计算机科学
COMPUTER SCIENCE

使用RAP生成可传输的对抗网络流量

杨有欢, 孙磊, 戴乐育, 郭松, 毛秀青, 汪小芹

引用本文

杨有欢, 孙磊, 戴乐育, 郭松, 毛秀青, 汪小芹. 使用RAP生成可传输的对抗网络流量[J]. 计算机科学, 2023, 50(12): 359-367.

YANG Youhuan, SUN Lei, DAI Leyu, GUO Song, MAO Xiuqing, WANG Xiaoqin. Generate Transferable Adversarial Network Traffic Using Reversible Adversarial Padding [J]. Computer Science, 2023, 50(12): 359-367.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221000155

使用 RAP 生成可传输的对抗网络流量

杨有欢[1,2]　孙　磊[2]　戴乐育[2]　郭　松[2]　毛秀青[2]　汪小芹[2]
1 郑州大学网络空间安全学院　郑州 450000
2 信息工程大学密码工程学院　郑州 450001
(202012332015247@gs.zzu.edu.cn)

摘　要　越来越多的深度学习方法被用于解决网络流量分类任务,但同时也带来了对抗网络流量(ANT)的威胁。对抗网络流量会使基于深度学习方法的网络流量分类器预测错误,进而导致安全防护系统做出错误的决策。视觉领域的对抗攻击算法虽然也可以运用于网络流量上产生对抗网络流量,但是这些算法产生的对抗扰乱会改变网络流量的头部信息,使得网络流量丢失了自己的特有属性和信息。文中分析了对抗样本在网络流量任务和视觉任务上的不同之处,提出了适用于对抗网络流量的攻击算法 Reversible Adversarial Padding(RAP)。RAP 利用网络流量 Packet 长度和网络流量分类器输入长度的不同,在尾部填充区域填充没有-ball 限制的对抗扰乱。并且,为了解决无法比较不同长度的对抗扰乱会导致不同攻击效果的问题,文中提出了指标收益,其综合考虑了对抗扰乱长度和对抗攻击算法强度对分类器攻击效果的影响。结果表明,RAP 不仅保留了网络流量可传递性的属性,而且获得了比传统对抗攻击算法更高的攻击收益。

关键词:深度学习;网络流量;对抗攻击
中图法分类号　TP181

Generate Transferable Adversarial Network Traffic Using Reversible Adversarial Padding

YANG Youhuan[1,2],SUN Lei[2],DAI Leyu[2],GUO Song[2],MAO Xiuqing[2] and WANG Xiaoqin[2]
1 School of Cyber Science and Engineering,Zhengzhou University,Zhengzhou 450000,China
2 School of Cryptography Engineering,Information Engineering University,Zhengzhou 450001,China

Abstract　More and more deep learning methods are used for network traffic classification,at the same time,it also brings the threat of adversarial network traffic(ANT). ANT will make network traffic classifier based on deep learning method predict incorrectly,and then cause the security protection system to make wrong decision. Although the adversarial algorithms in the vision field can be used to generate ANT,the perturbations generated by these algorithms will change the header information of the network traffic,causing the network traffic to lose its attributes and information. In this paper,the differences of adversarial examples between network traffic tasks and vision tasks are analyzed,and an attack algorithm suitable for generating ANT is proposed,i.e.,reversible adversarial padding(RAP). RAP uses the difference between the length of the network traffic packet and the input length of the network traffic classifier to fill the tail padding area with no -ball perturbations. Besides,to solve the problem that it is difficult to compare the effects of different lengths perturbations,this paper proposes gain on evaluating metrics,which comprehensively considers the impact of the length of the perturbations and the strength of the adversarial attack algorithm. Experimental results show that RAP not only retains the property of network traffic transferability but also obtains a higher gain of attack than traditional algorithms.

