

利用网络望远镜和 LSTM 神经网络的大数据环境网络异常探测模型

邓小亚[†]

(四川文理学院 智能制造学院, 四川 达州 635000)

摘 要:在当前大数据环境下,保障网络安全与稳定对于维护网络健康至关重要,而网络异常的及时探测则显得尤为迫切。为此,提出了一种基于网络望远镜技术和长短时记忆神经网络(LSTM)的网络异常探测模型。通过对中国某地区近期捕获的互联网流量进行实验验证,结果显示本文方法在不同程度的网络攻击检测中均表现出良好的性能。

关键词:网络异常;异常检测;网络望远镜;深度网络;大数据

中图分类号:TP393

文献标识码:A

Network Anomaly Detection Model in Big Data Environment Using Network Telescope and LSTM Neural Network

DENG Xiaoya[†]

(School of Intelligent Manufacturing, Sichuan University of Arts and Science, Dazhou, Sichuan 635000, China)

Abstract: In the current era of big data, ensuring network security and stability is of paramount importance for maintaining the overall health of networks. Prompt detection of network anomalies has become particularly urgent. To address this, a network anomaly detection model based on the combination of network telescope technology and long short-term memory (LSTM) neural networks is proposed. Through the experimental validation on recent captured Internet traffic in a specific region of China, the results demonstrate the efficacy of the proposed method in detecting network attacks of varying degrees.

Key words: network anomalies; anomaly detection; network telescopes; deep networks; big data

在大数据环境下,持续产生的大数据不断涌现,网络交互规模和复杂性持续攀升,这使得网络安全威胁变得更加严峻,网络异常的检测变得更为复杂和棘手。网络异常涵盖了恶意攻击、网络故障、网络拥堵等各种情形,这些异常行为往往隐匿在海量数据中,难以用传统手段精确捕获和识别^[1,2]。

现有的基于常规机器学习算法的入侵检测方法已无法满足日益复杂的网络攻击形式。相比之

下,基于深度学习的方法在接受大数据量训练时表现得更好,并且可适应不同的攻击场景。Terai 等人^[3]提出了一种使用长短时记忆的 ICS 入侵检测方法。LSTM 可以了解数据之间的时间上下文关系,并可以用于检测攻击数据序列。蹇诗婕等人^[4]针对入侵检测精度的问题,提出了一种基于稀疏自动编码器极限学习机入侵检测模型。李俊等人^[5]提出了一种基于门递归单元神经网络的网络入侵检测方法。周约翰逊等人^[6]利用长短期记忆网络,提

收稿日期:2023-10-06

基金项目:政务数据安全达州市重点实验室资助项目(ZSAQ202204, ZSAQ202201);智能制造产业技术研究院资助项目(ZNZZ202212)

作者简介:邓小亚(1977—),男,四川达县人,硕士,副教授,研究方向:软件工程,知识工程,数字图像处理。

[†]通信联系人, E-mail: dxy1234@163.com

出了一种利用移动流量数据检测网络异常的综合框架,从不同 eNodeBs 的 LTE 物理下行控制信道(PDCCH)收集数据,以半监督的方式实现深度学习算法,以检测可能产生的潜在流量异常。

然而,由于网络异常的多样性以及特征受干扰和变化的影响,上述基于规则和特征的异常检测方法已难以适应大数据环境中的异常探测。应对大数据环境下网络异常问题,近年来,人们开始将人工智能技术引入网络异常探测领域,特别是深度学习技术中的神经网络,如长短时记忆(LSTM)网络就以其出色的数据建模和特征提取能力受到广泛关注。通过训练神经网络模型,可以使其从大规模网络流量数据中学习规律和模式,从而实现对异常行为的自动识别和监测。此外,网络望远镜技术作为网络异常检测的重要手段,具备实时监测、数据采集和分析等功能,能够有效捕获异常行为特征。这些网络异常探测方法为解决大数据环境下的网络安全问题提供了全新的途径和思路。

1 基于网络望远镜和 LSTM 神经网络的网 络异常探测模型

本研究提出了一个完整的框架,用于监控网络望远镜收集的探测活动,并推断针对网络服务的异常流量。该模型通过对探测率(即暗网在一段时间内针对网络服务的接收数据包数量)建模来实现。

1.1 模型框架

图 1 总结了探测率建模和预测的方法。首先,输入暗网流量。然后,根据探测特征相似性,确定具有语义和时间上关联的探测活动的端口。这有助于优化预测模型的输入。为了实现这一点,本文提出了一种新的亲和度量,它考虑了端口探测率时间序列之间的交叉相关性(即时间探测相似性),以及来自顺序扫描端口的网络探测者的探测活动(即语义相似性)。为了降低模型特征空间的维度并减轻非信息特征带来的噪声,利用并评估了下面两种方法:

