

# 基于改进Cycle-GAN的样本数据生成在入侵检测中的应用研究

曾庆鹏, 郭航恺

(南昌大学数学与计算机学院, 江西 南昌 330031)

**摘要:** 针对入侵检测领域标准数据集存在数据更新缓慢、某些入侵类别数据样本少、正常数据集与异常数据集分布不平衡等问题, 从样本数据增强和检测模型优化两方面入手, 将入侵样本数据转为图数据作为Cycle-GAN的输入, 再将空间注意力机制引入Cycle-GAN的生成器, 保留数据流量特征的同时对关键特征信息进行抽取, 采用无监督学习优化原始数据集的分布。提出了一种基于全局注意力和残差结构的分类网络, 将预处理得到的图数据作为模型的输入, 经过串联通道注意力和空间注意力得到的全局注意力对输入特征分配权重, 最后模型输出为入侵分类。在CIC-IDS2017和NSL-KDD数据集上进行实验, 与原始数据在同类模型训练结果相比较, 在CIC-IDS2017数据集上,  $F1$  值从0.853 2提升到0.978 6, 召回率从0.914 8提升到0.984 2, 在NSL-KDD数据集上,  $F1$  值从0.646 2提升到0.844 3, 召回率从0.726 提升到0.876 8, 表明该方法可有效解决入侵检测某些样本类别更新缓慢及某些入侵类别样本少问题。

**关键词:** 入侵检测; 图数据转换; 循环生成对抗网络; 样本数据生成

**中图分类号:** TP391

**文献标志码:** A

**DOI:** 10.11959/j.issn.2096-109x.2025019

## Application research of sample data generation based on improved Cycle-GAN in intrusion detection

ZENG Qingpeng, GUO Hangkai

School of Mathematics and Computer Sciences, Nanchang University, Nanchang 330031, China

**Abstract:** The issue of slow data updates, insufficient data samples for certain intrusion categories, and imbalanced distributions between normal and abnormal data sets in standard intrusion detection data sets have been addressed through both data sample augmentation and detection model optimization. The intrusion sample data was converted into graph data to serve as input for Cycle-GAN, and a spatial attention mechanism was introduced into the Cycle-GAN generator. This approach preserved data traffic characteristics while extracting key feature information and utilized unsupervised learning to optimize the original data set distribution. A classification network based on

收稿日期: 2024-07-16; 修回日期: 2024-12-11

通信作者: 曾庆鹏, zengqingpeng@ncu.edu.cn

基金项目: 国家自然科学基金(62466035)

**Foundation Item:** The National Natural Science Foundation of China (62466035)

**引用格式:** 曾庆鹏, 郭航恺. 基于改进Cycle-GAN的样本数据生成在入侵检测中的应用研究[J]. 网络与信息安全学报, 2025, 11(2): 87-101.

**Citation Format:** ZENG Q P, GUO H K. Application research of sample data generation based on improved Cycle-GAN in intrusion detection[J]. Chinese Journal of Network and Information Security, 2025, 11(2): 87-101.

global attention and residual structure was proposed, taking the preprocessed graph data as input. After serializing channel attention and spatial attention to obtain global attention, input features were weighted. Finally, the model outputted the intrusion classification. Experiments on the CIC-IDS2017 and NSL-KDD data sets showed that, compared to similar models trained with the original data, the F1 score increased from 0.853 2 to 0.978 6 and the recall rate from 0.914 8 to 0.984 2 on the CIC-IDS2017 data set, and the F1 score increased from 0.646 2 to 0.844 3 and the recall rate from 0.726 to 0.876 8 on the NSL-KDD data set. This indicates that the proposed method effectively addressed the problems of slow data updates and insufficient samples for certain intrusion categories in intrusion detection.

**Keywords:** intrusion detection, graph data transformation, Cycle-GAN, sample data generation

## 0 引言

入侵检测技术作为网络安全的一道防线,通过监视主机行为或分析网络流量,及时发现并响应可疑活动。目前大多研究把入侵检测看作分类问题,主要判别样本数据是正常或异常。传统的机器学习算法由于其网络结构的局限性,无法应对层出不穷的网络攻击类型,越来越多的学者通过使用深度学习技术,来弥补传统机器学习算法的缺陷。虽然深度学习技术在入侵检测中取得了较高的准确率,但漏报率问题也不容忽视,导致漏报率较高的原因之一是数据不平衡问题。因此,研究如何处理类别不平衡的数据集,削弱不平衡数据对模型性能表现的影响变得十分重要。

Gu 等<sup>[1]</sup>提出了一种用于网络入侵检测的新型软原型自主模糊推理系统——SPAFIS, SPAFIS 以在线方式按块从网络流量数据流中学习,并自主识别一组有意义的、人类可解读的软原型,以构建用于分类的 IF-THEN 模糊规则库,并基于数据的集合属性和相互距离以人类可解读的方式进行内部推理和决策。Yuan 等<sup>[2]</sup>提出了一种基于动态图的实时入侵检测系统,该图根据消息的到达动态构建,并且同时提取特征,通过离线阶段提取的特征分布,构建的系统能够实时检测传入的消息,并识别被攻击消息的 ID。基于入侵检测逐步应用于微服务,Flora 等<sup>[3]</sup>基于微服务应用最新技术和工作负载技术提出了一种全面的方法来评估和比较微服务应用的不同入侵检测方法,并提出了一种基准测试方法,允许用户进行标准化代表性和可复现的评估过程。Li 等<sup>[4]</sup>对物联网环境中网络入侵检测系统的特征选择和特征提取技术进行了全面比较,使用 TON-IoT 和 BoT-IoT 数据集进行了二分类和多分类任务的测试,评估了包

括皮尔逊相关性和卡方在内的特征选择方法,以及主成分分析和自编码器等特征提取方法,并在 5 个经典机器学习模型上进行测试:决策树、随机森林、朴素贝叶斯、 $k$ -最近邻和多层感知器。Ouyang 等<sup>[5]</sup>提出了一种基于小样本学习的入侵检测系统 (few-shot intrusion detection system, FS-IDS),实验结果表明,在最坏的情况下,即只有少量恶意样本可用且包含各种已知或未知攻击的情况下,FS-IDS 保持了超过 90% 的检测准确率和召回率。Luigi 等<sup>[6]</sup>提出了一种基于多阶段深度学习的入侵检测系统,用于检测和分类汽车以太网网络中的网络攻击,第一阶段使用随机森林分类器快速检测网络攻击,第二阶段使用一种剪枝的卷积神经网络,该网络在分类不同类型的网络攻击时最小化了误报率。吴越等<sup>[7]</sup>总结了工业传感网络异常的成因,从时序特征、时空多尺度及非结构图表征 3 个视角,对工业传感网络异常检测的关键技术及典型方法进行阐述,分析现有各类方法的突破。Martins 等<sup>[8]</sup>提出了基于主机的入侵检测系统在物联网中异常检测的发展综述,并探讨了在物联网环境中使用主机信息进行入侵检测的方法和技术。Huang 等<sup>[9]</sup>介绍了一种用于 VANET 的增量生命周期学习入侵检测系统,利用增量学习技术来不断更新模型以应对新的入侵形式,提高了入侵检测的效果。

生成式对抗网络 (generative adversarial network, GAN)<sup>[10]</sup>是最近几年在深度学习领域新兴的热点研究方向之一。GAN 作为一种新的学习生成模型,在计算机视觉、语音语言领域、强化学习领域取得了不错的成果并衍生出许多改进模型。Zhao 等<sup>[11]</sup>提出了一种通过整合生成对抗网络来提高 NIDS 性能的新方法,利用 GAN 生成模

拟真实网络行为的合成网络流量数据的能力, 实现了3种不同的GAN模型 (Vanilla GAN、Wasserstein GAN和Conditional Tabular GAN), 生成专门用于表示异常活动的真实网络流量模式。深度卷积GAN<sup>[12]</sup>用卷积层代替了全连接层, 采用带步长的卷积代替上采样, 更好地提取图像特征, 判别器和生成器对称存在, 极大地提升了GAN训练的稳定性 and 生成结果的质量。Wasserstein GAN<sup>[13]</sup>从损失函数的角度对GAN进行了改进, 这种改进在全连接层也能得到很好的表现结果, 并采用Wasserstein距离来衡量数据分布和真实数据分布之间的距离, 理论上解决了训练不稳定的问题。最小二乘GAN (least squares GAN, LSGAN)<sup>[14]</sup>采用最小二乘损失函数代替原始GAN的交叉熵损失函数, 主要针对原始GAN生成器生成的图像质量不高和训练过程不稳定两个问题, 传统GAN以交叉熵作为损失, 使得生成器不会再优化那些被判别器识别为真实图片的生成图片, 即使这些生成图片距离真实数据比较远。循环生成对抗网络 (cycle generative adversarial networks, Cycle-GAN)<sup>[15]</sup>能够在两种类型图像之间进行转换, 而不需要两者对应关系, Cycle-GAN的创新点在于在源域和目标域之间, 无须建立训练数据间一对一的映射, 就可实现这种迁移。