Keywords　Deep learning,Netwok traffic,Adversarial attack

1　引言

近年来,随着互联网的快速发展,各种终端的网络流量错综复杂,因此,用网络流量分类对网络流量进行管理显得尤为重要。例如,网络运营商(ISPs)通过网络流量分类技术合理分配使用资源,针对不同类型的网络流量提供不同的服务质量。同时该技术与入侵检测[1]、恶意流量检测[2-3]结合能够进一步增强网络空间的安全性。为此,许多学者提出了不同的网络流量分类算法,这些算法可以归类为基于端口的方法、基于载荷的方法,以及基于机器学习(ML)/深度学习(DL)的方法。基于端口的方法,主要使用 TCP/UDP 头部的端口号进行分类,虽然该方法简单、快速,但是现在许多应用已经不再使用熟知端口号,该方法对这些应用的流量无法进行正确分类。基于载荷的方法,主要使用数据包中应用层的有效载荷,大多数的载荷检测法也被称为 Deep Packet Inspection(DPI),此类方法通过对不同协议设计类似于正则表达式的模式进行

到稿日期:2022-10-10　返修日期:2023-03-03
通信作者:孙磊(diyage0418@163.com)

分类，缺点是当有新协议发布时需要重新设计模式，而且通过对流量的载荷部分进行加密就可以使检测出现误判[4]。基于ML/DL的方法，都可以归为统计类方法，主要使用数据包或者数据流中的字节序、统计性的特征进行分类。本文主要使用基于深度学习的方法作为网络流量的分类器，该类方法因简便的部署方式和极高的分类准确率而被工业界及学术界广泛采用。

利用基于深度学习的方法对网络流量进行分类，虽然能够获得较高的分类准确率，但是Szegedy等[5]发现，现有的神经网络非常容易遭受到对抗样本的攻击，对抗样本会故意导致神经网络分类器分类出错，如将恶意流量分类为正常流量进行网络攻击，或者将Streaming类型的流量伪装成Chat类型的流量以减少流量计费，这种类型的对抗样本被称为对抗网络流量。直接将传统生成对抗样本的思想运用在网络流量上是不切合实际的，原因是，如Fast Gradient Sign Method (FGSM)[6]，Basic Iterative Method(BIM)[7]，Projected Gradient Descent(PGD)[8]等普通对抗攻击算法都是基于全局扰动的，这些算法会改变输入的报文头部信息，这就意味着虽然利用这些算法可以在网络流量上创造对抗样本，但其已经不是真正意义上的网络流量了(因为它无法在真实网络环境中进行传输)。为了解决上述问题，本文在普通对抗攻击算法上进行改进，提出了使用对抗填充的攻击算法RAP。该方法通过在网络流量数据包的尾部填充一部分解除了-ball限制的对抗扰乱，在实际的网络流量上实现了可传输/可逆的对抗网络流量(可逆指可以由网络流量图片逆向计算出真实网络流量，在本文和可传输的含义等价)。本文的主要贡献如下：

1)据我们所了解，本文提出的RAP方法首次在真实网络流量上实现了可传输的对抗网络流量，和基于网络流量的特征所产生的对抗网络流量相比，更具有实用价值。

2)提出了针对典型评价指标(Accuracy，Precision，Recall，F1-Score)的收益分数Gain，其很好地解决了无法比较不同长度的对抗扰乱将导致不同攻击效果的问题。并且和传统视觉领域的对抗攻击算法相比，RAP能够利用更少的对抗扰乱填充实现更高的攻击收益。

3)在基于深度学习方法分类网络流量的情况下，本文对网络流量数据包进行了Head-Data-Padding(HDP)抽象定义，该定义为研究产生可传输的对抗网络流量指明了一个可行的方向。

2 背景

网络流量分类、深度神经网络和对抗样本是本文研究内容的重要组成部分，本章主要介绍这三方面的内容。

2.1 网络流量分类

2.1.1 网络流量

广义的网络流量指网络流量数据包的信息(特征)，这一类型的网络流量数据有很多，如KDD CUP1999和NSL-KDD[9]等，它们使用的数据是从原始流量中提取的关键信息。这种数据只能用于网络流量的分析，并不适合用来产生对抗网络流量。本文所讨论的网络流量指的是原始流量(Raw Traffic)，利用的数据均为实际网络流量。Sadeghzadeh等[10]将网络流量正式定义为3种类型，包(Packet)、单向流(Unidirectional Flow)和双向流(Bidirectional Flow)。Packet是实际网络环境下的单个传输实体；Unidirectional Flow是一系列具有相同的源信息、目的信息及运输层协议的Packets集合；Bidirectional Flow是两个单向流的集合，其中IP地址和端口号刚好相反。不同的定义方式，实际上是对网络流量数据的不同粒度的划分，本文在后续的讨论中集中于对网络流量Packet的研究。

