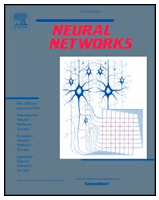
Neural Networks 184 (2025) 107064

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

DDP-DAR: Network intrusion detection based on denoising diffusion probabilistic model and dual-attention residual network

Saihua Cai [a,b,1,*], Yingwei Zhao [a,1], Jiaao Lyu [a], Shengran Wang [a,b], Yikai Hu [a], Mengya Cheng [a], Guofeng Zhang [c]

[a] School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, 212013, Jiangsu, China
[b] Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace, Jiangsu University, Zhenjiang, 212013, Jiangsu, China
[c] School of Information Science and Technology, Taishan University, Taian, 271000, Shandong, China

ARTICLE INFO

Keywords:
Network traffic detection
Data augmentation
Feature representation
Denoising diffusion probabilistic model
Dual-attention residual network

ABSTRACT

Network intrusion detection (NID) is an effective manner to guarantee the security of cyberspace. However, the scale of normal network traffic is much larger than intrusion traffic (i.e., appearing data imbalance problem), which leads to the training of NID model to be more towards the majority classes, thus affecting the detection effect. Although scholars have solved this problem by reducing normal network traffic or increasing intrusion traffic, while increasing the number of intrusion traffic can effectively expand the scale of datasets in the model training process, which is benefit for training a better NID model. In this paper, we propose a network intrusion detection based on denoising diffusion probabilistic model and dual-attention residual network (DDP-DAR) through feature representation, data augmentation and intrusion detection, respectively. In the feature representation phase, we propose a novel feature representation method to better represent network traffic in the format of RGB images by storing global features and local features. In the data augmentation phase, we utilize the denoising diffusion probabilistic model instead of traditional data augmentation models (e.g., VAE, GAN), and then introduce the cosine noise addition and learnable variance parameter strategies to improve the denoising diffusion model for generating RGB images with high quality. In the intrusion detection phase, we propose the detection method based on dual-attention residual network, which performs feature extraction through multi-layer network structure and dual-attention mechanism to get the higher level and more important information, thereby detecting intrusion traffic more accurately. Compared with the state-of-the-art data augmentation-based NID methods, a large number of experimental results show that DDP-DAR performs better in four metrics of Accuracy, F1-measure, FPR and ROC-AUC; Meanwhile, the detection results of DDP-DAR are more stable.

1. Introduction

In today's era of network communication, the number of organized and purposeful cyber-attacks such as data leakage, cyber fraud and ransom, viruses and other cyber security threats is increasing, which poses a serious cyber security issue. To cope with these cyber-attacks, network intrusion detection (NID) methods are used to detect malicious behaviors, thereby providing protection for cyberspace security. In recent years, statistical-based (Singh et al., 2022), port-based (Blaise et al., 2020) and labeling-based (Abdulganiyu et al., 2024) NID techniques are used to detect the behaviors of cyber-attacks, where statistics-based methods cannot effectively deal with new attack behaviors, port-based methods cannot effectively process multi-port attack activities, and label-based methods rely heavily on labeled samples and require long time to label the samples. With the rapid development of artificial intelligence (AI), machine learning methods (Yang et al., 2022; Chen et al., 2023a) and deep learning models (Gamage & Samarabandu, 2020; Cai et al., 2024; Chen et al., 2023c) have been widely researched due to their strong learning ability, and the traditional NID methods are gradually replaced by AI technologies.

Compared with machine learning methods, deep learning techniques are widely used in NID for their powerful automatic feature extraction capabilities when facing large-scale intrusion traffic. Currently, scholars have dedicated their researches on NID using deep learning models, such

* Corresponding author.
E-mail address: caisaih@ujs.edu.cn (S. Cai).
[1] Both authors contribute equally to this work.

https://doi.org/10.1016/j.neunet.2024.107064
Received 4 September 2024; Received in revised form 20 November 2024; Accepted 15 December 2024
Available online 18 December 2024

as BP neural network (Akgun et al., 2022), CNN (Landman & Nissim, 2021), LSTM (Wei et al., 2024), TCN (de Araujo-Filho et al., 2023), etc. However, traditional deep learning models have the problems such as appearing gradient explosion with the increase of network depth as well as rapid degradation of accuracy after gradual saturation; In addition, existing models usually consider all the features of network traffic when dealing with massive network traffic without exploring the important features that have a greater impact on NID, such as the features at network layer and application layer, resulting in poor detection performance. Therefore, it is necessary to design an efficient detection model that can guarantee the detection performance while increasing the depth of network as well as pay more attention to the important features, thereby improving the detection performance.

In the network traffic, the scale of various types of attack traffic is small compared with normal traffic, i.e., appearing data imbalance phenomenon, causing the deep learning-based NID models having a reduced detection effect due to unable to adequately learn the feature distribution. In order to solve the data imbalance problem, some scholars attempt to reduce the category imbalance rate by reducing the number of network traffic samples in most categories (Zhong et al., 2009). However, this manner makes deep learning model inefficient due to the small number of training samples. For this reason, scholars try to expand the network traffic samples using data augmentation techniques. Currently, data augmentation methods for NID can be broadly categorized into sampling-based method (Liu et al., 2021), variational autoencoder (VAE)-based method (He et al., 2022) and generative adversarial network (GAN)-based method (Huang & Jafari, 2020). Among them, the sampling method (especially over-sampling) may lead to overfitting of the training model to a few classes due to the fact that the samples generated from sampling in a few classes are too similar to the original samples, resulting in inefficient NID. The network traffic samples generated by VAE have the problems of blurring and distortion. The GAN needs to train both the generator and discriminator at the same time, which is prone to appear the problems of pattern collapse and training instability (Ahsan et, al.,2024). Compared with these data augmentation methods, the diffusion models (Rombach et al., 2022; Ho, Jain, & Abbeel, 2020) have the advantages of better stability, controllability and comprehensibility, and the generated network traffic samples are more realistic. However, existing diffusion models are mainly used in the field of computer vision, while the number of features extracted from network traffic is smaller than that in the tasks such as image recognition and classification, which makes the traditional diffusion model unable to effectively learn the original distribution of network traffic, leading to blurring of images generated by the diffusion model. Therefore, it is necessary to design an efficient diffusion model for network traffic to generate high-quality traffic images in small categories.

In the field of NID, network traffic is mainly characterized in the format of PCAP, CSV and gray-scale images. Among them, the PCAP format stores a large amount of byte information, which consumes large resources of hardware when processing network traffic in this format as well as has the problem of low detection efficiency; The CSV format mainly extracts the important features as well as filters some information of network traffic from PCAP files, leading to the model suffering from an inability to adequately learn the features. The gray-scale images usually contain the packet-level features at the application layer, network layer, etc., this form can better enable the deep learning-based detection model to achieve good detection performance, but it has the problems of rapid loss of edge information due to the reduction in the size of output image as the convolution kernel cannot completely cover the edges. Compared with the gray-scale images, each pixel point in the RGB images contains three bytes, and each pixel point stores more information, thus, the use of RGB images makes the training of the model more efficient. However, the features of network traffic usually cannot fill all pixel points in the RGB images and required to be filled using 0X00. For example, to convert the features of network traffic into RGB images for better representing the features, it is assumed that the size of RGB image is designed as 32×32, which means that 3072 (32×32×3) bytes are needed to fill the RGB image. In fact, the feature dimensions of network traffic are much smaller than 3072, thus, traditional methods use 0X00 for filling the missing features. Compared with the features of network traffic itself, the filled 0X00 bytes has no help for network intrusion detection and also causes many parts of RGB images appearing black, which ultimately has a negative impact on intrusion detection. Therefore, it is necessary to study a better feature representation of network traffic in the format of RGB images.

In this paper, we propose an efficient network intrusion detection method based on denoising diffusion probabilistic model and dual-attention residual network, namely DDP-DAR, to address the problems existing in network intrusion detection phase, data augmentation phase and feature representation phase. The main contributions of this paper are as follows:

1. **The innovative feature representation of network traffic:** We propose a novel feature representation of network traffic in the format RGB images, this representation manner of RGB image contains the global features, network layer local features and session layer local features while reducing the frequency of feature padding using 0X00, which can better represent the network traffic.
2. **The innovative data augmentation model:** We propose an improved denoising diffusion probabilistic model called NT-DDPM for network traffic, it introduces the cosine noise addition and learnable variance parameter strategy to improve the quality of generated RGB images, which in turn solves the data imbalance problem and improves the detection accuracy of network traffic.
3. **The innovative network intrusion detection model:** For the RGB images that store more feature information of network traffic, we propose a NID model based on dual-attention residual network (NID-DARN). On the basis of ResNet-34 network, we introduce a dual-attention layer including a channel-attention module and a spatial-attention module in each residual block. Through the dual-attention mechanism and deep network structure, more advanced and important features of network traffic can be extracted from RGB images, thereby providing network traffic dataset with better quality for NID.
4. **The extensive experimental validation:** We conduct a large number of experiments to verify the efficiency of DDP-DAR method, extensive experimental results show that compared with five state-of-the-art data augmentation-based NID methods, the proposed DDP-DAR achieves optimal results in four metrics of Accuracy, F1-measure, FPR and ROC-AUC as well as detection stability, which indicates that DDP-DAR is effective in detecting network intrusions in large-scale network traffic.

The remainder of this paper is organized as follows. Section 2 reviews the related work about NID and data augmentation for NID. Section 3 describes the details of DDP-DAR, a NID method based on denoising diffusion probabilistic model and dual-attention residual network, including the framework of DDP-DAR, the feature representation of network traffic, the data augmentation based on an improved denoising diffusion probabilistic model, and the NID based on a dual-attention residual network. Section 4 demonstrates the experimental setup, including the description of network traffic datasets, the introduction of baselines and evaluation metrics. Section 5 provides the experimental results to validate the effectiveness of DDP-DAR. Section 6 summarizes the full paper and gives the future directions.

2. Related work

This section firstly reviews the existing network intrusion detection (NID) methods and data augmentation methods for NID, and then briefly compares the proposed DDP-DAR method with existing data

augmentation-based NID methods in terms of feature representation, the used data augmentation model and the used detection model.

2.1. Network intrusion detection methods

With the rapid development of artificial intelligence, NID methods have gradually evolved from traditional statistical-based, port-based and signature-based techniques to machine learning and deep learning-based techniques. Since machine learning and deep learning can perform intrusion detection by learning the features of network traffic, their efficiency is better than traditional methods.

2.1.1. Machine learning-based NID methods

In recent years, machine learning techniques have been widely used in the field of NID, and scholars have proposed many machine learning technologies to learn the features of network traffic generated by the network intrusion behavior, thereby effectively detecting the intrusion networks.

In machine learning-based NID, feature selection is a very important process, and the good features can improve the model's understanding of features, thus leading to training better detection models. In recent years, scholars have tried to investigate more efficient feature selection methods. For example, the multi-stage optimization of NID method (Injadat et al., 2021) provided better features through Z-score standardized preprocessing and feature selection, and then optimized the hyper-parameters of different machine learning models using three different optimization methods, including stochastic search, particle swarm optimization and Bayesian optimization. The supervised machine learning-based NID model (Das et al., 2022) provided better features through ensemble screening, wrapping and embedding methods. M-MultiSVM (Turukmane & Devendiran, 2024) used the modified singular value decomposition to extract the basic, content and traffic features, and then selected the optimal features using a dyadic northern goshawk optimization algorithm to assist the detection of network intrusion. The hybrid meta-heuristic technique and weighted XGBoost classifier-based NID model combined the Whale Optimization algorithm and Sin Cosine algorithm to pick the most discriminative and optimal features. The NID model (Mohi-Ud-Din et al., 2023) based on hybrid enhanced crow search algorithm (CSA) and particle swarm optimization (PSO) algorithm selected the global optimal features using the search strategy of CSA and the fast convergence property of PSO, and then updated the weights of these features using weighted least squares. The above methods can realize more efficient NID on the basis of feature extraction together with detection model, but these methods are based on a single machine learning classifier, they cannot cope with several different categories of network intrusion at the same time.

In order to solve the problem of low efficiency of individual machine learning classifier, many scholars have introduced ensemble learning into the field of NID, which made the model to reduce the prediction error of normal traffic and attack behavior. For example, Apollonp (Paya et al., 2024) used multiple classifiers to detect intrusion behaviors and used a multi-armed bandit with Thompson sampling to dynamically select the best classifier or combination of classifiers for each input, which made it difficult for attackers to generate antagonistic samples by learning the behavior of intrusion detection system in order to evade detection. Liu et al. (2021) proposed a tabular data sampling method to solve the imbalanced learning problem by mixing the normal data after under-sampling and the attack data after over-sampling. Although the use of ensemble learning can obtain better detection performance than a single classifier, but the distribution of network traffic is very unbalanced, resulting in the detection model not being able to adequately learn the features, thereby affecting the detection results.

However, traditional machine learning methods are not so efficient in face of large-scale network traffic. In recent years, deep learning techniques are widely used to detect cyber-attack behaviors in a more automated manner due to their powerful automatic feature extraction capabilities.

2.1.2. Deep learning-based NID methods

Deep learning-based NID methods not only automatically extract the features from network traffic, but also learn the patterns to predict unknown network traffic. These capabilities enable NID systems to effectively recognize attacks in the network environment. In recent years, deep learning models such as CNN, ResNet, LSTM, GNN and their variants are often applied in NID tasks.

CNN models are able to identify anomalous behaviors by extracting advanced features from network traffic. For example, the deep learning-based DDoS attack intrusion detection model (Akgun et al., 2022) combined the data preprocessing, feature selection and other techniques with CNN model to detect attack behaviors; CANET (Zhang et al., 2019) formed the CA Block by mixing CNN with attention mechanism, the combination of multi-layer CA Blocks was able to adequately learn the multi-level spatio-temporal features of network attack behavior to improve detection accuracy. ResNet is an improved CNN-based algorithm that improves the detection performance while extending the depth of network structure. For example, Suja Mary et al. (2024) identified the network attacks by addressing the fundamental big data complexity associated with many heterogeneous security data types, it first picked the important features through combining the Aquila Optimizer and Fuzzy Entropy Mutual Information, and then introduced an optimized classifier based on ResNet152 to accurately classify the intrusion patterns. The LSTM model captures the temporal features by learning the sequence-to-sequence relationships to perform efficient NID operation. For example, LIO-IDS (Gupta et al., 2021) handled both frequent and infrequent network intrusions by using the LSTM model to improve One-vs-One technique; Bidirectional LSTM-based NID model (Imrana et al., 2021) handled both forward and backward network traffic inputs through a bi-directional structure, thus efficiently detecting unknown network intrusions. GNN characterizes the features and their relationships in the form of graph, and then learns from the graph to better discover the correlated relationship of features in the intrusion traffic, thus improving the detection efficiency. For example, Anomal-E (Caville et al., 2022) extended the GraphSAGE graph model to capture the edge features and topological patterns in the graph and then performed self-supervised learning by maximizing the local edge information between different parts in the latent space to more accurately detect intrusion traffic. The early NID method based on graph embedding technique (Hu et al., 2023) first learned the vectorial representation of features through graph2vec, and then used the random forest to classify graph vectors to detect cyber-attacks.

In addition, some scholars used the hybrid depth models for NID. For example, the time convolution-based model (Lopes et al., 2023) improved the detection efficiency by combining minimally random convolutional kernel transform, explainable convolutional neural network, one-dimensional convolution neural network and time series transformer. Remora Whale Optimization (RWO)-based hybrid deep model (Pingale & Sutar, 2022) utilized the CNN model for feature extraction as well as the RV coefficient for feature selection, and then the model was trained using RWO optimization algorithm through combining Deep Maxout network and Deep Auto Encoder. Liu et al. (2022b) proposed a multi-task learning based NID method, it first identified whether an attack has occurred and the specific type of attack by using the multi-layer perceptron (MLP), and then used the self-encoder based comparative learning method to improve detection accuracy by pairwise optimizing the reconstruction error to make the reconstruction error of attack data larger than the normal data.

Despite deep learning-based methods are fruitful in the field of NID, their performance depends largely on the quality of network traffic. In the network environment, network traffic is often highly imbalanced, leading to the detection methods favoring the traffic classes with larger data sizes and ignoring a few classes during the training process, thereby ultimately affecting the detection results.

2.1.3. Data augmentation methods for NID

The data imbalance phenomenon in network traffic seriously affects the detection efficiency, thus, data augmentation methods for NID has received more attention. The data augmentation methods in NID field are mainly based on sampling, autoencoder and generative adversarial network (GAN).

Sampling-based methods sample new traffic as augmented data with the distribution of original network. For example, Le et al. (2024) proposed a hybrid under-sampling and over-sampling to solve data imbalance problem, and then input the balanced network traffic samples into a detector integrating XGBoost, LightGBM and CastBoost to accurately detect intrusions. Although the use of ensemble learning as well as under-sampling and over-sampling strategies can effectively solve the problems of data imbalance and poor detection efficiency, this method faces the problem of high computational overhead when dealing with large-scale network traffic.

Autoencoder-based data augmentation methods first map the input data to a probability distribution in the latent space through an encoder, and then generates the data through a decoder. For example, the autoencoder-based data augmentation model (Wei et al., 2024) utilized a self-encoder to improve the features of network traffic as well as improved the detection accuracy by encoding the features of malicious traffic. The data level-based model Liu et al. (2022b) used the variable autoencoder (VAE)-based as well as conditional VAE-based schemes for data augmentation. Although the autoencoder-based data augmentation methods have better data generation capability, but the VAE tends to generate fuzzy reconstruction results, leading to the problem of unsatisfactory training results.