(1)端口排序法:根据端口与目标端口(即预测时间序列的端口)的亲和度得分对端口进行排序,并选择相关性最高的端口。

(2)端口聚类:使用谱聚类划分一个大型端口图,并根据亲和度得分进行加权。

最后,训练堆叠 LSTM 神经网络来预测探测率。该模型将与目标端口有亲缘关系的端口的探测活动历史记录作为输入,并输出目标端口的预期

探测率。

1.2 网络端口关联性

1.2.1 时间探测相似性

为了找到端口之间的时间相关性,本文使用从整个暗网 IP 地址空间的流量中提取的端口探测率时间序列。为了降低计算复杂度,本文考虑接收累计流量 50% 的端口的探测率时间序列,表示为 $I = \{I^{(u_x)}\}_{u_x \in D}$, I 的每条记录都是同一时间间隔内接收到的数据包数量的向量。该时间间隔是用户定义的,在本文的实验中,考虑 1 小时和 1 天的时间间隔。

为了测量两个端口 u_x 和 u_y 之间的时间探测相似性,在引入滞后 $l \in [-L, L]$ 之后,计算它们的探测率时间序列 $I^{(u_x)}$ 和 $I^{(u_y)}$ 之间的相关性。然后,考虑绝对值的最大相关性。用户定义的时间间隔 $[-L, L]$ 表示相关性中要捕捉的周期模式的数量。对于 1 小时分辨率的时间序列,设置 $L = 168$,相当于 1 周;对于 1 天分辨率的时间序列,选择 $L = 31$,相当于 1 个月。形式上,时间探测相似度的定义如下:

$$T_l(u_x, u_y) = \max_{-L \leq l \leq L} |\rho_l(u_x, u_y)| \in [0, 1] \quad (1)$$

其中, $\rho_l(u_x, u_y)$ 是端口 u_x 和端口 u_y 的探测率时间序列在引入滞后 l 后的无偏皮尔逊相关系数,定义为:

$$\rho_l(u_x, u_y) = \frac{1}{T - |l|} \sum_{z=\max(0, l)}^{\min(T, T+l)-1} \frac{I_z^{(u_x)} - \mu_{u_x}}{\sigma_{u_x}} \times \frac{I_{z-l}^{(u_y)} - \mu_{u_y}}{\sigma_{u_y}} \quad (2)$$

其中, T 是时间序列的长度; $I_z^{(\cdot)}$ 是第 z 个时间步长的探测率; $\mu_{\cdot}$ 和 $\sigma_{\cdot}$ 分别是时间序列 $I^{(\cdot)}$ 的平均值和标准偏差。

$T_l(u_x, u_y)$ 值接近 1 意味着,平均而言,端口 u_x 收到的探测流量与端口 u_y 在延迟 1 个时间步长之前或之后收到的探测流量高度相关。

1.2.2 语义探测相似性

采用提取和分析网络探测器连续扫描的端口。由于数据集中有多个 IP 地址(网络探测器),为了降低计算复杂度,从源 IP 地址池中随机(均匀)选择 5% 的 IP 地址,这些地址至少向两个目标端口发送了两个数据包。此外,只考虑探测活动最长持续 24 小时的 IP 地址,这是一个足够大的 DHCP 租约时间。如果有大量的 IP 地址顺序扫描过这两个端口,那么这两个端口就被认为在语义上是相似的。通常需要考虑以下几点:

- 1) 所选数据集的 24 小时流量子集,
 $\{D_1, D_2, \dots, D_w\}$ 。
 2) 如上所述选定的源 IP 地址集 G 。

3) $u_x \xleftrightarrow{xu} u_y$: 表示 IP 地址 ip 向端口 u_x 发送探测数据包后又向端口 u_y 发送数据包的事件, 反之亦然。

