomaly recognition and warning. The data

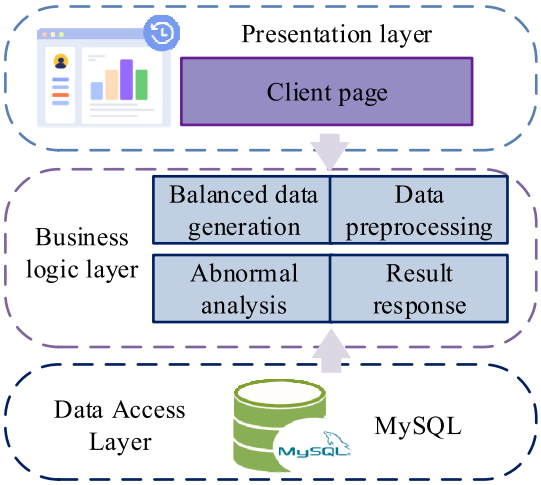

FIGURE 7. Architecture of NTAD system.

access layer interacts with the MySQL database through JDBC template to ensure persistent storage of data. The design of system functional modules is the core of an NTAD system, with the business logic layer as the key and modular development to reduce coupling and facilitate expansion. The system includes a system login module, data pre-processing module, data acquisition module, anomaly detection module, and alarm module. The system login module provides user access points, login, logout, and interception of illegal requests. The data acquisition module captures real-time network data and analyzes it. The data pre-processing module performs pre-processing operations. The anomaly detection module utilizes an improved RM-DDCG model for class balancing and combines it with the Trans-M model to identify abnormal behavior. The alarm module issues alert for detected anomalies and support timely response from security personnel.

IV. PERFORMANCE TESTING OF AN NTAD SYSTEM BASED ON GAN AND TRANSFORMERS

According to the above model construction, the effectiveness of an NTAD system based on GAN and Transformers proposed for testing in network security protection was studied. Simulation tests were conducted on the NTAD model and category balance model, respectively.

A. PERFORMANCE TESTING OF AN NTAD MODEL THAT INTEGRATES PSE AND TRANSFORMER

The research targeted to identify the effectiveness of the Trans-M model in detecting network traffic anomalies. CICIDS2017 and NSL-KDD were selected as experimental datasets. The CICIDS2017 dataset was sourced from the Canadian Institute of Cyber-security, which includes 2830608 pieces of data. In the past five days of network traffic data, 10 attack categories and normal categories were selected for experimentation. The NSL-KDD dataset is a revised version of the Data Mining and Knowledge Discovery

(KDD99) dataset, with attack types including denial of service, probing, user to root, and remote to local. Classify and sample the NSL-KDD dataset, totaling 148517 pieces of data. In the study, it was necessary to ensure the rationality of the experiment and to determine the hyper-parameters of the Trans-M model and to write the experimental code using the language in the computer system. Finally, based on the two datasets, select partial data to obtain binary and multi-classification data. Due to the imbalanced distribution of samples in the dataset, uniform sampling is required for each dataset. Concurrently, the quantity of data pertaining to normal categories was considerably greater than that of abnormal attacks. Consequently, the prediction error rate of attack categories was essentially negligible. A comparative experiment was conducted to determine the model hyper-parameters using variable control methods. Moreover, it used indicators such as accuracy, recall rate, and false positive rate to illustrate the evaluation performance of the experiment. The relationship between model training and performance is shown in Figure 8.

In Figure 8, Figure 8 (a) shows the relationship between model iteration times and accuracy, Figure 8 (b) shows the relationship between model hidden layer dimensions and accuracy, and Figure 8 (c) shows the relationship between model learning rate and accuracy. Overall comparison showed that RNN and Trans-M models had similar accuracy in the initial training stage, while the Bidirectional Long Short-Term Memory Network (BiLSTM) model had the highest initial accuracy. The selection of hyper-parameters for the model included the number of iterations (Epochs), the number of hidden layer neurons (E-hidden), and the learning rate (Learning_rate). To guarantee the optimal performance of the model, a variety of different combinations of hyper-parameters were analyzed in order to identify the optimal solution for the optimal combination of hyper-parameters. As the iterations (Epochs) increased to 1000, the Trans-M model had the highest accuracy, which was better than RNN and BiLSTM. Therefore, Epochs was set to 1000. In the experiment of changing the number of E-hidden, RNN and BiLSTM showed the best accuracy at an E-hidden of 128, while Trans-M outperformed other models when the E-hidden increased to 256. Therefore, an E-hidden of 256 was selected. The Learning_rate experiment showed that the accuracy of Trans-M was highest at 0.001, so the Learning_rate was determined to be 0.001. Based on the above parameters, Trans-M had the best performance in the configuration of Epochs 1000, E-hidden 256, and Learning_rate 0.001. Therefore, this parameter was used for performance testing experiments. The model parameters used in the binary network traffic experiment are denoted in Table 1.