GAN-based data augmentation methods make the generator to gradually learn to generate more realistic data as well as make the discriminator to better distinguish between real and generated data through adversarial learning of generator and discriminator. For example, the traditional GAN (Goodfellow et al., 2014) is consisted of two neural networks: a generator and a discriminator, the generator attempts to forge new data (fake data) that is identical to real data, while the discriminator attempts to distinguish between real data and fake data. TMG-GAN (Liu et al., 2022a) generates high-quality samples by using multiple generators and increasing the structure of classifiers, B-GAN (Ding et al., 2024) trains GANs based on the discrete values for data augmentation, APELID (Vo et al., 2024) improves the detection performance of intrusion by improving the quality of training datasets and employing a large number of powerful AI models. In addition, CGAN (Wang et al., 2020) introduces the conditional information on the basis of traditional GAN, it enables the generator to generate synthetic data that matches given conditions, thereby improving the controllability and targeting of generated data. The discriminator and generator of DCGAN (Radford, Metz, & Chintala, 2016) adopt deep learning models (especially for the convolutional neural networks (CNN)), which normalize the input of each neural unit to ensure zero mean and variance units, thereby effectively improving the consistency and efficiency of learning. CycleGAN (Zhu, Park, Isola, & Efros, 2020; He et al., 2023) is a method of automatically training an image to image translation model using GAN architecture without the need for paired examples, it utilizes a set of unrelated images from different source and target domains (such as X and Y domains), where the structure of model includes two generators and each generator is associated with a corresponding discriminator for binary classification. ByteSGAN (Wang et al., 2021) only requires a small number of labeled samples of network traffic, it achieves good classification performance by modifying the structure and loss function of discriminator in the conventional GAN model and obtaining a large number of samples through semi-supervised learning. However, traditional GAN is unable to effectively control the patterns of generated data. Although many scholars have proposed various variants of GANs to solve this problem, but the inherent structure of GAN models requires simultaneous training of generators and discriminators, making the training objectives very complex, which further leads to pattern collapse and model oscillation during the training process, thereby affecting the quality of the generated images and ultimately reducing the accuracy of network intrusion detection.

Although existing data augmentation methods (such as sampling, VAE, GAN, etc.) have the ability to generate data, but the quality of generated network traffic samples is not ideal. The diffusion model has emerged as one of the most mainstream generation models in the fields of text generation and image generation due to its advantages of stability, controllability and comprehensibility, it can be used in the field of NID to improve the quality of generated samples.

Condition guided-based diffusion model is a mainstream diffusion model, it achieves data augmentation through a forward process and a reverse process. This model first introduces the pre-trained classification model in the reverse process to generate the prediction label of image, and then calculates the gradient of cross-entropy loss based on the classification scores and target category for guiding the generation of samples. For example, the dual-granularity condition-guided method DiffMIC (Yang et al., 2023a), the dual-orbit transformer network-based method GFMDiff (Xu et al., 2024) and the multi-scale content aggregation module-based method FontDiffuser (Yang et al., 2023b) perform data augmentation by combining the global features and local relations of images to achieve better detection results. However, the effectiveness of condition guided-based diffusion model depends on the quality of pre-trained classification model, resulting in the inability to well guide the whole diffusion model to generate samples with high-quality once the classification model is poorly trained. Compared with condition guided-based diffusion models, denoising diffusion model generates high-quality images by gradually adding Gaussian noise to the real images and then gradually denoising the images with added noise, which has the advantages of high stability, strong controllability, diversity of generated images and good interpretability. Based on the advantages, the denoising diffusion models including SPD-DDPM (Li et al., 2023), DDPM-SKDNet (Wang et al., 2023), DiffSED (Bhosale et al., 2024), SU-DDPM (Lu et al., 2024) and GDDIM (Zhang, Tao, & Chen, 2022) have been proposed to improve the classification ability of deep learning models. However, existing diffusion models are mainly oriented to the field of computer vision, and to our best knowledge, no diffusion models have been used in the field of NID until now. Due to the format of network traffic is PCAP and CSV, while the diffusion model is usually used for processing the gray-scale images and RGB images, therefore, it is necessary to convert the format of network traffic to make the diffusion model applicable to the NID tasks.

The comparison of DDP-DAR method with the existing data augmentation-based NID methods is shown in Table 1.

Table 1
The comparison of data augmentation-based network intrusion detection methods.

Methods	Feature representation	Used data augmentation model	Used intrusion detection model
TMG-GAN (Liu et al., 2022a)	PCAP	GAN	Deep neural network
Le et al. (2024)	PCAP	Hybrid sampling	Ensemble learners
Liu et al. (2022b)	PCAP	Random unsampling CVAE	CNN/GRU
Wei et al. (2024)	PCAP	VAE	Dual classifier
B-GAN (Ding et al., 2024)	CSV	GAN	GAN
APELID (Cai et al., 2023)	CSV	Augmented wasserstein GAN	Ensemble learners
CycleGAN (He et al., 2023)	Gray-scale images	GAN	RNN
DDP-DAR (proposed)	RGB images	Denoising diffusion probabilistic model	Dual-attention residual network

3. Methodology

This section first describes the overall framework of the proposed DDP-DAR method, and then further describes the details of feature representation of network traffic, data augmentation based on denoising diffusion probabilistic model, and network intrusion detection based on dual-attention residual network.

3.1. The framework of DDP-DAR method

The framework of DDP-DAR method is shown in Fig. 1, it is consisted of feature representation module, data augmentation module and network intrusion detection module. Firstly, the network traffic is converted into RGB images by the feature representation module. And then, the data augmentation module is used to generate the samples in small category to balance the distribution of network traffic. Finally, the augmented network traffic samples are fed into the network intrusion detection module to detect network intrusions.

(1) **Feature representation module:** This module is mainly used to pre-process the network traffic, thereby representing the features of network traffic in the format of RGB images through traffic segmentation, traffic anonymization, traffic cleaning and feature transformation. Compared with traditional formats of PCAP, CSV and gray-scale images, RGB images can better store the global features and local features of network traffic to improve the performance of detection model.

(2) **Data augmentation module:** This module is mainly used to solve the data imbalance problem by expanding the smaller categories of network traffic through the forward process and inverse process of denoising diffusion probabilistic model to achieve data augmentation, thereby reducing the worse detection performance due to insufficient data samples.

Network intrusion detection module: This module is mainly used to detect the malicious traffic generated by intrusion behaviors. Through constructing a dual-attention residual network from both perspectives of RGB image channel and spatial channel, it focuses on the information related to network traffic to extract more important features, thus realizing more accurate NID.

3.2. Feature representation of network traffic

In the field of NID, network traffic is usually characterized in the format of PCAP, CSV and gray-scale images. However, PCAP can result in an inconspicuous feature representation due to the adulteration of a large amount of invalid or repetitive data, CSV rounds off some information of network traffic during feature extraction of PCAP, gray-scale images cannot represent the features well due to limited feature storage. Compared with above manners of feature representation, RGB image contains three channels of R, G and B, this manner can store more features than gray-scale images, thus, it can help the intrusion detection model to learn more features and thus benefit for obtaining better detection results.

In general, the feature representation method converts the network traffic into RGB images through traffic segmentation, traffic anonymization and traffic cleaning. Among them, the cleaned features have a uniform length of 1024 bytes (the first 1024 bytes are intercepted if the length is greater than 1024, otherwise 0X00 is supplemented at the end), and then the features with uniform-length are converted into an RGB image according to the binary form, where every three consecutive bytes corresponds to an RGB pixel point and each byte is mapped to [0, 255]. Because the RGB image has three channels, it requires extra 1024 bytes compared with gray-scale image, causing requiring more 0X00 for feature padding. However, the 0X00 used for padding only play the role of feature padding but has no effect or even an opposite effect on the features of RGB image, which results in the failure to fully ensure the effectiveness of feature representation in RGB images. To solve this problem, this paper proposes a novel feature representation method to convert the network traffic into RGB images, the specific process is shown in Fig. 2.

From a layering perspective, the intrinsic features of network traffic are reflected in the application layer of TCP/IP model (i.e., the 7th layer of OSI model), thus, only extracting the features in application layer can represent the intrinsic features of network traffic. However, the data of other layers also contain some information of network traffic. For example, the session layer information in the 5th layer of OSI model is responsible for establishing, managing and terminating sessions, it contains the correct source and destination of network traffic and can ensure the correct transmission of data between the source and destination. Through extracting the feature information in application layer

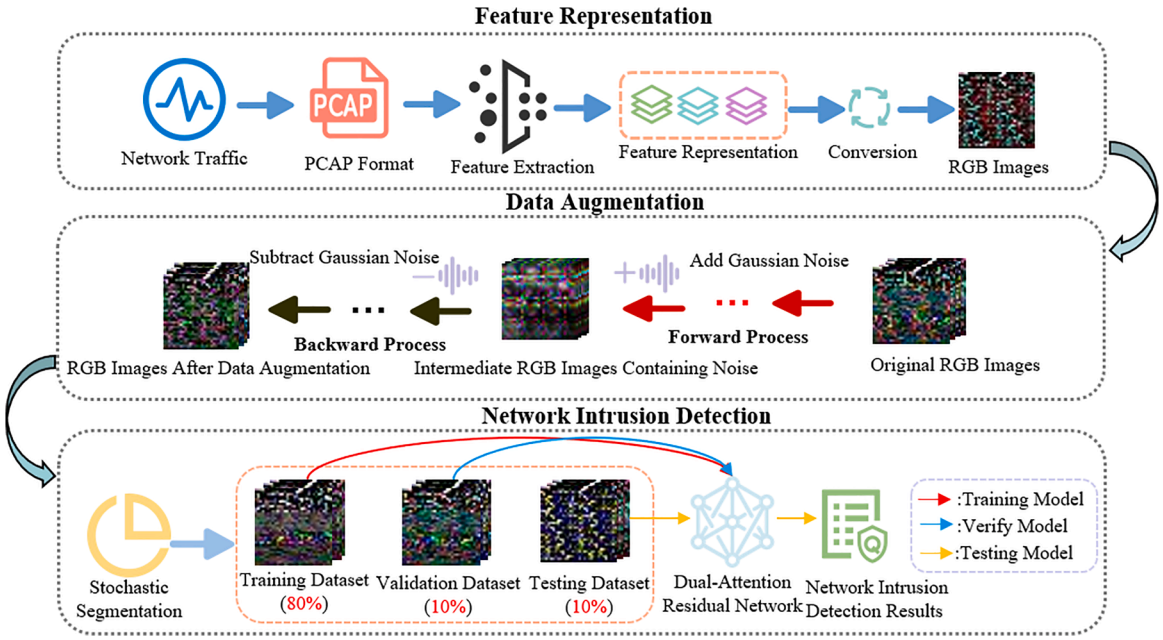


Fig. 1. The framework of DDP-DAR method.

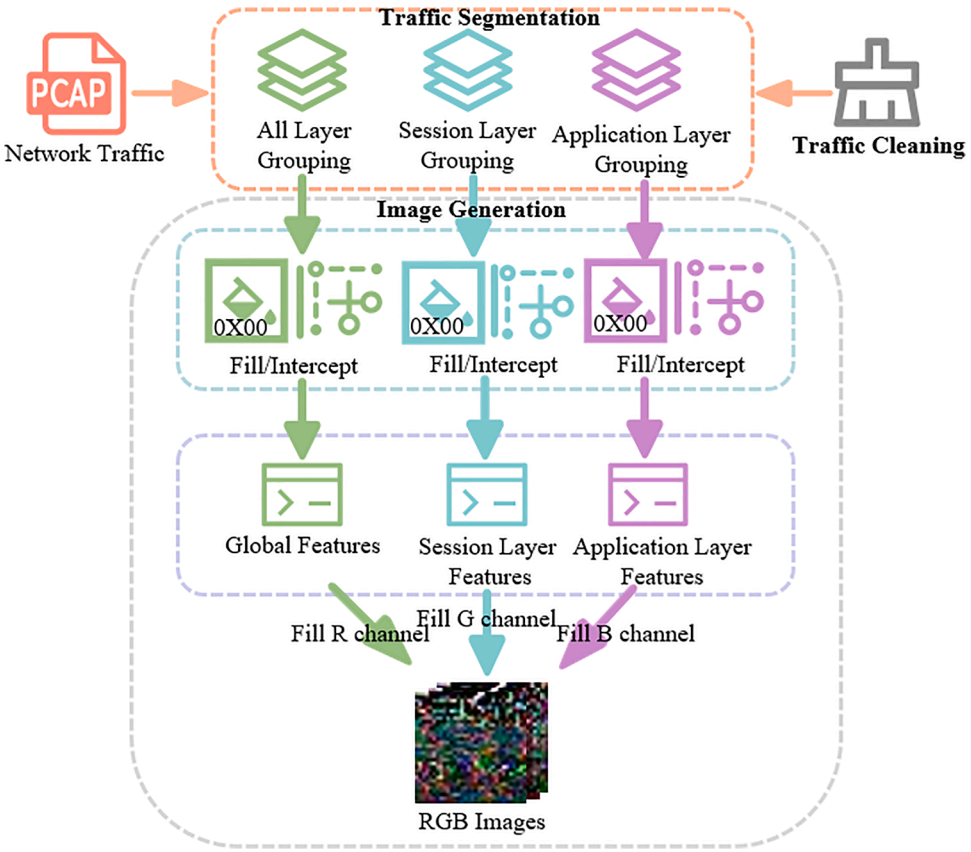


Fig. 2. The process of feature representation.

and the application information in session layer, network traffic can be grasped holistically at two levels of data content and data transmission direction. However, both categories of features are localized, thus, we additionally extract the global features of all layers again. Through fusing the above three categories of features, network traffic can be better characterized, which can further improve the detection performance.

Unlike the traditional feature representation in the format of RGB images that populates the features of network traffic in the R, G and B channels of each pixel point in turn, the proposed feature representation method starts from the perspective of three different categories of packet layer selection, including: all layers (L_{All}), layer 5 only (L_5) and layer 7 only (L_7), i.e., Session+L_{All}, Session+ L_5 and Session+ L_7. And then, each category of cladding features is populated in three channels of R, G and B, respectively, through traffic segmentation, traffic cleaning and image generation.

(1) **Traffic segmentation:** This step converts the continuous original network traffic into multiple discrete traffic groupings.
(2) **Traffic cleaning:** This step is mainly divided into traffic anonymization/cleaning and file cleaning. Among them, traffic anonymization/cleaning randomizes the MAC addresses and IP addresses in the data link layer and IP layer, respectively, thereby avoiding the IP and MAC information may damage the feature extraction process; File cleaning improves the quality of network traffic by cleaning up the empty and duplicate files.
(3) **Image generation:** This step firstly unifies the length of cleaned network traffic features (L_{All}, L_5, L_7) to 1024 bytes (the size of converted RGB image is 32×32). If the length is greater than 1024 bytes, the first 1024 bytes are intercepted; Otherwise, 0X00 is supplemented at the end to fill it to 1024 bytes. And then, the features of uniform length are saved as F_R, F_G and F_B according to the binary form. Finally, the features are converted to RGB images according to Eq. (1).

$$RGB=\begin{bmatrix} (F_{A_1},F_{7_1},F_{5_1}) & (F_{A_2},F_{7_2},F_{5_2}) & \cdots & (F_{A_{32}},F_{7_{32}},F_{5_{32}}) \\ (F_{A_{33}},F_{7_{33}},F_{5_{33}}) & (F_{A_{34}},F_{7_{34}},F_{5_{34}}) & \cdots & (F_{A_{64}},F_{7_{64}},F_{5_{64}}) \\ \vdots & \vdots & \vdots & \vdots \\ (F_{A_{993}},F_{7_{993}},F_{5_{993}}) & (F_{A_{994}},F_{7_{994}},F_{5_{994}}) & \cdots & (F_{A_{1024}},F_{7_{1024}},F_{5_{1024}}) \end{bmatrix} \tag{1}$$

This processing manner allows the detection model to learn not only the global features of network traffic, but also the local features. At the same time, this manner also reduces the frequency of feature padding using 0X00, which leads to a better representation of network traffic.

The specific process of feature representation of network traffic in the format of RGB images is shown in Algorithm 1. Firstly, from the perspective of three packet layers including all layers, session layer and application layer, network traffic is divided into groupings with fixed size through traversing each category of original network traffic packets (lines 2–4). The groupings are then cleaned up by deleting blank files, duplicate files, etc. (lines 5–7). Next, the length of each feature is fixed by intercepting or supplementing 0X00 to obtain the features of F_R, F_G and F_B (line 8), where the all-layer features F_R are populated into R channel, the session-layer features F_G are populated into G channel and the application-layer features F_B are populated into B channel (line 9), thereby obtaining RGB images (line 11).

3.3. Improved denoising diffusion probabilistic model

In recent years, data augmentation techniques such as sampling, autoencoder, GAN and diffusion model have been used to alleviate the decrease of detection performance due to data imbalance problem, and

Algorithm 1
Feature representation.

Input: Network Traffic X and their category *label* in PCAP format
Output: RGB images
01: **for** each label in X **do**
02: $L_{All}R$ = Splite(*label*) // Segmentation of all layer information
03: L_5G = Splite(*label*) // Segmentation of all layer information
04: L_7B = Splite(*label*) // Segmentation of all layer information
05: *FilteredSession*$\_R$ = Clean($L_{All}R$)// Clean up $L_{All}R$
06: *FilteredSession*$\_G$ = Clean(L_5G)// Clean up L_5G
07: *FilteredSession*$\_B$ = Clean(L_7B)// Clean up L_7B
08: Intercept or supplement 0X00 to obtain F_R, F_G and F_B
09: F_R, F_G and F_B are filled according to Eq. (1) to get RGB image
10: **end for**
11: *Result* = RGB images
12: **return** *Result*

diffusion model has emerged as one of the most mainstream generative models. At present, the widely used diffusion models mainly include conditional diffusion model (Niu et al., 2024), stable diffusion model (Fernando & Tsokos, 2022) and denoising diffusion probabilistic model (DDPM) (Rombach et al., 2022). Among them, the conditional diffusion model usually requires using two models to collaborate in sampling, resulting in lower sampling efficiency; The stable diffusion model controls the stability by introducing stable coefficients, but it sacrifices the diversity of some data samples. In contrast, DDPM can generate high-quality and diverse samples, thus, it is chosen as the base model.