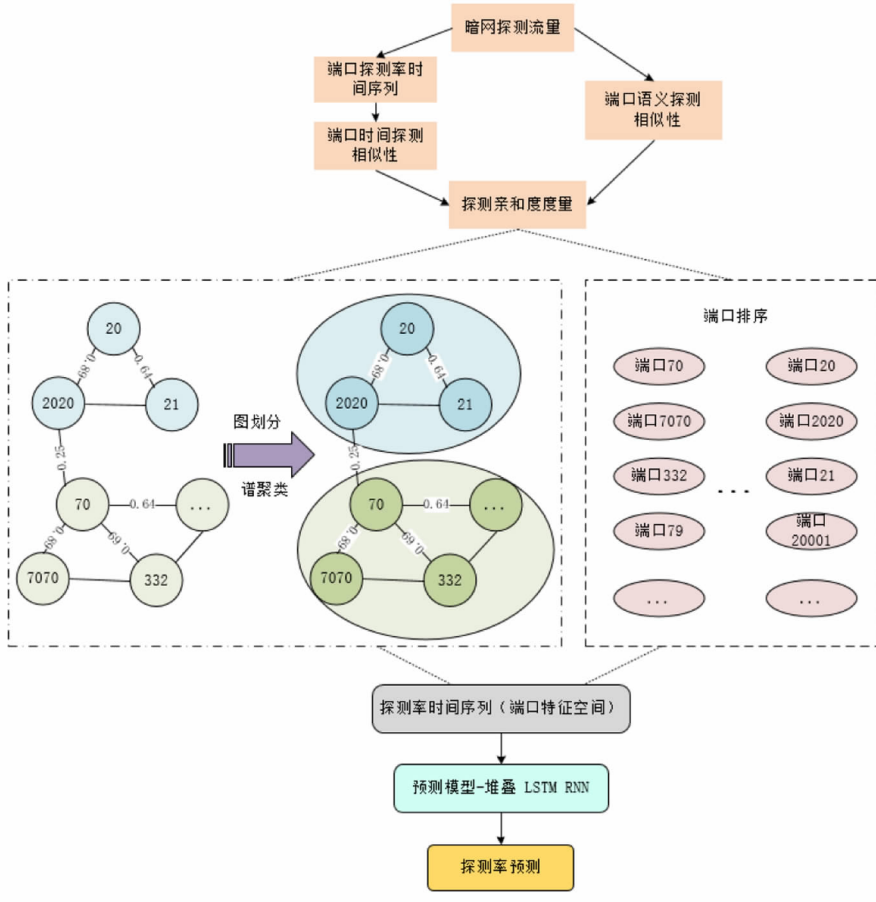


图 1 探测模型框架结构图

两个端口 u_x 和 u_y 之间的语义相似度定义为在 $\{D_x\}_{x \in [1, w]}$ 时间段内扫描这对端口号的 IP 地址的平均数量。

$$S(u_x, u_y) = \frac{1}{w} \sum_{z=1}^w \sum_{xu \in G} s_{xu}^{(D_z)}(u_x, u_y) \quad (3)$$

其中,

$$s_{xu}^{(D_z)}(u_x, u_y) =$$

$$\begin{cases} 1, & \text{if } u_x \xleftrightarrow{xu} u_y \text{ in probing traffic } D_z \\ 0, & \text{otherwise} \end{cases}$$

(4)

需要注意的是, 该度量并不像时间相似性度量那样绑定到区间 $[0, 1]$, 而是取决于所选 IP 地址的数量。为了使其与时间探测相似性的尺度保持一致, 通过对数值缩放然后应用最小-最大归一化来解决大值的偏差:

$$S_l(u_x, u_y) = \log(1 + S(u_x, u_y)) \quad (5)$$

$$\hat{S}(u_x, u_y) = \frac{S_l(u_x, u_y) - \min_{u_x, u_y} S_l(u_x, u_y)}{\max_{u_x, u_y} S_l(u_x, u_y) - \min_{u_x, u_y} S_l(u_x, u_y)} \in [0, 1] \quad (6)$$

得分 $\hat{S}(u_x, u_y)$ 接近 1 意味着端口 u_x 和 u_y 在语义上高度相似, 即它们被大量网络探测器连续扫描。

1.2.3 探测亲和度量

两个端口之间的亲和度量是衡量两个网络端口之间关系密切程度的指标。这个指标的定义方式是将“时间”和“语义探测相似性”这两个因素结合在一起, 通过计算它们的调和平均值来得出。

$$A(u_x, u_y) = 2 \cdot \frac{T(u_x, u_y) \times \hat{S}(u_x, u_y)}{T(u_x, u_y) + \hat{S}(u_x, u_y)} \quad (7)$$

1.2.4 端口聚类 and 排序

为了推断预测模型的特征空间,依靠定义的亲和度量,使用以下两种方法找到相似端口组。

(1) 谱聚类

亲和度矩阵 $\mathbf{G} \in \mathbb{R}^{|D| \times |D|}$ 是聚类算法的输入,其中元素 $G_{xy} = 1 - A(u_x, u_y)$ 如果 $x \neq y$, 以及 $G_{xx} = 0$, u_x 和 u_y 属于目标端口组的两个端口 D 。该矩阵是对称的,表示一个无向图。该图的每个顶点都是属于 D 的端口,如果 $G_{xy} > 0$, 则两个端口 u_x 和 u_y 之间存在边,且边的权重为 G_{xy} 。因此,通过对后一个图进行分区,可以找出具有相似探测特性的端口群。有很多聚类算法可以进行图划分。最稳健的算法之一是广泛用于此类任务的谱聚类算法。