总体来讲, 目前大部分入侵检测的研究重心在分类算法优化上, 如深度学习、联邦学习<sup>[16-17]</sup>, 侧重提高分类模型对异常数据的分类准确率, 使用的数据集也大多是标准数据集。但目前入侵检测领域的数据集存在数据更新慢、某些入侵类别数据样本过少等问题, 数据的不平衡会使分类器的判断趋向于数据占比更多的一类。所以, 利用生成样本的方法来扩充数据, 帮助训练分类模型是现在入侵检测领域的重要方向, 本文探究Cycle-GAN在入侵检测中的数据生成效果, 提出将原始数据集的数据特征经过处理后转为图数据格式, 然后借助Cycle-GAN在图像领域的应用优势来生成数据, 完成数据扩充后, 每个攻击类型的数据量得到了显著提升, 同时不平衡率也得到了有效控制, 使得数据集的数量分布更加均衡。该方法可有效缓解数据更新缓慢、某些入侵类别数据样本少、入侵数据由于敏感性不便于公

开等问题, 这对于建立入侵检测系统时提高模型的性能和鲁棒性具有重要意义。

## 1 循环生成对抗网络

### 1.1 网络模型结构

Cycle-GAN的优势在于只需要两种域的数据, 而不需要它们之间有严格的对应关系。其核心思想是从域 $X$ 生成域 $Y$ 的样本, 再从域 $Y$ 生成域 $X$ 的样本, 如此反复迭代循环。Cycle-GAN原理如图1所示。 $X$ 和 $Y$ 是两个不同的数据集, 生成器 $G$ 和 $F$ 分别是从小 $X$ 到 $Y$ 和从 $Y$ 到 $X$ 的映射, 同时需要两个判别器 $D_x$ 和 $D_y$ , 分别用于判别生成器生成图像的质量。

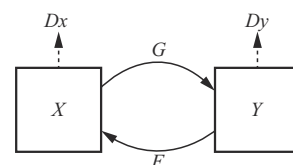


图1 Cycle-GAN原理

Figure1 Cycle-GAN schematic

### 1.2 Cycle-GAN损失

Cycle-GAN的损失分别由 $\text{Loss}_{\text{GAN}}$ 、 $\text{Loss}_{\text{cycle}}$ 和 $\text{Loss}_{\text{identity}}$ 组成, 即总的 $\text{Loss} = \text{Loss}_{\text{GAN}} + \text{Loss}_{\text{cycle}} + \text{Loss}_{\text{identity}}$ 。 $\text{Loss}_{\text{GAN}}$ 保证生成器和判别器可以进化, 进而保证生成器能产生更真实的数据。

$\text{Loss}_{\text{GAN}}$ 由两部分组成, 式(1)是由 $X \rightarrow Y$ 生成过程的损失, 式(2)是 $Y \rightarrow X$ 生成过程的损失。

$$L_{\text{GAN}}(G, D_Y, X, Y) = E_{y \sim p_{\text{data}}(y)} [D_Y(y)]^2 + E_{x \sim p_{\text{data}}(x)} [D_Y(G(x)) - 1]_{n \times n}^2 \quad (1)$$

$$L_{\text{GAN}}(F, D_X, Y, X) = E_{x \sim p_{\text{data}}(x)} [D_X(x)]^2 + E_{y \sim p_{\text{data}}(y)} [D_X(F(y)) - 1]_{n \times n}^2 \quad (2)$$

$\text{Loss}_{\text{cycle}}$ 是为了防止出现过拟合的情况, 由Zhu<sup>[15]</sup>提出, 即将从 $X \rightarrow Y$ 的生成图片 $y'$ 再放入生成器 $F$ 中, 进入 $Y \rightarrow X$ 的生成过程, 使产生的新数据 $x'$ 和最开始的数据 $X$ 尽可能相似, 目的是希望 $F(G(x)) = x$ , 所以Cycle Consistency Loss损失计算式为:

$$\text{Loss}_{\text{cycle}}(G, F) = E_{x \sim p_{\text{data}}(x)} \left[ \|F(G(x)) - x\|_1 \right] + E_{y \sim p_{\text{data}}(y)} \left[ \|G(F(y)) - y\|_1 \right]. \quad (3)$$

将  $\text{Loss}_{\text{GAN}}$  与  $\text{Loss}_{\text{cycle}}$  结合得到式(4):

$$L(G, F, D_X, D_Y) = L_{\text{GAN}}(G, D_Y, X, Y) + L_{\text{GAN}}(F, D_X, Y, X) + \lambda L_{\text{cycle}}(G, F) \quad (4)$$

$\text{Loss}_{\text{identity}}$  损失主要是训练网络的识别能力, 即将真实的  $y$  输入  $X$  生成  $Y$  的判别器中, 查看判别器的识别损失, 损失越小越好, 说明生成器网络真正地理解了  $B$  的结构, 通常记为  $A \rightarrow B(\text{real}B)$ , 同理存在  $B \rightarrow A(\text{real}A)$ , 其计算式为:

$$\text{Loss}_{\text{identity}} = E_{y \sim p_{\text{data}}(y)} \left[ (D_Y(G(y)) - y)_1 \right] + E_{x \sim p_{\text{data}}(x)} \left[ (D_X(G(x)) - x)_1 \right] \quad (5)$$

将上述式(4)与式(5)结合得到 Cycle-GAN 总的损失为:

$$\text{Loss} = L_{\text{GAN}}(G, D_Y, X, Y) + L_{\text{GAN}}(G, D_X, Y, X) + \lambda L_{\text{cycle}}(G, F) + \text{Loss}_{\text{identity}} \quad (6)$$

## 2 解决方案

### 2.1 整体框架

基于改进 Cycle-GAN 的样本数据生成框架如图 2 所示, 主要分为数据预处理阶段、数据生成阶段、神经网络预测模型 3 部分。

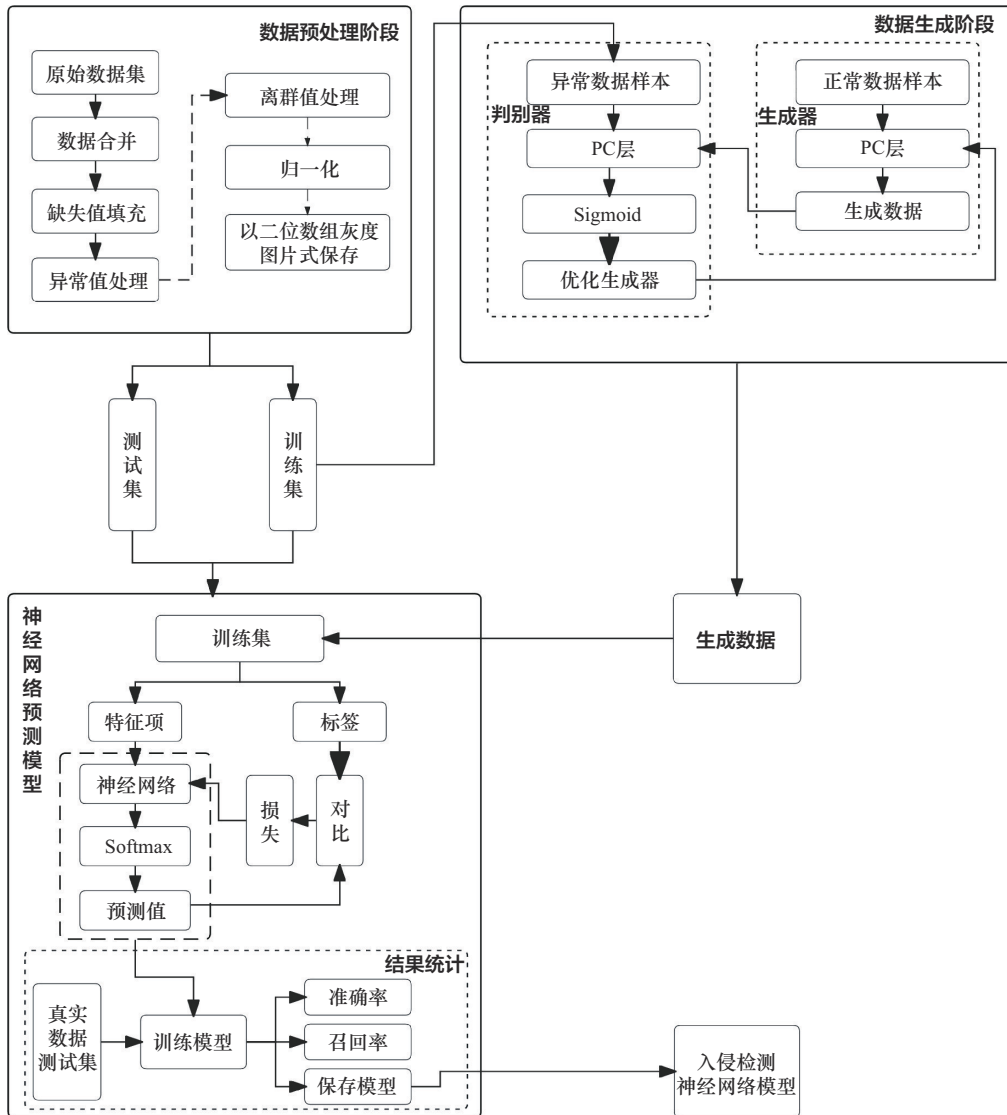


图2 基于改进Cycle-GAN的样本数据生成框架

Figure2 Framework of data generation based on improved Cycle-GAN

## 2.2 模块介绍

### 2.2.1 数据预处理

本文采用的数据集中包含连续值和离散值，且存在异常值、缺失值和离群值，因此需要先对数据集进行处理。数据预处理流程如图3所示，主要包括采用正常值的均值对异常值进行处理、采用线性插值法补充缺失值、基于数据分布的离群值识别和处理。数据集中的字符串特征值通过one-hot编码映射到欧几里得空间，特征距离为1。