Currently, DDPM is mainly active in the field of computer vision. Compared to the visual images in computer vision, the features of network traffic characterized by RGB images is very limited, which leads to the fact that DDPM cannot effectively learn the original distribution of network traffic. In order to more realistically increase the number of network traffic samples in small classes, we introduce the strategies such as variance learnability and modification of noise in the basic DDPM model to design an improved denoised diffusion probabilistic model called NT-DDPM for network traffic. The process of NT-DDPM model is shown in Fig. 3, it generates the high-quality network traffic samples by focusing on the important information in RGB images. The data augmentation operation is mainly performed through forward process and reverse process.

3.3.1. Forward process

The forward process of NT-DDPM is also known as the diffusion process, it is a Markov chain, i.e., the state of moment T is only related to the state of moment (T-1). Conventional DDPM gradually introduces the noise $\varepsilon \sim N(0, I)$ to initial data $x_0 \sim q(x_0)$ and gradually transforms the original data into Gaussian noise with random normal distribution as shown in Eq. (2), and then directly generates data x_T based on the initial data x_0 at any time step T by employing a reparameterization technique, as shown in Eq. (3).

$$q(x_T|x_{T-1}) = \mathrm{N}\left(x_T; \sqrt{1-\beta_T}x_{T-1}, \beta_T I\right) \tag{2}$$

$$x_T = \sqrt{\prod_{t=1}^{T}(1-\beta_t)}x_0 + \sqrt{1-\prod_{t=1}^{T}(1-\beta_t)}z, z \sim \mathrm{N}\left(0, \left(1-\prod_{t=1}^{T}(1-\beta_t)\right)I\right) \tag{3}$$

In Eqs. (2) and (3), $q(x_T|x_{T-1})$ represents the probability of x_T obtained by adding Gaussian noise to x_{T-1} in the forward process, β_T represents the variance that increases linearly with time step T, and z is a standard normally distributed random noise.

Considering that characterizing the network traffic as RGB images requires storing as many features as possible, therefore, the size of the model is set to 32×32 when performing RGB image conversion operation. The degree of noise addition in the DDPM exhibits a trend of increasing linearly over time, it is only applicable to high-resolution images, but performs not very well in the RGB images with a size of 32×32 due to the fact that too much forward noise addition does not improve the quality of samples but is counterproductive. For this reason, we use a nonlinear cosine method as shown in Eq. (4) to make $\prod_{t=1}^{T}(1-\beta_T)$ (denoted as α_T) in x_T decreases according to a nearly linear trend and smoother near to $T = 0$ or $T = T'$.

$$\alpha_T = \prod_{t=1}^{T}(1-\beta_t) = \frac{\cos\left(\frac{T/T'+s}{1+s}\cdot\frac{\pi}{2}\right)^2}{\cos\left(\frac{s}{1+s}\cdot\frac{\pi}{2}\right)^2} \tag{4}$$

In Eq. (4), s(=0.008) is a bias parameter.

Through adopting the nonlinear cosine noise addition manner, the quality of RGB image sampling can be improved to a great extent.

3.3.2. Reverse process

The reverse process of NT-DDPM is iterative denoising inference. In the diffusion process with infinite step size, the role of variance is much weaker than the effect of mean, thus, the σ_T in $\sum_{\theta}(x_T, x_0, T) = \sigma_T I$ of traditional DDPM is fixed to β_T as shown in Eq. (5). Specifically, the reverse process randomly samples a 2D Gaussian noise at moment T and progressively denoises it to obtain a new image, and then starts from the standard Gaussian distribution to calculate the posterior distribution

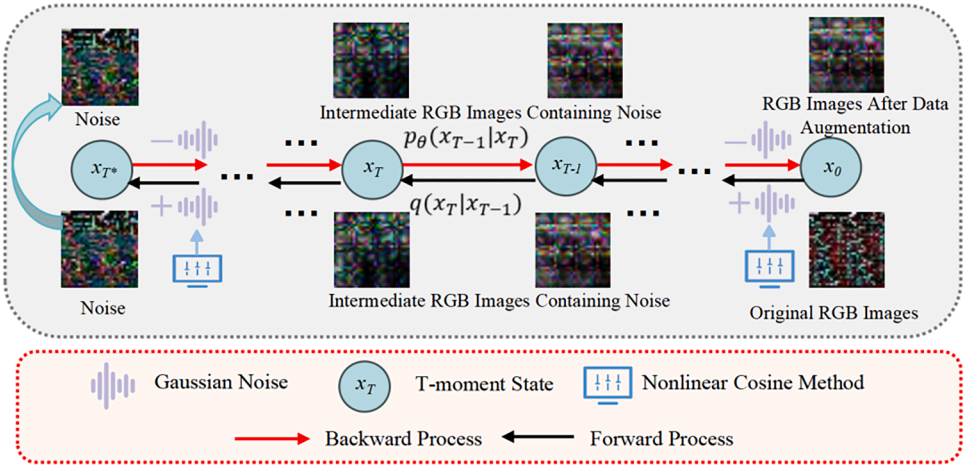


Fig. 3. The process of NT-DDPM model.

step by step according to the Markov chain and Bayesian formula, where the posterior distribution can be represented by a Gaussian distribution as shown in Eq. (5). These posterior probabilities are used to guide the step-by-step removal of noise from the noisy image to obtain a new RGB image.

$$p_\theta(x_{T-1}|x_T,x_0)=\mathrm{N}\left(x_{T-1};\mu_\theta(x_T,x_0,T),\sum_\theta(x_T,x_0,T)\right)$$
$$s.t.\quad \mu_\theta(x_T,x_0,T)=\frac{\sqrt{\prod_{t=1}^{T-1}(1-\beta_t)}(1-\beta_t)}{1-\prod_{t=1}^{T}(1-\beta_t)}x_0+\frac{\sqrt{1-\beta_t}\left(1-\prod_{t=1}^{T-1}(1-\beta_t)\right)}{1-\prod_{t=1}^{T}(1-\beta_t)}x_T$$
$$\sum_\theta(x_T,x_0,T)=\frac{\left(1-\prod_{t=1}^{T-1}(1-\beta_t)\right)(1-\beta_t)}{1-\prod_{t=1}^{T}(1-\beta_t)} \tag{5}$$

In the data augmentation process, we would like the generated RGB images as similar as possible to the original RGB images. The log-likelihood function can help quantify the fit of the model to actual observed data, i.e., the higher value of log-likelihood function indicates the better fit between the model's prediction and real data. Therefore, the quality of generated RGB images can be measured by the log-likelihood function value, thereby judging the realism degree of the generated RGB images. Although DDPM argues that fix σ_T can select the samples with better quality, but this operation does not focus on the log-likelihood function value, resulting in a poor measure of RGB image generation. Since the noise addition method and noise addition level at different times have an effect on the whole diffusion process, the use of non-fixed values of σ_T can improve the log-likelihood function value. For this purpose, the variance parameterization is defined as the interpolation of β_T or $\widetilde{\beta}_T$ in the logarithmic domain that varies with the output v and the change in noise addition at current moment as shown in Eq. (6), where v is a vector of model outputs and each dimension contains one component. Through logarithmic transformation, the variance values can be mapped to a more reasonable range, which improves the stability and efficiency of training; At the same time, this interpolation allows the model to adjust the variance more smoothly, thereby further improving the continuity and quality of the generated RGB images.

$$H_\theta=\sum_\theta(x_t,x_0,t)=\exp(v\log\beta_t+(1-v)\log\widetilde{\beta}_t) \tag{6}$$

With this operation, a more flexible posterior distribution can be obtained for denoising the noisy image, thus allowing the obtained log-likelihood function value to generate more realistic RGB images.

The specific process of NT-DDPM is shown in Algorithm 2. In the forward process (i.e., the model training phase), an image X^* is first randomly sampled from the original RGB image (line 4), and then a sample t is sampled from the uniform distribution $Uniform(\{1,\ldots,T\}$ (line 5) and adding t times to each RGB image X^* with Gaussian noise sampled from the standard normal distribution $N(0,I)(\varepsilon_1,\varepsilon_2,\ldots,\varepsilon_T)$(line 6), ε and the cosine addition mechanism is utilized to control the noise (line 7) to obtain the Gaussian noise X_T (line 8). Next, X_T is fed into unity network (U-Net) (line 9) and the output of U-Net is used to fit the noise ε in X until convergence (lines 10 and 11). In the inverse process, a noisy image X_T^* is first sampled from the standard normal distribution $N(0,I)$ (line 16), it is fed into U-Net to output the Gaussian noise ε_T (lines 18–19), and z is obtained through sampling from the standard normal distribution $N(0,I)$ (lines 20–24). Then, the denoising operation is carried out for X_T noise operation using the noisy images x_T, ε_T and z, (line 25), and the above process is repeated for T times to generate image X_0 (lines 27–28).

Compared with machine learning models, deep learning can obtain better efficiency with their powerful autonomous learning capability (Gamage & Samarabandu, 2020). With the increase of network depth, deep learning models can better fit potential mapping relationships and have stronger expressive ability. However, traditional deep learning-based NID models such as RNN, CNN, LSTM, etc. suffer from gradient explosion due to increasing network depth and rapid degradation of accuracy after gradual saturation. Compared with these models, deep residual network (ResNet) (Xia et al., 2022) has better convergence, it not only effectively solves the degradation problem of deep neural network, but also improves the detection performance while expanding the network depth.

However, the large amount of information in the filled RGB images makes the ResNet suffering from the problems of information overload and low processing efficiency of task. In order to solve these problems, we introduce a dual-attention mechanism in the ResNet to make the model pay more attention to the important regions of RGB images, thereby more accurately detecting network intrusions.

3.4. Dual attention mechanism

The RGB images have three feature channels of R, G and B, each feature has a different degree of importance. Traditional ResNet assigns same weight to the features in each channel, resulting in the inability to effectively differentiate the impact of each feature on intrusion detection. In order to obtain different important degrees of each channel in the RGB image to allow ResNet paying more attention to certain channel features, a channel attention mechanism is introduced into ResNet to efficiently model the channel-level relationships, thus enhancing its representation capability. The channel attention module consists of a global maximum pooling layer MaxPool, a global average pooling layer AvgPool and a multilayer perceptron layer MLP, it adjusts the weight of each important feature by calculating the channel attention weight coefficients F_c^* as shown in Eq. (7), thereby reducing the influence of noise and other disturbing information on the detection result.

$$F_c^*=F*Sigmoid(MLP(AvgPool(F))+MLP(MaxPool(F))) \tag{7}$$

In Eq. (7), F represents the features in the RGB image and Sigmoid is the activation function.

In the conversion process of RGB images, it is often filled by 0X00 when the features of network traffic are insufficient, while the filled information is unimportant compared to the information in the RGB image itself, i.e., the importance of information in different locations of RGB image is different. The use of channel attention mechanism can help ResNet to distinguish which information is useful, but it cannot distinguish the location of useful information, resulting in it cannot capture the important region, which in turn causes the decline of detection accuracy. To solve this problem, the spatial attention mechanism is introduced on the basis of channel attention mechanism to form a dual attention mechanism, which allows ResNet to focus on locally important features in the RGB images through the joint use of two attention mechanisms and suppresses the impact of other useless information on intrusion detection.

The spatial attention layer consists of global maximum pooling layer, global average pooling layer and convolutional layer, it emphasizes the different importance of different locations in the RGB images, and achieves the attention to different regions by learning the relationship between individual pixels, thereby obtaining the spatial attention weight coefficients F_s^* as shown in Eq. (8).

$$F_s^*=F*Sigmoid\left(C^{7\times7}(AvgPool(F),MaxPool(F))\right) \tag{8}$$

In Eq. (8), F represents the features in the RGB image, Sigmoid is the activation function, and C^{7*7} represents the 7×7 convolution operation.

The dual attention module connects the channel attention and the spatial attention serially, it first sends F to the channel attention module

Algorithm 2
NT-DDPM.

Input: Original RGB images X, Traffic category *label*
Output: RGB images
for each *label* **do**
  for each X in *label* **do**
    repeat
    $X^* \backsim q(X)$ // Image X^* is randomly sampled from the original image $q(X)$
    $t \backsim$ Uniform({1,…, T})
    $\varepsilon \backsim N(0,I)$
    Set the nonlinear cosine method to control the added noise according to Eq. (4)
    Sample the noise image X_t according to Eq. (3)
    U-Net $\leftarrow X_t$ // training the network
    take gradient descent step on
    $G = \nabla_\theta \| z - z_\theta\left(\sqrt{\prod_{t=1}^{T}(1-\beta_t)}x_0 + \sqrt{1-\prod_{t=1}^{T}(1-\beta_t)}z\right)\|^2$
    until converged
  end for
end for
return training weight
$X_T^* \backsim N(0,I)$
for $t = T,\ldots, 1$ **do**
  U-Net $\leftarrow X_T^*$
  Output ε_T
  if $t > 1$ **then**
    $z \backsim N(0,I)$
  else
    $z = 0$
  end if
  $x_{t-1} = \frac{1}{\prod_{t=1}^{T}(1-\beta_t)}\left(x_t - \frac{1-\prod_{t=1}^{T}(1-\beta_t)}{\prod_{s=0}^{t}\prod_{t=1}^{T}(1-\beta_t)}\varepsilon_\theta(x_t,t)\right) + \sigma_t z$
end for
Get the newly generated image X_0
Result = X_0
return *Result*

for maximum pooling and global average pooling operation, and then sends the result to the multilayer perceptual machine to learn and outputs the result to perform "+" operation. Then, it goes through the mapping process of Sigmoid function to obtain the weight coefficient H_c, H_c is then multiplied with F to obtain the scaled new feature F_c^* as shown in Eq. (7). Next, F_c^* is subjected to maximum pooling and global average pooling to obtain two channel descriptions of $H \times W \times 1$. The results of global pooling and average pooling are spliced by channel to obtain a feature image with the dimension of $H \times W \times 2$. Then, the splicing result is subjected to the convolution operation C of 7×7 convolutional layers to obtain a feature dimension of $H \times W \times 1$, and it is processed by the Sigmoid function to obtain the weight coefficients H_s. Finally, H_s is multiplied by F to obtain the new feature F^* as shown in Eq. (9).

$$F^* = F * Sigmoid\left(C^{7\times7}\left(AvgPool\left(F_c^*\right), MaxPool\left(F_c^*\right)\right)\right) \tag{9}$$

3.4.2. ELU activation function

The basic ResNet uses unsaturated activation function ReLU as shown in Eq. (10), which results in a very fast convergence of the model because it only has linear relationship. However, when $x < 0$, the neuron gradient of ReLU will be set to 0, leading to the phenomenon of "neuron death", i.e., the input features will not have any effect on the model, which reduces the detection performance.

$$ReLU(x) = \begin{cases} \max(0,x) & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{10}$$

To solve this problem, we modify the ReLU activation function to ELU as shown in Eq. (11). Compared to ReLU, ELU is a kind of activation function used to enhance the learning ability, which not only features the zero-mean distribution of outputs (which can speed up the training speed), but also is unidirectionally saturated. ELU avoids the "neuron death" problem by providing negative outputs with negative inputs.

$$ELU(x) = \begin{cases} x & x > 0 \\ \alpha(e^x - 1) & x \leq 0 \end{cases} \tag{11}$$

3.4.3. The structure of NID-DARN

The structure of proposed network intrusion detection method based on dual-attention residual network (NID-DARN) is shown in Fig. 4.

Among many ResNet models, the 34-layer network depth has fewer parameters and can be trained in a shorter period with better training results, thus, the construction of NID-DARN uses ResNet-34 as the basic model. Since a larger convolutional kernel can provide a larger sensory field with the same number of parameters to capture the global information of RGB images, the first two layers follow the structure of GoogLeNet, i.e., using a 7×7 convolutional layer with the number of output channels of 64 and a step size of 2, followed by a 3×3 maximal pooling layer with a step size of 2 to perform the down-sampling. Specifically, the NID-DARN consists of five modules, including conv1, conv2, conv3, conv4 and conv5 in order. Among them, the conv1 module includes a structure of one layer of 7×7 convolutional layers with a step size of 2, it is used to capture the global information T in RGB image to reduce the spatial dimensions that need to be processed by the subsequent layers, where T is computed as shown in Eq. (12). The conv2 module consists of a 3×3 maximum pooling layer with a step size of 2 and 3 residual block layers containing 2 residual blocks. The conv3 module consists of a 4-layer residual block layer containing 2 residual blocks. The conv4 module consists of a 6-layer residual block layer containing 2 residual blocks. The conv5 module consists of a 3-layer residual block layer containing 2 residual blocks. The following four modules reduce the size of features and increase the number of output channels by down-sampling, and the latter module extracts the higher-level features by its predecessor module to form a finer-grained feature T_f. The residual blocks in the residual block layer follow the complete 3×3 convolutional layer, we add the batch normalization layer BN and the activation function ELU after each convolutional layer, and then add a double attention layer after the last BN layer. Among them, BN layer makes the dimension corresponding to each channel of the feature matrix obtained after convolution satisfies the distribution law with mean 0 and variance 1, thus accelerating the convergence of network and improving the detection accuracy. After feature extraction and nonlinear mapping based on dual-attention mechanism in multiple residual block layers, the obtained feature $T_f = (z_1, z_2, \ldots, z_K)$ (where K is the number of categories) is input into the global average pooling layer, and then all pixel values of features are summed up for averaging to obtain the conditional probability of each category by normalization through Softmax function. Finally, the category with largest output value is selected as the input category L by argmax function, as shown in Eq. (13).

$$T = ELU\left(BN\left(C^{7\times7}(X_i, stride, padding)\right)\right) \tag{12}$$

$$L = \text{argmax}\left(\frac{\exp(z_1)}{\sum_k^K z_k}, \ldots, \frac{\exp(z_K)}{\sum_k^K z_k}\right) \tag{13}$$

In Eq. (12), stride is step size, padding is zero padding.

3.4.4. The workflow of NID-DARN

The workflow of NID-DARN is shown in Algorithm 3.