确定最佳簇数 k (即最优图割),使用轮廓方法。该指标通过测量每个数据点与其自己的聚类与最近的相邻聚类的相似程度来深入了解聚类分离的质量。

为了确定最佳聚类数量 z (即最佳图切割),使用了剪影法。该方法通过测量每个数据点与其所在的聚类与最近相邻聚类的相似程度,来了解聚类分离的质量。形式上,对于属于一个聚类 C_b 的给定端口 $u_x \in D$,剪影得分定义如下:

$$s(u_x) = \frac{h(u_x) - g(u_x)}{\max(h(u_x), g(u_x))} \quad (8)$$

其中, g 是簇内平均距离。

$$g(u_x) = \frac{1}{|C_b| - 1} \sum_{u_y \in C_b, u_y \neq u_x} A(u_x, u_y) \quad (9)$$

其中, h 是最近簇的平均距离。

$$h(u_x) = \min_{l \neq b} \frac{1}{|C_l|} \sum_{u_y \in C_l} A(u_x, u_y) \quad (10)$$

然后,簇的最佳聚类数量 z 就是能使平均剪影得分最大化的簇:

$$z = \underset{2 \leq z_t \leq \frac{|D|+1}{2}}{\operatorname{argmax}} \frac{1}{z_t} \sum_{1 \leq b \leq z_t} \left(\frac{1}{|C_b|} \sum_{u_x \in C_b} s(u_x) \right) \quad (11)$$

(2) 端口排序

作为推断给定目标端口预测模型特征空间的另一种方法,可以简单地考虑最相似的端口,即亲和度得分高于阈值 τ 的端口。选择 τ ,通过交叉验证,通过在平均值和最大亲和度得分之间变化,给出最佳决定系数 R^2 得分,步长等于标准差。

使用端口排序,给定目标端口 $u_x \in D$ 的端口特征空间为:

$$C_\tau(u_x) = \{u_y \in D \mid A(u_x, u_y) > \tau\} \quad (12)$$

和 τ 不同之处在于:

$$\{\mu_A, \mu_A + \sigma_A, \mu_A + 2 \times \sigma_A, \dots, \max_{u_y \in D} A(u_x, u_y)\} \quad \text{其中,}$$

$$\mu_A = \frac{1}{|D| - 1} \sum_{\substack{u_y \in D \\ u_y \neq u_x}} A(u_x, u_y) \quad (13)$$

和

$$\sigma_A = \sqrt{\frac{1}{|D| - 1} \sum_{\substack{u_y \in D \\ u_y \neq u_x}} (A(u_x, u_y) - \mu_A)^2} \quad (14)$$

1.3 端口探测率预测

LSTM 是一种广泛使用且性能已得到验证的 门控 RNN 单元,其结构如图 2 所示,LSTM 单元由三个门组成:输入门、遗忘门和输出门。

整个 LSTM 单元的信息流如下。在时间步长 n 时,单元状态 c_{n-1} 保存着记忆,即迄今为止已学习到的信息,而隐藏状态 b_{n-1} 则保存着短期信息。

遗忘门将之前的隐藏状态 b_{n-1} 和输入序列中的特征向量 i_n 作为输入。然后,通过 Sigmoid 函数 $(\sigma(k) = \frac{1}{1 + e^{-k}})$ 产生一个过滤向量 f_n ,该向量控制着存储在单元状态 c_{n-1} 中的哪些信息将被遗忘。

$$f_n = \sigma(i_n \cdot P_f + b_{n-1} \cdot M_f + h_f) \quad (15)$$

与遗忘门类似,输入门也会产生一个滤波器,决定要存储在单元状态中的信息。

$$x_n = \sigma(i_n \cdot P_x + b_{n-1} \cdot M_x + h_x) \quad (16)$$

输入数据 i_n 与之前的隐藏状态 b_{n-1} 相结合,用 tanh 激活函数压扁,tanh 激活函数作为正则化器,其导数可以抵抗抗长范围梯度消失。

$$a_n = \tanh(i_n \cdot P_a + b_{n-1} \cdot M_a + h_a) \quad (17)$$

由此产生的候选状态 a_n 包含新信息(由 i_n 和 b_{n-1} 汇总而成),并通过输入过滤器 x_n 来决定将哪些信息添加到存储器中,即单元状态 c_n 。

$$c_n = f_n c_{n-1} + x_n a_n \quad (18)$$

输出门控制单元状态存储器中存储的哪些信息将被保留以生成输出值,隐藏状态也会相应更新。

$$\begin{aligned} o_n &= \sigma(i_n \cdot P_o + b_{n-1} \cdot M_o + h_o) \\ b_n &= o_n \operatorname{ngtb}(c_n) \end{aligned} \quad (19)$$