2.1.2 数据包分类

网络流量分类可以被用在许多领域，如协议检测(Protocol Detection)[11-12]、网页指纹(Website Fingerprinting)[13-15]、应用识别(Application Identification)[16-19]，以及流量特征(Traffic Characterization)[17,20-22]。本文在USTC-TFC2016数据集上主要关注应用识别任务，在ISCXVPN-2016数据集上主要关注流量特征任务。因为这两种分类方法在很多应用场景下都非常重要，如网络流量的计费、服务质量的提供以及策略的执行等。

在数据包分类任务中，每个包都被分配一个标签，并且每个包的字节序被输入分类器，从而获得预测结果。Lotfollahi等[17]提出了Deep Packet(DP)框架，DP集成了Stacked Denoising Autoencoders(SDAE)和卷积神经网络(CNN)在ISCXVPN-2016网络流量数据集[20]上进行分类。结果表明CNN分类器在Application Identification和Traffic Characterization上的表现要优于SDAE，因此本文也采用CNN系列网络对网络流量数据包进行分类，具体的实施细节见第4章实验部分。

2.2 深度神经网络

深度神经网络(DNN)，如VGG[23]，Google Inception[24-27]以及Resnet[28]等由于具有强大的拟合能力、简单的部署模式，被广泛运用于视觉、音频以及文本等领域的相关任务，并取得了出色的表现。许多研究人员尝试将神经网络运用于网络流量分类任务，结果表明这种方法比传统方法更高效，更出色。因此本文的后续部分也采用DNN作为评估模型。

2.3 对抗样本

尽管各种DNN方法在解决复杂任务方面有很好的表现，但是其很容易遭受到对抗样本的攻击，而且这种对抗攻击对于基于梯度优化的神经网络而言是很难避免的。对抗样本的基本思想是通过对输入x增加对抗扰乱δ使分类器f得到一个非预期的结果，数学描述如下：

$$f(x+\delta)\neq f(x) \tag{1}$$

其中，

$$x+\delta\in\mathbb{R}^d$$

$$x\in\mathbb{R}^d$$

$$\|\delta\|_p\leqslant\epsilon$$

参数用来限制扰乱的大小，使得对抗扰乱不容易被人眼所辨别(一般也称其为-ball限制)。$\|\cdot\|_p$为L_p normal范数，其计算公式如下：

$$\|\delta\|_p=\left(\sum_{i=1}^{n}|\delta_i|^p\right)^{\frac{1}{p}} \tag{2}$$

特别的是：$\|\delta\|_0$为δ中非0元素的个数，$\|\delta\|_\infty$为$|\delta_i|$的最大值。

基于上述思想，Goodfellow 等[6]提出了 FGSM 算法，其使用单步、一阶梯度上升方法产生对抗扰乱。FGSM 为典型的白盒、L_∞ 攻击算法，当网络模型较为简单时，该算法有不错的攻击效果，数学表达式如下：

$$x_{adv}=x+\epsilon\ \mathrm{sgn}(\nabla_x J(\theta,f(x),y)) \tag{3}$$

其中，sgn 为符号函数，θ 为分类器 f 中的可学习参数，y 为真实标签。Kurakin 等[7]在 FGSM 的基础上提出了一种更强的多次迭代的 FGSM，称为 BIM。不同于最后一步进行ϵ限制的 FGSM，BIM 在每次迭代的过程中采用更小的步长 α 来计算，其计算式如下：

$$\begin{cases}x_0=x\\ x_{t+1}=Clip_{x,\epsilon}\{x_t+\alpha\,\mathrm{sgn}(\nabla_{x_t}J(\theta,f(x_t),y))\}\end{cases} \tag{4}$$

