文章编号：2096-1618(2022)06-0615-07

基于元学习的僵尸网络检测研究

郭楠馨[1,2]， 林宏刚[3]， 张运理[1,2]， 陈 麟[1,2]

(1. 成都信息工程大学网络空间安全学院，四川 成都 610225；2. 成都信息工程大学先进密码技术与系统安全四川省重点实验室，四川 成都 610225；3. 网络空间安全态势感知与评估安徽省重点实验室，安徽 合肥 230027)

摘要：针对现实网络中僵尸网络流量占比远小于正常网络流量，新出现的僵尸网络类型缺乏标记样本，以及传统深度学习依赖大量标记数据的问题，提出了基于元学习的僵尸网络检测模型，用于小样本场景下的僵尸网络检测。该模型分为特征提取模块和比较模块两个部分，都由CNN实现。特征提取模块从一对网络流量中学习流量特征，包含正常流量和僵尸网络流量，并引入非局部注意力机制，用来捕获长距离依赖关系，提高模型的准确率；比较模块用于获取这对网络流量特征图的相似度得分，进而判断两者是否为同一类型的样本。通过学习一定数量的小样本僵尸网络检测任务，使模型获得足够的先验知识，以便能通过极少量的标记样本实现对未知僵尸网络类型的检测。实验结果表明，1-shot设定下的小样本僵尸网络检测平均准确率达到96.79%，5-shot设定下的小样本僵尸网络检测平均准确率达到98.06%，验证了模型的有效性。

关 键 词：僵尸网络；深度学习；元学习；小样本；注意力机制；CNN

中图分类号：TP309 **文献标志码**：A

doi：10.16836/j.cnki.jcuit.2022.06.001

0 引言

由于互联网的普及，各种网络攻击也日益频繁[1]，攻击者利用僵尸网络执行各种网络犯罪活动[2]。近年来，僵尸网络的对抗性有所提升，针对蜜罐开源项目采取反制措施，在漏洞利用方面更加迅速[3]。现有的僵尸网络家族正在不断进化，各种新型攻击层出不穷[4]。因此，为保障网络环境的安全，对僵尸网络的检测至关重要。

国内外研究者提出了众多基于机器学习算法的僵尸网络检测的方案。周昌令等[5]通过研究校园网上的DNS流量后，提出了18个相关特征，并采用随机森林算法实现FFSN的检测。Khan等[6]提出了一种多层混合技术，根据僵尸网络流量的会话特征，采用基于机器学习的分类器，识别僵尸网络流量。上述方案基于传统机器学习算法，通过人工提取特征实现僵尸网络的检测。然而，面对复杂多变的网络世界，人工提取特征存在难度大、通用性不够等缺点。因此，有研究者提出基于深度学习的僵尸网络检测方法。McDermott等[7]使用双向长期短期记忆递归神经网络(BLSTM-RNN)，并结合词嵌入实现Mirai僵尸网络的检测。Chen等[8]使用CNN进行特征提取，并使用决策树算法进一步提高检测率，实验结果表明，卷积特征对于僵尸网络检测是有效的。牛伟纳等[9]提出了一种结合CNN和RNN两种神经网络的方法，可以提取时间与空间两个维度上的特征。

基于深度学习的僵尸网络检测方法虽然取得好的效果，但依赖大量的样本训练[10]。然而在现实的网络环境中标记的僵尸网络样本不够充分，且占比很低。例如，零日僵尸网络是在漏洞发现当天发起的僵尸网络攻击，短时间内难以获取足够多的攻击样本，可以将其视为小样本僵尸网络检测问题。小样本学习旨在通过几个甚至一个样本完成任务，现在常用元学习方法来解决小样本问题[11]。元学习能够通过学习大量的任务，获得足够的先验知识，从而快速学会新的任务[12]。Koch等[13]提出了孪生神经网络(siamese neural networks)，将两个结构相同的神经网络拼接，且共享权值，比较输入两个样本的相似度。Snell等[14]提出了原型网络(prototypical networks)，通过神经网络将D维数据映射到M维的特征空间，新的特征向量的均值心作为每类数据的原型点，并用欧几里得距离计算原型点之间的距离。Sung等[15]提出了关系网络(relation network)，通过特征嵌入模块提取样本特征，并用关系模块计算样本特征之间的相似度，两个模块都采用CNN。该方法在计算样本间度量没有预定义一个固定的度量方法，而是通过神经网络学习一个度量函数，使模型表达更准确。Vinyals等[16]提出了匹配网