为了消除指标之间的量级影响，使各项数据处于同一数量级，有助于模型进行训练。对数据集处理后的特征值利用式(7)进行归一化处理。

$$x^* = \frac{x - \min}{\max - \min} \quad (7)$$

完成数据集预处理后，将数据集划分为80%的训练集和20%测试集，对于CIC-IDS2017数据集，按标签攻击类型将数据集划分为15种。对于NSL-KDD数据集，按标签攻击类型将数据集划分为5种。

### 2.2.2 图数据转换

将高维数据转换为图像形式，以便利用Cycle-GAN的功能。由于数据集已经归一化，将每一个数值乘以255再向下取整，得到整数灰度值，这样有助于将数据转换为图像。Cycle-GAN将固定大小的图像作为输入，所以应该定义图像的最佳大小。以CIC-IDS2017数据集为例，一条数据有79个特征，除去最后一个攻击

类型标签后是78个特征，故可以定义一个9×9的矩阵。将处理后的数据按行方式放入9×9图像中来实现，CIC-IDS2017数据转换效果如图4所示。

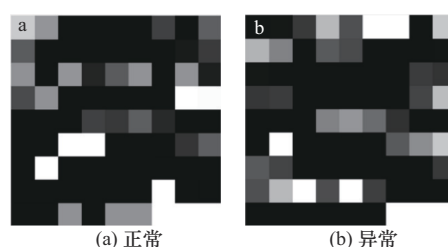


图4 CIC-IDS2017数据转换效果

Figure 4 CIC-IDS2017 data conversion effect

图4中最后3个像素白色部分是为了填充图像而扩展并替换为灰度值255的数据部分，其余不同像素灰度明暗各异是相关属性值取值不同导致的。利用此方法，将高维特征向量转为灰度图形式，可以更好地发挥Cycle-GAN在图像领域的优势。CIC-IDS2017数据经过预处理与图数据转换后的数据量如表1所示。

同样地，NSL-KDD数据集的每一条数据具有43个特征，最后两个特征是标签（正常或攻击）和评分（流量输入本身的严重性），故可以定义一个7×7的矩阵，同样将处理后的数据按行存储，得到如图5所示的效果图。图5中最后8个像素白色部分是为了填充图像而扩展并替换为灰度值255的数据部分，其余不同像素灰度明暗各异是相关属性值取值不同导致的。

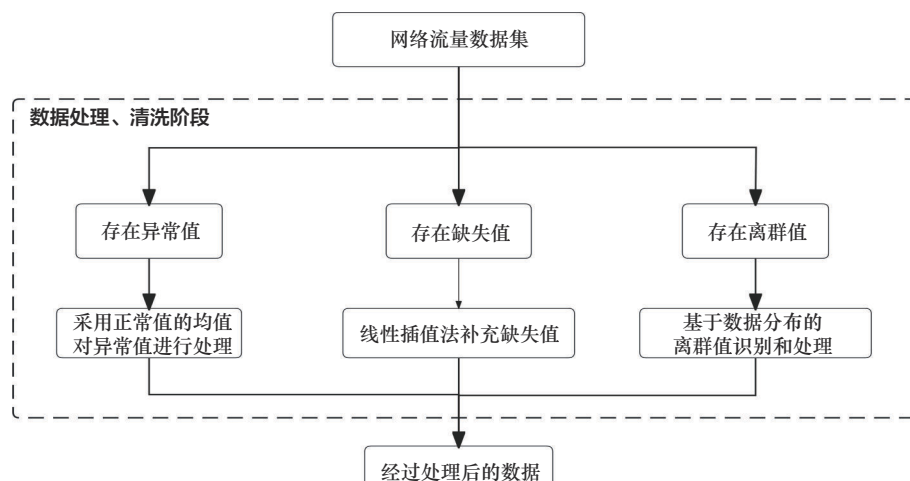


图3 数据预处理流程

Figure3 Data preprocessing workflow

表 1 CIC-IDS2017 数据经过预处理与图数据转换后的数据量

Table 1 CIC-IDS2017 data after preprocessing and graph data conversion

| 数据类型              | 训练集样本/条 | 测试集样本/条 |
|-------------------|---------|---------|
| BENIGN            | 36 352  | 9 089   |
| DDoS              | 20 204  | 5 677   |
| DoS GoldenEye     | 16 633  | 4159    |
| SSH-Patator       | 9 529   | 2 383   |
| Bot               | 3 161   | 791     |
| Infiltration      | 58      | 15      |
| DoS Slowhttptest  | 8 886   | 2 222   |
| DoS Hulk          | 36 819  | 9 205   |
| Heartbleed        | 18      | 5       |
| Web Attack XSS    | 1 053   | 264     |
| PortScan          | 25 414  | 6 354   |
| Web Brute Force   | 2 435   | 609     |
| DoS slowloris     | 9 366   | 2 342   |
| FTP-Patator       | 12 823  | 3 206   |
| Web Sql Injection | 34      | 9       |

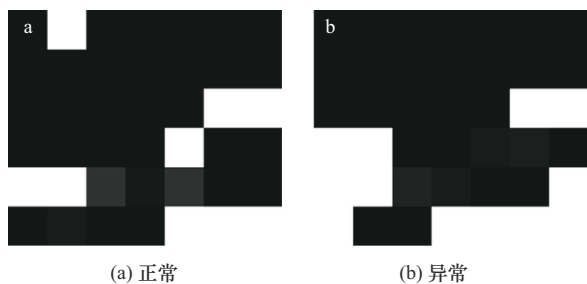


图 5 NSL-KDD 数据转换效果

Figure 5 NSL-KDD data conversion effect

NSL-KDD 数据经过预处理与图数据转换后的数据量如表 2 所示。

表 2 NSL-KDD 数据经过预处理与图数据转换后的数据量

Table 2 NSL-KDD data after preprocessing with graph data conversion

| 数据类型   | 训练集样本/条 | 测试集样本/条 |
|--------|---------|---------|
| BENIGN | 61 637  | 15 409  |
| DoS    | 42 851  | 10 712  |
| PROBE  | 11 253  | 2 813   |
| U2R    | 202     | 50      |
| R2L    | 2 856   | 713     |

### 2.2.3 数据生成

1) 数据准备：使用 CIC-IDS2017、NSL-KDD 数据集并通过预处理方法进行数据准备。

2) 生成过程：从数据样本中取出一定比例的数据作为模板数据。将所占比例较多的数据作为 Cycle-GAN 的  $X$  数据（通常入侵检测数据集中标记为 BENIGN 的类别占多数），将要生成的少数异常数据作为 Cycle-GAN 的  $Y$  数据。Cycle-GAN 的特性使无须关心两类数据之间的匹配关系，因此该网络可以通过判别器  $D_x$ 、 $D_y$  识别  $X$ 、 $Y$  的特征，并通过  $G_x$  将  $X$  转换为  $Y$ ，通过  $G_y$  将  $Y$  转换为  $X$ 。通过判别器  $D_x$ 、 $D_y$  的损失来调整生成器  $G_x$ 、 $G_y$ ，进而使其生成符合要求的数据。从数据生成到合并数据集并输入神经网络的流程如图 6 所示。基于空间注意力的 Cycle-GAN 内部过程如图 7 所示。

