计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(8):162−170 [doi: 10.15888/j.cnki.csa.009194]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

融合 GRU 和 CNN 的轻量级网络入侵检测模型①

周　璨, 杨　栋, 魏松杰

(南京理工大学 计算机科学与工程学院, 南京 210094)
通信作者: 魏松杰, E-mail: swei@njust.edu.cn

摘　要: 当前网络流量数据呈现出高维、多态、海量的特点, 这对入侵检测是一个新挑战. 针对传统入侵检测模型中检测效率低、缺乏轻量化考虑等局限性, 提出了一种融合 GRU 和 CNN 的轻量级网络入侵检测模型. 首先使用极度随机树删除数据集中的冗余特征; 其次使用 GRU 进行特征提取. 考虑到数据中的长短期依赖关系, 将所有隐藏层输出作为序列特征信息进行下一步处理; 再通过带有逆残差、深度可分离卷积、空洞卷积等结构的轻量化 CNN 模型进行空间特征提取; 为了加速模型收敛加入了通道注意力机制. 最后在 CIC-IDS2017 数据集上的实验表明, 该方法具有优秀的检测性能, 同时也具有模型参数量少、模型体积小、训练时间短、检测时间短等优点, 适用于网络流量的入侵检测工作.

关键词: 网络入侵检测; 门控循环单元; 卷积神经网络; 轻量级模型; 极度随机树

引用格式: 周璨,杨栋,魏松杰.融合 GRU 和 CNN 的轻量级网络入侵检测模型.计算机系统应用,2023,32(8):162–170. http://www.c-s-a.org.cn/1003-3254/9194.html

Integrating GRU and CNN for Light-weighted Model in Network Intrusion Detection

ZHOU Can, YANG Dong, WEI Song-Jie

(School of Computer Science and Engineering, Nanjing University of Science & Technology, Nanjing 210094, China)

Abstract: Current network traffic data show high-dimensional, polymorphic, and massive characteristics, which is a new challenge for intrusion detection. In order to address the limitations of low detection efficiency and lack of lightweight consideration in traditional intrusion detection models, a lightweight network intrusion detection model incorporating GRU and CNN is proposed. Firstly, redundant features in the dataset are removed by using extremely randomized trees. Secondly, feature extraction is performed by using GRU. By taking into account the long and short-term dependencies in the data, all hidden layer outputs are treated as sequence feature information for the next step; then a lightweight CNN model with structures such as inverse residual, depthwise separable convolution, and dilated convolution are used for spatial feature extraction; a channel attention mechanism is added to accelerate model convergence. Finally, experiments on the CIC-IDS2017 dataset show that the method has excellent detection performance, as well as the advantages of few model parameters, small model size, short training time, and short detection time, which is suitable for intrusion detection of network traffic.

Key words: network intrusion detection; gated recurrent unit (GRU); convolutional neural network (CNN); lightweight models; extremely randomized trees

① 基金项目: 工信部 2020 年工业互联网创新发展工程 (TC200H01V); 国家自然科学基金 (61802186)
收稿时间: 2023-01-05; 修改时间: 2023-02-03, 2023-02-27; 采用时间: 2023-03-14; csa 在线出版时间: 2023-06-09
CNKI 网络首发时间: 2023-06-13

飞速发展的互联网给人们带来了便捷的服务, 同时也面临着各种各样的安全问题. 比如 2016 年一场超大规模的 DDoS 攻击使得美国互联网瘫痪. 世界范围内频发的恶意软件攻击事件使得网络安全问题渐渐得到了人们的关注.

为了避免或者减轻网络攻击对设备造成的危害, 人们需要使用有效的入侵检测系统对各种网络数据流进行检测. 入侵检测系统通过对网络流量特征进行分析, 主动检测网络此时是否遭受了恶意攻击, 从而方便相关人员做出及时的响应对策.

近年来, 深度学习技术已经被应用在了多个领域中, 比如语音识别、图像识别、文字翻译等[1], 由此可见深度学习技术对数据的分析和处理有着一定的优势. 它可以通过各种非线性的转换从原始数据中提取到特征信息, 进而用于分类等场景. 入侵检测的本质是一个分类问题, 其目的是检测出网络中的恶意流量并判断其对应的类别, 这与深度学习非常契合. 因此越来越多的深度学习方法被应用在网络流量入侵检测中[2]. 可以看出研究人员投入了大量的精力在入侵检测系统的检测性能上, 但是对于模型的参数量、模型大小以及训练时间的研究较少. 由于网络设备中资源存在着局限性, 复杂的入侵检测模型可能无法顺利部署并且运行在网络设备中, 一旦攻击者成功入侵网络系统, 将会造成难以估量的损失[3]. 因此, 如何同时兼顾检测性能和模型的轻量化是目前入侵检测系统中亟需解决的问题.

本文提出了一种融合 GRU 和 CNN 的轻量级入侵检测模型 (integrated GRU and light-weighted CNN, IGRU-LiCNN), 从特征降维和模型结构两方面实现了模型的轻量性, 同时模型也表现出了优秀的检测性能. 主要贡献如下.

(1) IGRU-LiCNN 针对时序数据中长短期的依赖关系, 将 GRU 每个时间步的隐藏层进行拼接作为卷积神经网络的输入.