收稿日期：2022-01-11
基金项目：网络空间安全态势感知与评估安徽省重点实验室开放课题资助项目(CSSAE-2021-002)

络(matching network),该方案将注意力机制和长短期记忆网络(long short-term memory,LSTM)相结合,将支持集和查询集输入 CNN 提取图像特征,然后输入双向 LSTM 中获取图像的特征向量,最后用余弦距离注意力判断样本相似度。上述方法都是基于度量的元学习,在小样本分类问题上取得了不错的成果。

近几年,注意力机制在小样本学习中的应用也越来越广泛。注意力机制的核心思想是忽略无关信息关注重点信息[17],能够增强模型的鲁棒性、泛化性和可解释性[18]。Ren 等[19]提出了注意力吸引网络,在预训练好的初始分类器面对新的分类任务时,会训练一组新的权重,在不遗忘旧的分类任务的前提下实现对新类型的分类。Gao 等[20]提出了一种基于混合注意力机制的原型网络,包括实例级注意力用于在支持集中选择拥有更多信息的实力,降低噪声干扰的问题;特征级注意力着重关注特征的重要维度,缓解特征的稀疏性。Wu 等[21]在关系网络的基础上引入自注意力机制,用于获取特征的非局部信息。

为解决传统深度学习对未经学习的任务泛化能力不足、依赖大量标记数据的问题,本文根据关系网络[15]的思想,提出基于度量的元学习的僵尸网络检测方法。通过简单的卷积神经网络提取网络流量特征,并学习出一个度量函数,用于比较样本间的相似度。在面对未知的僵尸网络类型时,能通过少量的样本信息快速完成分类任务。由于卷积神经网络中的感受野仅覆盖局部信息,因此在提取特征时引入非局部注意力机制[22],可以提取网络流量特征的全局信息,考虑网络流量各点之间的联系,使僵尸网络的检测结果更加准确。

1 基于元学习的僵尸网络检测方法

提出基于元学习和注意力机制的僵尸网络检测方法,整体流程如图 1 所示。首先对网络流量数据进行预处理,将处理后的数据输入特征提取模块,在得到特征图后还需将其输入比较模块,获取两个样本之间的相似度。当模型获得足够多的先验知识,便能利用很少的样本完成新的分类任务。

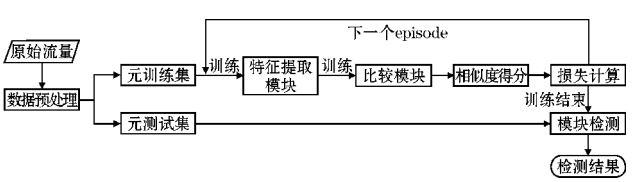

图 1 检测方法流程图

元学习在训练和测试阶段的基本单元为元任务,而非单个样本。元任务可以表示为 N-way、K-shot 小样本问题,其中 N-way 表示 N 个类型的数据,K-shot 表示每种类型数据包含 K 个样本数量,且 K 较小。本文数据集区别于传统机器学习的数据集,由多个元任务组成元训练集和元测试集,其中元训练集和元测试集的样本类型不同。每个元任务包含支持集 $S=\{(x_i,y_i)\}_i^{N\times K}$,$(y_i=0,1)$,查询集 $Q=\{(x_j,y_j)\}_j^{N\times B}$,$(y_j=0,1)$。支持集表示 $N\times K$ 个已知标签的样本,查询集表示 $N\times B$ 个待检测的样本。

Vinyals 等[16]提出了一种用于元学习模型的训练策略。在训练阶段,每个元任务都模拟测试阶段的 N-way、K-shot 问题,进行周期性的迭代训练,学习足够的元任务 $T=\{\text{task}_1,\text{task}_2,\cdots,\text{task}_n\}$,直至收敛。在面对新任务时,通过极少量的新样本,便能快速学会新任务 task_{n+1}。