Firstly, the input RGB images are sent to the conv1 module for down-sampling to obtain feature T_1 (lines 3 and 4). Then, T_1 is fed to the conv2 module (line 5), which undergoes convolutional layer for feature extraction, normalization layer for normalization, activation function for activation (line 6), and then calculates the weight coefficients in dual attention mechanism layer (line 7) to obtain the final weight coefficients, which then do "×" operation with the input features to get higher attention feature T_2 (line 8). Because the modules of conv2, conv3, conv4 and

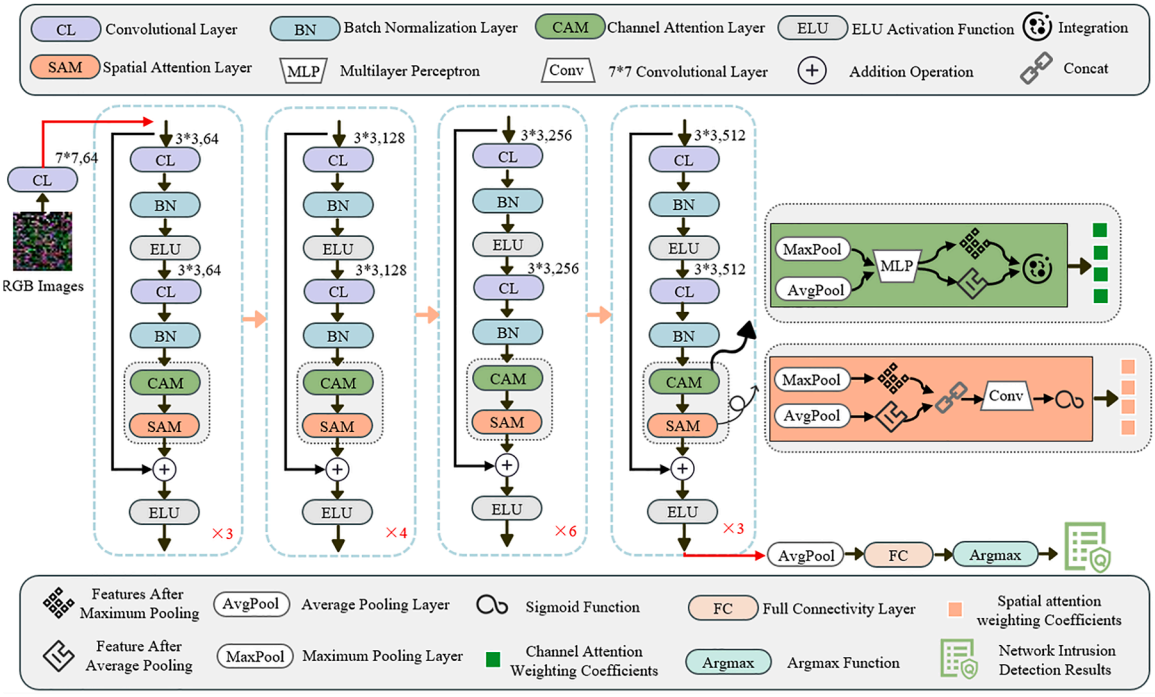


Fig. 4. The structure of NID-DARN.

conv5 are connected serially, the similar feature extraction method is used to sequentially get the features T_3, T_4 and T_5 (line 9). Finally, T_5 is input into the global average pooling layer for pooling (line 10) and the category with largest output value is selected as the input category of network traffic by Softmax function (line 11) and argmax function (line 12), thereby obtaining the final detection result (line 16).

3.4.5. *The explanation of DDP-DAR architecture and training process*

The proposed DDP-DAR model is composed of three phases: feature representation phase, data augmentation phase and network intrusion detection phase. The flowchart of DDP-DAR model is shown in Fig. 5.

(1) **Feature representation phase:** At this phase, the pcap packets of network traffic are first selected at the packet level from three different perspectives: all layers, 5th layer and 7th layer. And then, the traffic segmentation is performed for each packet layer selection to convert the continuous network traffic into multiple discrete traffic groups to obtain all layer groups, 5th layer groups and 7th layer groups. Next, the obtained layer groups are performed traffic cleaning to randomize the MAC addresses and IP addresses of data link layer and IP layer to avoid IP and MAC information damaging the feature extraction process; At the same time, the quality of network traffic is improved by cleaning empty and duplicate files to obtain the cleaned layer groups. Then, the length of network traffic features for layer groups is unified to 1024 bytes (that is, the converted RGB image size is 32 * 32) to generate the RGB images of network traffic; If the length is greater than 1024 bytes, the first 1024 bytes are intercepted; Otherwise, 0X00 is added to fill it to 1024 bytes. According to the binary form, the uniformly long features are saved as F_R, F_G and F_B. Finally, F_R, F_G and F_B are respectively filled in the R, G and B channels to obtain RGB images of network traffic.

(2) **Data augmentation phase:** At this phase, the denoising diffusion probability model DDPM is used to focus on important information in the RGB images of network traffic and generate high-quality network traffic samples. This process is mainly

Algorithm 3
NID-DARN.

Input: RGB images after data augmentation, Traffic category *label*
Output: Intrusion network
01: **for** each *label* **do**
02: **for** each RGB image X_i **do**
03: Capture the global information T_1 via conv1
04: $T_1 = \mathrm{ELU}(\mathrm{BN}(\mathrm{C}^{7\times7}(X_i, stride, padding)))$
05: $T_1 \rightarrow$ conv2 // T_1 is input to conv2
06: $T_2 = \mathrm{ELU}(\mathrm{BN}(\mathrm{C}^{3\times3}(T_1, stride, padding)))$ // undergo convolution, normalization and activation operations
07: W = Dual-attention_mechanism(T_2) // calculate the weight coeffi-cients
08: $T_2 = \mathrm{W} \times T_1$ // obtain the features after conv2
09: Sequentially input higher-level information of T_2 into conv3, conv4 and conv5 to obtain T_3, T_4 and T_5 respectively through equations (7)-(9)
10: Feed T_5 into the global average pooling layer to obtain $z_1, z_2, \ldots\ldots, z_K$
11: $P_1, \ldots, P_K = \mathrm{Softmax}(z_1, z_2, \ldots\ldots, z_K)$ // get the corresponding probabilities P for each category
12: $\mathrm{L} = \mathrm{Argmax}(P_1, \ldots, P_K)$ // select the category with largest output value as the input category L
13: Save L in R
14: **end for**
10: **end for**
11: *Result* = R
12: **return** *Result*

divided into two stages: In the forward process, DDPM gradually adds noise to the RGB images of original network traffic to generate two-dimensional Gaussian noise, and then introduces a cosine noise addition strategy in the model to control the size of subsequent noise addition, thereby avoiding distortion of the final generated image; In the backward process, DDPM randomly samples two-dimensional Gaussian noise and gradually denoises it to obtain a new image, and then, it calculates the posterior distribution probability from the standard Gaussian distribution step by step based on Markov chain and Bayesian formula to guide the gradual removal of noise from the noisy image, thereby obtaining a new RGB image of network traffic. In addition, the model also introduces a learnable variance strategy to generate more realistic network traffic.

(3) **Network intrusion detection phase:** At this phase, a ResNet model with a 34-layer network structure is used for network intrusion detection, where the ELU activation function is used instead of traditional ReLU activation function to solve the problems of gradient vanishing and "neuron death" in the model; In addition, the model also introduces a dual attention mechanism (including channel attention mechanism and spatial attention mechanism) in each residual block, where the channel attention mechanism is used to obtain different importance levels of each channel in the RGB image of network traffic to make ResNet pays more attention to the features of certain channels, the spatial attention mechanism is used to focus more on the effective feature information in the RGB image of network traffic, ultimately improving the detection effect of network intrusion.

4. Experimental setup

To validate the effectiveness of the proposed DDP-DAR method, we conduct extensive experiments on four network traffic datasets.

4.1. Description of datasets

This subsection describes four network traffic datasets used during the experiments, including CTU, USTC-TFC, ISAC and UJS-IDS2024. Table 2 shows the distribution of different categories of network traffic in these four datasets.

CTU dataset: CTU is intercepted from a dataset of botnet traffic captured at Czech Technical University (CTU) in 2011. This dataset consists of 9 categories of intrusion traffic (including Ursnif, Coinminer, Htbot, Trickbot, Tinba, Zeus, Artemis, Dridex and Miuref) and 1 category of normal traffic.

USTC-TFC dataset: USTC-TFC contains both normal and malicious traffic collected by the device from 2011 to 2015. Among them, malicious traffic is collected by CTU researchers from real network environments, while normal traffic is collected using the IXIA BPS network traffic emulation device. In this dataset, we select 9 categories of malicious traffic (including Cridex, Geodo, Htbot, Miuref, Neris, Shifu, Tinba, Zeus and Virut) and 1 category of normal traffic.

ISAC dataset: ISAC is consisted of malware and normal data captured over a long period of time by Malware Capture Facility Project. In this dataset, we select 9 categories of malicious traffic (including DownloadGuide, Dyreza, Enotet, Ghost_RAT, Ncurse, OpenCandy, Ramnit, Sennoma and Shifu) and 1 category of normal traffic.

UJS-IDS2024 dataset: UJS-IDS2024 is produced by multiple normal software and malware in the real environment collected by the Key Laboratory of Industrial Cyberspace Security Technology of Jiangsu University from August 15, 2024 and August 20, 2024. In this dataset, we select 9 categories of malicious traffic (including ArkeiStealer, AsyncRAT, Thunder, Loki, MassLogger, Njrat, OskiStealer, RedLineStealer and SmokeLoader) and 1 category of normal traffic.

The purpose of the experiment is to test whether DDP-DAR method can accurately detect intrusion traffic, thus, we treat all normal network traffic as one class. In the experiments, we divide these network traffic datasets into ten parts, where eight parts are used as training datasets, one part is used as validation dataset and the remaining one part is used as testing dataset.

4.2. Baselines

In order to test the effectiveness of the proposed DDP-DAR method, five data augmentation-based NID models including CVAE-IDS-ResNet, BAGAN-GP-ResNet, DiffTPT-ResNet, SU-DDPM-ResNet, GDDIM-ResNet, CGSA-RNN (Cai et al., 2023) and BiTCN (Chen et al., 2023b) are selected as baseline methods. Because these data augmentation models of CVAE-IDS (Liu et al., 2022a), BAGAN-GP (Huang & Jafari, 2020), DiffTPT (Feng et al., 2023), SU-DDPM (Lu et al., 2024) and GDDIM (Zhang, Tao, & Chen, 2022) only provide data augmentation function, thus, we combine them with ResNet model (having better detection efficiency) to make them obtain better detection performance. These methods are described in details as follows:

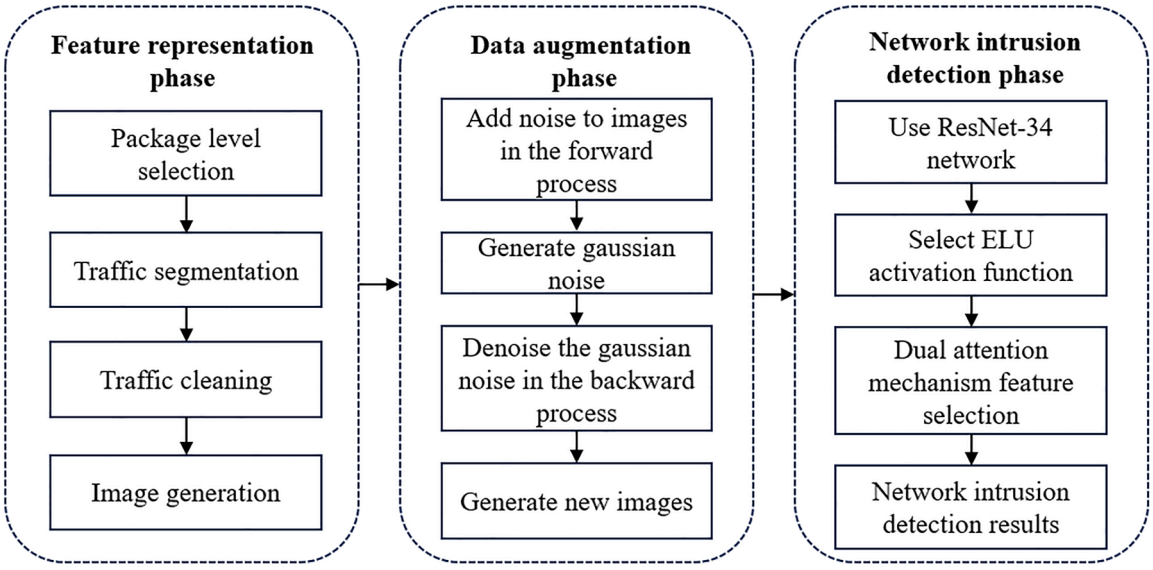


Fig. 5. The process of DDP-DAR model.

Table 2
The distribution of different categories of network traffic.

CTU dataset				USTC-TFC dataset				ISAC dataset				UJS-IDS2024 dataset			
Categories	Training Set	Validation Set	Testing Set	Categories	Training Set	Validation Set	Testing Set	Categories	Training Set	Validation Set	Testing Set	Categories	Training Set	Validation Set	Testing Set
Ursnif	20,604	2576	2576	Cridex	13,108	1639	1639	DownloadGuide	225	28	28	ArkeiStealer	15,575	1947	1947
Coinminer	253	32	32	Geodo	32,757	4095	4095	Dyreza	14	2	2	AsyncRAT	1811	227	227
Htbot	7504	938	938	Htbot	5093	637	637	Enotet	48,000	6000	6000	Thunder	5052	632	632
Trickbot	9605	1201	1201	Miuref	10,785	1348	1348	Ghost_RAT	24	3	3	Loki	2357	295	295
Tinba	41,072	5134	5134	Neris	27,033	3379	3379	Ncurse	6425	803	803	MassLogger	5893	737	737
Zeus	2005	251	251	Shifu	7708	963	963	OpenCandy	8155	1024	1024	Njrat	17,876	2235	2235
Artemis	6956	869	869	Tinba	6804	850	850	Ramnit	661	82	82	OskiStealer	7711	964	964
Dridex	2507	313	313	Zeus	8776	1097	1097	Sennoma	48,000	6000	6000	RedLineStealer	8948	1119	1119
Miuref	10,785	1348	1348	Virut	26,483	3310	3310	Shifu	20,701	2588	2588	SmokeLoader	1588	199	199

(1) **CAVE-IDS-ResNet:** CAVE-IDS is an improved VAE-based algorithm, it utilizes the labeling information of samples to generate the samples of specified categories.
(2) **BAGAN-GP-ResNet:** BAGAN-GP utilizes a supervised autoencoder with an intermediate embedding model to disperse the labeled latent vectors; Meanwhile, it establishes a BAGAN architecture with gradient penalties through enhanced autoencoder initialization, thereby overcoming the instability problem in the original BAGAN as well as converging faster.
(3) **DiffTPT-ResNet:** DiffTPT combines the method that relies on data augmentation and confidence selection with a pre-trained stabilizing diffusion model, it improves the model's ability to adapt unknown data through exploiting their respective advantages to increase the data size.
(4) **SU-DDPM-ResNet:** SU-DDPM improves the quality of augmented images by combining the degraded image with a reference image in the diffusion stage to create a fusion DDPM model.
(5) **GDDIM-ResNet:** GDDIM uses some specific fractional approximations to construct the non-Markovian noise process when solving the corresponding stochastic differential equations, and makes a small but subtle modification in parameterizing the fractional network, thereby producing an implicit model that generates high-quality samples faster.
(6) **CGSA-RNN:** CGSA-RNN utilizes the migration advantage of CycleGAN for data augmentation on the small-scale network traffic, it also introduces a self-attention mechanism to help it better capture the important features of network traffic to further improve the capability of data augmentation, where the RNN model is selected for the network intrusion detector.
(7) **BiTCN:** BiTCN uses a temporal convolutional network (TCN) model to better capture the sequential features of network traffic, and then uses an exponential linear unit (ELU) activation function instead of ReLU in the model training phase to avoid the problem of "neuron death"; In addition, it also improves the original unidirectional model to a bidirectional model to capture the bidirectional semantic fusion features of network traffic.

4.3. Evaluation metrics

To test the efficiency of the proposed DDP-DAR method, four evaluation metrics including Accuracy, F1-measure, FPR and ROC-AUC are used in the experiments. These four metrics are obtained through confusion matrix as shown in Table 3.

In Table 3, *TP* represents the intrusion traffic is classified as intrusion traffic; *FN* represents the intrusion traffic is classified as normal network traffic; *FP* represents the normal network traffic is classified as intrusion traffic; *TN* represents the normal network traffic is classified as normal network traffic.

(1) Accuracy: It indicates the ratio of the number of accurately detected network traffic samples to the total number of network traffic samples, which is calculated as shown in Eq. (14).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{14}$$

(2) F1-measure: It is the reconciled average of precision and recall, which is calculated as shown in Eq. (15).

$$F1 - measure = \frac{2 * \frac{TP}{TP+FP} * \frac{TP}{TP+FN}}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}} \tag{15}$$

(3) FPR: It indicates the number of mis-detected network intrusion traffic to the total number of network traffic samples, which is calculated as shown in Eq. (16).

$$FPR = \frac{FP}{FP + TN} \tag{16}$$

(4) ROC-AUC: It indicates the area of AUC under ROC curve, which measures the overall performance of detection method. The AUC value ranges from 0 to 1 and the larger value indicates the better model performance. The vertical coordinate of ROC curve is true positive rate (TPR), and the horizontal coordinate is false positive rate (FPR). The ROC-AUC is calculated as shown in Eq. (17).

$$ROC - AUC = \int_0^1 \frac{TP}{TP + FN} d\left(\frac{FP}{FP + TN}\right) \tag{17}$$

4.4. Experimental environment and parameter settings

The experiments are run on an Intel(R) Xeon(R) Platinum 8168 CPU and 4*NVIDIA A40, the deep learning framework is Pytorch (python version 3.8). The main parameters of NT-DDPM model and NID-DARN model in the DDP-DAR method are listed in Table 4.

5. Experimental results and analysis

To better evaluate the detection efficiency of DDP-DAR method, we set the following four research questions (RQs):

RQ1: Whether the feature representation in the format of improved RGB images can better store the features of network traffic and thus improving the efficiency of NID?
RQ2: Can the proposed NT-DDPM model obtain better data augmentation effect than other data augmentation models?
RQ3: Does the DDP-DAR method achieve better detection results than other data augmentation-based NID methods?
RQ4: Whether the DDP-DAR method can achieve more stable detection results than other data augmentation-based NID methods?