(2) 在空间特征提取时, 本文采用了逆残差、深度可分离卷积、空洞卷积和通道洗牌等结构对数据进行多尺度的特征提取. 这在降低模型参数量的同时能够对网络流量数据特征进行充分提取. 并通过通道注意力机制对各特征通道分配不同的权重, 从而提高数据的表示能力, 加快模型的收敛速度.

(3) 极度随机树和轻量化的 CNN 算法的使用缩短了模型的攻击检测时间, 提高了模型的攻击检测性能.

1 相关工作

研究人员最早采用一些传统的机器学习算法来解决分类问题. 例如, Deng 等人[4] 使用 KNN 算法对大数据和医学成像数据进行分类实验. Garg 等人[5] 使用 SVM 算法进行车联网数据的异常检测. Kiss 等人[6] 探索了很多种聚类算法用于时序数据的分类任务, 最终选择了 K-means 算法对时序特征进行聚合, 从而对物理系统的网络攻击进行入侵检测.

近年来, 人们越来越关注深度学习技术在入侵检测领域的应用. Jiang 等人[7] 提出了一种多通道的 LSTM 攻击检测方法, 并且通过一种投票算法来判断攻击类型. 实验表明所提出的方法优于 SVM 等浅层算法. Kasongo 等人[8] 采用深度门控循环单元进行无线网络入侵检测. 实验表明他们提出的方法要优于一些传统的机器学习方法. Azizjon 等人[9] 设计了一个 1D-CNN 模型. 他们对互联网协议的数据包进行序列化作为训练数据. 在 UNSW NB15 IDS 数据集中的结果表明其提出的模型优于传统的机器学习分类器. Zhang 等人[10] 提出了一个基于流的入侵检测模型. 他们将不平衡处理和卷积神经网络结合在一起, 并研究了卷积核数量对模型检测性能的影响. 同样的, 开发人员也将一些混合模型运用在了网络流量入侵检测中. Kunhare 等人[11] 提出了一种混合逻辑回归和决策树的流量异常检测模型. 他们先通过遗传算法来进行特征选择, 再使用混合分类器进行网络流量分类. 最后通过 NSL-KDD 数据集验证了所提出模型的有效性. Sun 等人[12] 使用 CNN 和 LSTM 的混合模型提取特征. 分类时采用了类权重的概念来解决数据类别不平衡的问题.

这些基于深度学习方法在入侵检测中取得了一定的进展, 但是对于入侵检测模型在网络中的实际部署时没有考虑到模型的复杂性以及模型的大小. 由于存储空间和计算能力的限制, 在没有 GPU 的设备中部署神经网络模型仍然是一个问题. 最近, 一些研究人员注意到了这个问题并展开了一些研究. Ren 等人[13] 首先使用递归特征消除的方法来降低数据特征维度, 再使用深度强化学习模型作为入侵检测的分类器. Popoola 等人[14] 使用长短期记忆自动编码器的编码阶段来降低特征维度, 然后使用 BiLSTM (bidirectional long short-term memory) 模型分析低维数据中的关联信息从而区

分不同类别的流量数据. 上述的轻量级模型都是通过降低输入特征的维度来降低模型的复杂度. 但是在特征提取步骤中, 他们仍然使用一些复杂的模型来达到特征提取的目的.

2 轻量化的入侵检测模型

本文提出的轻量化的入侵检测模型 IGRU-LiCNN 的整体架构如图 1 所示, 主要包括数据预处理、特征提取和分类输出这 3 个部分.

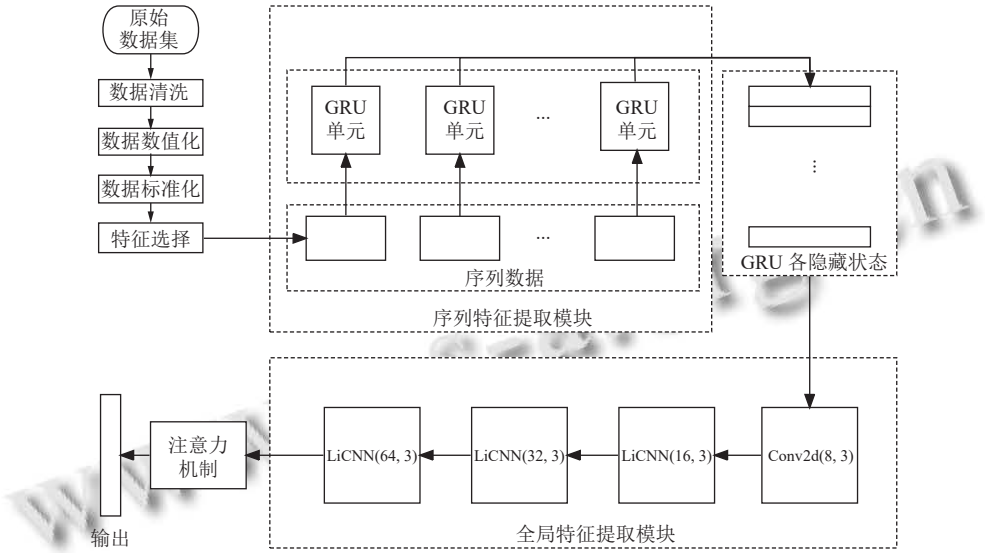

图 1 模型整体架构

IGRU-LiCNN 模型实现方法如算法 1 所示.