In the testing of the binary classification model, to prevent over-fitting, the dropout rate was set to 0.4 and Epochs was determined to be 1000 times. To achieve effective learning, E-hidden was set to 256 and Learning_rate was set to 0.001. The model adopted an encoder module stacked with 6 Encoders and 8 multi head AMs to capture complex

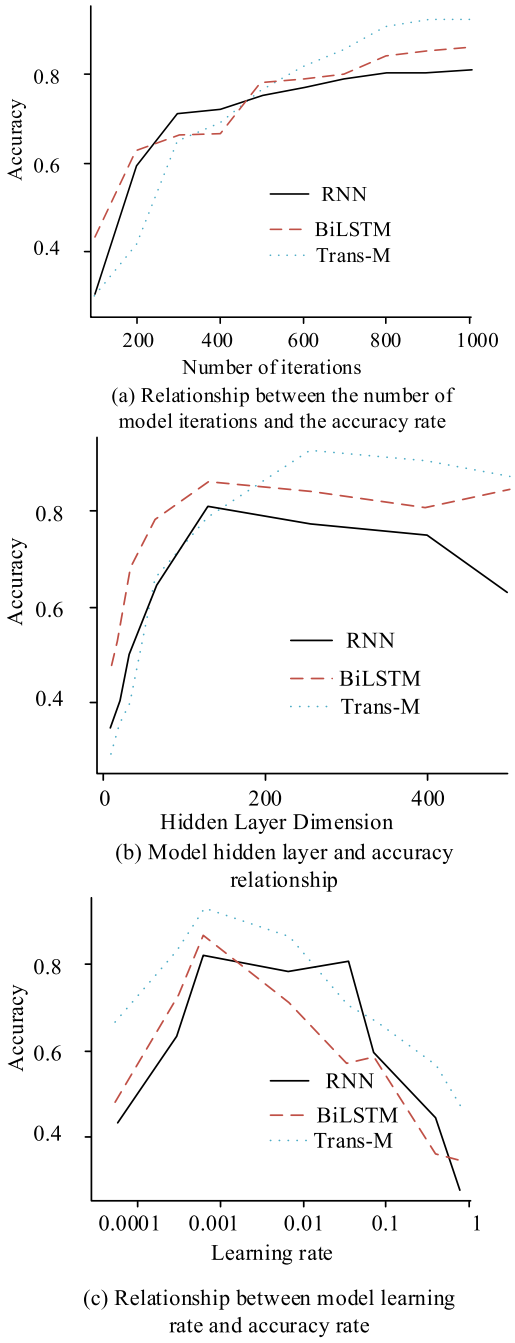

FIGURE 8. Relationship between model training and performance.

data features. To ensure the accuracy of the model optimization direction, the optimizer used Adam, and the Loss was cross entropy loss. The binary classification test results based on the above parameter settings are shown in Figure 9.

In Figure 9, Figures 9 (a) and 9 (b) show the binary classification test results on the CICIDS2017 and NSL-KDD datasets, respectively. In the binary classification task of the CICIDS2017 dataset, the accuracy of the Trans-M model reached 93.65%, which was about 13% higher than the Decision Tree (DT) model, about 9% higher than the

TABLE 1. Model parameters.

Parameter	Parameter settings
Dropout	0.4
Epochs	1000
Learning_rate	0.001
Multi-head	8
E_hidden	256
Optimizer	Adam
Loss	Cross entropy loss
Encoder	6
Input features/each	78

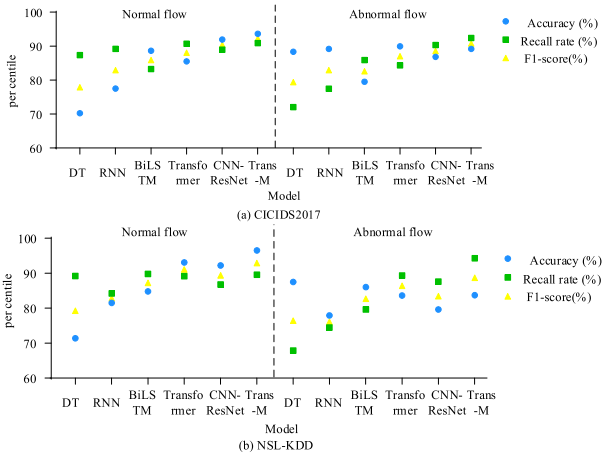

FIGURE 9. Results of binary classification test.