To eliminate the chance of experiment, each experiment is conducted 30 times and the average result is calculated as the final experimental result, where the value before "±" symbol indicates the average of 30 repetitions, and the value after "±" symbol indicates the standard deviation of 30 repetitions.

5.1. Answer to RQ1

To answer RQ1, the network traffic in four formats of PCAP, Gray-scale images, RGB images and improved RGB images are compared in the experiments, where six widely used deep learning-based NID models (including CNN (Landman & Nissim, 2021), LSTM (Wei et al., 2024), VGG (Manjula & Baghavathi Priya, 2022), ViT (Dosovitskiy et al., 2020), TCN (de Araujo-Filho et al., 2023) and ResNet (Xia et al., 2022)) are adopted to comprehensively test the efficiency. The experimental results are shown in Table 5, where the best detection efficiency is indicated in bold and the best detection result achieved by each model under four feature representations is marked in blue.

It can be seen from Table 5 that on the four network traffic datasets, compared with three network traffic feature representation formats (including PCAP, Gray-scale image and RGB image), all six network intrusion detection models obtain the highest Accuracy, F1-measure and ROC-AUC value as well as lowest FPR value, and their standard deviation is also lowest on the improved RGB image representation. This is due to the fact that compared with three traditional formats of network traffic feature representation, the improved RGB feature representation stores all layer feature information, session layer feature information and application layer feature information of network traffic in the R, G and B channels of RGB image, respectively, this representation allows for storing a large amount of feature information in the network traffic, which enables the intrusion detection model to focus on the global features of network traffic as well as its local features, thereby better learning the feature relationships among network traffic. Meanwhile, it also can be seen from Table 5 that compared with five network intrusion detection models (including CNN, LSTM, VGG, ViT and TCN), ResNet model shows better detection performance in four evaluation metrics in most cases. This is due to the fact that the ResNet model introduces the residual structure to build a deep network, which can extract higher-quality features from network traffic and thus improving the detection performance. However, the ResNet model is slightly inferior to the TCN in some evaluation metrics on the CTU dataset, which is owing to that TCN model extracts the information in the time-dimension through the convolution operation and captures the temporal relationships and long-term dependencies in the network traffic by utilizing the inflated convolution operation, thus, it shows a better detection performance. As shown in Table 5, the high data imbalance rate on the ISAC dataset results in these six network intrusion detection models not being able to learn the features of few classes of samples well, which leads to the generally low average F1-measure value.

Table 3
Confusion matrix.

Confusion matrix		Predicted results	
		Positive	Negative
Real results	Positive	TP	FN
	Negative	FP	TN

5.2. Answer to RQ2

To answer RQ2, we compare the proposed NT-DDPM model with six data augmentation models (including CVAE-IDS (Liu et al., 2022a), BAGAN-GP (Huang & Jafari, 2020), DiffTPT (Feng et al., 2023), SU-DDPM (Lu et al., 2024), GDDIM (Zhang, Tao, & Chen, 2022), all these data augmentation models are used to generate the RGB images) as well as CGSA (Cai et al., 2023) (it is used to generate Gary-scale images) on four network traffic datasets. In order to better show the effect achieved by each data augmentation model, we use the ResNet model with the best detection results (shown in Table 5) as the NID model (except for DDP-DAR, CGSA-RNN and BiTCN), where the experimental result in the columns of "BiTCN" and "DDP-DAR" are not considered in this subsection). The experimental result is shown in Table 6, where the best detection efficiency is indicated in green.

Table 6 shows that compared with six data augmentation-based network intrusion detection models including CVAE-IDS-ResNet, BAGAN-GP-ResNet, SU-DDPM-ResNet, DiffTPT-ResNet, GDDIM-ResNet and CGSA-RNN, NT-DDPM-ResNet performs best on these three publicly available network traffic datasets and one real-world network traffic. On the USTC-TFC dataset, the detection accuracy of NT-DDPM-ResNet is 99.48 %, F1-measure is 99.32 %, and ROC-AUC is 99.98 %, all of which are the highest; In addition, the FPR is 0.06 %, which is the lowest among six compared models. On the CTU dataset, NT-DDPM-ResNet outperforms GDDIM-ResNet model by 0.21 % in accuracy, 0.22 % in F1-measure, and 0.03 % in FPR; Meanwhile, compared to CVAE-IDS-ResNet model with worst performance, the accuracy is 1.08 % higher, the F1-measure is 2.68 % higher, the ROC-AUC is 0.02 % higher, and the FPR is 0.15 % lower. On the ISAC dataset, the accuracy of NT-DDPM-ResNet is 99.65 %, which is 0.58 % higher than CVAE-IDS-ResNet, 0.49 % higher than BAGAN-GP-ResNet, 0.45 % higher than SU-DDPM-ResNet, 0.33 % higher than DiffTPT-ResNet, 0.31 % higher than GDDIM-ResNet, and 0.31 % higher than CGSA-RNN; Meanwhile, the F1 measure of NT-DDPM ResNet is as high as 99.91 %, while the F1-measure of other six models is slightly worse. On the UJS-IDS2024

Table 4
The parameters of NT-DDPM model and NID-DARN model.

Models	Parameter	Parameter value	Selection principle
NT-DDPM	diffusion step	50	A short diffusion step can quickly generate a large number of samples, but it may lead to a decrease in the quality of generated network traffic samples; a long diffusion step can improve the quality of generated samples, but it requires the model to spend more time to gradually remove the noise and restore the original data. In our model, the diffusion step is chosen as 50, which can ensure the quality of generated network traffic samples while avoiding unnecessary computational overhead.
	noise schedule	cosine	The choice of noise schedule directly affects the training effectiveness of the model and the quality of generated samples. Cosine schedule can affect the training stability of model and the diversity of generated samples by controlling the rate and pattern of noise addition. In our model, the cosine is chosen as noise schedule, which can ensure that the changes in noise during diffusion process are smooth, thereby better helping the model control the diffusion process.
	dropout	0.3	A small dropout value means fewer neurons are discarded, which helps the model fully utilize the information in the training data with less network traffic, but it can lead to overfitting of the model on the training data, especially when the training data is limited or noisy. A large dropout value can cause more neurons to be randomly discarded during training, which helps the model learn more robust feature representations and improve generalization ability, but it can also lead to underfitting of the model. In our model, the dropout value is chosen as 0.3, which can prevent overfitting without causing significant impact on model performance, that is, achieving a good balance between computational resources and model performance.
	learning rate	0.0001	A low learning rate can ensure the stability of the model, but it will significantly increase the training time. A high learning rate can accelerate the training speed, but it may cause the model to oscillate near the optimal solution and even fail to converge. In our model, the learning rate is chosen as 0.0001, which can increase training stability, prevent overfitting, adapt to complex network traffic, and balance training speed and performance.
	batch_size	64	A small batch size may cause the model to fall into local optima, while a large batch size may lead to overfitting of the model on the training data. In our model, the batch size is chosen as 64, which is usually suitable for various deep learning models as it can more effectively utilize parallel computing resources and improve data processing speed.
NID-DARN	batch_size	128	A small batch size can provide more frequent weight updates, which helps the detection model converge faster, but it may lead to a more unstable training process. A large batch size can more fully utilize hardware parallelism, thereby accelerating the training speed of a single epoch, but it means consuming more memory. In our model, the batch size is chosen as 128, which can achieve a good balance between training speed and memory consumption, as well as avoid the problems such as slow training speed, memory fragmentation, insufficient video memory and excessive gradient updates.
	epoch	20	A small epoch can reduce the training cycle of detection model, but it may not fully learn the distribution and features of network traffic, making it prone to underfitting. A large epoch improves the performance of detection model, but it consumes more training time. In our model, the epoch is chosen as 20, which enables NID-DARN to achieve optimal training results while avoiding unnecessary time and computational resource consumption.
	loss function	Corss-entropy loss function	Due to the presence of nine types of malicious traffic and one type of normal traffic in network intrusion detection tasks, the loss function should be suitable for handling multi class tasks; In addition, the loss function should also have numerical stability to avoid the problems such as underflow or overflow during the calculation process. The selected cross-entropy loss function has the above advantages and can handle imbalanced network traffic to some extent, which can reduce the impact of data imbalance on model training.
	learning rate	0.0001	Although the low learning rate is stable, but the convergence speed of detection model is slow, which may require more training time and iteration times. A high learning rate enables the detection model to converge quickly, but it may also lead to over adjustment or even divergence. In our model, the learning rate is chosen as 0.0001, which can avoid the problem of gradient vanishing while maintaining the stability of training.
	activation function	ELU	The ELU activation function effectively solves the "neuron death" problem of traditional ReLU activation function through its negative value smoothing feature, it also provides faster convergence speed and better training performance, therefore, we choose ELU as activation function in our model.

dataset, the accuracy of NT-DDPM-ResNet is 97.57 %, which is 1.33 % higher than CVAE-IDS-ResNet, 0.69 % higher than BAGAN-GP-ResNet, 0.06 % higher than SU-DDPM-ResNet, 0.15 % higher than DiffTPT-ResNet, 0.1 % higher than GDDIM-ResNet, and 2.36 % higher than CGSA-RNN.

It can be known from Table 6 that these four evaluation metrics (including Accuracy, F1 measure, ROC-AUC, and FPR) of the proposed NT-DDPM-ResNet model are the best among six compared models, this is mainly due to the proposed traffic denoising diffusion probability model introduces cosine to increase noise and a learnable variance strategy to generate high-quality RGB images of network traffic, which solves the imbalance problem of network traffic as well as improves the detection efficiency of network intrusion.

5.3. Answer to RQ3

To answer RQ3, we use the confusion matrix to exhibit the metric of Accuracy of eight data augmentation-based NID methods of CVAE-IDS-ResNet, BAGAN-GP-ResNet, DiffTPT-ResNet, SU-DDPM-ResNet, GDDIM-ResNet, CGSA-RNN, BiTCN and DDP-DAR, and the experimental results are shown in Fig. 5–Fig. 9 and Table 6, where the best detection efficiency are indicated in bold and blue in the table (except for the NT-DDPM, the experimental result in this column is not considered in the subsection).

As shown in Table 6 that on the USTC-TFC dataset, the detection accuracy of DDP-DAR is 99.85 %, F1-measure is 99.40 %, ROC-AUC is 99.99 % and FPR is 0.04 %, which is better than seven advanced network intrusion detection models. On the CTU dataset, DDP-DAR has an accuracy of 1.26 % higher than CVAE-IDS-ResNet, 1.15 % higher than BAGAN-GP-ResNet, 0.89 % higher than SU-DDPM-ResNet, 0.51 % higher than DiffTPT-ResNet, 0.39 % higher than GDDIM-ResNet, 2.17 % higher than CGSA-RNN, and 0.18 % higher than BiTCN. On FPR metric, DDP-DAR is 0.16 % lower than CVAE-IDS-ResNet, 0.14 % lower than BAGAN-GP-ResNet, 0.13 % lower than SU-DDPM-ResNet, 0.05 % lower than DiffTPT-ResNet, 0.04 % lower than GDDIM-ResNet, 0.24 % lower than CGSA-RNN, and 0.01 % lower than BiTCN. On the ISAC dataset, DDP-DAR outperforms CVAE-IDS-ResNet by 0.86 % in accuracy, 0.93 % in F1-measure, 0.01 % in ROC-AUC, and 0.05 % in FPR compared to DiffTPT-ResNet. On the UJS-IDS2024 dataset, DDP-DAR performs better than seven compared NID models, with a detection accuracy of 97.89 %, while BiTCN performs the worst and with a detection accuracy of only 93.54 %. In contrast, due to DiffTPT-ResNet, SU-DDPM-ResNet and GDDIM-ResNet use diffusion models for data augmentation, they have slightly lower detection accuracy than DDP-DAR with the detection accuracy of 97.42 %, 97.51 % and 97.47 %, respectively.

Figs. 5–9 show the confusion matrices of experimental results of eight models on three publicly available network traffic datasets and one real-world network traffic, where the horizontal axis represents the predicted classes of models and the vertical axis represents the true

Table 5
Detection efficiency of different NID models on four formats of network traffic (%).

Datasets	Metrics	Format	CNN	LSTM	VGG	ViT	TCN	ResNet
CTU	Accuracy	PCAP	86.86±0.71	80.79±0.70	85.60±0.65	90.85±0.07	91.54±4.54	92.14±1.23
		Gray-scale images	88.16±0.82	87.76±1.00	91.08±1.15	93.23±0.15	92.64±2.69	93.77±1.73
		RGB images	91.15±0.72	93.16±1.51	95.51±0.67	93.29±0.16	94.07±0.79	96.97±0.41
		Improved RGB images	**93.01±0.66**	**95.30±0.67**	**97.67±0.62**	**95.47±0.07**	**97.82±0.67**	**97.96±0.39**
	F1-measure	PCAP	55.72±0.96	59.64±2.32	65.48±7.48	69.77±0.69	73.17±9.11	74.44±1.61
		Gray-scale images	62.89±0.33	65.74±2.22	73.94±2.73	82.28±0.27	84.78±1.87	81.59±1.27
		RGB images	70.68±3.08	76.95±4.15	85.33±6.15	82.29±0.26	85.48±2.04	90.80±3.62
		Improved RGB images	**78.75±0.30**	**79.85±1.89**	**94.39±1.11**	**84.26±0.32**	**94.46±1.33**	**95.40±0.29**
	FPR	PCAP	1.75±0.42	2.54±0.48	1.95±0.26	1.25±0.22	1.12±0.59	1.04±0.21
		Gray-scale images	1.53±0.24	1.58±0.15	1.11±0.10	0.88±0.23	0.91±0.31	0.72±0.18
		RGB images	1.11±0.19	0.80±0.18	0.52±0.16	0.77±0.12	0.70±0.05	0.35±0.05
		Improved RGB images	**1.01±0.15**	**0.62±0.09**	**0.28±0.09**	**0.53±0.14**	**0.31±0.05**	**0.27±0.02**
	ROC-AUC	PCAP	93.17±0.18	93.91±0.48	93.57±0.17	94.21±1.03	96.48±2.71	94.99±2.04
		Gray-scale images	93.64±0.31	96.82±0.40	97.65±0.83	95.17±1.78	98.87±0.21	98.95±0.08
		RGB images	98.06±0.20	98.96±0.34	99.58±0.12	95.18±1.75	99.07±0.13	99.79±0.12
		Improved RGB images	**98.58±0.16**	**99.05±0.11**	**99.82±0.10**	**95.29±1.00**	**99.61±0.09**	**99.93±0.07**
USTC-TFC	Accuracy	PCAP	84.41±0.65	84.39±0.63	89.18±0.45	82.86±0.30	94.65±0.43	95.10±0.14
		Gray-scale images	85.57±2.35	87.84±2.03	94.13±0.25	85.99±0.34	94.89±0.04	95.62±0.17
		RGB images	87.56±0.32	88.93±0.75	94.19±0.09	86.12±0.07	95.91±0.35	96.11±0.12
		Improved RGB images	**89.61±0.23**	**89.63±0.18**	**97.05±0.05**	**90.12±0.06**	**98.01±0.04**	**98.74±0.04**
	F1-measure	PCAP	73.83±1.42	73.04±1.57	83.39±0.54	69.72±0.47	90.73±0.64	92.35±0.26
		Gray-scale images	80.06±4.74	82.28±4.55	94.40±0.27	79.23±0.50	95.81±0.21	96.03±0.12
		RGB images	83.51±0.90	82.57±2.97	94.72±0.16	81.17±4.28	96.91±0.04	97.02±0.06
		Improved RGB images	**84.91±0.80**	**84.99±0.42**	**97.02±0.16**	**82.85±0.22**	**97.03±0.04**	**97.18±0.03**
	FPR	PCAP	2.08±0.06	2.06±0.13	1.50±0.03	2.07±0.05	0.61±0.06	0.56±0.02
		Gray-scale images	1.68±0.29	1.40±0.26	0.68±0.02	1.56±0.04	0.59±0.01	0.49±0.01
		RGB images	1.44±0.03	1.27±0.07	0.67±0.02	1.56±0.03	0.46±0.04	0.44±0.02
		Improved RGB images	**1.20±0.03**	**1.19±0.02**	**0.35±0.02**	**1.06±0.01**	**0.22±0.03**	**0.14±0.01**
	ROC-AUC	PCAP	97.12±0.03	97.04±0.34	98.68±0.16	90.95±0.31	99.49±0.12	99.66±0.01
		Gray-scale images	98.64±0.24	98.48±0.70	99.73±0.02	95.88±0.56	99.52±0.02	99.79±0.01
		RGB images	99.06±0.05	98.69±0.25	99.74±0.01	97.63±0.74	98.54±2.25	99.86±0.01
		Improved RGB images	**99.12±0.03**	**98.70±0.14**	**99.91±0.01**	**97.69±0.23**	**99.86±0.02**	**99.91±0.01**
ISAC	Accuracy	PCAP	96.04±1.14	95.24±0.39	95.95±0.32	95.96±0.17	95.17±0.12	96.60±0.58
		Gray-scale images	96.11±0.68	96.07±0.74	96.39±2.24	97.63±0.12	96.69±0.22	97.79±0.12
		RGB images	96.95±0.82	96.25±0.70	97.43±0.25	98.06±0.13	97.32±0.41	98.36±0.17
		Improved RGB images	**98.08±0.44**	**98.56±0.20**	**97.52±0.15**	**98.37±0.06**	**97.33±0.12**	**98.67±0.07**
	F1-measure	PCAP	50.72±6.74	51.42±1.57	50.51±5.29	57.69±0.39	55.37±1.48	61.37±0.38
		Gray-scale images	57.91±3.55	53.38±2.12	58.86±6.72	60.53±0.37	59.14±0.96	61.87±1.16
		RGB images	58.65±0.58	56.16±4.96	62.65±0.38	60.52±0.47	59.68±5.00	63.72±0.74
		Improved RGB images	**59.01±0.51**	**57.44±0.40**	**63.39±0.30**	**62.61±0.37**	**59.73±0.71**	**63.79±0.74**
	FPR	PCAP	0.50±0.18	0.65±0.04	0.48±0.04	0.52±0.02	0.63±0.11	0.39±0.19
		Gray-scale images	0.47±0.06	0.50±0.09	0.47±0.29	0.28±0.02	0.42±0.23	0.27±0.01
		RGB images	0.38±0.03	0.37±0.03	0.30±0.02	0.25±0.01	0.32±0.24	0.18±0.01
		Improved RGB images	**0.34±0.03**	**0.19±0.03**	**0.22±0.01**	**0.24±0.01**	**0.20±0.06**	**0.17±0.01**
	ROC-AUC	PCAP	92.72±1.98	94.52±2.36	95.35±0.31	92.52±0.64	92.89±5.62	94.76±0.24
		Gray-scale images	93.54±5.88	96.33±0.87	95.55±0.83	94.45±1.39	93.13±0.52	97.04±0.20
		RGB images	97.07±1.41	96.42±0.56	95.52±0.46	95.21±0.59	94.50±3.08	99.05±0.47
		Improved RGB images	**97.71±1.10**	**96.74±0.54**	**98.26±0.12**	**97.66±0.34**	**97.94±0.44**	**99.13±0.12**
UJS-IDS2024	Accuracy	PCAP	71.84±0.48	66.20±0.53	75.57±1.08	63.36±0.42	71.07±0.16	81.80±0.02
		Gray-scale images	76.26±0.93	70.86±0.32	79.90±0.12	73.49±0.38	76.16±0.93	87.26±0.10
		RGB images	85.97±0.04	83.31±0.47	87.54±1.13	84.26±0.47	85.81±0.02	88.02±0.01
		Improved RGB images	**94.20±0.98**	**93.04±1.99**	**95.93±0.20**	**93.64±1.99**	**94.16±0.98**	**96.02±0.13**
	F1-measure	PCAP	54.69±1.88	58.65±0.47	69.23±1.42	55.45±0.47	54.24±0.18	79.43±0.13
		Gray-scale images	61.86±0.91	58.97±0.56	75.70±0.10	67.14±0.54	61.64±0.91	84.70±0.03
		RGB images	82.50±0.36	78.94±0.52	84.11±2.27	80.49±0.52	74.72±0.02	85.09±0.09
		Improved RGB images	**89.47±2.37**	**86.18±3.65**	**92.31±0.68**	**85.26±3.65**	**89.40±2.37**	**92.82±0.32**
	FPR	PCAP	3.72±0.07	4.48±0.05	3.18±0.17	4.70±0.07	3.43±0.01	2.33±0.04
		Gray-scale images	3.08±0.14	3.83±0.06	2.56±0.02	3.06±0.06	3.00±0.14	1.59±0.03
		RGB images	1.79±0.01	2.1 ± 0.07	1.54±0.13	1.9 ± 0.07	1.72±0.01	1.51±0.04
		Improved RGB images	**0.66±0.11**	**0.83±0.23**	**0.49±0.02**	**0.90±0.23**	**0.67±0.11**	**0.45±0.02**
	ROC-AUC	PCAP	92.06±0.39	91.89±0.36	95.83±0.50	88.74±0.59	95.24±0.02	97.89±0.04
		Gray-scale images	95.61±0.16	93.70±0.19	97.57±0.05	93.98±0.35	95.49±0.16	98.87±0.04
		RGB images	98.61±0.09	97.76±0.08	99.06±0.23	96.57±0.08	98.32±0.01	99.10±0.40
		Improved RGB images	**99.61±0.20**	**99.30±0.33**	**99.70±0.01**	**99.10±0.33**	**99.58±0.20**	**99.72±0.01**