算法 1. IGRU-LiCNN 模型实现方法

输入: 训练集 D_a; 测试集 D_b; 训练周期 n
输出: IGRU-LiCNN 入侵检测模型

1. 数据预处理(D_a,D_b)
2. 从数据集中删除异常值
3. 将类别标签转化为数字表示
4. 计算数据集的非线性标准化结果
5. D_c←使用 ETR 对训练集进行降维处理
6. D_d←使用 ETR 对测试集进行降维处理
7. return D_c,D_d→得到新的 k 维特征空间
8. IGRU-LiCNN(D_c,D_d)
9. while $i \leqslant n$ do
10. 载入轻量级网络
11. 将训练集D_c输入到轻量级网络中进行训练
12. 使用交叉熵损失函数更新参数
13. 保存模型参数 CE_i
14. end while
15. 保存模型参数 CE_n 作为 IGRU-LiCNN 模型的训练结果
16. 使用测试集D_d测试 IGRU-LiCNN 模型
17. return IGRU-LiCNN 模型

2.1 检测原理

网络遭受攻击时, 产生的流量与正常流量存在着一定的差异性, 并且相同攻击产生的网络流量具有相似性而不同攻击产生的流量具有差异性, 根据这一特点可以通过分析捕获到的数据包特征来判断网络是否遭受到了攻击以及攻击的具体类型. DoS 攻击是一种常见的攻击类型, 其目的是使计算机或者网络无法提供正常的服务. 常见的 DoS 攻击分为计算机网络带宽攻击和连通性攻击. 其主要原理是向目的主机发送大量的连接请求, 但是不向服务器发送确认连接的数据包, 这使得服务器一直处于等待连接的状态, 最后导致服务器资源耗尽. 而 DDoS 与 DoS 的区别在于 DoS 是单机之间的攻击模式, 而 DDoS 攻击是利用一批受控制的主机同时向目标主机发起攻击, 这使得攻击的规模更大, 具体表现为传输的数据包更多, 数据包的传输速率更快等. 端口扫描攻击与一般的 DoS 攻击不同, 其主要目的是通过对主机中的指定端口进行扫描, 发送数据包信息, 从而判断该端口是否正常工作, 记录存在漏洞的端口方便后续发起攻击. 通常端口扫描攻击中会向目标主机发送 PSH、 FIN、 URG 标识且值为 1 的数据包以确认端口的状态. 因此可以从数据流中的数据传输速率以及带有标识数据包的数量等特征来判断端口扫描攻击.

2.2 数据预处理

网络流量的样本表示为$T = [t_1, t_2, \cdots, t_n, C]$, 其中 t_i表示第 i 个流量特征, C 表示流量样本对应的标签信

息. 因此, 整个网络流量数据集可以表示为:

$$TA=\begin{vmatrix} t_1^1 & t_2^1 & \cdots & t_n^1 \\ t_1^2 & t_2^2 & \cdots & t_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ t_1^m & t_2^m & \cdots & t_n^m \end{vmatrix} \tag{1}$$

其中, n 和 m 分别代表网络流量数据的特征数和样本数. 数据预处理包括 4 个步骤.

(1) 数据清洗. 对数据集中的空值、缺失值或者无穷大值 (inf) 所在的行进行删除.

(2) 数据数值化. 对于数据集中的一些非数值特征, 比如数据的标签信息需要使用独热编码将分类值映射成整数值.

(3) 数据标准化. 由于各个特征的取值范围不同, 有些特征之间相差多个数量级, 这会对分类结果产生影响, 因此要对特征值进行归一化, 本次采用的是非线性归一化的方法, 采用 log 函数对特征值进行映射. 其公式可以表示为:

$$t_j^{i'}=\log_{10}\left(t_j^i\right) \tag{2}$$

再使用 min-max 归一化的方法将数据区间映射到 [0, 1] 之间, 其公式为:

$$t_{\text{norm}}=\frac{t-t_{\min}}{t_{\max}-t_{\min}} \tag{3}$$

其中, t_{norm}为归一化后的结果, $t_{\min}$和$t_{\max}$分别表示数据中的最小值和最大值, t为需要归一化的数据.

(4) 特征选择. 它是在高维的数据特征中删除一些冗余特征, 从而达到减少数据集特征维度的目的. 以此来减少入侵检测模型的训练时间并且优化模型性能. 具体体现在节省了模型在特征提取时的额外时间, 这些额外时间是在处理那些对分类任务没有贡献的特征时产生的. 本次研究采用的特征选择技术为极度随机树 (ERT)[15], 是一种集成学习技术. 它通过对数据随机抽样和随机划分来构建多个决策树, 然后利用这些决策树的特征重要性指标来评估每个特征的重要性, 从而实现特征选择. 相较于其他特征选择方法, ERT 具有高效性和稳健性等优点, 特别适用于高维度数据的特征选择. 本次选用的特征重要性指标为基尼重要性. 它是指一个特征对于分类的贡献程度, 计算方法是在每个节点上比较分裂前后的基尼系数的变化. 对于每个特征, ERT 都会计算基尼重要性, 然后根据重要性大小对特征进行排序. 通过 ERT 技术, 将 CIC-IDS2017 数据集的特征维度降到 30 维.

2.3 IGRU-LiCNN 入侵检测模型

在网络设备中部署入侵检测模型时需要考虑模型的时间开销、计算开销以及模型的检测性能. 为了部署成功, 一些入侵检测模型不得不使用一些计算复杂度低、检测精度低的传统机器学习方法. 这使得入侵检测系统的防御能力大打折扣. 因此本文从轻量性和检测性能两个角度出发, 设计了一个轻量级的模型 IGRU-LiCNN 来准确地检测网络入侵行为. 我们提出模型的入侵检测模块可以分为 3 个部分, 分别是序列特征提取模块、全局特征提取模块和注意力机制模块.