3) 合并数据集：将生成的数据与原来不平衡的训练数据结合，从而在数据集的正常类和异常

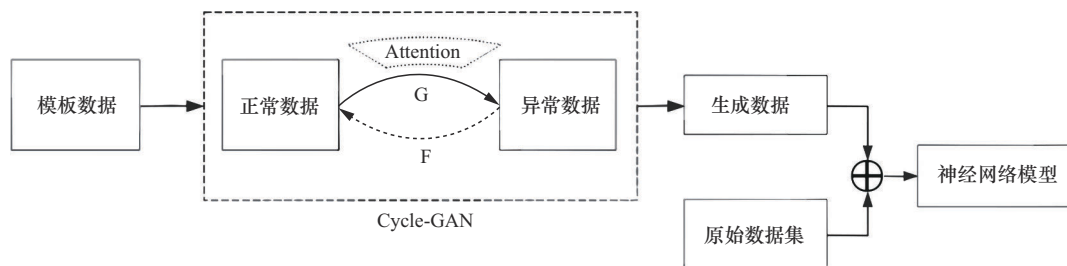


图 6 从数据生成到合并数据集并输入神经网络的流程

Figure 6 Flow from data generation to merging datasets and feeding them into a neural network

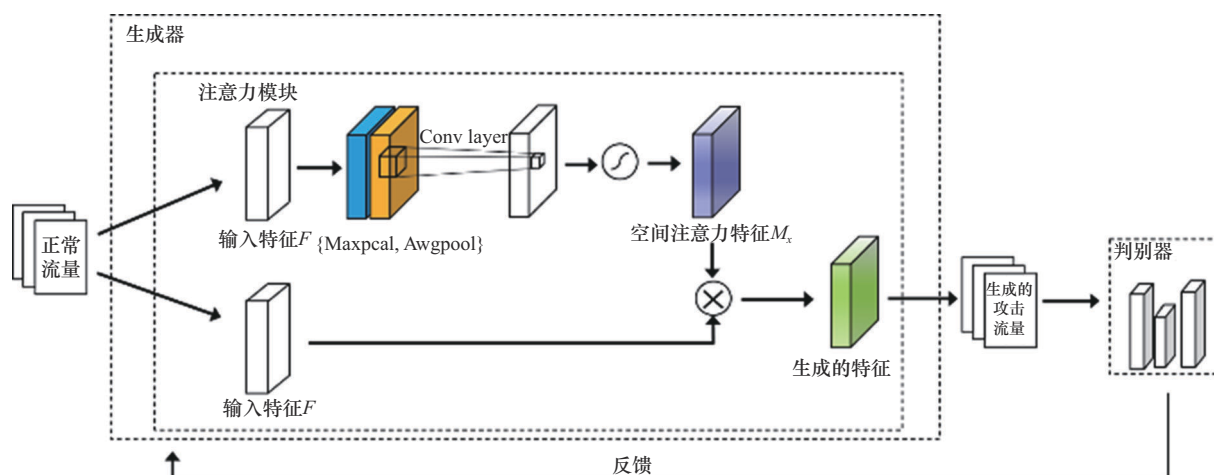


图7 基于空间注意力的Cycle-GAN内部过程

Figure 7 Spatial attention-based Cycle-GAN internal processes

类之间建立平衡。最后，使用平衡数据集来训练神经网络模型，并使用测试集来评估分类。

4) 网络的参数设定：Cycle-GAN模型使用两个全连接神经网络分别作判别器和生成器，判别器和生成器的激活函数采用ReLU，判别器的损失函数使用二元交叉熵，生成器的损失函数包括对抗损失 $Loss_{GAN}$ 和循环一致性损失 $Loss_{cycle}$ ，判别器与生成器的学习率随着训练动态调整，初始为0.1。

#### 2.2.4 数据生成前后指标分析

本文使用基于空间注意力的循环生成对抗网络方法对CIC-IDS2017和NSL-KDD的训练集进行样本数据的扩充，得到扩充后的训练数据集。

在CIC-IDS2017数据集中，攻击类型为Infiltration、Heartbleed、Web Sql Injection的不平衡率分别高达622、1 975、1 056，少样本数据会影响到分类器的综合性能，所以选择不平衡率超过5的类别（Bot、Infiltration、Heartbleed、Web Attack XSS、Web Attack Brute Force、Web Attack Sql Injection）进行扩充，得到CIC-IDS2017数据集平衡前后的数据量，如表3所示。

同样地，在NSL-KDD数据集中，攻击类型为PROBE、U2R、R2L的不平衡率分别为5.5、305.7、21.6，说明这几类样本占比较少，使用相同的方法对NSL-KDD的数据进行扩充，平衡前后的数据量如表4所示，主要对少样本攻击类型数据（PROBE、U2R、R2L）进行扩充。

表3 CIC-IDS2017数据集平衡前后的数据量

Table 3 The quantity of data before and after balancing the CICI-DS2017 dataset

| 攻击类型                            | 平衡前/条  | 平衡后/条    |
|---------------------------------|--------|----------|
| BENIGN                          | 36 352 | 36 352   |
| DDoS                            | 20 204 | 20 204   |
| DoS GoldenEye                   | 16 633 | 16 633   |
| SSH-Patator                     | 9 529  | 9 529    |
| <b>Bot</b>                      | 3 161  | 10 581 ↑ |
| <b>Infiltration</b>             | 58     | 10 064 ↑ |
| DoS Slowhttptest                | 8 886  | 8 886    |
| DoS Hulk                        | 36 819 | 36 819   |
| <b>Heartbleed</b>               | 18     | 10 019 ↑ |
| <b>Web Attack XSS</b>           | 1 053  | 11 173 ↑ |
| PortScan                        | 25 414 | 25 414   |
| <b>Web Attack Brute Force</b>   | 2 435  | 12 712 ↑ |
| DoS slowloris                   | 9 366  | 9 366    |
| FTP-Patator                     | 12 823 | 12 823   |
| <b>Web Attack Sql Injection</b> | 34     | 10 037 ↑ |

表4 NSL-KDD数据集平衡前后的数据量

Table 4 The quantity of data before and after balancing the NSL-KDD dataset

| 攻击类型   | 平衡前/条  | 平衡后/条  |
|--------|--------|--------|
| BENIGN | 61 637 | 61 637 |
| DoS    | 42 851 | 42 851 |
| PROBE  | 11 253 | 26 645 |
| U2R    | 202    | 15 052 |
| R2L    | 2 856  | 15 995 |

### 2.2.5 神经网络预测

在卷积神经网络结构中, 为了提高网络性能, 大多采取增加卷积网络的层数, 但这里存在一个问题, 随着网络层数的提升, 可能会出现梯度消失和梯度爆炸。因此, 在残差网络出现之前, 单纯增加深度的模型层次可能会导致更高的训练误差。

残差网络的思想是通过跳跃连接的方式将低层信息传递到高层, 以此解决网络退化的问题。借助残差网络, 神经网络的层数得以加深, 基于残差网络在图像分割、图像识别、目标检测等领域取得的显著成果, 本文使用 ResNet50 神经网络对原始数据集和平衡后的数据集分别进行预测, 得到预测结果。这里输入神经网络模型的数据集已经是经过转换后的图数据格式, 并且是使用 Cycle-GAN 进行数据增强后的样本数据, 主要步骤如下。

1) 将图片数据作为神经网络模型的特征输入, 其类型作为标签。

2) 数据通过神经网络的计算得到一个预测值, 并将其与标签值进行计算得到 loss, 根据所得的 loss 反向传播调整神经网络模型的参数, 使模型的 loss 值稳定。

3) 得到经过平衡数据集训练后的神经网络模型, 使用该模型对原始数据集进行预测, 得到各项评估指标并进行对比。

### 2.2.6 性能评估

在平衡训练数据集之后, 为了评估分类结果, 使用准确率、精确率、召回率及  $F1$  值作为对结果的评价指标。

## 3 实验与分析

### 3.1 实验数据

机器学习算法性能的优劣, 一般需要在常用公共数据集上来评价。在入侵检测领域中, 常用的数据集有 CIC-IDS2017、NSL-KDD、KDD99 数据集, 本文实验使用 CIC-IDS2017 和 NSL-KDD 数据集。

#### (1) CIC-IDS2017 数据集简介

CICIDS-2017 是由通信安全机构与加拿大网络安全研究所合作项目采集的入侵检测数据集, 共约 500 万条数据, 每条记录包含 79 个特征, 其中 78 个特征指的是流量特性 (分别对应目的端

口、源端口等), 最后 1 个是标签 (正常或攻击), 其中正常数据的占比远远大于异常数据的占比 (达到 4:1)。攻击类型包括暴力 FTP、暴力 SSH、DoS、Heartbleed、Web 攻击、渗透、僵尸网络和 DDoS, 细分为 14 种攻击类型和正常类型。本文取其中部分数据集作为样本集, 表 5 为 CIC-IDS2017 数据集的样本数据示例。