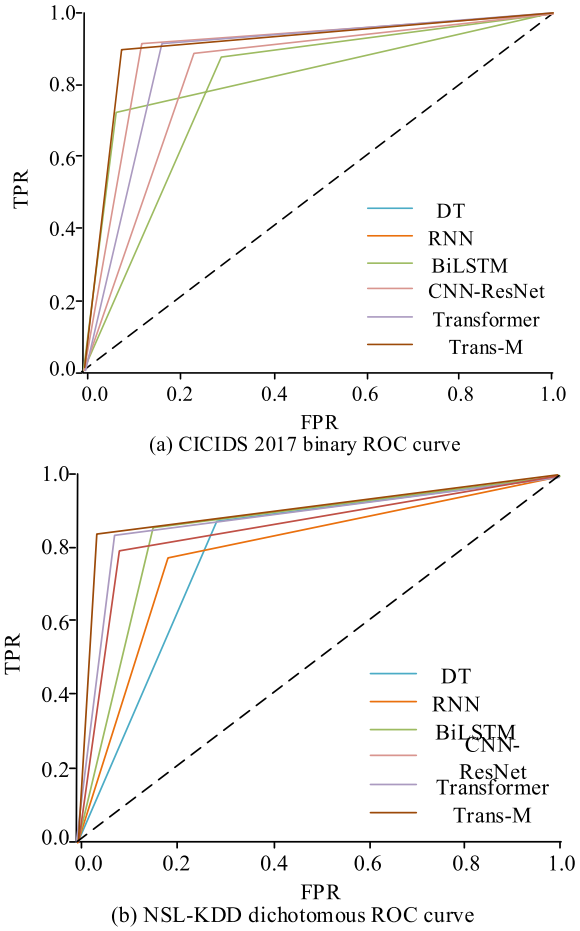

FIGURE 10. Binary ROC curve.

RNN model, about 7% higher than the BiLSTM model, about 4% higher than the native Transformer, and about 2% higher than the Convolutional Neural Network Residual Network (CNN-ResNet). In the F1-score evaluation, the Trans-M model achieved 92.24% performance on normal traffic and 90.77% on attack traffic, showing a significant performance improvement compared to Transformer. The test results on the NSL-KDD dataset indicated that the Trans-M model performed well on multiple metrics. In normal traffic detection, the accuracy of Trans-M reached 96.50%, which was 3.30% higher than the closest CNN-ResNet model. In attack traffic detection, the accuracy was 83.72%, which was 4.09% higher than the CNN-ResNet model. The recall rate of Trans-M was 89.55% in normal traffic and 94.29% in attack traffic, which was 3.69% and 6.42% higher than the CNN-ResNet model, respectively. The F1-score of the Trans-M model was 92.90% in normal traffic and 88.69% in attack traffic, both higher than other models and 1.82% and 2.31% higher than the Transformer model, respectively. The comparative test results showed that the Trans-M model could effectively improve the accuracy of NTAD and had significant performance advantages in identifying attack traffic. The comparison outcomes of ROC curves are indicated in Figure 10.

In Figure 10, Figures 10 (a) and 10 (b) show the comparison of binary ROC curves on the CICIDS2017 and NSL-KDD datasets, respectively. In the ROC curve of the CICIDS2017 dataset, the Trans-M model had the best effectiveness in classification performance, with an AUC value of 0.91. The AUC values for CNN-ResNet and native Transformer models were 0.89 and 0.87, respectively. The AUC values of both RNN and BiLSTM models were 0.83, slightly better than the DT model's 0.79, but the overall performance was weak. In the binary classification task of the NSL-KDD dataset, the Trans-M model performed the best with an AUC value of 0.90. The AUC value of the Transformer model was 0.88, which performed better than BiLSTM's 0.85 and CNN-ResNet's 0.86. The AUC values of RNN and DT models were slightly lower, at 0.80 and 0.79, respectively. The ROC comparison outcomes verified that the Trans-M model had excellent classification performance on different datasets. The experiment of detecting abnormal traffic in multi-class networks is shown in Table 2.

In Table 2, the multi-class NTAD experiment was conducted on the CICIDS2017 dataset, and the Trans-M model performed well. Overall comparison showed that the Trans-M model achieved 91.75% accuracy, 91.98% recall, and 91.82%

TABLE 2. Multi-class NTAD experiment.

Model	Accuracy (%)	Recall rate (%)	F1-score (%)
DT	81.89	86.19	83.61
RNN	75.25	83.42	78.86
BiLSTM	81.56	86.23	83.38
CNN-ResNet	88.72	90.38	88.98
Transformer	86.06	90.73	88.04
Trans-M	91.75	91.98	91.82

F1-score, significantly higher than other comparative models. Compared with DT, the Trans-M model showed significant improvement in all evaluation metrics. Compared to the CNN-ResNet model, Trans-M improved recall by 5.07%, with better accuracy and F1-score performance. The experimental results of multi-class NTAD demonstrated the efficiency and accuracy of the Trans-M model in processing network traffic data, as well as its robustness in distinguishing multiple types of network traffic anomalies.

B. CATEGORY BALANCE MODEL AND PERFORMANCE TESTING OF NTAD SYSTEM

When conducting research on NTAD, it was necessary to combine RM-DDCG and Trans-M models. The research first sampled network traffic data and divided them into categories based on traffic characteristics. In view of the category imbalance problem in the dataset, a small number of class samples were selected and the RM-DDCG algorithm was used to enhance their data to achieve class balance. The optimizer of the RM-DDCG model used Adam, with a learning rate of 0.0001, a loss function using cross entropy loss, and an iteration count of 14000. The comparison chart of binary classification indicators before and after balance is shown in Figure 11.