2.3.1 序列特征提取模块

该部分采用 GRU 提取序列数据间的依赖关系, 防止出现梯度消失和梯度爆炸的问题. 与 LSTM 相比, GRU 在保证性能的同时有着更少的参数量. 将数据划分成宽度为 W 的窗口大小, 该窗口包括 W 个连续的网络流量样本数据, 可以表示为$S=[x_{t-W+1},x_{t-W+2},\cdots,x_t]$. 将 S 输入到 GRU 中, 对于每一个时间步数据都会产生一个隐藏状态的向量化表示h_1. 考虑到序列数据中的长短期依赖关系, 将所有的隐藏层信息进行输出, 而不仅是输出最后的隐藏层信息, 对应的公式如下:

$$h_1,h_2,\cdots,h_n=GRU(x_1,x_2,\cdots,x_n) \tag{4}$$

2.3.2 空间特征提取模块

在传统的 CNN 结构中往往采用最大池化层进行特征的下采样工作, 通过选择局部单位中的最大值来减少数据的维度. 如果想要从多个维度对数据特征进行提取则需要进行多次的池化操作. 本文在卷积中加入空洞卷积的结构来达到这一目的, 如果需要从多个感受野对数据特征进行提取, 只需改变空洞卷积中的膨胀系数, 从而避免了多次池化操作带来的计算成本. 另外本文采用逆残差结构来防止梯度消失和网络退化. 如果采用普通的残差结构先对特征图进行压缩操作, 只能提取到很有限的特征信息. 而逆残差结构是先将特征图从低维映射到高维空间再进行特征提取工作, 提取完成后再对特征图进行压缩. 本文将深度可分离卷积结构拆分成两个独立的部分, 分别是深度卷积和逐点卷积. 使用深度卷积进行特征提取工作, 使用逐点卷积进行扩充和压缩特征图的操作. 最终将这些轻量化的结构结合起来构成了本文的轻量级单元 LiCNN, 其结构如图 2 所示.

如图 1 所示, 空间特征提取模块中主要包括 1 个

普通卷积层、3 个 LiCNN 轻量级单元. 其主要的设计思路如下: 首先将上一个模块输出的所有隐藏层信息进行拼接作为本模块的输入. 然后采用一个步长为 2 的普通卷积层实现降采样以及特征图大小调整的功能. 最后将处理后的特征图输入到 3 层轻量级单元 LiCNN 中进行特征提取工作.

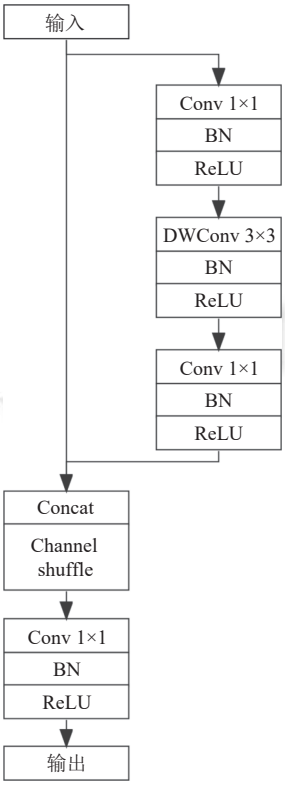

图 2 LiCNN 结构图

LiCNN 的主要实现流程如下.

(1) 将输入的特征图按照通道数等分为x_{f1}和x_{f2}, 特征图x_{f1}做同等映射, 特征图x_{f2}进行特征提取.

(2) 使用 1×1 的逐点卷积将特征图x_{f2}从低维空间映射到高维空间, 得到特征图x_{f2_h}.

(3) 使用带有空洞卷积结构的深度卷积对特征图x_{f2_h}进行特征提取, 得到特征图x_{f2_h2}.

(4) 使用 1×1 的逐点卷积对特征图x_{f2_h2}进行压缩使其特征层数与x_{f2}相同, 得到特征图x_{f2_out}.

(5) 对特征图x_{f1}和x_{f2_out}进行张量拼接, 然后使用通道洗牌技术实现特征图之间的信息交互, 从而消除边界效应, 得到特征图x_{c_s}.

(6) 最后采用 1×1 的逐点卷积调整通道数目, 得到最终的输出x_{Li_out}.

在 LiCNN 结构中每个卷积步骤后均使用 BN 层进行规范化处理. 这样使得输出能够满足或者近似服从正态分布, 以此来加快模型的收敛速度. 并且能够防止梯度消失现象的发生. 然后使用 ReLU 激活函数放大特征间的差异, 得到最终的输出. 对于 3 层的 LiCNN 结构中深度卷积加入了混合空洞卷积结构, 其中膨胀系数分别设置为 [1, 2, 3]. 各计算式如下:

$$x_{out\_c} = CNN(Concatenation(h_1, h_2, \cdots, h_n)) \tag{5}$$

$$x_{out\_gf} = LiCNN\left(x_{out\_c}\right) \tag{6}$$

2.3.3 注意力机制

在通道注意力机制中为了避免全连接层中的大量参数, 使用全局平均池化代替进行特征压缩任务, 其公式为:

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i, j) \tag{7}$$

其中, u_c为特征层中的特征点, H 和 W 为特征层大小. 再使用两个全连接层来融合各通道的信息, 其公式为:

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2\delta(W_1 z)) \tag{8}$$

其中, 第 1 个全连接层将特征通道压缩为C/r, r 为缩放参数. 然后使用 ReLU 激活函数. 接着是第 2 个全连接层, 将特征通道数恢复到 C, 再使用 Sigmoid 函数, 得到 s. 最后将权重和对应通道特征相乘, 其公式为:

$$\tilde{x}_c = F_{\text{scale}}(u_c, s_c) = s_c \cdot u_c \tag{9}$$

最后对于加权后的特征图, 使用全连接层对加权后的特征图进行处理得到输出 Z, 最后使用 Softmax 函数进行网络流量分类.