其中, M_o 、 P_o 和 h_o 是神经网络训练过程中可学习的参数集。

用于建模和预测探测率的堆叠 LSTM 神经网络如图 3 所示。

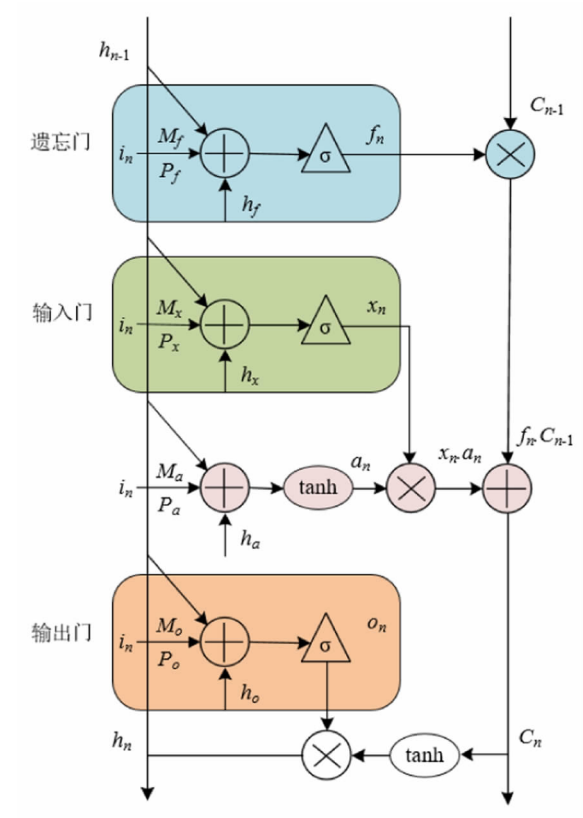


图 2 LSTM 单元架构

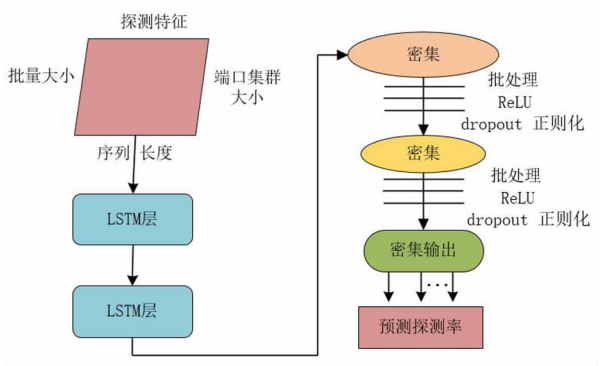


图 3 LSTM 神经网络结构图

该架构由具有 256 个和 128 个 LSTM 单元的 2 个堆叠 LSTM 层组成。在评估中,两个 LSTM 层足以在限制学习时间的同时获得精确的结果。LSTM 层后面是两个完全连接的区块,分别有 128 和 96 个隐藏单元。每个区块都从一个批处理归一化层开始,该层可以缓解内部协变量偏移的问题,并允许使用更高的学习率来加快训练速度。该块还包含 ReLU 激活层 ($\text{ReLU}(i) = \max(0, i)$),可降低训练过程中梯度消失的可能性。ReLU 之后是 dropout 正则化,即随机将一部

分隐藏单元设置为 0,以通过模拟不同的网络架构(单一网络架构中的集合学习)来防止过拟合。输出层返回探测率的预测结果,其大小与目标端口的数量相对应。最后,评估部分将讨论与训练模型相关的超参数。

该模型将属于同一集群(或排名靠前)的端口的探测率序列作为输入,并输出相关端口的探测率预测值。探测特征空间是三维的(探测速率序列的小批量),并且特征值被归一化为零均值和单位标准差。

1.4 探测流量异常推断

网络望远镜能检测异常流量,如扫描、蠕虫、反向散射数据包等,但也可能捕获无害的流量,如配置错误或统计扫描。为此,提出了以下两个指标用以识别异常流量。

A. 探测异常推断指标

为了推断探测异常,引入了一个依赖模型预测误差的指标。如果预测的探测率与实际探测率(包括不规则的恶意探测)之间有显著差异,这可能是本文模型没有学习到的异常流量。因此,这种误差指标代表了潜在威胁。

对于时间步长为 n 的给定端口 u ,使用以下指标:

$$E_n^{(u)} = \max\left(0, \frac{I_n^{(u)} - I_n^{(\hat{u})}}{I_n^{(\hat{u})}}\right)$$
 (20)

其中, $I_n^{(\hat{u})}$ 是使用训练有素的模型预测的探测率, $I_n^{(u)}$ 是暗网记录的探测率。

B. 网络风险预警指标

该指标依赖预测模型的性能,可根据暗网传感器记录的过去和当前探测的动态准确预测未来的探测率。因此,当使用本文的模型预测的探测率与使用简单线性模型得出的探测率(在前一个时间窗口中拟合)之间存在显著差异时,即网络风险较高。

对时间步长为 n 的端口 u ,相关网络服务的网络风险预警指标定义如下:。

$$M_n^{(u)} = \max\left(0, \frac{I_{n+1}^{(\hat{u})} - L_{n+1}^{(u)}}{L_{n+1}^{(u)}}\right)$$
 (21)