1.1 数据预处理

1.1.1 网络流量处理

实验使用的数据为原始的网络流量,需经过预处理后输入模型,具体流程如图 2 所示。将原始流量按照五元组(源 IP、目的 IP、源端口、目的端口、协议)切分为多个子集[23],每个子集包含多个数据包,按照时间顺序排列组成一个数据流。由于每条数据流的大小不一,而检测模型要求输入固定大小的数据,因此还需对数据流进行采样。数据流的前面部分包含了主要的建立连接过程和内容交换信息,后面部分能提供的特征信息较少,所以本文截取数据流的前 784 字节,不足则补 0x00。此外,还需对数据流进行匿名化和图像化处理。

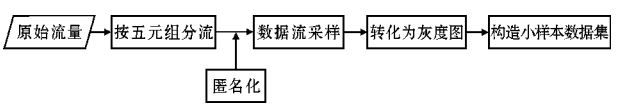

图 2 数据预处理流程图

数据匿名化:本文使用的数据集为实验室网络采集的流量数据,IP 地址和 MAC 地址单一且固定,在模型进行特征提取时会干扰实验结果,因此对 IP 地址和 MAC 地址进行随机化处理。

转换为灰度图:将统一大小后的数据按照每个字节二进制值转换为灰度图,其中 0xFF 代表黑色,0x00 代表白色[23]。

1.1.2 小样本数据集构造

在僵尸网络检测中,每次训练都从元训练集中任意选取一种僵尸网络类型,并从选取的僵尸网络类型中随机采样 $K+B$ 个恶意样本。再从元训练集中随机采样 $K+B$ 正常样本。其中,K 个恶意样本和 K 个正常

样本作为支持集,剩下的 $2B$ 个样本作为待检测的查询集。这样的一次采样训练过程称为一个 episode。

元测试集中的任务是在训练过程中没有出现的新任务,其支持集和查询集构造方法与元训练集相同。

1.2　模型总体框架

本文提出的僵尸网络检测模型分为特征提取模块和比较模块两个部分,模型总体框架如图 3 所示。将通过预处理后的查询集中的样本 x_i 和支持集中的样本 x_j 输入特征提取模块中,生成特征图 $f(x_i)$ 和 $f(x_j)$,再将特征图串联得到 $[f(x_i),f(x_j)]$。将 $[f(x_i),f(x_j)]$ 输入到比较模块中,得到两个样本的相似性得分 $g([f(x_i),f(x_j)])$,范围是[0, 1]。样本集包含正常流量样本(负样本)和僵尸网络样本(正样本),标签分别为 0 和 1。

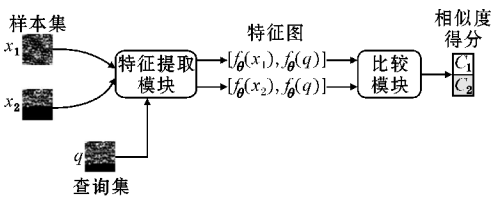

图 3　模型总体框架

查询集中的样本会和支持集中的样本一一比较,得到与 K 个正样本的相似性得分均值:

$$C_m=\frac{1}{K}\sum g_\alpha([f_\theta(x_i),f_\theta(x_{j_m})]) \tag{1}$$

与 K 个负样本的相似性得分均值:

$$C_n=\frac{1}{K}\sum g_\alpha([f_\theta(x_i),f_\theta(x_{j_n})]) \tag{2}$$

式(1)、(2)中 x_{j_m} 和 x_{j_n} 分别表示正样本和负样本。

在训练过程中得到 C_m 和 C_n 后,进行误差反向传播,直至收敛(见图 1)。在测试过程中,将元测试集输入训练好的模型中,得到 C_m 和 C_n 后,比较 C_m 和 C_n 的大小。若 C_m 大,预测标签为 1;反之,标签为 0。