Answer to RQ1:

Extensive experimental results show that different intrusion detection models obtain higher detection efficiency and better stability when processing network traffic in the format of improved RGB images, which indicates that the proposed feature representation method can better store the features of network traffic and thus further promoting the efficiency and stability of NID.

Table 6
Detection efficiency with different data augmentation models and different data augmentation-based NID methods (%).

Dataset	Metric	CVAE-IDS-ResNet	BAGAN-GP-ResNet	DiffTPT-ResNet	SU-DDPM-ResNet	GDDIM-ResNet	CGSA-RNN	BiTCN	NT-DDPM-ResNet	DDP-DAR
USTC-TFC	Accuracy	98.79±0.04	98.98±0.07	99.32±0.02	99.05±0.21	99.22±0.05	99.07 ±0.16	98.76 ±0.41	99.48±0.04	99.58 ±0.02
	F1-measure	98.78±0.04	99.17±0.04	99.30±0.03	99.29±0.16	99.31±0.04	99.33 ±0.11	99.11 ±0.28	99.32±0.05	99.40 ±0.03
	FPR	0.13±0.01	0.11±0.01	0.07±0.01	0.11±0.02	0.09±0.01	0.11 ±0.02	0.15 ±0.05	0.06±0.02	0.04 ±0.01
	ROC-AUC	99.98±0.01	99.97±0.04	99.98±0.03	99.98±0.03	99.98±0.02	99.99 ±0.01	99.98 ±0.01	99.98±0.02	99.99 ±0.01
CTU	Accuracy	98.56±0.17	98.67±0.07	99.31±0.11	98.93±0.38	99.43±0.15	97.65 ±0.57	99.64 ±0.09	99.64±0.03	99.82 ±0.02
	F1-measure	96.95±0.09	96.95±0.26	99.31±0.11	97.41±0.79	99.41±0.15	86.73 ±1.01	91.68 ±2.38	99.63±0.02	99.82 ±0.02
	FPR	0.18±0.03	0.16±0.01	0.07±0.01	0.15±0.03	0.06±0.02	0.26 ±0.06	0.03 ±0.01	0.03±0.01	0.02 ±0.01
	ROC-AUC	99.96±0.02	99.97±0.01	99.97±0.04	99.77±0.13	99.98±0.03	99.76 ±0.24	99.99 ±0.01	99.98±0.04	99.99 ±0.01
ISAC	Accuracy	99.07±0.12	99.16±0.07	99.32±0.02	99.20±0.26	99.34±0.05	99.34 ±0.36	99.89 ±0.02	99.65±0.05	99.93 ±0.01
	F1-measure	98.97±0.33	99.01±0.71	99.32±0.03	99.12±0.42	99.58±0.34	89.24 ±0.97	97.30 ±1.22	99.91±0.08	99.94 ±0.01
	FPR	0.10±0.01	0.10±0.01	0.07±0.01	0.05±0.01	0.08±0.01	0.07 ±0.04	0.02 ±0.01	0.03±0.01	0.02 ±0.01
	ROC-AUC	99.95±0.06	99.95±0.04	99.98±0.03	98.98±0.03	99.76±0.01	99.82 ±0.17	99.99 ±0.01	99.99±0.03	99.99 ±0.01
UJS-IDS2024	Accuracy	96.24±0.25	96.88±1.52	97.42±0.10	97.51±0.11	97.47±0.12	95.21 ±0.26	93.54 ±1.29	97.57±0.07	97.89 ±0.07
	F1-measure	93.45±0.52	96.87±1.53	97.43±0.11	97.51±0.11	97.47±0.13	92.46 ±0.33	91.08 ±1.47	97.57±0.07	97.89 ±0.06
	FPR	0.44±0.03	0.35±0.17	0.29±0.02	0.28±0.02	0.28±0.01	0.58 ±0.04	0.81 ±0.17	0.27±0.01	0.23 ±0.01
	ROC-AUC	99.83±0.18	99.84±0.25	99.90±0.02	99.93±0.02	99.93±0.02	99.79 ±0.19	99.74 ±0.11	99.93±0.01	99.96 ±0.01

classes of network traffic. It can be seen from Fig. 6f, Fig. 7f, Fig. 8f and Fig. 9f that DDP-DAR has good performance in network intrusion detection with an accuracy of nearly 99 %, which is better than other seven compared models.

The experimental results show that compared with seven state-of-the-art data augmentation-based network intrusion detection models, due to the use of deep residual network with dual attention mechanism, DDP-DAR can effectively avoid the problem of network performance degradation while expanding network depth. At the same time, by utilizing channel attention mechanism and spatial attention mechanism, it is possible to efficiently model channel level and spatial level relationships to enhance the representation ability of deep residual networks, thereby obtaining higher quality and more important features of network traffic, as well as improving the detection efficiency of network intrusion.

5.4. Answer to RQ4

To answer RQ4, we compare the stability of the proposed DDP-DAR method with seven state-of-the-art data augmentation-based NID methods (including CVAE-IDS-ResNet, BAGAN-GP-ResNet, DiffTPT-ResNet, SU-DDPM-ResNet, GDDIM-ResNet, CGSA-RNN, BiTCN) using box plots, and the experimental results are shown in Fig. 10.

It can be seen from Fig. 10 that on these four network traffic datasets, compared with eight advanced data augmentation-based network intrusion detection models, the proposed DDP-DAR model has the smallest interquartile range (box length) among four indicators and does not have any outlier, which indicates that the DDP-DAR model has better detection stability. As is shown in Fig. 10a that DDP-DAR outperforms CVAE-ResNet, BAGAN-GP-ResNet, SU-DDPM-ResNet, DiffTPT-ResNet, GDDIM-ResNet, BiTCN, CGSA-RNN and NT-DDPM-ResNet models in terms of accuracy; Among them, SU-DDPM-ResNet model has poor stability and may encounter outliers. As is shown in Fig. 10b and Fig. 10d, DDP-DAR still maintains the highest position in F1-measure and ROC-AUC, indicating that DDP-DAR has better performance in network intrusion detection tasks; Among them, SU-DDPM-ResNet, GDDIM-ResNet, CGSA-RNN and BiTCN perform relatively weakly and show outliers in some cases. As is shown in Fig. 10c, DDP-DAR model is at the lowest position on the FPR, which indicates that DDP-DAR can reduce the probability of detecting normal network activity as malicious attack behavior.

Compared with eight data augmentation-based network intrusion detection models, DDP-DAR model has better stability for the following reasons: (1) It replaces the traditional ReLU activation function with ELU to solve the "neuron death" problem of traditional neural networks, thereby helping DDP-DAR better capture the features of network traffic; (2) The deep residual network utilizing dual attention mechanism not

Answer to RQ2

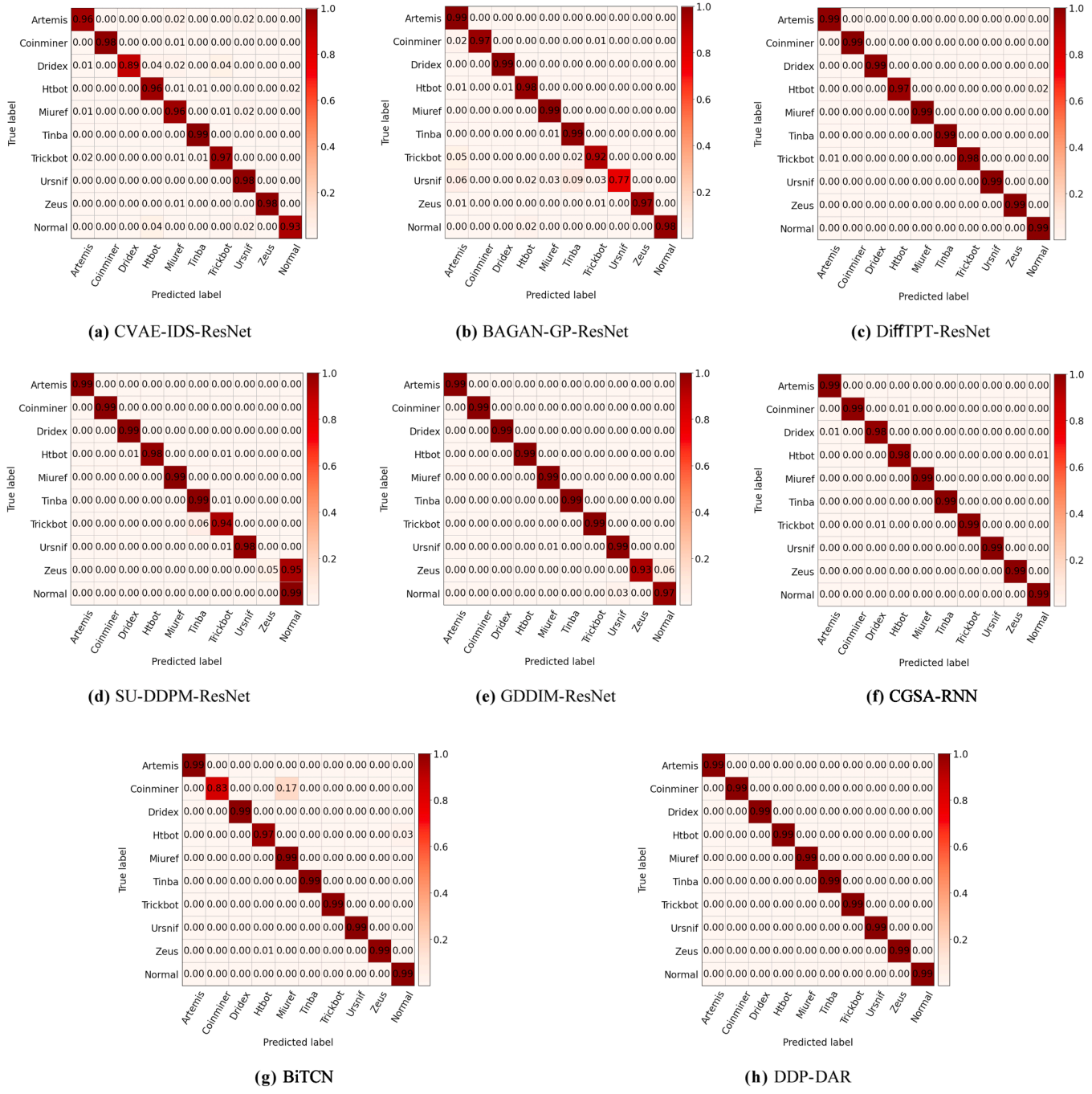
Extensive experimental results show that the augmented RGB images by our NT-DDPM model can make the ResNet to achieve higher detection efficiency as well as better stability. The experiments verify that NT-DDPM model can obtain better effect of data augmentation than the state-of-the-art data augmentation models.

only obtains high-quality features of network traffic through multi-layer convolution, but also extracts some features that are more helpful for network intrusion detection based on different important levels of network traffic features.

5.5. Discussion

In this paper, we construct a novel network intrusion detection model called DDP-DAR from three perspectives: network traffic RGB image representation, network traffic data augmentation and network intrusion detection. Firstly, the model extracts the features of original network traffic by selecting three different types of packet layers including all layers, application layers and session layers, and then represents the extracted features in the form of RGB images, thereby obtaining more information of network traffic. Meanwhile, the distribution of different categories of network traffic is balanced through using a denoising diffusion probability model to learn the features of network traffic. In addition, the joint use of dual attention mechanisms allows the deep residual network to focus on locally higher-level and more important features, while suppressing the influence of other useless information on network intrusion detection, thereby improving the detection accuracy.

Although the proposed DDP-DAR model can achieve good detection results, it still has the following problems, including the practical usability and scalability of the proposed model, deployment challenges, and limitations of the dataset used. The limitations of DDP-DAR model and future improvement options are summarized below: **(1) Practical**

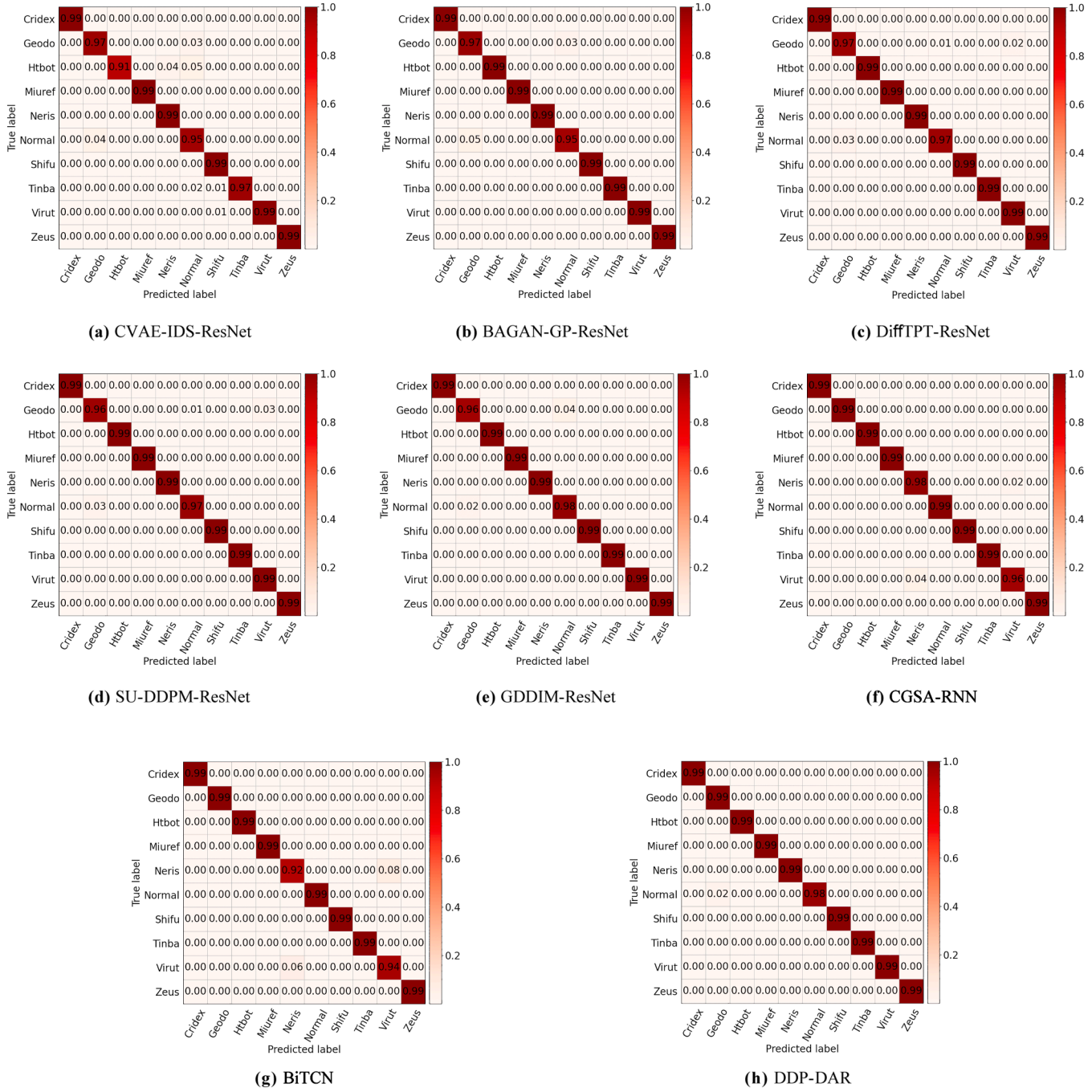


(a) CVAE-IDS-ResNet **(b)** BAGAN-GP-ResNet **(c)** DiffTPT-ResNet

(d) SU-DDPM-ResNet **(e)** GDDIM-ResNet **(f)** CGSA-RNN

(g) BiTCN **(h)** DDP-DAR

Fig. 6. The confusion matrix of eight data augmentation-based NID methods on CTU dataset.