表 5 CIC-IDS2017 数据集的样本数据示例

Table 5 Example of CIC-IDS2017 dataset sample data

| 特征                                                                                                                                                                                                                                                                                                                                         | 标签       |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| 54 586, 3, 2, 0, 12, 16, 84, 363, 0, 0, 0, 0, 0, 1, 0, 0, 0, 40, 0, 0, 0, 30, 0, 0, 1, 1, 0.00, 0.00, 0.00, 0.00, 1.00, 1.50, 0.00, 104, 66, 0.63, 0.03, 0.01, 0.00, 0.00, 0.00, 0.00, 0.00, 4 444, 1, 223, 0, 0, 0.5, 441, 33, 1, 21, 0, 0, 0, 81, 5521, 0.61, 0.99, 1, 2, 0, 0, 0, 0.22, 16, 66, 73, 811, 1.00, 21, 108, 22, 0, 0, 11, 2 | BENIGN   |
| 3000, 43, 1, 1, 2, 6, 2, 2, 2, 0, 6, 6, 6, 0, 186046.5116, 46511.62791, 43, 0, 43, 43, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 20, 23 255.813 95, 23 255.813 95, 2, 6, 3.45233, 2.309 401 077, 5.331 333, 0, 0, 0, 1, 0, 0, 0, 0, 1, 5, 2, 6, 24, 0, 0, 0, 0, 0, 0, 1, 2, 1, 6, 1024, 0, 0, 24, 0, 0, 0, 0, 0, 0, 0, 0               | PortScan |

#### (2) NSL-KDD 数据集简介

NSL-KDD 是 KDD99 数据集的修订版本, 该数据集每条记录包含 43 个特征, 其中 41 个特征指的是流量输入本身, 最后 2 个是标签 (正常或攻击) 和评分数 (流量输入本身的严重性)。该数据集存在 4 种攻击类型: 拒绝服务 (DoS)、探测、用户到根 (U2R) 和远程到本地 (R2L)。本文采用 NSL-KDD 全部数据集作为样本集, 表 6 为 NSL-KDD 数据集的样本数据示例。

### 3.2 实验评价标准

本文实验使用准确率、精确率、召回率和  $F1$  值作为评价指标。

### 3.3 实验模型参数

#### (1) 入侵检测神经网络模型参数

神经网络分类模型使用 ResNet50 模型。模型使用交叉熵损失函数, 激活函数选取 ReLU, 梯度优化器使用 Adam 优化器, 初始学习率使用 0.1 并动态下降, 训练次数为 2 000 次。

表6 NSL-KDD数据集的样本数据示例

Table 6 Example of NSL-KDD dataset sample data

| 特征                                                                                                                                                                                                           | 标签     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| 2.0, 12983.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 1.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 134.0, 86.0, 0.61, 0.04, 0.61, 0.02, 0.0, 0.0, 0.0, 0.0, 0.0 | BENIGN |
| 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 229.0, 10.0, 0.0, 0.0, 1.0, 1.0, 0.04, 0.06, 0.0, 255.0, 10.0, 0.04, 0.06, 0.0, 0.0, 0.0, 0.0, 1.0, 1.0, 1.0  | DoS    |

## (2) Cycle-GAN 模型参数

Cycle-GAN模型使用两个全连接神经网络分别作判别器和生成器，判别器和生成器的激活函数使用Leaky\_ReLU，判别器的损失函数使用二元交叉熵，生成器的损失函数包括对抗损失 $Loss_{GAN}$ 和循环一致性损失 $Loss_{cycle}$ ，判别器与生成器的学习率随着训练动态调整，初始为0.1。

## 3.4 实验结果与分析

### 3.4.1 数据生成阶段的 loss

对Cycle-GAN进行100轮生成过程的损失分析，得到如图8所示的生成器 $G_x$ 与 $G_y$ 以及判别器 $D_x$ 与 $D_y$ 的损失曲线。其中， $G_x$ 代表从正常样本数据到异常样本的生成过程( $X \rightarrow Y$ )， $G_y$ 为异常样本数据到正常样本数据的生成过程( $Y \rightarrow X$ )， $D_x$ 是正常样本( $X$ )的判别器， $D_y$ 是异常样本( $Y$ )的判别器。下面对损失值具体分析。

1)  $G_x$ 损失逐步下降：通过观察 $G_x$ 的损失曲线，可以看到在经过40轮训练后， $G_x$ 的损失逐步下降。这表明Cycle-GAN在训练过程中逐渐学到生成符合要求的异常数据，为后续的入侵检测系统提供了有效的数据增强。

2)  $G_x$ 损失波动性较大：首先，GAN的训练过程本质上是不稳定的，生成器和判别器的损失函数在训练过程中出现波动属于正常现象；其次，学习率、批次大小等超参数设置不当可能导致生成器损失波动；第三，训练数据分布不均匀，生成器可能难以学习到数据的真实分布，也会导致损失波动。与 $G_y$ 相比， $G_x$ 的损失呈现更大的波动性，可能的原因是异常数据在原始数据集中的分布较少，导致 $G_x$ 能够学到的特征相对较少，因此生成器的损失值更不稳定。尽管波动较大，但整体趋势仍然是向下的，表明生成器在训练中逐渐适应生成异常数据的

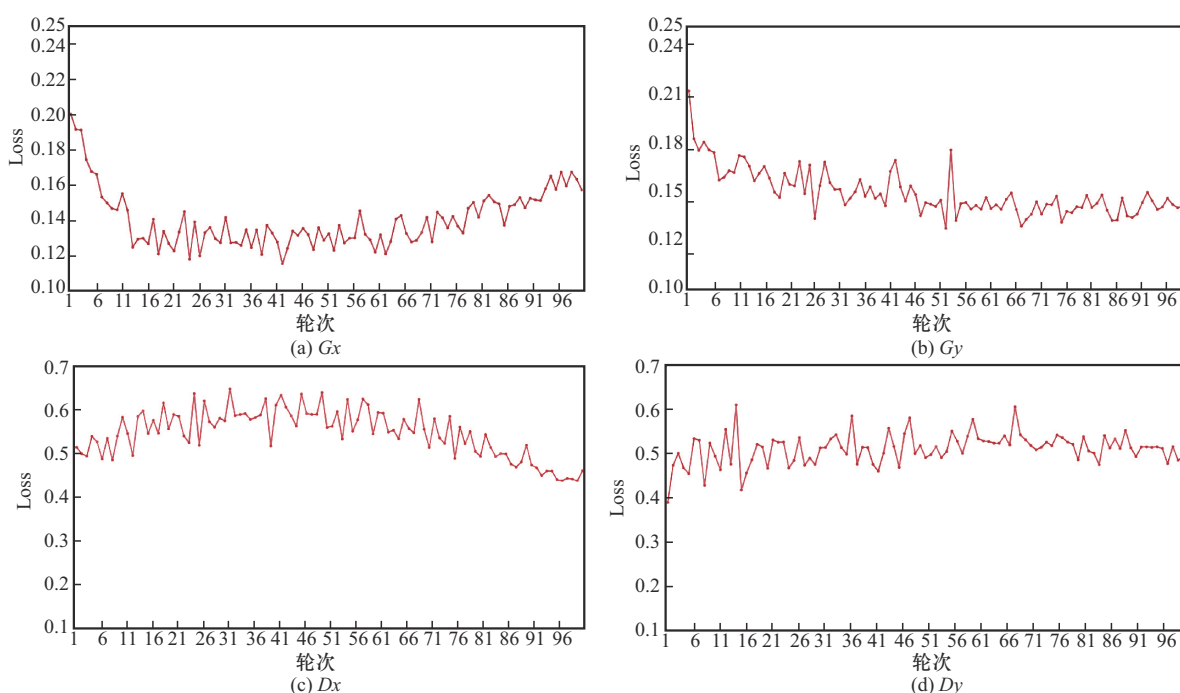


图8 损失值  
Figure 8 Loss value

特征。为了减少生成器  $G$  的损失波动, 后期可以采取不同的优化算法、调整学习率、梯度惩罚等措施予以优化。

3) 判别器  $D_x$  和  $D_y$  的损失波动相似: 在损失曲线中, 判别器  $D_x$  和  $D_y$  的损失呈现与  $G_x$ 、 $G_y$  之间相似的波动趋势。这表明判别器在学习识别生成的异常数据和原始正常数据之间的差异, 不断调整以提高鉴别性能。

综上所述, Cycle-GAN 生成数据的损失分析结果表明, 在 100 轮的训练中, 生成器逐步学到了生成符合要求的异常数据, 虽然  $G_x$  的损失波动较大, 但整体趋势是向下的。这进一步验证了 Cycle-GAN 作为数据增强手段的有效性, 为入侵检测分类器的训练提供了更多样本和更全面的数据特征, 有望提高分类模型的性能和鲁棒性。

### 3.4.2 CIC-IDS2017 数据集上的结果分析

#### (1) 使用 CIC-IDS2017 原始数据集

使用 ResNet50 在原始的训练集上训练, 并在测试集上分类, 得到初步的实验结果, CIC-IDS2017 原始数据集性能指标如表 7 所示, 加粗类别为在数据集中占比较少、不平衡率较高的类别, 对应的混淆矩阵如图 9 所示。