1.3　模型网络结构

特征提取模块和比较模块都由卷积神经网络实现,网络结构如图 4 所示。特征提取模块由 4 个卷积层和注意力机制组成,其中 Block 表示卷积层,Attention Block 为非局部注意力机制。图 4 中“Conv,3×3,64”表示卷积操作,卷积核的大小为3×3,通道数为 64;“BN,64”表示批量标准化,通道数为 64;“ReLU”表示使用修正线性单元作为激活函数;“Max-Pool,2”表示最大池化,池化核大小为2×2。比较模块中“Concatenation”表示将特征提取模块输出的两个特征图串联起来;“FC,8”和“FC,1”表示全连接层维度分别为 8 和 1;输出层通过 Sigmoid 函数得到一个实数。训练过程中,不使用预设的固定的线形度量算法,如欧式距离、余弦距离等,而是学习出一个非线性的相似度度量,使模型在面对多种僵尸网络类型时效果更好。

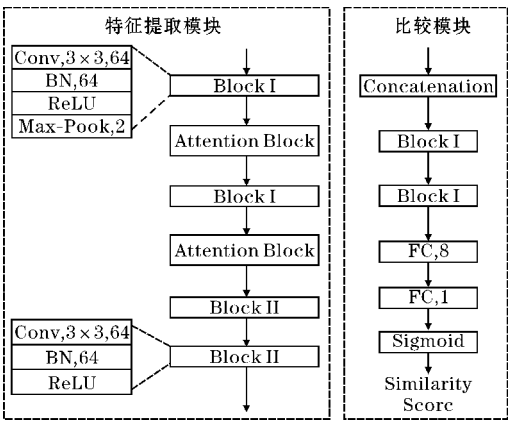

图 4　僵尸网络检测模型网络框架图

特征提取模块是一个双路处理的卷积神经网络,将两个尺寸为 28×28×1 的样本图片输入到特征提取模块中得到两个尺寸为 5×5×64 的特征图;再将特征提取模块输出的两个特征图输入到比较模块中,通过 Concatenation 将两个特征图串联,得到尺寸为 5×5×128 的特征图;特征图经过两个卷积层和两个全连接层,最后得到两个样本的相似度得分。

1.4　非局部注意力机制

在卷积神经网络中,感受野只考虑了局部范围,如卷积和池化操作。为捕获远距离位置的依赖关系,通常做法是叠加多个卷积模块。但导致感受野的效率低,且增加网络层数会提高网络设计难度,因此本文在特征提取模块引入非局部(Non-Local)注意力机制[22]。Non-local 操作对于二维图片,可以捕获空间位置的长范围依赖,实现远距离的信息传递,获取更多信息,具体的实现为

$$Y_i=\frac{1}{N}\sum_{\forall j}f(X_i,X_j)g(X_j) \tag{3}$$

式中:X 表示输入特征图,Y 表示输出特征图,且两者大小相同;i 表示 X 某一像素点的索引,j 表示除 i 外所有位置的索引;N 表示 X 中像素点个数,用来进行归一化处理。函数 f 计算 i 和 j 之间的相似关系,采用嵌入高斯公式:

$$f(X_i,X_j)=e^{\theta(X_i)^T\phi(X_j)} \tag{4}$$

式中,公式嵌入项 $\theta X_i=W_\theta X_i$,$\phi(X_j)=W_\phi X_j$。函数 g 用来计算 j 在 X 中的特征表示,可以看作 1×1 的卷积:

$$g(X_j)=W_gX_j \tag{5}$$

图 5 展示了 Non-Local 模块实现的具体过程。输

入特征图 X 的长和宽分别为 H、W,通道数为 C,批量大小为 N。X 首先经过 3 个 1×1 的卷积核,即经过 3 个线性变化 θ、ϕ、g 得到 $\theta(X_i)$、$\phi(X_j)$、$g(X_j)$,使通道数减半,减少计算量;对 $\theta(X_i)$、$\phi(X_j)$ 进行维度变换,再用 softmax 函数进行归一化处理:

$$f' = \text{softmax}(\theta(X_i) \otimes \phi(X_j)) \tag{6}$$

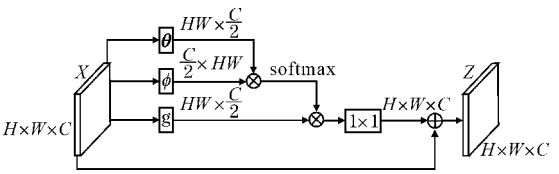

图 5 Non-local 模块示意图

对得到的 $g(X_j)$ 先进行维度变换,再和 f' 进行矩阵相乘得到 Y;最后将 Y 再进行维度变换,并经过一个 1×1 的卷积核得到 Z:

$$Z_i = W_Z Y_i + X_i \tag{7}$$

Z 和 X 的大小和通道数相同,这样便于将 Non-Local 块引入现有的网络结构中。

2 实验结果与分析

2.1 实验数据与环境

实验使用的数据集为 ISOT Botnet 数据集[24] 和 CTU-13 数据集[25]。ISOT Botnet 数据集包含了 P2P 僵尸网络和正常流量。CTU-13 数据集包含了 13 个不同场景下的僵尸网络流量。本文实验采用其中的正常流量和部分僵尸网络流量(Neris、Rbot、Waledac、Zeus、Virut、Fast-Flux)。

实验环境为:深度学习框架 Pytorch;处理器是 CPU:i7-7700,GPU:1080TI。

2.2 评价指标

本文采用准确率(accuracy,ACC)、检测率(detection rate,DR)、误报率(false alarm rate,FAR)作为评价指标。

$$\text{ACC} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \tag{8}$$

$$\text{DR} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{9}$$

$$\text{FAR} = \frac{\text{FP}}{\text{FP}+\text{TN}} \tag{10}$$

式中:TP 表示僵尸网络流量被分类为僵尸网络流量个数;FP 表示僵尸网络流量被分类为正常流量个数;TN 表示正常流量被分类为正常流量个数;FN 表示正常流量被分类为僵尸网络流量个数。

2.3 实验设置

实验采用 2.1 节的数据集,其中 Neris 和 Rbot 为 IRC 僵尸网络;Waledac 和 Zeus 为 P2P 僵尸网络;Virut 和 Fast-Flux 为 HTTP 僵尸网络。为了模拟小样本场景下的僵尸网络检测,训练数据和测试数据不能重叠,且僵尸网络类型不能相同。因此将上述按照协议分类的 3 种僵尸网络中的 1 种作为元测试集,剩余 2 种僵尸网络作为元训练集,共有 3 种组合方式。例如,将 P2P 僵尸网络(Waledac、Zeus)和 HTTP 僵尸网络(Virut、Fast-Flux)作为元训练集,IRC 僵尸网络(Neris、Rbot)作为元测试集。由于在训练过程中没有使用 IRC 僵尸网络样本,因此在测试过程中将 IRC 僵尸网络视为新的僵尸网络类型,即小样本场景下的僵尸网络检测。

在训练和测试过程中,设置 $K=2$,$N=1$、5,$B=10$。通过完成 2-way、1-shot 和 2-way、5-shot 的小样本实验,来验证小样本场景下僵尸网络检测的可行性。

虽然将僵尸网络检测看作二分类问题,最后输出 0 或 1,但是比较模块输出的是预测的相似度得分,不是标签,可以看作回归问题。因此,使用均方误差(MSE)训练模型:

$$\text{MSE} = \frac{1}{2N}\sum_{i=1}^{N}\sum_{j=1}^{K}(\hat{y}_{ij} - y_{ij})^2 \tag{11}$$

式中,N 为 episode 个数,K 为小样本数据集中每类样本个数,$\hat{y}_{ij}$ 为模型预测的待测样本相似度得分,y_{ij} 为待测样本标签。训练过程中使用 Adam 优化算法,并将学习率设置为0.001。图 6 表示在训练过程中准确率的变化曲线,在 150 个 episode 左右时,模型已接近收敛。因此,采用 150 个 episode 训练模型。

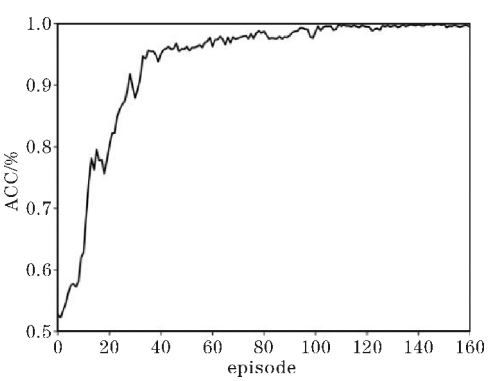

图 6 训练准确率变化曲线注意力机制设置

为了选择合适的位置引入 Non-local 注意力机制,在 2-way、5-shot 的小样本场景下做了实验验证。Baseline,不引入注意力机制。为了验证注意力机制位置对模型的影响,在特征提取模块的不同位置引入注意力机制。