Figure 11 shows the comparison of binary classification indicators before and after balance on the CICIDS2017 dataset. In the dataset after balancing RM-DDCG and DD-GAN, most models performed better. RM-DDCG improved the robustness and generalization ability of the model by incorporating RMD. All models achieved higher accuracy, recall, and F1-score on the RM-DDCG balanced dataset. The Trans-M model achieved the optimal effectiveness with an accuracy of 99.23%, a recall rate of 90.87%, and an F1-score of 90.71% in the balanced dataset. In contrast, the performance metrics of imbalanced datasets were lower, indicating the effectiveness of the RM-DDCG balancing strategy in improving model detection performance. From the comprehensive evaluation of various indicators, the RM-DDCG balanced model had higher accuracy and robustness in handling traffic detection tasks, and was superior to other balancing methods. The loss variation diagram of the RM-DDCG model's multi-class discriminator is shown in Figure 12.

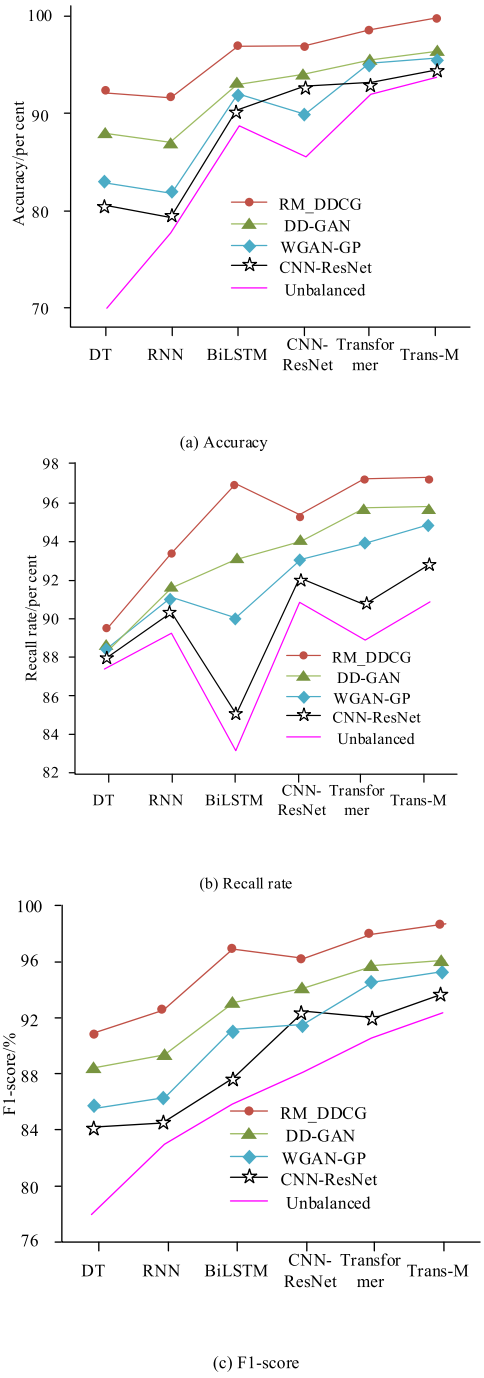

FIGURE 11. Comparison of binary classification indicators before and after balance.

In the multi-class discriminator loss variation diagram of the RM-DDCG model denoted in Figure 12, the discriminator loss values of FTP Patator, SSH Patator, DoS slowloris, DoS Slowhttptest, Bot, and Web Attack gradually decreased. This indicated that during the iterative process of model training, the discriminator's performance in distinguishing attacks of various categories improved and tended to be stable. This downward trend in loss indicated that the model's ability to classify data was improving, and the generator and

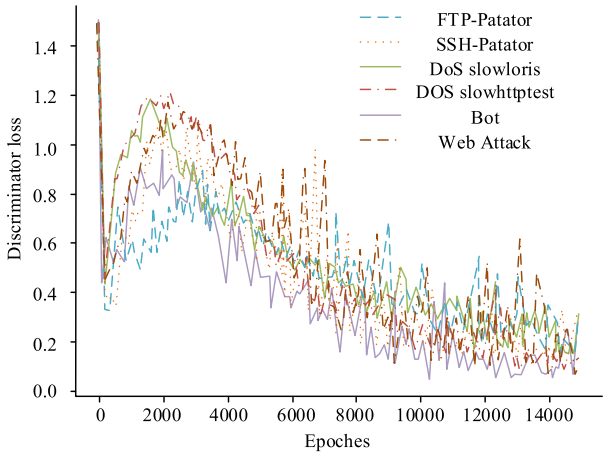

FIGURE 12. Changes in Loss of RM-DDCG model's multi-class discriminator.

TABLE 3. Comparison of multi-class indicators after balance on the CICIIDS2017 dataset.