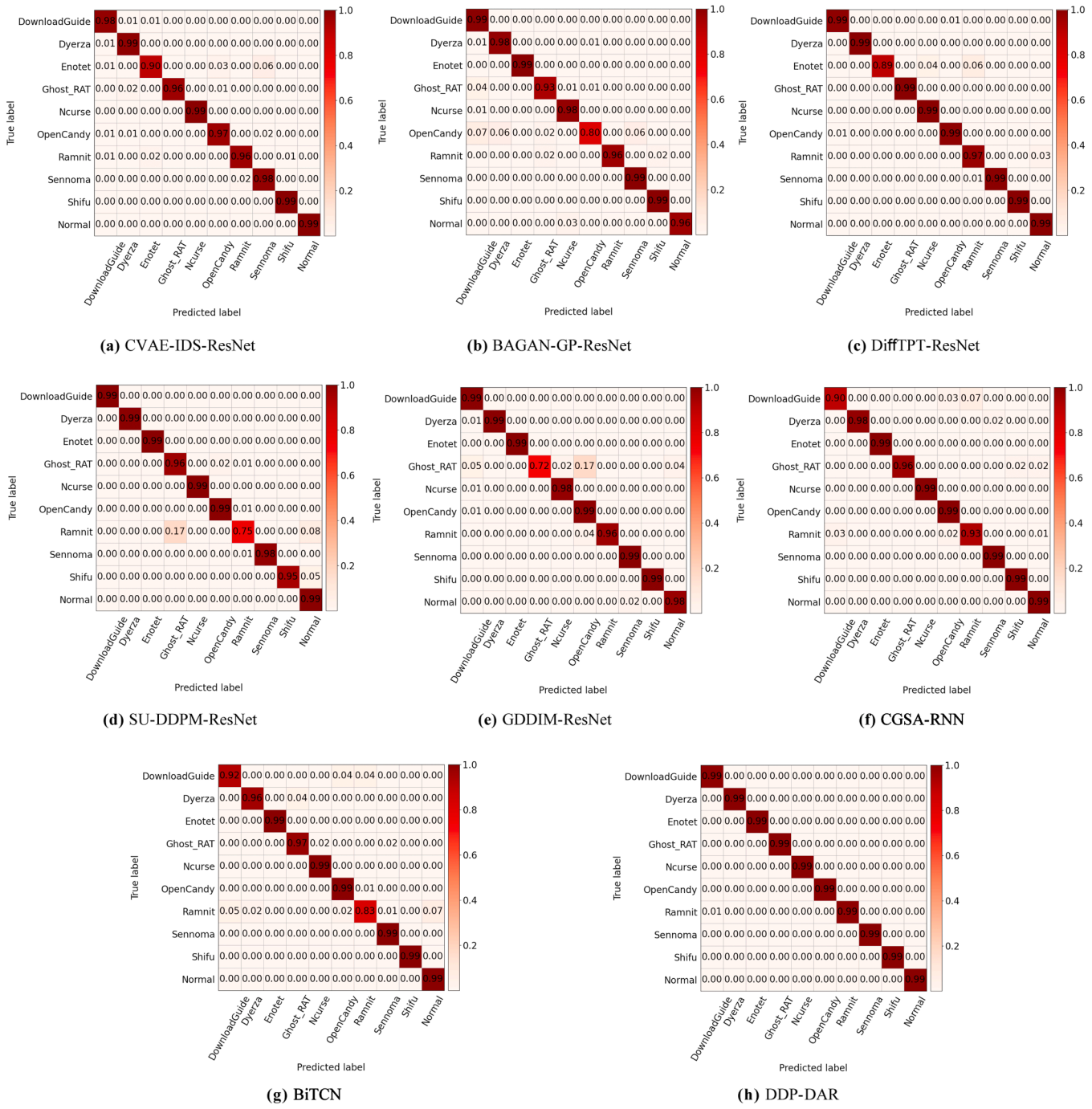
(a) CVAE-IDS-ResNet **(b)** BAGAN-GP-ResNet **(c)** DiffTPT-ResNet

(d) SU-DDPM-ResNet **(e)** GDDIM-ResNet **(f)** CGSA-RNN

(g) BiTCN **(h)** DDP-DAR

Fig. 7. The confusion matrix of eight data augmentation-based NID methods on USTC-TFC dataset.

usability and scalability: For the proposed DDP-DAR model, its detection performance is verified on three public network traffic datasets and one real-life network traffic dataset, and the experimental results show that DDP-DAR model has better performance in terms of Accuracy, F1-measure, FPR, ROC-AUC and stability. Therefore, DDP-DAR has strong practical usability in network intrusion detection. However, due to the fact that the proposed DDP-DAR model requires the use of diffusion models for data augmentation, and the network intrusion detection model has a complex network structure, it is difficult for the model to meet real-time requirements when dealing with large-scale network traffic. In the future, we plan to improve the time efficiency of DDP-DAR model through the following strategies: (I) Change the initial sampling distribution of network traffic to reduce the number of iterations in the denoising stage as well as accelerate the inference process of denoising diffusion probability model, thereby reducing the time consumption; (II) Reduce the number of network layers in the detection model appropriately to accelerate the extraction of network traffic features; (III) Accelerate the inference speed of model by utilizing parallel computing and distributed processing, and then assign the computing tasks to multiple nodes in a large network environment, thereby speeding up the processing speed of DDP-DAR model. (2) **Deployment challenges:** (I) Existing network infrastructure has weak computing power: DDP-DAR model uses dual attention residual network and network traffic denoising diffusion probability model to improve the performance of network intrusion detection, which requires high computing resources, therefore, it is necessary to evaluate whether the existing network infrastructure has sufficient computing power to support the operation of DDP-DAR model. (II) Weak inference ability for



(a) CVAE-IDS-ResNet **(b)** BAGAN-GP-ResNet **(c)** DiffTPT-ResNet

(d) SU-DDPM-ResNet **(e)** GDDIM-ResNet **(f)** CGSA-RNN

(g) BiTCN **(h)** DDP-DAR

Fig. 8. The confusion matrix of eight data augmentation-based NID methods on ISAC dataset.

real-time processing of network traffic: When deploying DDP-DAR models in high-speed network environments, it is necessary to optimize the inference speed of model through parallel processing and other methods; In addition, it is also necessary to preprocess network traffic to ensure the efficiency of model in high-speed traffic environments; These two requirements make it difficult for the DDP-DAR model to detect malicious traffic in large-scale network traffic in real-time after deployment. (III) The model has weak adaptability to real network traffic: In different network environments, the detection performance of DDP-DAR may change with the emergence of new cyber-attacks, which requires regular retraining of DDP-DAR model to adapt to new network environments. **(3) Limitations of the used dataset:** Due to the dynamic nature of network behavior, the distribution of data samples in network traffic will change over time, resulting in concept drift phenomenon, which leads to a decrease detection efficiency of DDP-DAR; However, the proposed DDP-DAR model does not consider the situation of concept drift, it may not be suitable for handling the network traffic with concept drift. In the future, a concept drift detection and adaptive method can be integrated on the basis of DDP-DAR to adapt to the phenomenon of concept drift in network traffic, thereby improving the accuracy of network intrusion detection.

In addition, when deploying the DDP-DAR model in actual network environments, the following practical factors need to be considered: **(1) The computing power of existing network infrastructure:** DDP-DAR model uses a dual attention residual network and a network traffic denoising diffusion probability model to improve the detection accuracy

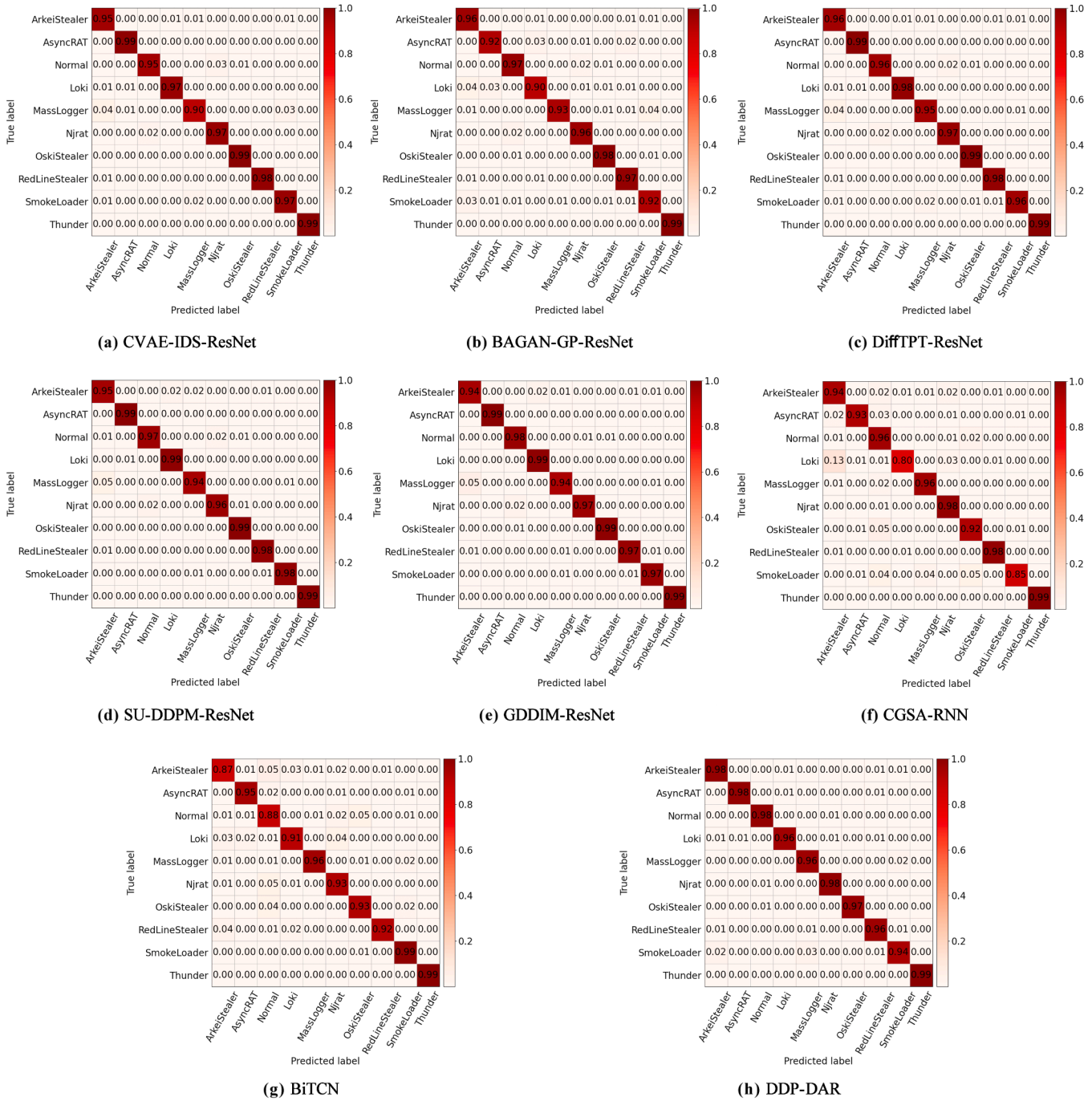


Fig. 9. The confusion matrix of eight data augmentation-based NID methods on UJS-IDS2024 dataset.

Answer to RQ3:

Extensive experimental results show that compared with six state-of-the-art data augmentation-based NID methods and one NID method, the proposed DDP-DAR method performs better in four metrics of Accuracy, F1-measure, FPR and ROC-AUC.

of network intrusion, which requires relatively high computing resources, therefore, it is necessary to evaluate whether the existing network infrastructure has sufficient computing power to support the real-time operation of DDP-DAR model. **(2) The inference ability of the model for real-time processing of network traffic:** When deploying DDP-DAR model in high-speed traffic network environment, it is necessary to optimize the inference speed of the model through parallel processing and other methods; In addition, to ensure the efficiency of model in high-speed network traffic environment, the preprocessing of network traffic is also necessary. **(3) The adaptability of**

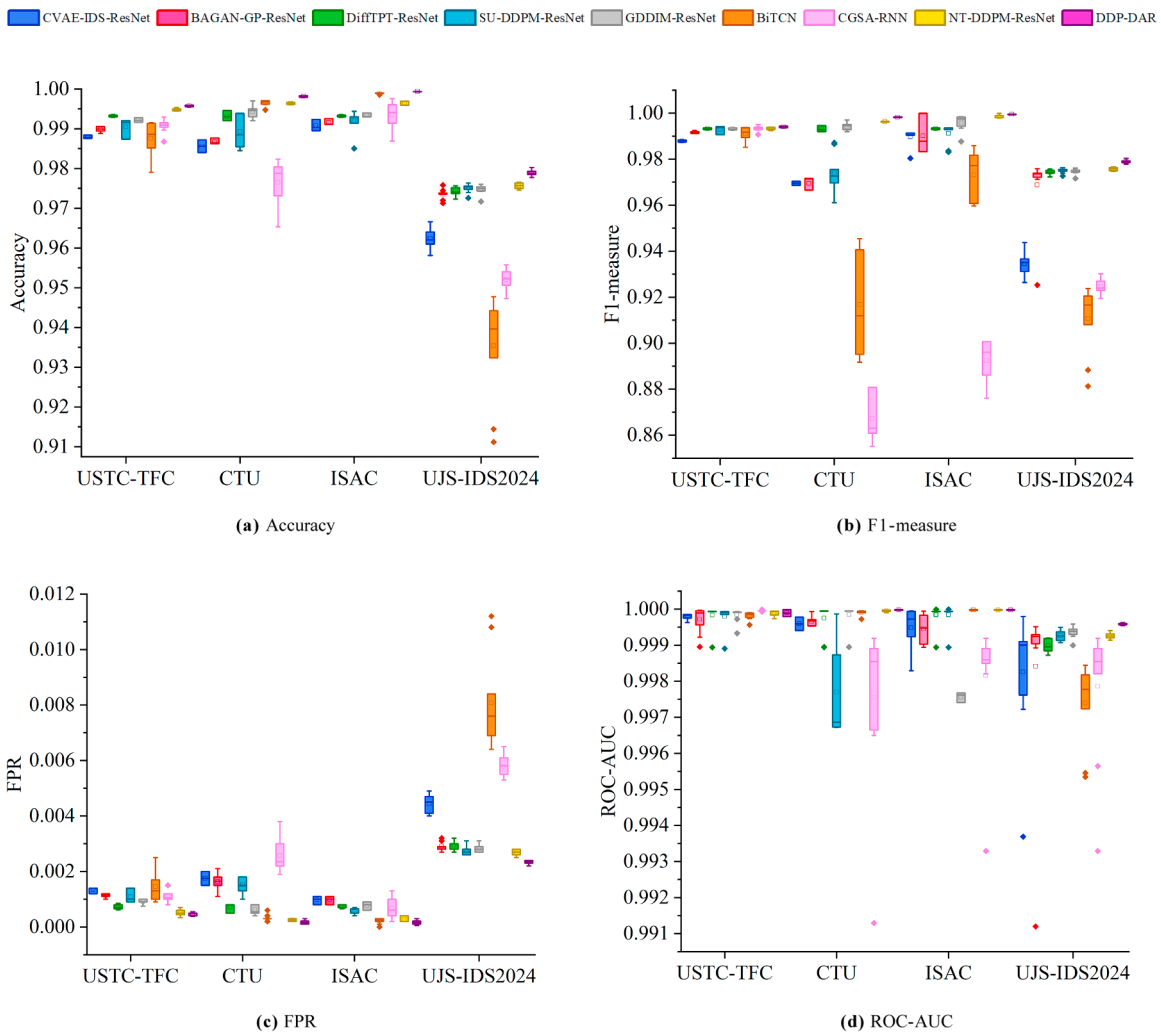


Fig. 10. Stability of different data augmentation-based NID methods.

Answer to RQ4:

Extensive experimental results show that the proposed DDP-DAR method performs better in stability than six data augmentation-based NID methods and one NID method on four network traffic datasets.

the model to real network traffic: In different network environments, the performance of DDP-DAR may change with the emergence of new types of network attacks, so it is necessary to regularly retrain DDP-DAR model to adapt to new network traffic environment; In addition, other factors such as the interpretability of model need to be considered when deploying the model, thereby further improving the effectiveness and reliability of DDP-DAR model in practical network environment.