其中, $I_{n+1}^{(\hat{u})}$ 是使用本文的模型预测的探测率, $L_{n+1}^{(u)}$ 是使用线性模型预测的探测率,两者都提前一步。

较高的 $M_n^{(u)}$ 值意味着预测模型可能已经识别出端口特征空间中的探测模式,所以,预测出的探测率明显高于正常的探测趋势,这可能是威胁迫在

眉睫的指示。

2 实验与分析

2.1 实验设置

数据集由设在中国某地区的两个网络望远镜收集的网络流量组成。捕获的流量包括传入 TCP 和 UDP 数据包的报头,没有有效载荷。对于每个数据包,记录的信息包括时间戳、协议、源和目标 IP 地址、源和目标端口,以及 TCP 数据包的标志。为了降低计算复杂度,本研究考虑了 1074 个最有针对性的端口。探测活动从每天早上 6 点开始,在 24 小时内记录了 50 天(2022 年 12 月 22 日至 2023 年 2 月 9 日)流量的数据包跟踪。为了评估预测模型,探测流量被分为两个子集:75% 用于训练集,25% 用于测试集。

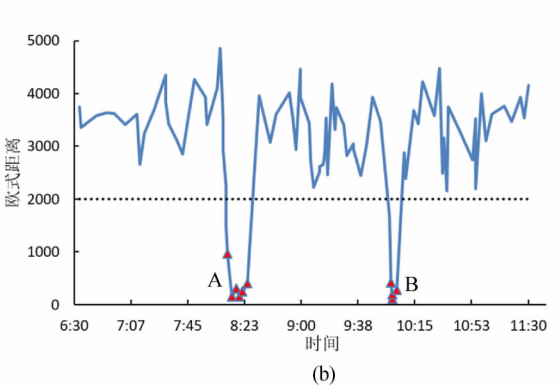
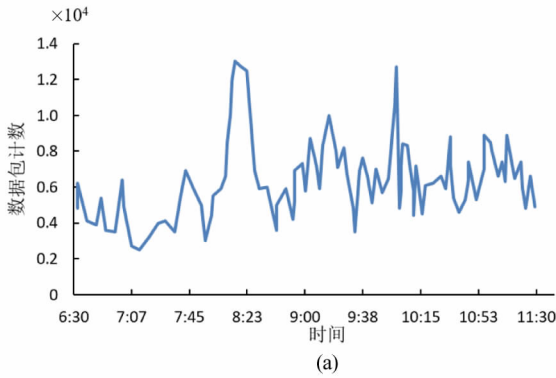


图 4 周日上午 8:15—10:00 出现异常

图 5(a)显示了第一周周一下午 13:00 至 18:00 流量中的数据包计数。起初,流量平面上的行为表现正常,然后在下午 15:15 时,ACK 风暴攻击作为异常情况出现,并持续了约 25 分钟。然后攻击结束,平面上恢复正常表现。图 5 (b)显示本文提出的方法成功检测到 2 个异常(C 和 D),每个都

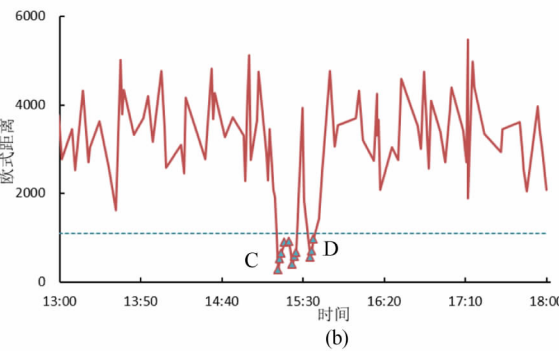
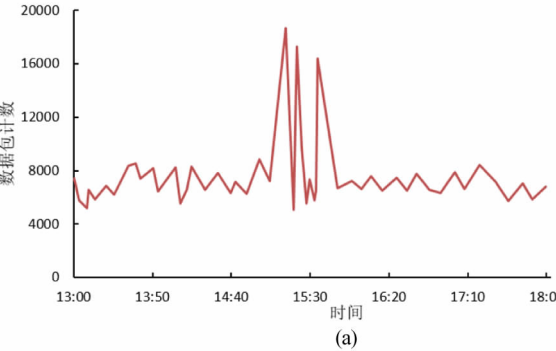