Model	Accuracy (%)	Recall rate (%)	F1-score (%)	Time efficiency (M)	False positive prediction (%)
DT	79.67	83.97	81.39	3.252M	16.47
RM-DDCG-DT	91.81	94.02	91.96	0.036M	11.24
RNN	73.03	81.20	76.64	3.121M	13.16
RM-DDCG-RNN	92.00	93.19	92.46	0.034M	13.28
BiLSTM	79.34	84.01	81.16	0.213M	14.64
RM-DDCG-BiLSTM	94.99	95.02	94.99	0.054M	11.26
CNN-ResNet	86.50	88.16	86.76	2.011M	10.28
RM-DDCG-CNN-ResNet	95.90	95.87	95.87	0.076M	10.04
Transformer	83.84	88.51	85.82	1.465M	12.64
RM-DDCG-Transform er	96.10	96.40	96.13	0.034M	9.46
Trans-M	89.53	89.76	89.60	0.118M	5.78
RM-DDCG-Trans-M	98.12	97.86	98.46	0.015M	3.04

discriminator reached a good balance. After data balancing, the processed data was applied to train the detection model, and the accuracy of different models on the CICIDS2017 dataset also showed a significant improvement. The comparison of various indicators for multi-classification after balancing on CICIDS2017 is indicated in Table 3.

The performance of each model on multi-class tasks after balancing using RM-DDCG is shown in Table 3. From the data in the table, after category balancing, the accuracy, recall, and F1-score of each model were all been significantly improved. The accuracy of the Trans-M model on the balanced dataset reached 98.12%, an improvement of 8.59% compared to before balance. The recall rate of the Trans-M model was 97.86%, which was 8.62% higher than before balance. On F1-score, the highest score of the Trans-M model was 98.46%, which was 8.18% higher than before

TABLE 4. Performance comparison of different algorithms on the Balanced IDS2018 dataset.

Method	Accuracy (%)	False alarm rate (%)	Missed alarm rate (%)	Susceptibility (%)	Specificity (%)
1D-CNN [38]	97.06%	3.46%	3.51%	96.41%	5.48%
CGAN [39]	97.02%	4.52%	4.06%	95.64%	5.03%
MLP [40]	94.46%	5.35%	5.13%	93.15%	6.98%
RF [41]	93.58%	4.67%	4.16%	92.04%	8.47%
RM-DDCG-Trans-M	98.14%	2.36%	2.42%	97.13%	2.05%

equilibrium. Moreover, a comparison of the floating-point operation speeds per second of different models revealed that traditional network models exhibited lower computational efficiency, whereas models incorporating AM modules demonstrated enhanced detection efficiency. In comparison to other models, the Trans-M model demonstrated the most favorable outcomes across all indicators, with a detection efficiency value of 0.015M. This indicates the significance and efficacy of RM-DDCG in enhancing the model's capacity to address imbalanced data.

Afterwards, to verify the classification and detection accuracy of the system model, other algorithms were selected for the analysis of evaluation indicators on the Balanced IDS2018 dataset. The compared algorithm models included One-dimensional Convolutional Neural Network (1D-CNN) [38], Conditional Generative Adversarial Network (CGAN) [39], Multi-layer Perceptron (MLP) [40], random forest (RF) [41], and the algorithm proposed in this study. The results are shown in Table 4.

In Table 4, the evaluation indicators of 1D-CNN, CGAN, and RM-DDCG-Trans-M were good, with false positive and false negative rates of 3.46% and 3.51%, respectively, reflecting the efficient processing effect of their algorithms. The method proposed by the study, RM-DDCG-Trans-M, had a high accuracy among all methods, with an accuracy of 98.14%, while the false alarm rate and false alarm rate were 2.36% and 2.42%, respectively. Concurrently, the method exhibited a sensitivity and specificity of 97.13% and 2.05%, thereby demonstrating that the RM-DDCG-Trans-M method has an excellent capacity for data processing and enhances the accuracy and resilience of NTAD.

V. CONCLUSION

With the continuous evolution of network attack methods, traditional rule-based and signature-based security strategies are becoming increasingly hard to deal with increasingly complex network threats. To improve the accuracy and robustness of NTAD, an NTAD system based on GAN and Transformers was studied. The performance test results showed that Trans-M performed best in the configuration with Epochs of 1000, E-hidden of 256, and Learning_rate of 0.001. In binary

classification tasks, the accuracy of Trans-M model reached 93.65%, which was about 13% higher than DT model. In the ROC curve, Trans-M model showed the best classification effectiveness with an AUC value of 0.91. The Trans-M model achieved 91.75% accuracy, 91.98% recall, and 91.82% F1-score, significantly higher than other comparative models. All models achieved higher accuracy, recall, and F1-score on the RM-DDCG balanced dataset. The Trans-M model achieved the optimal effectiveness with an accuracy of 99.23%, a recall rate of 90.87%, and an F1-score of 90.71% in the balanced dataset. After using RM-DDCG balance, the performance of each model on multi-class tasks was significantly improved, including accuracy, recall, and F1-score. The accuracy of the Trans-M model on the balanced dataset reached 98.12%, an improvement of 8.59% compared to before balance. The recall rate of the Trans-M model was 98.12%, with an improvement of 8.36%. On F1-score, the highest score of the Trans-M model was 98.12%, which was 8.52% higher than before equilibrium. The performance test results showed that the RM-DDCG balancing method could raise the accuracy and reliability of NTAD in the Trans-M model. The proposed NTAD system could meet the actual network security protection requirements. Although the proposed model exhibits excellent performance in detecting network traffic anomalies, it is important to note that there are still some limitations. In particular, the adaptability of the current model in complex network environments and its ability to detect new types of attacks still need to be verified. Further investigation is required to ascertain the efficacy and significance of incorporating discriminators into convolutional GANs with dual discriminators. With regard to the issue of anomalous network traffic, it is necessary to implement detection methods for real-world scenarios in order to ensure the robustness and effectiveness of the model. At the same time, the real-time detection efficiency and performance of the model on large-scale datasets also need to be further improved.