模型 1:在第 1 个卷积层后引入一个 Non-local 注意力机制;模型 2:在第 2 个卷积层后引入一个 Non-local 注意力机制;模型 3:在第 3 个卷积层后引入一个 Non-local 注意力机制;模型 4:在第 4 个卷积层后引入一个 Non-local 注意力机制。

实验结果如图 7 所示,引入 Non-local 注意力机制模块对模型的检测准确率有所提升,在第一个卷积层后引入 Non-Local 模块,即模型 1 的效果最好。

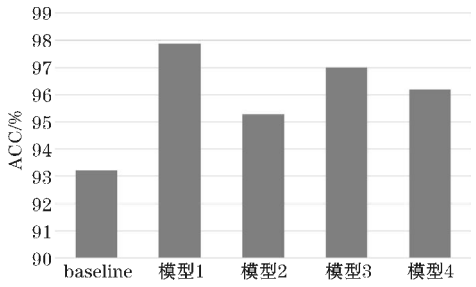

图 7　不同位置的 Non-local 模块测试结果对比图

为验证注意力机制个数对模型的影响,在特征提取模块中引入不同个数的 Non-Local 注意力机制。

模型 5:在第 1 个卷积层后引入一个 Non-local 注意力机制;模型 6:在前 2 个卷积层后都引入一个 Non-local 注意力机制;模型 7:在前 3 个卷积层后都引入一个 Non-local 注意力机制;模型 8:在前 4 个卷积层后都引入一个 Non-local 注意力机制。

实验结果如图 8 所示,Non-local 注意力机制模块数量增加到 2 个以上时,对模型的检测准确率提升很小。根据图7和图8的结果对比,考虑到模型检测效果和计算复杂度,选择模型 6 作为本文方法,具体的网络结构如图 4 所示。

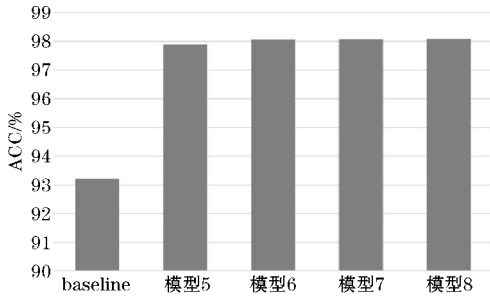

图 8　不同数量的 Non-local 模块测试结果对比图

2.4　实验结果

实验采用2.2节的 3 个指标,将 2way-1shot 和 2way-5shot 小样本场景下的僵尸网络检测进行 100 次实验,将平均值作为最后的结果汇总于表 1。对于 IRC 僵尸网络和 HTTP 僵尸网络的检测,1shot 和 5shot 实验都取得了较好的结果。但对于小样本场景下 P2P 僵尸网络的检测准确率还有待提高。在 1shot 的僵尸网络检测任务中,平均准确率为96.79%,平均检测率为96.68%,误报率为3.32%。在 5shot 的僵尸网络检测任务中,平均准确率为98.06%,平均检测率为98.58%,平均误报率为2.43%。从实验结果来看,对于缺少样本的情况下,小样本场景下的僵尸网络检测是可行的。

表 1　小样本场景下的僵尸网络检测结果汇总表　　单位:%

僵尸网络	类型	ACC		DR		FAR	
		1shot	5shot	1shot	5shot	1shot	5shot
Neris	IRC	97.70	99.25	97.53	99.19	2.22	0.56
Rbot	IRC	99.28	99.32	98.62	99.17	0.68	0.78
Waledac	P2P	93.38	95.78	93.39	97.56	6.92	5.59
Zeus	P2P	93.05	95.31	93.19	97.60	7.68	7.14
Virut	HTTP	98.84	99.28	98.83	98.83	1.16	0.36
Fast-Flux	HTTP	98.92	99.36	98.86	98.95	1.08	0.22
均值		96.79	98.06	96.68	98.58	3.32	2.43