其中，t 为迭代次数，$Clip$ 操作是为了将每次迭代过程中的对抗样本限制在$\epsilon-L_\infty$邻域内以及确保其是有效值。

Madry 等[8]提出了理论上最强的一阶白盒攻击算法 PGD。和 BIM 一样，该算法也是多步迭代类型攻击算法，计算流程如下：

$$\begin{aligned}&x_0=x+\delta_0\\ &x_{t+1}=\prod_{x+S}(x_t+\alpha\,\mathrm{sgn}(\nabla_{x_t}J(\theta,f(x_t),y)))\end{aligned} \tag{5}$$

两者之间的主要区别是，PGD 产生对抗样本的起始点不再是原始样本，而是在其基础上加上了一个随机扰动δ_0。$\prod$为 projected 操作，目的是将更新过程中的对抗样本限制在$\epsilon-L_\infty$邻域内以及确保其是有效值（其中 S 为对抗扰乱在$\epsilon-L_\infty$邻域内的可变范围）。

除上述对抗样本生成方法外，还有 Jacobian-based Saliency Map Attack(JSMA)[29]，Carlini & Wagner(C&W)[30]，DeepFool[31]，Zoo[32]，UAP[33]等其他攻击算法，它们都是直接或者间接利用模型的梯度来计算对抗扰乱，本文将 RAP 与典型的攻击算法 FGSM，BIM 以及 PGD 进行对比。

3 可逆对抗填充

基于深度学习的网络流量分类任务同样会遭到对抗攻击的威胁，2.3 节所介绍的对抗样本生成方法一般用在视觉任务上，也可以用在网络流量分类任务上。图像领域的对抗样本计算对抗扰乱时一般是基于全图扰动的，且会设置超参数来限制对抗扰乱在一个范围之内变化。但是网络流量上的对抗样本（即对抗网络流量）和图像上的对抗样本又有不同之处。首先，网络流量在真实世界中传递，对于网络流量包的头部信息，如果对其添加了对抗扰乱，则该网络流量数据包将失去在真实网络中的传输性且该网络流量包将不可被解析。其次，网络流量没有-ball 限制（因为网络流量不存在图片失真的问题），这使得对抗扰乱可以在可行的域内随意变换。针对这些不同之处，本文提出了 Reversible Adversarial Padding (RAP)方法，该方法在网络流量包的尾部空白字段生成对抗扰乱而不改变数据包的头部信息，使得 RAP 在不失去网络流量的传输性的情况下，也可以获得较强的攻击性。

3.1 HDP 定义

为了更好地介绍 RAP 如何产生对抗扰乱，本文对网络流量包进行了 Head-Data-Padding 定义（HDP）：

$$\begin{cases}Packet=(Head,Data)\\ Packet^+=(Packet,Padding)\end{cases} \tag{6}$$

其中，$Packet$ 为实际的网络流量数据包，$Packet^+$ 为神经网络的抽象输入数据。和实际上的输入数据相比，它们的数据长度是一致的，只是内容上稍有不同，这主要是因为一些数据进行了预处理操作。

Head 泛指 Packet 的整个头部信息，一般包括数据链路层、网络层、运输层等。在本文的定义下，Head 部分是不能被对抗扰乱篡改的，保护好 Head 的信息是本文产生的对抗流量可逆的关键。

Data 指应用层数据，该部分是网络流量包的重要部分，它是链路两端需要传递的真正信息。虽然在 Data 上产生对抗扰乱并不会影响对抗流量是否可逆（不会影响到 Head 部分），但是本文并没有选择在 Data 上添加对抗扰乱，因为在 Data 上稍加改动，数据到达应用层之后将无法解析（这在一些特定应用场景下尤为重要）。

Padding 的存在对于计算 RAP 至关重要，它位于实际网络流量包的尾部，其数据长度（P Bytes）取决于网络流量分类器的输入长度（Q Bytes）和实际网络流量数据包的长度（T Bytes），即：

$$P=Q-T \tag{7}$$

根据 P 的大小，可以分为如下两种情况：

1）当 $P>0$ 时，Padding 位置的数据实际存在，这也是本文产生对抗扰乱所选择的位置。

2）当 $P\leqslant 0$ 时，Padding 位置的数据虚拟存在，此时可以将其视为网络流量包被切割部分的数据。

情况 2 由于没有实际的 Padding 位置，无法产生对抗扰乱（事实上，此时依旧可以通过对数据包进行人为分片来满足情况 1 的条件），因此在不加特别说明的情况下，本文后续的讨论及实验均在情况 1 下进行。