表 7 CIC-IDS2017 原始数据集性能指标

Table 7 CIC-IDS2017 raw dataset performance metrics

| 攻击类型              | 精确率          | 召回率          | F1 值         | 准确率     |
|-------------------|--------------|--------------|--------------|---------|
| BENIGN            | 1.000        | 1.000        | 1.000        | 1.000   |
| DDoS              | 1.000        | 0.957        | 0.978        | 0.995   |
| DoS GoldenEye     | 0.998        | 0.998        | 0.998        | 1.000   |
| SSH-Patator       | 0.992        | 0.995        | 0.993        | 0.999   |
| Bot               | 1.000        | 0.998        | 0.999        | 1.000   |
| Infiltration      | <b>0.054</b> | 0.870        | <b>0.102</b> | 0.994   |
| DoS Slowhttptp    | 0.993        | 0.984        | 0.988        | 0.999   |
| DoS Hulk          | 0.999        | 0.996        | 0.998        | 0.999   |
| Heartbleed        | <b>0.265</b> | <b>0.300</b> | <b>0.281</b> | 1.000   |
| Web Attack XSS    | 0.953        | 0.972        | 0.962        | 1.000   |
| PortScan          | 0.994        | 1.000        | 0.997        | 1.000   |
| Web Brute Force   | 0.992        | 1.000        | 0.996        | 1.000   |
| DoS slowloris     | 0.995        | 0.990        | 0.993        | 0.999   |
| FTP-Patator       | 0.997        | 0.995        | 0.996        | 0.999   |
| Web Sql Injection | <b>0.421</b> | <b>0.667</b> | <b>0.516</b> | 1.000   |
| 平均值               | 0.843 5      | 0.914 8      | 0.853 2      | 0.991 3 |

#### (2) 使用基于空间注意力的循环生成对抗网络方法平衡后的 CIC-IDS2017 数据集

使用改进后的循环生成对抗网络方法生成数据后, 使用 ResNet50 在平衡后的训练集上训练, CIC-IDS2017 平衡数据集性能指标如表 8 所示, 对应的混淆矩阵如图 10 所示。

表 8 CIC-IDS2017 平衡数据集性能指标

Table 8 CIC-IDS2017 balanced dataset performance metrics

| 攻击类型              | 精确率     | 召回率     | F1 值    | 准确率     |
|-------------------|---------|---------|---------|---------|
| BENIGN            | 1.000   | 1.000   | 1.000   | 1.000   |
| DDoS              | 1.000   | 1.000   | 1.000   | 1.000   |
| DoS GoldenEye     | 1.000   | 1.000   | 1.000   | 1.000   |
| SSH-Patator       | 1.000   | 0.999   | 0.999   | 1.000   |
| Bot               | 1.000 ↑ | 0.998 ↑ | 0.999 ↑ | 1.000   |
| Infiltration      | 1.000 ↑ | 1.000 ↑ | 1.000 ↑ | 1.000   |
| DoS Slowhttptest  | 0.994   | 1.000   | 0.997   | 1.000   |
| DoS Hulk          | 1.000   | 1.000   | 1.000   | 1.000   |
| Heartbleed        | 1.000 ↑ | 0.800 ↑ | 0.889 ↑ | 1.000   |
| Web Attack XSS    | 0.966 ↑ | 0.993 ↑ | 0.979 ↑ | 1.000   |
| PortScan          | 1.000   | 0.999   | 0.999   | 1.000   |
| Web Brute Force   | 1.000   | 0.979   | 0.989   | 1.000   |
| DoS slowloris     | 0.997   | 0.995   | 0.996   | 1.000   |
| FTP-Patator       | 1.000   | 1.000   | 1.000   | 1.000   |
| Web Sql Injection | 0.711   | 1.000 ↑ | 0.831   | 1.000   |
| 平均值               | 0.977 8 | 0.984 2 | 0.978 6 | 0.999 1 |

使用 Cycle-GAN 生成数据的实验结果表明, 增强后的数据集在各性能指标上取得了显著提升。对比原始数据集, 使用基于空间注意力的 Cycle-GAN 生成的数据在多数攻击类型上取得了平衡和较好鲁棒性。对于占比过少的样本 (如 Infiltration 和 Heartbleed) 的识别较为准确, 两者的召回率提升巨大, 其值分别为 1.0 和 0.800。

通过对比以上结果, 可以观察到以下效果。

1) 模型整体性能提升: 通过对比平衡数据集之前和之后的结果, 可以观察到模型整体性能显著提升。平衡数据集后, 模型的平均召回率从 0.914 8 提高到 0.984 2、F1 值从 0.853 2 提高到 0.978 6 (如表 9 所示), 说明平衡数据集有助于

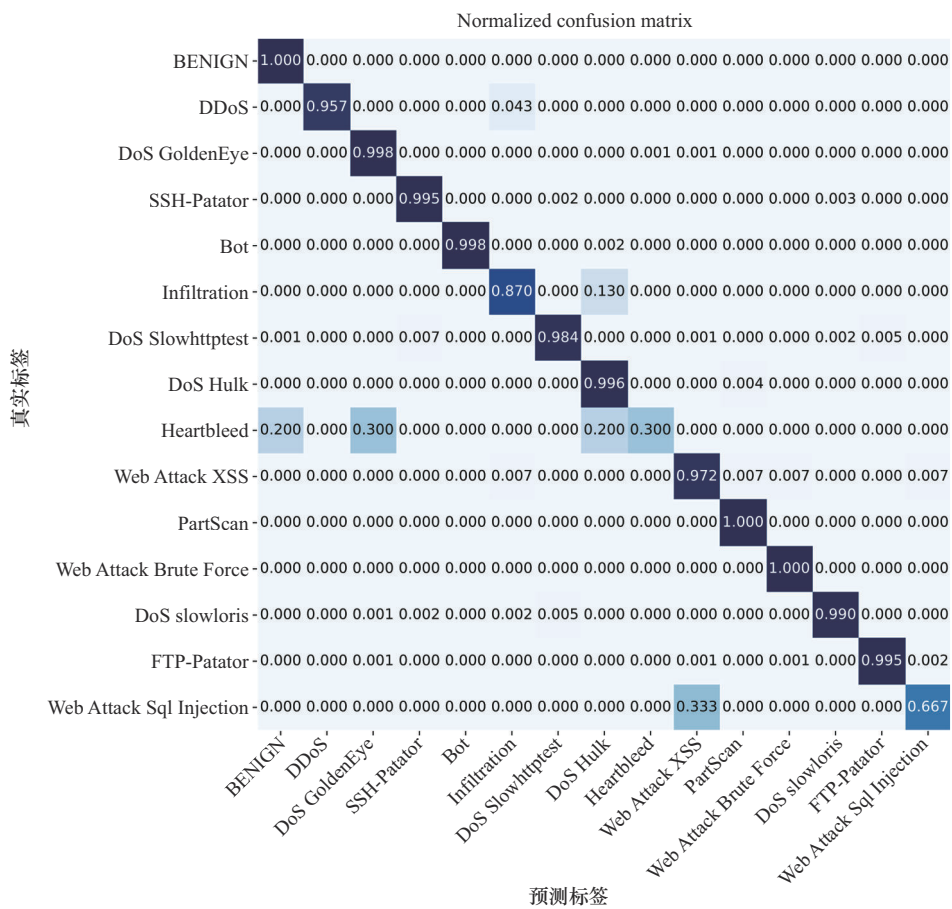


图9 CIC-IDS2017原始数据测试集的混淆矩阵

Figure 9 Confusion matrix for the CIC-IDS2017 raw data test set

模型更好地学习少数类别的特征，从而改善各种攻击类型的分类效果。

表9 CIC-IDS2017数据集结果汇总  
Table 9 Summary of results from the  
CIC-IDS2017 dataset

| 数据形式  | 精确率            | 召回率            | F1 值           | 准确率            |
|-------|----------------|----------------|----------------|----------------|
| 原始数据集 | 0.843 5        | 0.914 8        | 0.853 2        | 0.991 3        |
| 平衡数据集 | <b>0.977 8</b> | <b>0.984 2</b> | <b>0.978 6</b> | <b>0.999 1</b> |

2) 对少数类别的改善：在平衡数据集之前，一些攻击类型如 Infiltration、Heartbleed 的  $F1$  值过低，分别为 0.102 和 0.281。而在使用 SA-Cycle-GAN 平衡数据集并在 ResNet50 上训练，这些攻击类型的召回率和  $F1$  值得到了显著提升，Infiltration、Heartbleed 的  $F1$  值分别提升到 1.0 和 0.889。

### 3.4.3 NSL-KDD 数据集结果分析

#### (1) 使用 NSL-KDD 原始数据集

在原始的 NSL-KDD 数据集上使用 ResNet50 进行训练分类，得到该数据的初步结果，NSL-KDD 原始数据集性能指标如表 10 所示，其混淆矩阵如图 11 所示。