由于基于小样本的僵尸网络检测还处于比较新的领域,目前还没有相关研究成果进行直接对比。因此,将本文方法与其他基于深度学习并使用 ISOT Botnet 数据集和 CTU-13 数据集的僵尸网络检测方法进行比较,对比结果如表 2 所示。Ahmed 等[26]采用简单的人工神经网络(artificial neural network,ANN),对僵尸网络的检测准确率仅 95%。牛伟纳等[9]和 Nugraha 等[27]将 CNN 和 LSTM 算法相结合,用来提取流量的空间和时间两种特征,虽然取得了不错的效果,但这需要大量的标记数据训练,且模型的计算复杂度较高。而本文方法在面对新的僵尸网络类型时,仅需 5 个样本便能取得很好的检测效果,相对于其他方法不需要

太多标记样本。

表 2 本文方法与使用相同数据集的其他方法对比

研究者	方法	数据集	样本量	ACC/%
Ahmed 等[26]	ANN	CTU-13	3000	95.00
牛伟纳等[9]	CNN+LSTM	CTU-13 和 ISOT	8232	98.36
Nugraha 等[27]	LSTM	CTU-13	84117	97.33
Nugraha 等[27]	CNN	CTU-13	84117	97.64
Nugraha 等[27]	CNN+LSTM	CTU-13	84117	98.67
郭楠馨等	本文方法	ISOT 和 CTU-13	5	98.06

3 结束语

为检测小样本场景下的僵尸网络,提出了一种基于度量元学习僵尸网络检测模型。该模型通过卷积神经网络学习一个非线性的相似度度量算法,能更加准确地表达样本之间的相似度关系。同时,为了在获取网络流量特征的全局信息,引入了非局部注意力机制,进一步提升了对僵尸网络检测的准确率。在后续的研究工作中,还会继续改进和优化模型,特别是提高对 P2P 僵尸网络的检测效果。

参考文献:

[1] 冯贵兰,李正楠,周文刚. 大数据分析技术在网络领域中的研究综述[J]. 计算机科学,2019,46(6):1-20.

[2] 张蕾,李井泉,曲武,等. 基于 SparkStreaming 的僵尸主机检测算法[J]. 计算机应用研究,2016,33(5):1497-1503.

[3] 姜建国,王继志,孔斌,等. 网络攻击源追踪技术研究综述[J]. 信息安全学报,2018,3(1):111-131.

[4] 高见,王安. 面向网络攻击的能力评估分类体系研究[J]. 计算机应用研究,2020,37(8):2449-2454.

[5] 周昌令,陈恺,公绪晓,等. 基于 Passive DNS 的速变域名检测[J]. 北京大学学报(自然科学版),2016,52(3):396-402.

[6] Khan R U,Zhang X,Kumar R,et al. An adaptive multi-layer botnet detection technique using machine learning classifiers[J]. Applied Sciences,2019,9(11):2375.

[7] McDermott C D,Majdani F,Petrovski A V. Botnet detection in the internet of things using deep learning approaches[C]. 2018 international joint conference on neural networks(IJCNN). IEEE,2018:1-8.

[8] Chen S C,Chen Y R,Tzeng W G. Effective botnet detection through neural networks on convolutional features[C]. 2018 17th IEEE International Conference On Trust,Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering(TrustCom/BigDataSE). IEEE,2018:372-378.

[9] 牛伟纳,蒋天宇,张小松,等. 基于流量时空特征的 fast-flux 僵尸网络检测方法[J]. 电子与信息学报,2020,42(8):1872-1880.

[10] Hospedales T,Antoniou A,Micaelli P,et al. Meta-learning in neural networks:A survey[J]. arXiv preprint arXiv:2004.05439, 2020.

[11] Huisman M,Van Rijn J N,Plaat A. A survey of deep meta-learning[J]. Artificial Intelligence Review,2021:1-59.

[12] Peng H. A Comprehensive Overview and Survey of Recent Advances in Meta-Learning[J]. arXiv preprint arXiv:2004.11149,2020.