3 实验验证

3.1 实验环境和超参数设置

本文的模型训练和测试均在 Windows 上进行, CPU 为 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40 GHz, GPU 版本为 NVIDIA GeForce MX450, RAM 为 16.0 GB, PyTorch 版本为 1.2.0, Python 版本为 3.6.12. 本文提出的模型使用 SGD 优化器, 其中 weight_decay 设置为 0.000 1. 学习率设置为 0.001. GRU 中隐藏层的节点数设置为 32. 批量大小设置为 8. epoch 设置为 15.

3.2 数据集选取

本文主要使用 CIC-IDS2017 公共数据集[16] 来评估模型. CIC-IDS2017 数据集中主要包括 8 种攻击类

型, 包括基于网络的攻击 (Web attack)、暴力破解攻击 (brute force FTP 和 brute force SSH)、拒绝服务攻击 (DoS)、分布式拒绝服务攻击 (DDoS)、渗透攻击 (Infiltration)、Heart-bleed 攻击、僵尸网络攻击 (Bot) 和端口扫描攻击 (Port Scan). 共有八十多个特征维度. 其数据分布如表 1 所示.

表 1 原始数据分布

数据类型	数据量
BENIGN	2273097
DoS	252661
DDoS	128027
Port Scan	158930
Bot	1966
FTP-Patator	7938
SSH-Patator	5897
Heartbleed	11
Infiltration	36

网络流特征是从原始的数据包信息中提取出来的, 能够反映数据包的结构和对应的网络行为.其具体可以分为统计特征、时序特征、协议特征和有效载荷特征. 通过统计数据包中的信息得到了统计特征, 例如数据流的持续时间 (flow duration)、后向数据包的总长度 (total length of Bwd packets) 等. 时序特征是统计数据包之间的时间关系得来的. 该特征对于一些与时间相关的攻击 (如分布式拒绝服务攻击) 的检测结果有着重要的影响. 例如两个流之间的平均时间 (flow IAT mean)、前向发送的两个数据包之间的总时间 (Fwd IAT total) 等. 协议特征则是统计传输层协议的数据字段得来的. 由于这类特征包含了协议相关的特征信息, 因此对于针对协议发起的攻击的检测有着重要的作用, 比如分布式拒绝服务攻击. 该类特征有带有 FIN 标志的数据包数量 (FIN flag count)、带有 ACK 标志的数据包数量 (ACK flag count) 等. 有效载荷特征是指数据包中携带数据信息的特征, 通过统计有效载荷特征也能识别出特定的攻击流量, 比如在 ping 报文中发现了超过 32 字节长度的数据部分, 则能判断该报文为异常报文.

由于数据分布不平衡, 有些类别样本量过少. 本次只选取了其中样本数量较多的 4 种, 分别是 BENIGN、DoS、DDoS 和 Port Scan 样本. 因此数据集中仅挑选了与这几类攻击样本相关的数据集文件. 进行数据预处理之后各类数据分布如表 2 所示.

3.3 评价指标

为了对本文提出的模型进行评估, 选取准确率、召回率、$F1$ 值以及参数量作为本次实验的评价指标. 首先介绍 TP、FN、FP 和 TN, 其中 N 代表负例样本, P 代表正例样本, 因此 TP 代表预测为正例样本实际也为正例样本的数量; FN 代表预测为负例样本实际为正例样本的数量; FP 代表预测为正例样本实际为负例样本的数量; TN 代表预测为负例样本实际为负例样本的数量. 各评估指标的计算公式如下:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

$$Recall = \frac{TP}{TP+FN} \tag{11}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{12}$$

其中, 准确率表示模型正确识别的样本与总样本的比值. 召回率表示模型预测正确的正样本数量与总正样本数量的比值. $F1$ 值是精度和召回率的调和平均值. 在入侵检测系统中追求的是更高的准确率、召回率、$F1$ 值和更低的参数量.

表 2 提取后数据集各类样本数量

数据类型	训练集	测试集
BENIGN	466430	198856
DoS	176782	75879
DDoS	89563	38464
Port Scan	111173	47757

3.4 实验结果

本文主要使用的 CIC-IDS2017 公共数据集为近年来较新的网络流量数据集, 与 KDD99 数据集相比, 它通过搭建新的实验环境从而增加了一些新的攻击类型. 图 3 为模型训练的损失收敛曲线. 本文采用的优化器为 SGD. 训练时首先采用了预热学习率的方式, 在训练的前 3 个周期将学习率从一个较小值逐渐增加到设定值, 这样能够让模型快速学习到数据的特征信息, 也能够避免训练初期较大的学习率会导致模型容易陷入局部最优解的问题; 然后在余下周期训练中使用余弦退火的学习率衰减方法逐步调整学习率, 这样能够极大程度的减少模型的震荡和过拟合风险.