表10 NSL-KDD原始数据集性能指标  
Table 10 NSL-KDD raw dataset performance metrics

| 攻击类型   | 精确率          | 召回率          | F1 值         | 准确率     |
|--------|--------------|--------------|--------------|---------|
| BENIGN | 1.000        | 1.000        | 1.000        | 1.000   |
| DoS    | 0.962        | 0.881        | 0.920        | 0.944   |
| PROBE  | 0.728        | 0.756        | 0.742        | 0.950   |
| U2R    | <b>0.099</b> | <b>0.396</b> | <b>0.158</b> | 0.993   |
| R2L    | <b>0.314</b> | <b>0.597</b> | <b>0.412</b> | 0.959   |
| 平均值    | 0.620 5      | 0.726 0      | 0.646 2      | 0.923 2 |

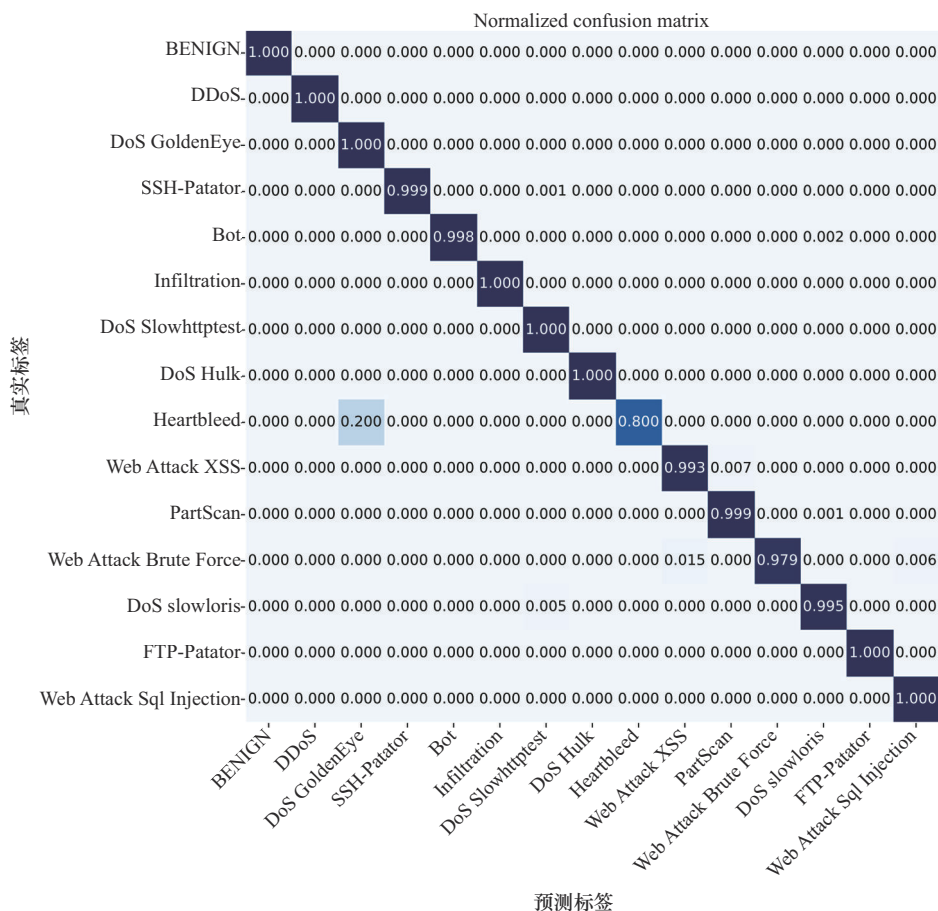


图 10 CIC-IDS2017 平衡数据集测试集的混淆矩阵

Figure 10 Confusion matrix for the CIC-IDS2017 balanced dataset test set

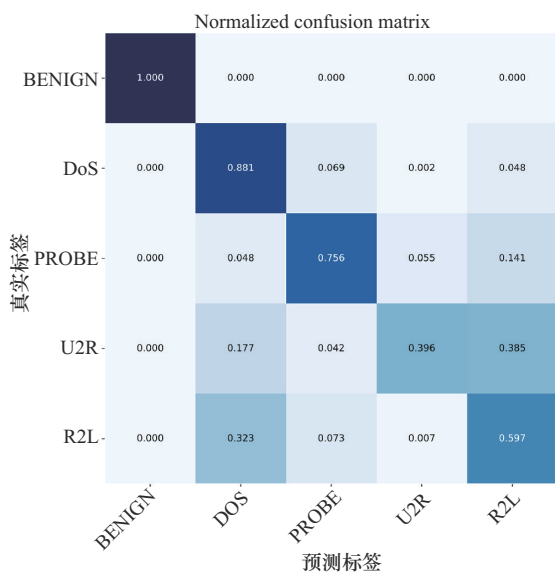


图 11 NSL-KDD 原始数据测试集上的混淆矩阵

Fig.11 Confusion matrix on the NSL-KDD raw data test set

(2) 使用循环生成对抗网络方法平衡后的 NSL-KDD 数据集

使用基于空间注意力的循环生成对抗网络生成数据后得到平衡数据集，将其作为训练集来训练 ResNet50，NSL-KDD 平衡数据集性能指标如表 11 所示，对应的混淆矩阵如图 12 所示。

表 11 NSL-KDD 平衡数据集性能指标

Table 11 NSL-KDD balanced dataset performance metrics

| 攻击类型   | 精确率     | 召回率     | F1 值    | 准确率     |
|--------|---------|---------|---------|---------|
| BENIGN | 0.996   | 1.000   | 0.998   | 0.998   |
| DoS    | 0.963   | 0.944   | 0.954   | 0.967   |
| PROBE  | 0.826   | 0.848 ↑ | 0.837 ↑ | 0.969   |
| U2R    | 0.624 ↑ | 0.851 ↑ | 0.720 ↑ | 0.999   |
| R2L    | 0.687 ↑ | 0.742 ↑ | 0.713 ↑ | 0.986   |
| 平均值    | 0.819 3 | 0.876 8 | 0.844 3 | 0.958 9 |

使用基于空间注意力的 Cycle-GAN 对 NSL-KDD 数据集进行平衡后，通过训练 ResNet50 得

到的结果表明,整体性能相对原始数据集有了较大提升。整体的召回率和 $F1$ 值得到了提升,整体的 $F1$ 值从0.646 2提升到0.844 3。在一些性能表现的攻击类型上,如U2R和R2L,模型在平衡后的数据集上表现更为出色,U2R和R2L的 $F1$ 值分别为0.720和0.713,相较原始数据集有了显著提升。

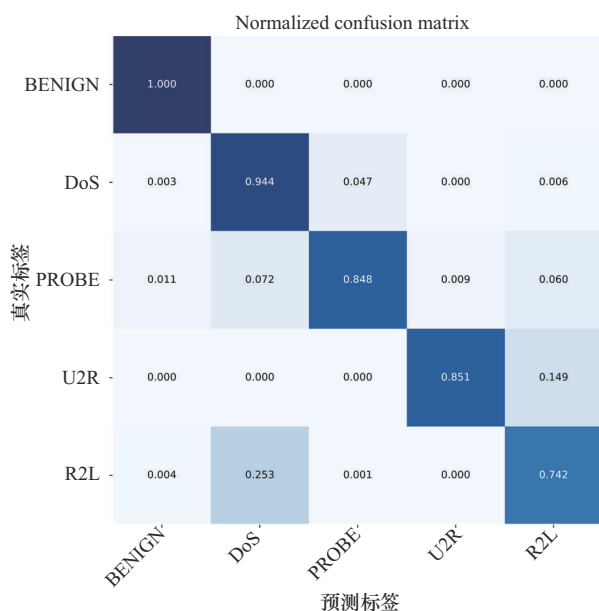


图12 NSL-KDD平衡数据集测试集的混淆矩阵

Fig.12 Confusion matrix for NSL-KDD balanced dataset test set

对比以上结果,通过SA-Cycle-GAN进行样本扩充得到的平衡数据集,整体性能表现取得了显著提升。在平衡数据集后,召回率从0.726提升到0.876 8, $F1$ 从0.646 2提升到0.8443,准确率从0.923 2提升到0.958 9(如表12所示)。

本文基于循环生成对抗网络的数据扩充方法成功地改善了模型对于不平衡和复杂数据集的分类性能,为进一步提升入侵检测系统的效果提供了有效手段。

### 3.4.4 横向对比结果

为检验综合性能,采用CIC-IDS2017和NSL-KDD数据集对实验结果进行横向比较,如表13所示。

表12 NSL-KDD数据集结果汇总

Table 12 Summary of results for the NSL-KDD dataset

| 数据形式  | 精确率     | 召回率     | $F1$ 值  | 准确率     |
|-------|---------|---------|---------|---------|
| 原始数据集 | 0.620 5 | 0.726 0 | 0.646 2 | 0.923 2 |
| 平衡数据集 | 0.819 3 | 0.876 8 | 0.844 3 | 0.958 9 |