[13] Koch G,Zemel R,Salakhutdinov R. Siamese neural networks for one-shot image recognition[C]. ICML deep learning workshop,2015.

[14] Snell J, Swersky K, Zemel R. Prototypical networks for few-shot learning[J]. Advances in neural information processing systems,2017,30.

[15] Sung F,Yang Y,Zhang L,et al. Learning to compare: Relation network for few-shot learning[C]. Proceedings of the IEEE conference on computer vision and pattern recognition,2018:1199-1208.

[16] Vinyals O,Blundell C,Lillicrap T,et al. Matching networks for one shot learning[J]. Advances in neural information processing systems,2016,29.

[17] Ji Y,Zhang H,Wu Q J. Salient Object Detection via Multi-Scale Attention CNN[J]. Neurocomputing,2018,322(17):130-140.

[18] 刘颖,雷研博,范九伦,等. 基于小样本学习的图像分类技术综述[J]. 自动化学报,2021,47(2):19.

[19] Ren M,Liao R,Fetaya E,et al. Incremental few-shot learning with attention attractor networks[J]. Advances in Neural Information Processing Systems,2019,32.

[20] Gao T, Han X, Liu Z, et al. Hybrid attention-based prototypical networks for noisy few-shot re-

lation classification[C]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(1):6407-6414.

[21] Wu Z, Li Y, Guo L, et al. PARN: Position-aware relation networks for few-shot learning[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019:6659-6667.

[22] Wang X, Girshick R, Gupta A, et al. Non-local neural networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018:7794-7803.

[23] 王伟. 基于深度学习的网络流量分类及异常检测方法研究[D]. 安徽:中国科学技术大学, 2018.

[24] Saad S, I Traoré, Ghorbani A A, et al. Detecting P2P botnets through network behavior analysis and machine learning[C]. Ninth International Conference on Privacy. IEEE, 2011.

[25] Garcia S, Grill M, Stiborek J, et al. An empirical comparison of botnet detection methods[J]. computers & security, 2014, 45:100-123.

[26] Ahmed A A. Botnet Detection Using a Feed-Forward Backpropagation Artificial Neural Network[C]. International Conference on Computational Intelligence in Information System. Springer, Cham, 2018:24-35.

[27] Nugraha B, Nambiar A, Bauschert T. Performance Evaluation of Botnet Detection using Deep Learning Techniques[C]. 2020 11th International Conference on Network of the Future (NoF). IEEE, 2020:141-149.

Botnet Detection Method based on Meta-Learning Network

GUO Nanxin[1,2], LIN Honggang[3], ZHANG Yunli[1,2], CHEN lin[1,2]

(1. College of Cyberspace Security, Chengdu Univ. of Info. Technol., Chengdu 610225, China; 2. Advanced Cryptography and System Security Key Lab. of Sichuan Province, Chengdu Univ. of Info. Technol, Chengdu 610225, China; 3. Anhui Province Key Lab. of Cyberspace Security Situation Awareness and Evaluation, Hefei 230027, China)

Abstract: In view of the fact that the proportion of botnet traffic in real network world is far less than that of normal network traffic, the new types of botnet lack of labeled sufficient samples, and the traditional deep learning relies on a large number of labeled data for training, a botnet detection model based on metric meta-learning is proposed for botnet detection in few-shot scenarios. The model is divided into feature extraction module and comparison module, which are implemented by convolutional neural network (CNN). In the feature extraction module, network traffic features are learned from a pair of network traffic as the input of the comparison module, including normal traffic and botnet traffic, and Non-Local attention mechanism is introduced to capture long-range dependencies and improve the accuracy of the detection model; The comparison module is used to obtain the similarity score of the two network traffic feature maps, and then judge whether they are the same type of samples. By learning a certain number of small sample botnet detection tasks, the model can obtain enough prior knowledge to detect unknown botnet types through a very small number of traffic samples. The experimental results show that the average accuracy of few-shot botnet detection under 1-shot setting is 96.79%, and the average accuracy of few-shot botnet detection under 5-shot setting is 98.06%, which verifies the effectiveness of the model.

Keywords: botnet; deep learning; meta-learning; few-shot; attention mechanism; CNN