图 4 为窗口大小对模型的准确性的影响. 从图中可以看出, 当窗口大小为 10 时, 模型在 CIC-IDS2017 数据集上实现最佳性能. 此后, 随着窗口大小的增加, 数据集中的检测性能存在着下降的趋势. 对于较大的窗口, 模型的检测性能基本没有提高, 并且会因为参数量的增多而变得难以训练.

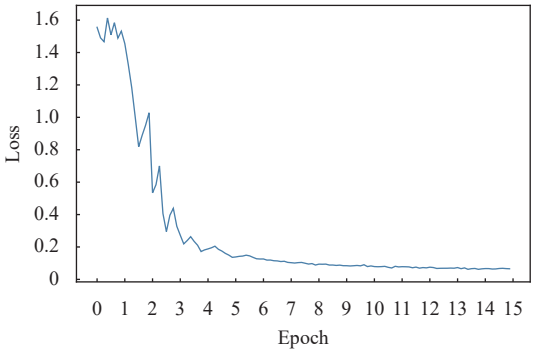

图 3 模型训练的损失收敛曲线

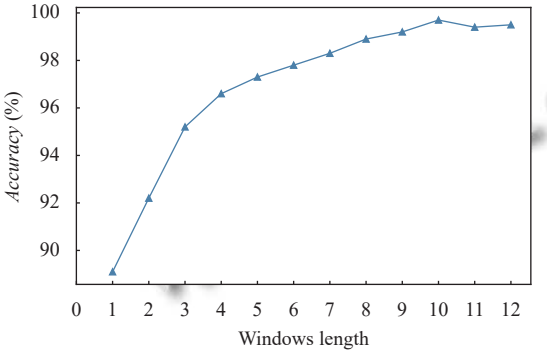

图 4 窗口大小对模型检测性能的影响

本文分析了特征选择技术对实验结果的影响. 如表 3 和表 4 所示, 显示了数据集特征降维前后的混淆矩阵. 在第 1 个实验中, 使用全部特征数据 (G1) 来训练和评估提出的模型. 在第 2 个实验中, 使用最优的 30 个特征 (G2) 来训练和评估模型. 可以看出基于 G2 的入侵检测模型的准确率要高于在 G1 上训练的准确率. 这说明特征选择算法 ERT 删除了对分类无用的特征, 从而使得模型能够更好地提取特征信息.

表 3 测试集 G1 的混淆矩阵

真实值	预测值			
	BENIGN	DDoS	DoS	Port Scan
BENIGN	197966	57	674	159
DDoS	7	38447	10	0
DoS	968	5	74906	0
Port Scan	243	0	25	47489

表 4 测试集 G2 的混淆矩阵

真实值	预测值			
	BENIGN	DDoS	DoS	Port Scan
BENIGN	198201	32	525	98
DDoS	10	38445	9	0
DoS	784	4	75091	0
Port Scan	224	0	21	47512

3.5 对比分析

本文将 IGRU-LiCNN 模型与 4 种模型进行了对比, 分别是 IGWO-SVM[17]、ANN-CFS[18]、KNN-PCA[19] 和 OCNN-HMLSTM[20]. 结果如表 5 所示. 可以看出基于机器学习的检测方法已经具备一定的检测性能, 比如 KNN-PCA, 但是与神经网络模型相比仍有较大差距. 这说明面对高维复杂的数据, 传统的机器学习算法已经不再适用于入侵检测模型. 相比较而言, ANN-CFS 模型的各项指标都有一定的提升. 这是因为神经网络具有自学习和构建非线性关系的能力, 但是其没有考虑到数据中多维的特征信息. OCNN-HMLSTM 解决了这一问题, 分别使用 CNN 和 LSTM 进行特征提取. 但是其使用的模型结构单一并且没有考虑到网络优化的问题. 而 IGRU-LiCNN 考虑到了数据中的长短期依赖关系, 并且采用 LiCNN 结构对网络流量特征进行提取, 具有更加丰富的特征表示能力. 解决了传统 CNN 结构中特征提取不充分以及深层模型参数量过大的问题. 再加上 BN 层, 注意力机制等结构的引入加快了模型的收敛速度. 因此本文有着更少的训练时间为 855.1 s. 同样的, 本文提出模型在对整个测试集进行分类反馈的平均时间为 18 s, 这表明模型对于一个批次数据的检测时间约为 3.9 ms, 对于入侵检测来说这个时间消耗是可以接受的.

表 5 数据集中各模型的检测结果

模型	准确率 (%)	召回率 (%)	*F*1值 (%)	训练时间 (s)
IGWO-SVM	91.45	90.81	93.19	1954.4
ANN-CFS	94.60	92.75	95.06	1476.3
KNN-PCA	92.51	90.15	94.14	1634.5
OCNN-HMLSTM	96.48	96.12	97.20	1396.7
IGRU-LiCNN	**99.53**	**99.52**	**99.53**	**855.1**

表 6 为 CIC-IDS2017 数据集中各类样本的召回率和 *F*1 值. 从表中可以看出, 模型对于正常类型样本的召回率为 99.67%, 对 Port Scan 样本的召回率为 99.49%; 对于 DDoS 样本的召回率为 99.95%; 对于 DoS 样本的召回率为 98.96%. 从 *F*1 值可以看出模型对于 Port Scan 样本和 DDoS 样本的分类性能最高, 分别是 99.64% 和 99.93%. IGRU-LiCNN 仅在极少数样本中存在错误分类的情况, 这是由于在攻击发生的初期, 捕获到的数据流信息中攻击相关特征体现不明显, 存在着错误分成正常流量的情况. 但是攻击是一个持续的过程, 在之后捕获到的数据流信息中则能很好地检测出攻击. 因此可以得出, IGRU-LiCNN 实现了稳健的检测性能.