图 5 周一下午 15:15-15:40 出现异常

2.2 异常检测性能分析

为了评估所提出的模型在异常网络检测中的有效性,将提出的方法应用于 4 个案例中进行分析,如图 4~图 7 所示。图 4 (a) 显示了第一周周日上午 6:30 至 11:30 流量中的数据包计数。起初,流量平面上的行为一切正常,后来在上午 8:15 左右出现了一些异常情况,并在该窗口停留了约 10 分钟,然后再次表现正常,直到上午 10:00 再次出现行为差异。图 4(b)显示了本文所提方法成功检测到了这些异常情况。有 2 个异常(A 和 B),每个都有 0 个正常点,且分别有 6 个和 4 个连续的异常值。本文提出的方法将 A 和 B 都视为异常点,且确定率为 100%,表明本文模型可以百分之百检测到该案例中的网络异常情况。

包含至少 3 个连续的异常值。其中,检测出的 C 包含 8 个异常点、1 个正常点,检测准确率为 89%; D 包含 3 个异常点、0 个正常点,检测准确率为 100%。本文提出的方法将 C 和 D 都视为异常点,并且具有约 95% 的检测准确率。

图 6(a)显示了第一周周三上午 7:00 至 12:00 流量中的数据包计数。由图可以观察到流量平面上刚开始表现正常,然后在上午 10:00 时,ACK 风暴攻击作为异常情况出现,并持续了约 10 分钟。随后攻击结束,流量平面再次表现正常。图 6(b)

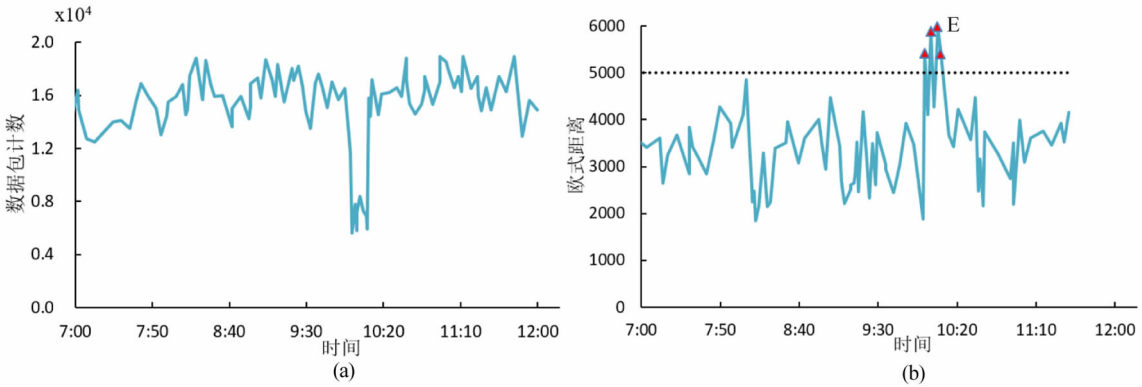


图 6 周三上午 10:00-10:10 出现异常

图 7(a)显示了第一周周四晚上 19:30 至 23:30流量中的数据包计数。一开始,平面的数据包数量表现相似,在晚上 20:10 左右,平面的数据包数量突然都有所增加。本文将这种突然增加视为异常。异常现象在窗口上持续了约 15 分钟。晚上 21:50 时,幅度增加再次出现。一般的相关性统

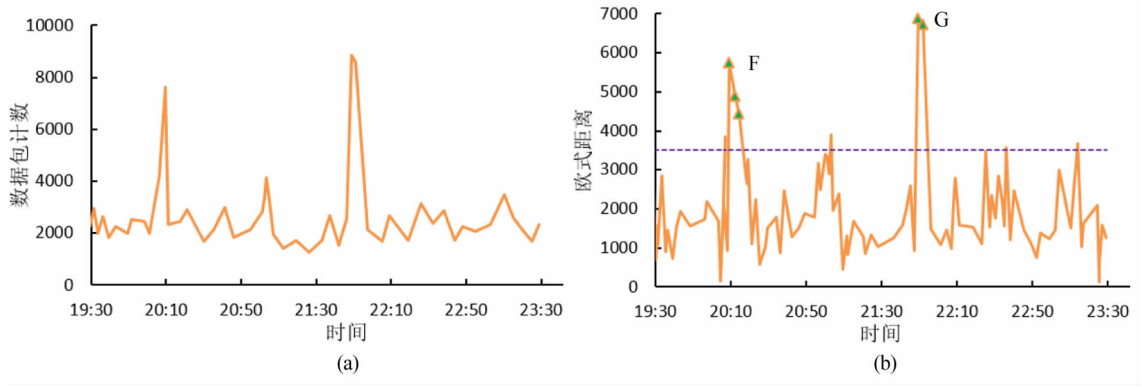


图 7 周四晚上 20:10、21:50 出现异常

实验结果证明本文模型可以有效检测到 4 个案例(图 4~图 7)中的网络流量异常情况。还有一些程度不同的攻击出现在流量平面上,本文方法也能检测到。

2.3 性能对比分析

将本文提出的模型性能与文献[7]、文献[8]、文献[9]中的模型进行比较,结果如表 1 所示。使用四个指标来比较性能,即准确率、精确率、召回率和 F1 值。表 1 显示,本文提出的模型准确率达到 95.5%,精确率达到 96.3%,召回率为 98.6%,F1 值为 92.7%,在每个指标上均完胜其他对比模型,