3.2 RAP 的计算流程

本文基于 HDP 定义提出了 RAP 方法，在 $Packet^+$ 的 $Padding$ 域上产生对抗扰乱，RAP 的计算流程如图 1 所示。

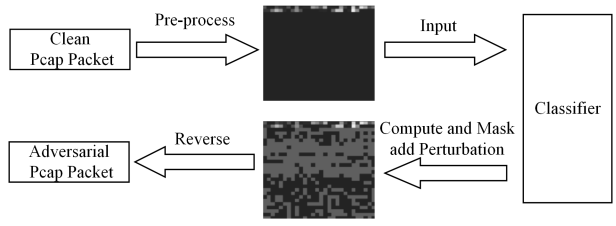


图 1 RAP 计算流程图

Fig. 1 Calculation flow chart of RAP

1）预处理数据包

任何一个不加处理的网络流量包都不适合直接作为分类器的输入，需要经过一些预处理操作。首先随机化 IP 地址、MAC 地址等会导致分类器过拟合的字段；然后在网络流量数据包的尾部填充 0 元素，使其数据长度达分类器的输入的长度 Q（不考虑 $P\leqslant 0$ 的情况）；最后将 $Packet^+$ 中的每个字节作为分类器输入的一个像素点（本文将一个 $Packet^+$ 视为一张单通道的灰度图片）。

2)输入分类器

在对原始的网络流量包完成 1)中的预处理操作后,将网络流量图片经过简单的归一化操作(归一化到[0,1])便输入到分类器中进行预测。本文并没有对网络流量图片采取一些传统图片的预处理操作(随机旋转、切割等),因为那样会改变 $Packet^+$ 的 *Head*,导致网络流量不再可逆。

3)利用掩码计算 ANT

当获得分类器的输出之后,利用损失函数通过反向传播计算出梯度和对抗扰乱,最后利用 *padding_mask* 即可得到在 $Packet^+$ 上的可传输对抗网络流量。

算法 1 展示了生成可逆对抗样本的过程,一共进行 I 轮迭代,每轮迭代都利用梯度和 *padding_mask* 来计算可逆对抗扰乱。其中 *padding_mask* 是可逆的关键,它的形状大小和分类器的输入一致,其在 $Packet^+$ 的 *padding* 所对应的位置取值为 1,其他位置取值为 0。函数 $Rand(Domain(x))$ 代表随机生成一个和 x 具有相同形状大小和取值范围的矩阵(在本文中若未特别说明则输入的取值范围均为[0,1])。

算法 1 可传输对抗网络流量生成算法

输入:数据预处理之后的 x,标签 l,填充掩码 padding_mask,分类器 f 及其内部可学习参数 θ,损失函数 J,最大攻击迭代次数 I 和梯度放缩因子 α

输出:ANT r_x_adv

1. r_x_adv←x+padding_mask? Rand(Domain(x))
2. for i←0 to I do
3. grad←∇J(θ,f(r_x_adv),l)
4. r_x_adv←r_x_adv+α? padding_mask? sign(grad)
5. r_x_adv←$clip_{Domain(x)}$(r_x_adv)
6. end for
7. return r_x_adv

算法 1 中不同的梯度放缩因子 α 对 RAP 的攻击效果影响差异较大,因此本文分别在 ISCXVPN-2016[20] 和 USTC-TFC2016[2] 两个数据集上对 4 种模型结构(Small MLP,Large MLP,Small CNN,Large CNN)进行 α 取值的探索。各种网络模型结构如表 1 所列。