表 6 数据集中各类样本的召回率和 *F*1 值

数据类型	*F*1值	召回率
BENIGN	0.995 8	0.996 7
DDoS	0.999 3	0.999 5
DoS	0.991 1	0.989 6
Port Scan	0.996 4	0.994 9

3.6 消融实验分析

我们通过消融实验来评估各个模块的贡献, 如表 7 所示. GRU 模型采用最后一个隐藏层的输出作为分类类依据; CNN 模型使用普通卷积进行特征提取; 从实验结果可以看出, GRU 和 CNN 相结合的模型与 CNN 模型相比准确率提升了 2.1%, 这是因为模型考虑到了数据中的时空特征信息. LiCNN 结构的加入使得模型的准确率提升了 1.8%, 这是因为 LiCNN 与原有的 CNN 结构相比做了一些改进, 首先是使用逆残差结构以及深度卷积和逐点卷积, 在保证从高维特征图中充分提取特征信息的同时减少了模型的参数量. 混合空洞卷积的结构在不使用池化层的情况下从不同的感受野对特征信息进行提取. 以及通道洗牌结构的加入消除了特征图中的边界效应. 通过运用逐层的逆残差学习有效解决了网络退化的问题, 也更利于模型的收敛. 加入注意力机制层后模型的准确率提升了 0.9%. 因为注意力机制层使得模型更加关注对分类有用的特征层而忽略无用的特征层.

表 7 消融实验中各模型的检测结果 (%)

模型	准确率	*F*1值	召回率
GRU	94.35	94.26	93.89
CNN	94.74	94.53	94.17
GRU+CNN	96.84	96.35	96.73
GRU+LiCNN	98.65	98.12	98.23
IGRU-LiCNN	99.53	99.53	99.52

3.7 模型的轻量化性能

从前面的对比实验可以看出, IGRU-LiCNN 有着出色的分类性能. 除此之外, IGRU-LiCNN 还专注于轻量化的性能, 使其能够更好地部署在网络设备上. 考虑到网络设备资源有限的特点, 我们通过实验比较了模型改进前后的参数量及其训练时间.

表 8 为两个模型的具体结构参数. 表 9 为模型的参数量对比. 从表中可以看出, IGRU-LiCNN 有着更少的参数量. 没有使用特征选择算法时与 GRU+CNN 模型相比, IGRU-LiCNN 的参数量减少了 49.94%. 可以得出 IGRU-LiCNN 占用的存储空间也更小, 能够更好地缓解网络设备资源受限的问题. 此外, 如果在数据输入模型之前进行特征选择工作能够进一步减少模型的参数量和模型大小. 从表中可以看出, 与不使用 ERT 的模型相比, IGRU-LiCNN 的参数量减少了 23.7%. 由此可得, 轻量化的特征提取模型和特征选择算法都能为模型的轻量化做出贡献. 通过实验可得在 CIC-IDS-2017 数据集中 GRU+CNN 模型的训练时间为 1 265.3 s, IGRU-LiCNN 在训练时间上减少了 32.4%. 另外从检测结果可以看出 IGRU-LiCNN 在准确率上提升了 2.69%. 由此可得 IGRU-LiCNN 为轻量化的网络入侵检测模型提供了一种解决方案.

表 8 两个模型的具体结构

层	GRU+CNN	IGRU-LiCNN
Input	Input	Input
Layer1	GRU	GRU
Layer2	Conv2d1(8, 3)	Conv2d1(8, 3)
Layer3	Conv2d2(16, 3)	LiCNN1(16, 3)
Layer4	Conv2d3(32, 3)	LiCNN2(32, 3)
Layer5	Conv2d4(64, 3)	LiCNN3(64, 3)
Layer6	GAP	CA
Layer7	FC	FC
Output	Output	Output

表 9 模型的参数量对比

模型	参数量
GRU+CNN (ERT)	34 184
Ours (ERT)	**14 808**
GRU+CNN	38 792
Ours	**19 416**

4 结论与展望

本文提出了一种轻量化的入侵检测模型 IGRU-LiCNN. 首先使用极度随机树对高维的数据特征进行降维处理. 再采用 GRU 和轻量化的 CNN 模型进行特征提取, 其中轻量化的 CNN 模型是本文的研究重点. 它采用了逆残差结构和通道洗牌方法, 增强特征层之间的关联性, 从而更有效地提取特征, 也避免了网络退化的问题. 通过深度可分离卷积大大降低了模型的参数量. 空洞卷积的结构使得模型能够从多个维度进行有效的特征提取工作, 并且不会增加模型的计算成本. 实验表明, 本文提出的模型不仅在检测方面有着优异的性能, 而且能够降低模型的参数量、减少模型的体积、训练时间, 同时保持较低的检测时间. 并且通过实验发现本文提出的模型也能在 CPU 环境下进行入侵检测. 因此, 针对在资源受限的网络设备中部署入侵检测模型的问题, 本文提出的方法能够作为一种可行的解决办法.