显示本文方法成功检测到 1 异常(E),且 E 包含 4 个异常点和 0 个正常点。实验结果证明本文方法 100%检测到该案例中的网络异常情况,具有较高的检测准确率。

计容易忽略这种异常情况,然而本文提出的方法却成功检测到了此类异常情况,如图 7(b)所示。虽然有些异常点最初不满足异常条件,但本文方法仍能检测到 2 个异常(F 和 G)。F 包含 3 个异常点和 0 个正常点,G 包含 2 个异常点和 0 个正常点,结果表明本文方法 100%检测到异常。

表明本文模型具有最优的性能。

表 1 本文模型与其他模型在四个指标上的比较				
方法	准确率 /%	精确率 /%	召回率 /%	F1 值 /%
文献[7]	84.6	87.1	76.8	81.3
文献[8]	88.3	86.4	80.7	89.5
文献[9]	90.2	88.6	94.5	90.4
本文方法	95.5	96.3	98.6	92.7

图 8 显示了所提模型与其他对比模型的 ROC 曲线。本文的模型通过产生 AUC(ROC 曲线下面

积)分数($AUC=0.98$)证实了其良好的性能,真阳性率非常高,假阳性率较低。

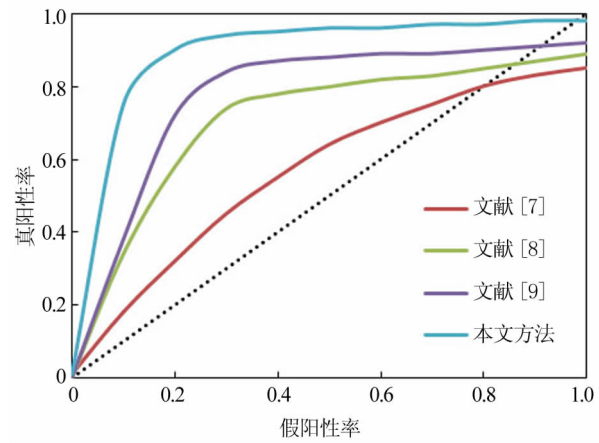


图 8 不同模型的 ROC 曲线

3 结 论

为了充分发挥网络望远镜技术和 LSTM 神经网络的优势,提出了一种创新的网络异常探测方法。该方法通过进行端口聚类 and 排序,推断出具有最高关联度的网络服务,从而运用 LSTM 神经网络对探测率进行建模和预测。所设计的亲和度量能够量化不同端口之间的时间和语义相似性,从而降低预测模型特征空间内的维度和噪声影响。此外,详细介绍了如何借助探测率模型改进对网络望远镜记录中探测活动的监测方法。并且,基于预测模型提出了新的指标,这些指标能够有效地推断与恶意攻击相关的异常流量,并在网络风险较高时触发预警。通过在大数据网络环境中对所提出的模型进行实验评估,验证了其在异常探测方面的能力。通过与其他方法进行对比,进一步证明了所提出模型的优越性和有效性。

参考文献

[1] 张赛男,孙彪. 基于机器学习的网络异常检测方法综述[J]. 吉林大学学报(信息科学版), 2021, 39(6): 732—742.

[2] 翟亚红,崔峻玮. 软件定义网络安全研究进展[J]. 科技导报, 2023, 41(13): 76—88.

[3] TERA I A, CHIBA T, SHINTANI H, et al. Intrusion detection using long short-term memory model for industrial control system[J]. International Journal of Safety and Security Engineering, 2020, 10(2): 183—189.

[4] 蹇诗婕,卢志刚,杜丹,等. 网络入侵检测技术综述[J]. 信息安全学报, 2020, 5(4): 96—122.

[5] 李俊,夏松竹,兰海燕,等. 基于 GRU-RNN 的网络入侵检测方法[J]. 哈尔滨工程大学学报, 2021, 42(6): 879—884.

[6] 周翰逊,陈晨,冯润泽,等. 基于值导数 GRU 的移动恶意软件流量检测方法[J]. 通信学报, 2020, 41(1): 102—113.

[7] 麻文刚,张亚东,郭进. 基于 LSTM 与改进残差网络优化的异常流量检测方法[J]. 通信学报, 2021, 42(5): 23—40.

[8] 董书琴,张斌. 基于深度特征学习的网络流量异常检测方法[J]. 电子与信息学报, 2020, 42(3): 695—703.

[9] 李小剑,谢晓尧,徐洋. 网络流量异常检测方法:SSAE-IWELM-AdaBoost[J]. 武汉大学学报(理学版), 2020, 66(2): 126—134.