5.6. Threats to validity

Extensive experimental results in subSections 5.1-5.4 show that the proposed DDP-DAR method can effectively expand the number of smaller-scale intrusion traffic, which helps the detection model to obtain better detection performance and more stable detection results. However, our research also faces the following threats to validity. (1) In the selection of data augmentation model and network intrusion model, we mainly focus on the detection accuracy; However, the time efficiency is also an important factor in NID, which requires us to adopt the lightweight deep learning framework for model design. (2) The proposed DDP-DAR method mainly incorporates an improved denoising diffusion probabilistic model for data augmentation and a dual-attention ResNet for intrusion detection, it is still unknown whether these two deep learning models are inherently safe and thus able to prevent hackers from launching network attacks through these two methods themselves. (3) We select default parameters in the experiments, while different hyper-parameter settings may lead to different detection results, thus, whether the DDP-DAR model can achieve better detection performance under different parameters remains to be further verified. (4) The diffusion model used in the DDP-DAR method is mainly used in the field of computer vision, and no diffusion model has been used in the field of NID; In order to validate the effect of the proposed NT-DDPM model on the data augmentation, the DiffTPT, SU-DDPM and GDDIM model used in the computer vision field are as comparisons, the reasonableness of this choice is still debatable. (5) Because some models (such as DiffTPT,

SU-DDPM and GDDIM) do not disclose source code, we reproduce these methods based on our understanding, this process inevitably generates errors due to our cognitive limitations. (6) In the process of validating the effectiveness of DDP-DAR method, three commonly used network traffic datasets are selected in the experiments, these three datasets only cover some categories of network traffic, resulting in unknown whether the DDP-DAR method is applicable for other types of network traffic, which requires deploying it to real-world scenarios in the future to validate its practicality. (7) Currently, hackers utilize existing deep learning-based NID models to carry out reverse design to avoid detection, which may lead to misreporting certain intrusion traffic as normal traffic with the DDP-DAR method; In the future, we can ensemble multiple NID models to effectively counteract failures due to the breakthroughs in individual detection models, thus providing more solid protection for cyberspace security.

6. Conclusion

In recent years, network intrusion has brought many challenges to network security protection, and the data imbalance phenomenon in network traffic leads to the fact that existing network intrusion detection methods cannot accurately detect malicious attacks. In order to more accurately detect intrusion traffic among a large amount of network traffic, this paper proposes a network intrusion detection method called DDP-DAR based on denoising diffusion probabilistic model and dual-attention residual network, which is composed of three modules: feature representation module, data augmentation module and network intrusion detection module, these three modules work together to complete network intrusion detection tasks. Firstly, the feature representation module fills all-layer features, application-layer features and session-layer features of network traffic into the R, G and B channels of RGB image, thereby converting the original network traffic in the PCAP form into an RGB image, facilitating subsequent data augmentation and network intrusion detection. And then, the converted RGB image of network traffic is sent to data augmentation module based on the denoising diffusion probability model, which learns the data distribution of RGB image of original network traffic to generate malicious traffic samples, thereby solving the problem of data imbalance. Finally, the generated RGB images and original RGB images of original network traffic are shuffled and sent to the improved ResNet model-based network intrusion detection module for network intrusion detection, thereby improving the detection efficiency of network intrusion.

In order to verify the effectiveness of the proposed DDP-DAR model for network intrusion detection, a large number of experiments are conducted on three publicly available network traffic, including CTU, USTC-TFC and ISAC, and the experimental results show that DDP-DAR model outperforms the existing data augmentation-based NID methods in the four metrics of Accuracy, F1-measure, FPR and ROC-AUC, and it also has better stability.

However, the dynamic and real-time changes in network result in the trained NID models not being able to adapt well to current network environment. In the future, we can improve our proposed methodology in the following ways, summarized as follows: (1) The introduction of denoising diffusion probability model consumes a lot of time to generate the RGB images of small-scale network traffic, therefore, it is better to further optimize the efficiency of DDP-DAR model in the future to process network traffic, especially in the large-scale network environments. Specifically, we can start with model optimization and deployment: (I) Model optimization: The time consumption of model can be reduced by changing the initial sampling distribution to reduce the number of iterations in the denoising stage of the model as well as accelerating the inference process of denoising diffusion probability model. (II) Model deployment: DDP-DAR models can be deployed on distributed network environments or edge devices to achieve low latency, high-throughput network intrusion detection. (2) The emergence of concept drift phenomenon can affect the feature distribution of network traffic, ultimately affecting the effectiveness of network intrusion detection. Therefore, the proposed DDP-DAR model can be combined with a concept drift detection model in the future to adaptively detect concept drift phenomena in the network environment using concept drift detection model and provide timely feedback to DDP-DAR, thereby reducing the situation that detecting the network traffic with concept drift phenomena as intrusion traffic by DDP-DAR and ultimately improving the effectiveness of DDP-DAR model for network intrusion detection. (3) Although the proposed DDP-DAR model can accurately detect intrusion traffic in four network traffic datasets, it should be acknowledged that these datasets only contain some types of intrusion traffic. Therefore, it is unknown whether the DDP-DAR model can adapt to more types of intrusion traffic. In the future, we plan to combine the detection model in DDP-DAR with other detection models (such as deep learning models or rule-based models) to form a hybrid detection system, thereby providing higher detection accuracy and coverage in different attack scenarios as well as improving the robustness and adaptability of the model.

CRediT authorship contribution statement

Saihua Cai: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Funding acquisition. **Yingwei Zhao:** Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation. **Jiaao Lyu:** Writing – review & editing, Validation, Data curation. **Shengran Wang:** Writing – review & editing, Data curation. **Yikai Hu:** Writing – review & editing, Data curation. **Mengya Cheng:** Writing – review & editing, Data curation. **Guofeng Zhang:** Writing – review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) (Grant no. 62202206), the China Postdoctoral Science Foundation (Grant no. 2023T160275), the Natural Science Foundation of Jiangsu Province (Grant no. BK20220515), the Shandong Provincial Natural Science Foundation (Grant no. ZR2021QF056), the Shandong Province Science and Technology Small and Medium sized Enterprise Innovation Ability Enhancement Project (Grant no. 2023TSGC0588), and the Graduate Research and Innovation Project of Jiangsu Province (Grant no. KYCX24_3981).

Data availability

Data will be made available on request.

References

Abdulganiyu, O. H., Tchakoucht, T. A., & Saheed, Y. K. (2024). Towards an efficient model for network intrusion detection system (IDS): Systematic literature review. *Wireless Networks, 30*, 453–482.

Ahsan, M. M., Ali, M. S., & Siddique, Z. (2024). Enhancing and improving the performance of imbalanced class data using novel GBO and SSG: A comparative analysis. *Neural Networks, 173*, Article 106157.

Akgun, D., Hizal, S., & Cavusoglu, U. (2022). A new ddos attacks intrusion detection model based on deep learning for cybersecurity. *Computers & Security, 118*, Article 102748.

Bhosale, S., Nag, S., Kanojia, D., Deng, J., & Zhu, X. (2024). DiffSED: Sound event detection with denoising diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 792–800).

Blaise, A., Bouet, M., Conan, V., & Secci, S. (2020). Detection of zero-day attacks: An unsupervised port-based approach. *Computer Networks, 180*, Article 107391.

Cai, S., Xu, H., Liu, M., Chen, Z., & Zhang, G. (2024). A malicious net-work traffic detection model based on bidirectional temporal convolutional network with multi-head self-attention mechanism. *Computers & Security, 136*, Article 103580.
Cai, S., Zhao, W., Tang, H., Chen, J., & Guo, W. (2023). CGSA-RNN: Ab-normal network traffic detection model based on cyclegan and self-attention mechanism. In *2023 IEEE 23rd International Conference on SoftwareQuality, Reliability, and Security (QRS)* (pp. 541–549).
Caville, E., Lo, W. W., Layeghy, S., & Portmann, M. (2022). Anomal-E: A self-supervised network intrusion detection system based on graph neural networks. *Knowledge-Based Systems, 258*, Article 110030.
Chen, J., Chen, Y., Cai, S., Yin, S., Zhao, L., & Zhang, Z. (2023a). An optimized feature extraction algorithm for abnormal network traffic detection. *Future Generation Computer Systems, 149*, 330–342.
Chen, J., Lv, T., Cai, S., Song, L., & Yin, S. (2023b). A novel detection model for abnormal network traffic based on bidirectional temporal convolutional network. *Information and Software Technology, 157*, Article 107166.
Chen, J., Song, L., Cai, S., Xie, H., Yin, S., & Ahmad, B. (2023c). TLS-MHSA: An efficient detection model for encrypted malicious traffic based on multi-head self-attention mechanism. *ACM Transactions on Privacy and Security, 26*, 1–21.
Das, S., Saha, S., Priyoti, A. T., Roy, E. K., Sheldon, F. T., Haque, A., et al. (2022). Network intrusion detection and comparative analysis using ensemble machine learning and feature selection. *IEEE Transactions on Network and Service Management, 19*, 4821–4833.
de Araujo-Filho, P. F., Naili, M., Kaddoum, G., Fapi, E. T., & Zhu, Z. (2023). Unsupervised GAN-based intrusion detection system using temporal con-volutional networks and self-attention. *IEEE Transactions on Network and Service Management, 20*, 4951–4963.
Ding, H., Sun, Y., Huang, N., Shen, Z., & Cui, X. (2024). TMG-GAN: Generative adversarial networks-based imbalanced learning for network intrusion detection. *IEEE Transactions on Information Forensics and Security, 19*, 1156–1167.
Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16×16 words: Transformers for image recognition at scale. *International Conference on Learning Representations, abs/2010.11929*.
Feng, C.-M., Yu, K., Liu, Y., Khan, S., & Zuo, W. (2023). Diverse data augmentation with diffusions for effective test-time prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 2704–2714).
Fernando, K. R. M., & Tsokos, C. P. (2022). Dynamically weighted balanced loss: Class imbalanced learning and confidence calibration of deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 33*, 2940–2951.
Gamage, S., & Samarabandu, J. (2020). Deep learning methods in network intrusion detection: A survey and an objective comparison. *Journal of Network and Computer Applications, 169*, Article 102767.
Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. et al. (2014). Generative Adversarial Networks. Communications of the ACM, abs/1406.2661.
Gupta, N., Jindal, V., & Bedi, P. (2021). LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system. *Computer Networks, 192*, Article 108076.
He, J., Wang, X., Song, Y., Xiang, Q., & Chen, C. (2022). Network intrusion detection based on conditional wasserstein variational autoencoder with generative adversarial network and one-dimensional convolutional neural networks. *Applied Intelligence, 53*, 12416–12436.
He, X., Chen, Q., Tang, L., Wang, W., & Liu, T. (2023). CGAN-based collaborative intrusion detection for UAV networks: A blockchain-empowered distributed federated learning approach. *IEEE Internet of Things Journal, 10*, 120–132.
Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Proceedings of the 34th International Conference on Neural Information Processing Systems, abs/2006.11239.
Hu, X., Gao, W., Cheng, G., Li, R., Zhou, Y., & Wu, H. (2023). Toward early and accurate network intrusion detection using graph embedding. *IEEE Transactions on Information Forensics and Security, 18*, 5817–5831.
Huang, G., & Jafari, A. (2020). Enhanced balancing GAN: Minority-class image generation. *Neural Computing & Applications, 35*, 5145–5154.
Imrana, Y., Xiang, Y., Ali, L., & Abdul-Rauf, Z. (2021). A bidirectional LSTM deep learning approach for intrusion detection. *Expert Systems with Applications, 185*, Article 115524.
Injadat, M., Moubayed, A., Nassif, A. B., & Shami, A. (2021). Multi-stage optimized machine learning framework for network intrusion detection. *IEEE Transactions on Network and Service Management, 18*, 1803–1816.
Landman, T., & Nissim, N. (2021). Deep-Hook: A trusted deep learning-based framework for unknown malware detection and classification in linux cloud environments. *Neural Networks, 144*, 648–685.
Le, T.-T.-H., Shin, Y., Kim, M., & Kim, H. (2024). Towards unbalanced multiclass intrusion detection with hybrid sampling methods and ensemble classification. *Applied Soft Computing, 157*, Article 111517.
Li, Y., Yu, Z., He, G., Shen, Y., Li, K., Sun, X., et al. (2023). SPD-DDPM: Denoising diffusion probabilistic models in the symmetric positive definite space. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
Liu, C., Antypenko, R., Sushko, I., & Zakharchenko, O. (2022a). Intrusion detection system after data augmentation schemes based on the VAE and CVAE. *IEEE Transactions on Reliability, 71*, 1000–1010.
Liu, J., Gao, Y., & Hu, F. (2021). A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM. *Computers & Security, 106*, Article 102289.
Liu, Q., Wang, D., Jia, Y., Luo, S., & Wang, C. (2022b). A multi-task based deep learning approach for intrusion detection. *Knowledge-Based Systems, 238*, Article 107852.
Lopes,I., Zou, D., Abdulqadder, I.H., Akbar, S., Li, Z., Ruambo, F. et al. (2023). Network intrusion detection based on the temporal convolutional model. *Computers & Security*, 135, 103465.
Lu, S., Guan, F., Zhang, H., & Lai, H. (2024). Speed-up ddpm for real-time underwater image enhancement. *IEEE Transactions on Circuits and Systems for Video Technology, 34*, 3576–3588.
Manjula, P., & Baghavathi Priya, S. (2022). An effective network intrusion detection and classification system for securing WSN using VGG-19 and hybrid deep neural network techniques. *Journal of Intelligent & Fuzzy Systems, 43*, 6419–6432.
Mohi-Ud-Din, G., Zhiqiang, L., Jiangbin, Z., Sifei, W., Zhijun, L., Asim, M., et al. (2023). Intrusion detection using hybrid enhanced CSA-PSO and multivariate wls random-forest technique. *IEEE Transactions on Network and Service Management, 20*, 4937–4950.
Niu, A., Pham, T. X., Zhang, K., Sun, J., Zhu, Y., Yan, Q., et al. (2024). ACDMSR: Accelerated conditional diffusion mod- els for single image super-resolution. *IEEE Transactions on Broadcasting, 70*, 492–504.
Paya, A., Arroni, S., Garca-Daz, V., & Gomez, A. (2024). Apollon: A robust defense system against adversarial machine learning attacks in intrusion detection systems. *Computers & Security, 136*, Article 103546.
Pingale, S. V., & Sutar, S. R. (2022). Remora whale optimization-based hybrid deep learning for network intrusion detection using CNN features. *Expert Systems with Applications, 210*, Article 118476.
Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *Computer Science, abs/1511.06434*.
Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 10674–10685).
Singh, N. K., Majeed, M. A., & Mahajan, V. (2022). Statistical machine learning defensive mechanism against cyber intrusion in smart grid cyberphysical network. *Computers & Security, 123*, Article 102941.
Suja Mary, D., Jaya Singh Dhas, L., Deepa, A., Chaurasia, M. A., & Jaspin Jeba Sheela, C. (2024). Network intrusion detection: An optimized deep learning approach using big data analytics. *Expert Systems with Applications, 251*, Article 123919.
Turukmane, A. V., & Devendiran, R. (2024). M-multisvm: An efficient feature selection assisted network intrusion detection system using machine learning. *Computers & Security, 137*, Article 103587.
Vo, H. V., Du, H. P., & Nguyen, H. N. (2024). APELID: Enhancing re al-time intrusion detection with augmented WGAN and parallel ensemble learning. *Computers & Security, 136*, Article 103567.
Wang, P., Li, S., Ye, F., Wang, Z., & Zhang, M. (2020). PacketCGAN: Ex-ploratory study of class imbalance for encrypted T*raffic* classification using CGAN. In *2020 IEEE International Conference on Communications (ICC)* (pp. 1–7).
Wang, P., Wang, Z., Ye, F., & Chen, X. (2021). ByteSGAN: A semi-supervised Generative Adversarial Network for encrypted tr*affic* classification in SDN Edge Gateway. *Computer Networks, 200*, Article 108535.
Wang, Y., Liu, B., Yang, B., Li, J., Li, Y., & Pei, Y. (2023). DDPM-SKDNet: A deep learning method for ICG image classification. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 3204–3209).
Wei, N., Yin, L., Tan, J., Ruan, C., Yin, C., Sun, Z., et al. (2024). An autoencoder-based hybrid detection model for intrusion detection with small-sample problem. *IEEE Transactions on Network and Service Management, 21*, 2402–2412.
Xia, B., Han, D., Yin, X., & Na, G. (2022). RICNN: A ResNet& inception convolutional neural network for intrusion detection of abnormal traffic. In *Computer Science and Information Systems, 19* pp. 309–326).
Xu, C., Wang, H., Wang, W., Zheng, P., & Chen, H. (2024). Geomet ric-facilitated denoising diffusion model for 3D molecule generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
Yang, Y., Fu, H., Avils-Rivero, A. I., Schonlieb, C.-B., & Zhu, L. (2023a). DiffMIC: Dual-guidance diffusion network for medical image classification. *Medical Image Computing and Computer Assisted Intervention - MICCAI 2023*.ArXiv:abs/2303.10610.
Yang, Z., Liu, X., Li, T., Wu, D., Wang, J., Zhao, Y., et al. (2022). A systematic literature review of methods and datasets for anomaly-based network intrusion detection. In *Computers & Security, 116*, Article 102675.
Yang, Z., Peng, D., Kong, Y., Zhang, Y., Yao, C., & Jin, L. (2023b). Font-Diffuser: One-shot font generation via denoising diffusion with multi-scale content aggregation and style contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
Zhang, C., Lin, G., Liu, F., Yao, R., & Shen, C. (2019). CANet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *2019 IEEE/CVF Conference on Computer Vision and Pat- tern Recognition (CVPR)* (pp. 5212–5221).
Zhang, Q., Tao, M., & Chen, Y. (2022). gDDIM: Generalized denoising diffusion implicit models. *International Conference on Learning Representations, abs/2206.05564*.
Zhong, W., Raahemi, B., & Liu, J. (2009). Learning on class imbalanced data to classify peer-to-peer applications in IP traffic using resampling techniques. In *2009 International Joint Conference on Neural Networks (IJCNN)* (pp. 3548–3554).
Zhu, J.-Y., Park, T., Isola, P., & Efros, A.A. (2020). Unpaired image-to-image translation using cycle-consistent adversarial networks. 2017 IEEE International Conference on Computer Vision (ICCV), abs/1703.10593.