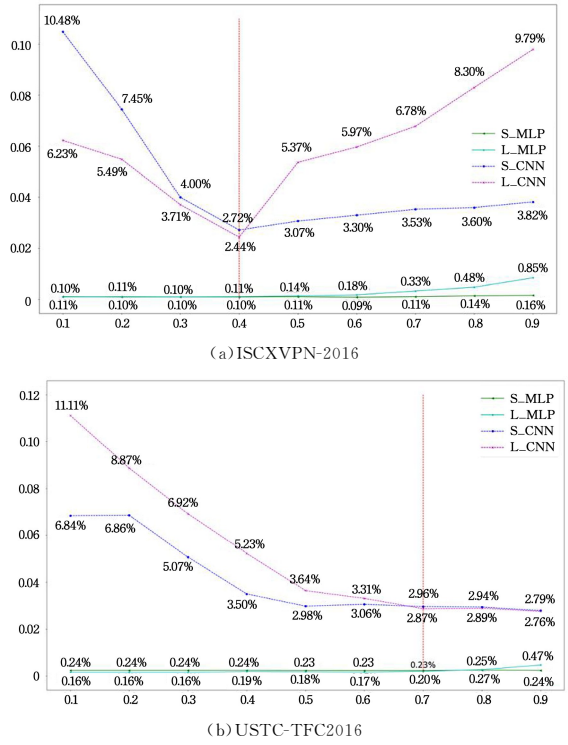
表 1 Small MLP,Large MLP,Small CNN,Large CNN 模型结构

Table 1 Structures of Small MLP, Large MLP, Small CNN and Large CNN

Module	Small MLP	Large MLP	Small CNN	Large CNN
Block1	—	—	Conv(32) MaxPooling(2) Conv(64) MaxPooling(2)	Conv(32) Conv(32) MaxPooling(2) Conv(64) Conv(64) MaxPooling(2)
Block2	FC(1024) FC(512) FC(class_num)	FC(1024) FC(512) FC(256) FC(128) FC(class_num)	FC(1024) FC(512) FC(class_num)	FC(1024) FC(512) FC(256) FC(class_num)

本文将网络模型划分为两部分,即提取特征部分(Block1,对高纬度数据进行特征提取,一般指卷积模块)和分类部分(Block2,对低纬数据进行分类,一般指全连接模块)。表 1 中,Conv()为卷积层,代表卷积核的通道数;MaxPooling()为最大池化层,代表采样窗口大小;FC()为全连接层,代表神经元个数,class_num 代表类别数量。每个卷积层之后都伴随着一个批量正则化层(BN)和 ReLu 激活操作,每个全连接层(不包括最后一个全连接层)之后也伴随着一个 ReLu 激活操作。

此处对数据集的划分和预处理操作和 4.1 节一致,本文将探索不同 α 取值对 RAP 攻击效果影响的实验放在验证集上进行,表 1 中 4 种模型在 ISCXVPN-2016 和 USTC-TFC2016 数据集上均有不错的分类表现。在 ISCXVPN-2016 的验证集上正确率分别达 86.66%,87.50%,83.85%,81.50%;在 USTC-TFC2016 的验证集上正确率分别达 93.49%,92.56%,91.55%,91.11%。不同取值的 α 在迭代次数 $I=20$ 时对 4 种网络模型的攻击效果影响如图 2 所示。



(a)ISCXVPN-2016

(b)USTC-TFC2016

图 2 不同取值的 α 在 ISCXVPN-2016 和 USTC-TFC2016 上的攻击效果

Fig. 2 Attack effects with different α on ISCXVPN-2016 and USTC-TFC2016

在图 2 中,横轴为 α 的不同取值,纵轴为模型在验证集上的正确率。结果表明,在 MLP 模型上,不同取值的 α 对于 RAP 攻击效果来说表现差距不大;但是在 CNN 模型上,α 的不同取值对 RAP 攻击效果的影响明显。在没有特别说明的情况下,后续的实验部分,在 ISCXVPN-2016 上本文选取 α 的值为 0.4,在 USTC-TFC2016 上选取 α 的值为 0.7。