## 4 结束语

随着机器学习技术在各领域的深入应用,研究者开始探索将其融入入侵检测领域。特别是,深度学习技术已经成为构建高效入侵检测系统的核心技术之一。但目前广泛部署的入侵检测系统仍面临着较高的误报率和漏报率。本文专注于探讨入侵检测数据不平衡问题对模型性能产生的负面影响,旨在提出相应的解决策略。

针对数据不平衡导致模型性能下降问题,本文提出了基于空间注意力机制的循环生成对抗网络方法来生成数据。本文还对非数值进行转换、缺失值进行填充,对数据进行归一化处理,并将处理好的高维数据存到二维数组中进而转换为灰

表13 分类结果比较

Table 13 Comparison of classification results

| 数据集         | 方法                                     | 精确率            | 召回率            | $F1$ 值         | 准确率            |
|-------------|----------------------------------------|----------------|----------------|----------------|----------------|
| CIC-IDS2017 | ANN <sup>[18]</sup>                    | 0.645 0        | 0.500 0        | 0.437 5        | 0.502 5        |
|             | EP-LSTM <sup>[19]</sup>                | 0.921 0        | 0.912          | 0.776 0        | 0.904 0        |
|             | MLP <sup>[20]</sup>                    | 0.956 2        | 0.978 0        | 0.967 0        | 0.986 0        |
|             | Cycle-GAN增强后                           | <b>0.977 8</b> | <b>0.984 2</b> | <b>0.978 6</b> | <b>0.999 1</b> |
| NSL-KDD     | ANN <sup>[21]</sup>                    | 0.823 6        | 0.592 4        | 0.632 6        | 0.799 0        |
|             | 2-class Classification <sup>[22]</sup> | <b>0.965 6</b> | 0.637 3        | 0.768 0        | 0.780 6        |
|             | AE+logistic classifier <sup>[23]</sup> | 0.846 0        | <b>0.928 0</b> | <b>0.887 0</b> | 0.872 0        |
|             | Cycle-GAN增强后                           | 0.819 3        | 0.876 8        | 0.844 3        | <b>0.958 9</b> |

度图的形式,以便作为Cycle-GAN的输入。在数据生成阶段,将占比较多的正常数据取出一部分作为模板数据,通过Cycle-GAN的生成器来得到占比较少的数据,以达到扩充数据集的目的。在生成器中引入注意力机制,保留数据流量的特征,并对关键特征信息抽取,使模型更有效地生成数据。

最后,在CIC-IDS2017和NSL-KDD数据集上进行了实验,本文提出的基于空间注意力的循环生成对抗网络在入侵检测领域多分类的具体效果如下。

1) 数据集性能综合提升:在CIC-IDS2017数据集上,准确率从0.958 7增至0.999 1,精确率从0.826 4增至0.977 8,召回率从0.915 增至0.984 2,  $F1$  值从0.816 7提高至0.978 6。在NSL-KDD数据集上,准确率从0.925 增至0.958 9,精确率从0.603 1增至0.819 3,召回率从0.632 2增至0.876 8,  $F1$  值从0.609 3提高至0.844 3。

2) 提升某些攻击类型检测性能:对抗生成方法对不同攻击类型的适应性良好,尤其对于NSL-KDD数据中少数类别攻击类型(如U2R)的性能提升明显。生成对抗网络在样本扩充中的应用不仅仅提升整体性能,更在处理少数类别和复杂攻击场景上展现了优越性。

3) 增强隐私保护:采用对抗生成网络的样本扩充方法不仅提高了整体性能,还解决了敏感性数据无法公开的问题,为实际入侵检测系统的部署提供了强大的解决方案。

本文在入侵检测领域有较好的表现,但在模型训练时间、数据生成的有效率等方面还是有不足之处,下一步研究可进一步优化对抗生成网络模型,对内部的生成器和判别器结构进行修改,简化内部模型结构。对于提高数据生成的有效性,可以对一些性能表现不佳的攻击类型进行深入研究,在预处理阶段采用较好的特征工程对数据进行清洗,并可以获取网络流数据,对更原始的数据集进行分析,提取其中可以帮助模型训练的特征。本文的数据扩充方法也可以应用于其他入侵检测数据集,甚至应用于其他领域,验证其在不同场景下的通用性和稳定性。

## 参考文献:

- [1] GU X W, HOWELLS G, YUAN H Y. A soft prototype-based autonomous fuzzy inference system for network intrusion detection[J]. Information Sciences, 2024, 677:1-19.
- [2] YUAN J R, QIN G H, et al. DGIDS: Dynamic graph-based intrusion detection system for CAN[J]. Computers&Security, 2024,147:1-14.
- [3] FLORA J, ANTUNES N. Evaluating intrusion detection for microservice applications: Benchmark,dataset, and case studies[J]. The Journal of Systems and Software, 2024,216:1-16.
- [4] LI H, CHEN W, et al. Enhancing IoT security: A comparative study of feature reduction techniques for intrusion detection system[J]. Intelligent Systems with Applications, 2024,23:1-40.
- [5] OUYANG Y, LI B, KONG Q, et al. FS-IDS: A novel few-shot learning based intrusion detection system for SCADA networks[J]. IEEE Transactions on Computers, 2021.
- [6] LUIGI F M, PAULO F A, DIVANILSON R C. Multi-stage deep learning-based intrusion detection system for automotive Ethernet networks[J]. Ad Hoc Networks,2024,162:1-15.
- [7] 吴越, 曹国彦. 面向工业传感网络的时间序列异常检测综述[J]. 网络与信息安全学报, 2024, 10(4):17-36.  
WU Y, CAO G Y. Survey of time series anomaly detection for industrial sensor networks[J]. Chinese Journal of Network and Information Security, 2024, 10(4):17-36.
- [8] MARTINS I, RESENDE J S, et al. Host-based IDS: A review and open issues of an anomaly detection system in IoT[J]. Future Generations Computer systems:FGCS, 2022,133:95-133.
- [9] HUANG Y, MA M. ILL-IDS:S An incremental lifetime learning IDS for VANET[J]. Computers&Security, 2023, 124: 28-42.
- [10] WANG Z W, SHE Q, et al. Generative adversarial networks in computer vision: A survey and taxonomy[J]. arXiv Preprint arXiv: 1906.01529, 2019.
- [11] ZHAO X X, FOK K W, et al. Enhancing network intrusion detection performance using generative adversarial networks[J]. Computers & Security, 2024,145:1-12.
- [12] RADFORDA, METZ L, CHINTALA S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. Computer Science, 2015.
- [13] ARJOVSKY M, CHINTALA S L, et al. Wasserstein generative adversarial networks[C]//Proceedings of Machine Learning Reseach (PMLR). 2017.
- [14] XIE H, LI Q, LAU R Y K, et al. Least squares generative adversarial networks[J]. IEEE Transactions on Computers, 2016.
- [15] ZHU J Y, PARK T, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[J]. IEEE Transactions on Computers, 2017.
- [16] TERMOS M, GHALMANE Z, et al. GDLC: A new graph deep learning framework based on centrality measures for intrusion detection in IoT networks[J].Internet of Things, 2024,26:1-21.
- [17] HELIO N. CUNHA NETO, et al. FedSBS: federated-learning participant-selection method for intrusion detection systems[J]. Computer Networks, 2024,244:1-17.
- [18] KAMIL Z, YUSOF R, et al. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 data-

set[J]. IEEE Access, 2021.

- [19] LEE J, KIM J, et al. Cyber threat detection based on artificial neural networks using event profiles[J]. IEEE Access, 2019.
- [20] GORYUNOV M N, MATSKEVICH A G, RYBOLOVEV D. Synthesis of a machine learning model for detecting computer attacks based on the CICIDS2017 dataset[J]. Proceedings of the Institute for System Programming of RAS, 2017, 32(5):81-94
- [21] GURUNG S, GHOSE M, SUBEDI A. Deep learning approach on network intrusion detection system using NSL-KDD dataset[J]. International Journal of Computer Network and Information Security, 2019.
- [22] HOUDA J, MOHAMED I K. Adversarial machine learning for network intrusion detection: A comparative study[J]. Computer Networks, 2022, 214:1-14.
- [23] WANG L H, DAI Q, et al. Lightweight intrusion detection model based on CNN and knowledge distillation[J]. Applied Soft Computing Journal, 2024, 165:1-15.

#### [作者简介]



曾庆鹏（1974-），男，江西吉水人，南昌大学副教授，主要研究方向为网络与信息安全、信息检索与数据挖掘。



郭航恺（1998-），男，辽宁沈阳人，南昌大学硕士生，主要研究方向为网络安全、数据分析与数据增强。