ABBREVIATIONS

Abbreviations	Full name.
GAN	Generative Adversarial Networks.
Trans-M	Transformer Multi Receive Field Fusion.
DDoS	Distributed Denial of Service.
GRU	Gate Recurrent Unit.
DRTormer	Dual Relationship Transformer.
SS	Subset Simulation.
CLM	Causal Language Modeling.
MRR	Mean Reciprocal Rank.
RNN	Recurrent Neural Network.
CBAM	Convolutional Block Attention Module.
MRFF	Multi Receptive Field Fusion.
AE	Auto Encoder.
RMD	Random Masked data blocks.
DT	Decision Tree.
AI	Artificial intelligence.
F1-score	Balanced F Score.
CBIR	Content-based Image Retrieval.
RL	Reinforcement Learning.
ViT	Visual Transformer.
ARM	Attribute Relationship Module.
NPSO	Niche Particle Swarm Optimization Algorithm.
ECLM	Extended Context Learner Model.
DCGAN	Deep Convolutional Generative Adversarial Networks.
LSTM	Long Short-Term Memory.
PSE	Patch Segmentation.
Seq2Img	Sequence to Image.
VAE	Variational Autoencoders.
NTAD	Network Traffic Anomaly Detection.
CNN-ResNet	Convolutional Neural Network Residual Network.

REFERENCES

[1] M. Hammad, N. Hewahi, and W. Elmedany, "MMM-RF: A novel high accuracy multinomial mixture model for network intrusion detection systems," *Comput. Secur.*, vol. 120, Sep. 2022, Art. no. 102777, doi: 10.1016/j.cose.2022.102777.

[2] J. Wang, "Full-scene network security protection system based on ubiquitous power Internet of Things," *Int. J. Commun. Syst.*, vol. 35, no. 5, pp. 95–108, Mar. 2022, doi: 10.1002/dac.4695.

[3] Y. Kang, J. Zhong, R. Li, Y. Liang, and N. Zhang, "Classification method for network security data based on multi-featured extraction," *Int. J. Artif. Intell. Tools*, vol. 30, no. 1, Feb. 2021, Art. no. 2140006, doi: 10.1142/s0218213021400066.

[4] S. Wang, C. Zhao, L. Huang, Y. Li, and R. Li, "Current status, application, and challenges of the interpretability of generative adversarial network models," *Comput. Intell.*, vol. 39, no. 2, pp. 283–314, Apr. 2023, doi: 10.1111/coin.12564.

[5] Y. Zhang, J. Shen, C. Yu, and C. Wang, "Views meet labels: Personalized relation refinement network for multiview multilabel learning," *IEEE Multimedia Mag.*, vol. 29, no. 2, pp. 104–113, Apr. 2022, doi: 10.1109/MMUL.2022.3142154.

[6] K. Mabodi, M. Yusefi, S. Zandiyan, L. Irankhah, and R. Fotohi, "Multi-level trust-based intelligence schema for securing of Internet of Things (IoT) against security threats using cryptographic authentication," *J. Supercomput.*, vol. 76, no. 9, pp. 7081–7106, Jan. 2020, doi: 10.1007/s11227-019-03137-5.

[7] K. Dlshad and S. Askar, "Deep learning models for cyber security in IoT networks: A review," *Int. J. Sci. Bus.*, vol. 5, pp. 61–70, Apr. 2021, doi: 10.5281/zenodo.4497017.

[8] B. Sun, "Wireless network for computer puzzle online software cloud platform based on CBIR and sustainable computing," *Int. J. Netw. Virtual Org.*, vol. 25, no. 2, pp. 197–212, Jun. 2021, doi: 10.1504/ijnvo.2021.119069.

[9] H. Xu, L. Chai, Z. Luo, and S. Li, "Stock movement prediction via gated recurrent unit network based on reinforcement learning with incorporated attention mechanisms," *Neurocomputing*, vol. 467, pp. 214–228, Jan. 2022, doi: 10.1016/j.neucom.2021.09.072.

[10] S. Chhabra, M. K. Aiden, S. M. Sabharwal, and M. Al-Asadi, "5G and 6G technologies for smart city," in *Enabling Technologies for Effective Planning and Management in Sustainable Smart Cities*, vol. 26. Cham, Switzerland: Springer, Feb. 2023, pp. 335–365, doi: 10.1007/978-3-031-22922-0_14.