未来仍有几项工作要做: 第一, 针对不平衡数据中的分类问题展开研究; 第二, 重点研究如何在真实的网络环境中部署入侵检测模型; 第三, 通过对真实网络数据的抓取实现一个完整的入侵检测系统.

参考文献

1 Dong S, Wang P, Abbas K. A survey on deep learning and its applications. Computer Science Review, 2021, 40: 100379. [doi: 10.1016/j.cosrev.2021.100379]

2 Yang Z, Liu XD, Li T, *et al*. A systematic literature review of methods and datasets for anomaly-based network intrusion detection. Computers & Security, 2022, 116: 102675. [doi: 10.1016/j.cose.2022.102675]

3 Kan X, Fan YX, Fang ZJ, *et al*. A novel IoT network intrusion detection approach based on adaptive particle swarm optimization convolutional neural network. Information Sciences, 2021, 568: 147–162. [doi: 10.1016/j.ins.2021.03.060]

4 Deng ZY, Zhu XS, Cheng DB, *et al*. Efficient KNN classification algorithm for big data. Neurocomputing, 2016, 195: 143–148. [doi: 10.1016/j.neucom.2015.08.112]

5 Garg S, Kaur K, Kaddoum G, *et al*. Sec-IoV: A multi-stage anomaly detection scheme for Internet of vehicles. Proceedings of the 2019 ACM MobiHoc Workshop on Pervasive Systems in the IoT Era. Catania: ACM, 2019. 37–42. [doi: 10.1145/3331052.3332476]

6 Kiss I, Genge B, Haller P, *et al*. Data clustering-based anomaly detection in industrial control systems. Proceedings of the 10th IEEE International Conference on Intelligent Computer Communication and Processing. Cluj-Napoca: IEEE, 2014. 275–281.

7 Jiang F, Fu YS, Gupta BB, *et al*. Deep learning based multi-channel intelligent attack detection for data security. IEEE Transactions on Sustainable Computing, 2020, 5(2): 204–212. [doi: 10.1109/TSUSC.2018.2793284]

8 Kasongo SM, Sun YX. A deep gated recurrent unit based model for wireless intrusion detection system. ICT Express, 2021, 7(1): 81–87. [doi: 10.1016/j.icte.2020.03.002]

9 Azizjon M, Jumabek A, Kim W. 1D CNN based network intrusion detection with normalization on imbalanced data. Proceedings of 2020 International Conference on Artificial Intelligence in Information and Communication. Fukuoka: IEEE, 2020. 218–224. [doi: 10.1109/ICAIIC48513.2020.9064976]

10 Zhang HP, Huang LL, Wu CQ, *et al*. An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. Computer Networks, 2020, 177: 107315. [doi: 10.1016/j.comnet.2020.107315]

11 Kunhare N, Tiwari R, Dhar J. Intrusion detection system using hybrid classifiers with meta-heuristic algorithms for the optimization and feature selection by genetic algorithm. Computers and Electrical Engineering, 2022, 103: 108383. [doi: 10.1016/j.compeleceng.2022.108383]

12 Sun PF, Liu PJ, Li Q, *et al*. DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system. Security and Communication Networks, 2020, 2020: 8890306. [doi: 10.1155/2020/8890306]

13 Ren KZ, Zeng YF, Cao ZQ, *et al*. ID-RDRL: A deep reinforcement learning-based feature selection intrusion detection model. Scientific Reports, 2022, 12(1): 15370. [doi: 10.1038/s41598-022-19366-3]

14 Popoola SI, Adebisi B, Hammoudeh M, *et al*. Hybrid deep learning for botnet attack detection in the Internet-of-Things networks. IEEE Internet of Things Journal, 2021, 8(6): 4944–4956. [doi: 10.1109/JIOT.2020.3034156]

15 Shams E A, Rizaner A, Ulusoy A H. A novel context-aware feature extraction method for convolutional neural network-based intrusion detection systems. Neural Computing and Applications, 2021, 33(20): 13647–13665. [doi: 10.1007/s00521-021-05994-9]

16 Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proceedings of the 4th International Conference on Information Systems Security and Privacy. Madeira, Portugal: SciTePress, 2018. 108–116.

17 Safaldin M, Otair M, Abualigah L. Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. Journal of Ambient Intelligence and Humanized Computing, 2021, 12(2): 1559–1576. [doi: 10.1007/s12652-020-02228-z]

18 Sumaiya Thaseen I, Saira Banu J, Lavanya K, *et al*. An integrated intrusion detection system using correlation-based attribute selection and artificial neural network. Transactions on Emerging Telecommunications Technologies, 2021, 32(2): e4014. [doi: 10.1002/ett.4014]

19 Benaddi H, Ibrahimi K, Benslimane A. Improving the intrusion detection system for NSL-KDD dataset based on PCA-fuzzy clustering-KNN. Proceedings of the 6th International Conference on Wireless Networks and Mobile Communications. Marrakesh: IEEE, 2018. 1–6. [doi: 10.1109/WINCOM.2018.8629718]

20 Kanna PR, Santhi P. Unified deep learning approach for efficient intrusion detection system using integrated spatial-temporal features. Knowledge-based Systems, 2021, 226: 107132. [doi: 10.1016/j.knosys.2021.107132]

(校对责编: 牛欣悦)