[11] L. Yadav, M. Mitra, A. Kumar, B. Bhushan, and M. A. Al-Asadi, "Nullifying the prevalent threats in IoT based applications and smart cities using blockchain technology," in *Low Power Architectures for IoT Applications*, vol. 5. Singapore: Springer, Apr. 2023, pp. 241–261, doi: 10.1007/978-981-99-0639-0_14.

[12] O. PJena, B. Bhushan, N. Rakesh, P. N. Astya, and Y. Farhaoui, *Machine Learning and Deep Learning in Efficacy Improvement of Healthcare Systems*, vol. 19. Boca Raton, FL, USA: CRC Press, May 2022, p. 395, doi: 10.1201/9781003189053-7.

[13] S.-W. Lee, H. Mohammed Sidqi, M. Mohammadi, S. Rashidi, A. M. Rahmani, M. Masdari, and M. Hosseinzadeh, "Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review," *J. Netw. Comput. Appl.*, vol. 187, Aug. 2021, Art. no. 103111, doi: 10.1016/j.jnca.2021.103111.

[14] M. Y. Aldarwbi, A. H. Lashkari, and A. A. Ghorbani, "The sound of intrusion: A novel network intrusion detection system," *Comput. Electr. Eng.*, vol. 104, Dec. 2022, Art. no. 108455, doi: 10.1016/j.compeleceng.2022.108455.

[15] G. Chandrasekaran, P. R. Karthikeyan, N. S. Kumar, and V. Kumarasamy, "Test scheduling of system-on-chip using dragonfly and ant lion optimization algorithms," *J. Intell. Fuzzy Syst.*, vol. 40, no. 3, pp. 4905–4917, Mar. 2021, doi: 10.3233/jifs-201691.

[16] G. Chandrasekaran, V. Kumarasamy, and G. Chinraj, "Test scheduling of core based system-on-chip using modified ant colony optimization," *J. Européen des Systèmes Automatisés*, vol. 52, no. 6, pp. 599–605, Dec. 2019, doi: 10.18280/jesa.520607.

[17] G. Chandrasekaran, S. Periyasamy, and P. R. Karthikeyan, "Test scheduling for system on chip using modified firefly and modified ABC algorithms," *Social Netw. Appl. Sci.*, vol. 1, no. 9, pp. 1–12, Sep. 2019, doi: 10.1007/s42452-019-1116-x.

[18] A. Tian, Y. Zhang, C. Ma, H. Chen, W. Sheng, and S. Zhou, "Noise-robust machinery fault diagnosis based on self-attention mechanism in wavelet domain," *Measurement*, vol. 207, Feb. 2023, Art. no. 112327, doi: 10.1016/j.measurement.2022.112327.

[19] Z. Tang and J. Huang, "DRFormer: Learning dual relations using transformer for pedestrian attribute recognition," *Neurocomputing*, vol. 497, pp. 159–169, Aug. 2022, doi: 10.1016/j.neucom.2022.05.028.

[20] M. Li, G. Jia, Z. Cheng, and Z. Shi, "Generative adversarial network guided topology optimization of periodic structures via subset simulation," *Compos. Struct.*, vol. 260, Mar. 2021, Art. no. 113254, doi: 10.1016/j.compstruct.2020.113254.

[21] H. Ezaldeen, S. K. Bisoy, R. Misra, and R. Alatrash, "Semantics-aware context-based learner modelling using normalized PSO for personalized E-learning," *J. Web Eng.*, vol. 21, pp. 1187–1223, Apr. 2022, doi: 10.13052/jwe1540-9589.2148.

[22] A. Oseni, N. Moustafa, G. Creech, N. Sohrabi, A. Strelzoff, Z. Tari, and I. Linkov, "An explainable deep learning framework for resilient intrusion detection in IoT-enabled transportation networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 1000–1014, Jan. 2023, doi: 10.1109/TITS.2022.3188671.

[23] V. Shrivastava and M. Kamble, "A comparative study on deep learning-based algorithms for intruder detection systems and cyber security," *SAMRIDDHI, J. Phys. Sci., Eng. Technol.*, vol. 15, no. 1, pp. 154–160, Jan. 2023, doi: 10.18090/samriddhi.v15i01.30.

[24] P. Dixit and S. Silakari, "Deep learning algorithms for cybersecurity applications: A technological and status review," *Comput. Sci. Rev.*, vol. 39, Feb. 2021, Art. no. 100317, doi: 10.1016/j.cosrev.2020.100317.

[25] J. Clements, Y. Yang, A. A. Sharma, H. Hu, and Y. Lao, "Rallying adversarial techniques against deep learning for network security," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2021, pp. 1–8, doi: 10.1109/SSCI50451.2021.9660011.

[26] P. Tzirakis, J. Chen, S. Zafeiriou, and B. Schuller, "End-to-end multimodal affect recognition in real-world environments," *Inf. Fusion*, vol. 68, pp. 46–53, Apr. 2021, doi: 10.1016/j.inffus.2020.10.011.

[27] X. Qian, Y. Wang, and A. Karanth, "Guest editors' introduction to the special issue on machine learning architectures and accelerators," *IEEE Trans. Comput.*, vol. 69, no. 7, pp. 929–930, Jul. 2020, doi: 10.1109/TC.2020.2997574.

[28] C. Zhang, H. Wan, X. Shen, and Z. Wu, "PVT: Point-voxel transformer for point cloud learning," *Int. J. Intell. Syst.*, vol. 37, no. 12, pp. 11985–12008, Dec. 2022, doi: 10.1002/int.23073.

[29] Q. Lu, W. Ye, and L. Yin, "ResDenIncepNet-CBAM with principal component analysis for wind turbine blade cracking fault prediction with only short time scale SCADA data," *Measurement*, vol. 212, May 2023, Art. no. 112696, doi: 10.1016/j.measurement.2023.112696.

[30] L. Chen, H. Yao, J. Fu, and C. Tai Ng, "The classification and localization of crack using lightweight convolutional neural network with CBAM," *Eng. Struct.*, vol. 275, Jan. 2023, Art. no. 115291, doi: 10.1016/j.engstruct.2022.115291.

[31] M. Zhu and A. Towne, "Recursive one-way Navier–Stokes equations with PSE-like cost," *J. Comput. Phys.*, vol. 473, Jan. 2023, Art. no. 111744, doi: 10.1016/j.jcp.2022.111744.

[32] Q. Chen, W. Hao, and J. He, "Power series expansion neural network," *J. Comput. Sci.*, vol. 59, Mar. 2022, Art. no. 101552, doi: 10.1016/j.jocs.2021.101552.

[33] S. M. Iranmanesh and N. M. Nasrabadi, "HGAN: Hybrid generative adversarial network," *J. Intell. Fuzzy Syst.*, vol. 40, no. 5, pp. 8927–8938, Apr. 2021, doi: 10.3233/jifs-201202.

[34] Y. Han, P. Zhang, W. Huang, Y. Zha, G. D. Cooper, and Y. Zhang, "Robust visual tracking based on adversarial unlabeled instance generation with label smoothing loss regularization," *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 107027, doi: 10.1016/j.patcog.2019.107027.

[35] H. Ding, Z. Cui, E. Maghami, Y. Chen, J. P. Matinlinna, E. H. N. Pow, A. S. L. Fok, M. F. Burrow, W. Wang, and J. K. H. Tsoi, "Morphology and mechanical performance of dental crown designed by 3D-DCGAN," *Dental Mater.*, vol. 39, no. 3, pp. 320–332, Mar. 2023, doi: 10.1016/j.dental.2023.02.001.

[36] H. Zhong, S. Yu, H. Trinh, Y. Lv, R. Yuan, and Y. Wang, "Fine-tuning transfer learning based on DCGAN integrated with self-attention and spectral normalization for bearing fault diagnosis," *Measurement*, vol. 210, Mar. 2023, Art. no. 112421, doi: 10.1016/j.measurement.2022.112421.

[37] S. Basalama, A. Sohrabizadeh, J. Wang, L. Guo, and J. Cong, "FlexCNN: An end-to-end framework for composing CNN accelerators on FPGA," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 16, no. 2, pp. 1–32, Jun. 2023, doi: 10.1145/3570928.

[38] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han, M. M. Iqbal, and K. Han, "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018, doi: 10.1109/ACCESS.2018.2863036.

[39] J. Lan, X. Liu, B. Li, Y. Li, and T. Geng, "DarknetSec: A novel self-attentive deep learning method for darknet traffic classification and application identification," *Comput. Secur.*, vol. 116, May 2022, Art. no. 102663, doi: 10.1016/j.cose.2022.102663.

[40] Q. Jiang, L. Zhu, C. Shu, and V. Sekar, "Multilayer perceptron neural network activated by adaptive Gaussian radial basis function and its application to predict lid-driven cavity flow," *Acta Mechanica Sinica*, vol. 37, no. 12, pp. 1757–1772, Dec. 2021, doi: 10.1007/s10409-021-01144-5.

[41] R. R. Chowdhury, A. C. Idris, and P. E. Abas, "Identifying SH-IoT devices from network traffic characteristics using random forest classifier," *Wireless Netw.*, vol. 30, no. 1, pp. 405–419, Jan. 2024, doi: 10.1007/s11276-023-03478-3.

HUI CAO was born in March 1973. He received the degree in computer science from Hunan Railway Professional Technology College (formerly Zhuzhou Railway Electrical Machinery School), in July 1992, the bachelor's degree in computer science and technology from Hunan University, in October 2006, and the master's degree in software engineering from Wuhan University, in October 2008.

He started working, in August 1992, and was initially assigned to the college library to manage and maintain computers. In 1998, when the college established the Network Center, he became one of the key management personnel, responsible for network management. Since March 2006, he has been a Network Engineer with the Technical Center. From 1992 to 2022, he was a Network Engineer with Hunan Railway Professional Technology College. From 2022 to now, he works as a Senior Network Security Engineer with Library and Information Center, Hunan Railway Professional Technology College. He has published six academic articles.

Mr. Cao received the Intermediate Professional Title, in 2013. He was certified as a Senior Network Security Engineer by the Hunan Provincial Cyberspace Administration, in 2022.