4)逆向对抗网络流量

通过算法 1 可以计算得到 r_x_adv,即可逆的对抗样本/对抗网络流量,只需要对 $Packet^+$ 中 *Head* 的长度、校验等字段进行修改即可。

4 实验

4.1 数据集及处理方法

本文的所有实验均在 ISCXVPN-2016 和 USTC-

TFC2016 数据集上进行。两个数据集都来自于没有经过特征提取的原始真实流量(Pcap 格式)。对数据集的预操作及划分方法如下。

Draper 等[20]提出了由 6 种类型(FileTransfer，Torrent，Chat，Streaming，Email，VoIP)组成的网络流量数据集 ISCX-VPN-2016，每种类型都包含 VPN 以及非 VPN 类型的流量(在一些论文中，研究人员将这两种类型的流量视为不同类别，但本文将这两者视为同类型的网络流量)。ISCXVPN-2016 数据集是高度不平衡的，直接使用全部样本会降低分类器的性能。本文没有采用在较少样本的类别上进行过采样，在较多样本的类别上欠采样的方法[34]，因为这样会导致各类别样本数量不一致。本文采取的做法是，在每个类别上随机选取 30000 个样本(共计 180000 个样本)，之后按类别将数据集划分为 train(60%)，validation(20%)，test(20%)，即在数据集的 3 个子集上也保证了各个类别的样本数量一致。

Wang 等[2]提出了恶意流量数据集 USTC-TFC2016，该数据集包含两类流量，Benign 和 MalWare。Benign 包含 10 种类型的正常流量(Skype，Facetime，WorldOfWarcraft，FTP，Weibo，SMB，MySQL，Gmail，BitTorrent，Outlook)，MalWare 包含 10 种类型的恶意流量(Cridex，Tinba，Virut，Shifu，Neris，Htbot，Miuref，Geodo，Zeus，Nsis-ay)。该数据集是由 CTU 研究者从真实网络环境的公共网站上采集得到的。和 ISCXVPN-2016 一样，该数据集上的各类别样本分布也极不均衡，因此本文从正常流量和恶意流量中的每个类别(共计 20 类)随机挑选 6000 个样本(共计 120000 个样本)，之后每个类别的样本也按照 6∶2∶2 的比例进行划分。

在完成对数据集的划分之后，本文对每个样本都做了 3.2 节中的 Packet 预处理操作。此外，在训练网络模型时本文使用的是 train 集，此时不用考虑样本是需要填充还是截断(即 train 集上的数据被 100%选取)；validation 集用来调整超参数和挑选表现较好的分类器；在测试 RAP 以及其他典型的攻击算法在不同指标上的表现时使用的是 test 集。$P\leqslant 0$ 的样本因不满足 RAP 的条件而被从 test 集中剔除了。在 test 集上的样本选取/剔除情况如表 2 所列。

表 2 test 集上填充长度 P 分布

Table 2 Distribution of P on test set

$Domain(P)$	ISCXVPN-2016	USTC-TFC2016
$(-\infty,0]$	11328(31.47%)	5218(21.74%)
(0,100]	151(0.42%)	400(1.66%)
(100,200]	100(0.28%)	364(1.52%)
(200,300]	285(0.79%)	45(0.19%)
(300,400]	266(0.74%)	295(1.23%)
(400,500]	404(1.11%)	933(3.89%)
(500,600]	649(1.80%)	597(2.49%)
(600,700]	9742(27.06%)	3656(15.23%)
(700,784]	13078(36.33%)	12492(52.05%)

表中第一列为 P 的不同取值范围(范围在$(-\infty,0]$内的数据包即为被剔除的对象)，后两列统计了 ISCXVPN-2016 和 USTC-TFC2016 在各范围内的样本数量以及在整个测试集中所占的百分比。

4.2 实验设置

在 ISCXVPN-2016 数据集和 USTC-TFC2016 数据集上的实验设计基本保持一致，本文选择 LargeCNN 模型作为分类器(该模型在网络流量分类任务中表现出色且能在一定程度上代表卷积神经网络，因此被选为被攻击者模型)。本文对该模型进行了 10 次随机初始化参数然后训练，并保存了 5 个表现较好的模型进行测试取其平均值作为最终结果。在训练过程中 batch size 设为 128，优化器为 Adam[35]，学习率设为 1×10^{-3}，损失函数为 CrossEntropy，训练集上最大迭代次数设为 100。本文没有为分类器设置一些其他的 tricks(如 Dropout[36]，Gradient Clip[37]等)以提高模型的性能/表现，因为这并不是本文研究的重点。

本文使用 3 种典型的白盒对抗攻击算法(FGSM，BIM，PGD)和 RAP 进行攻击效果对比，超参数的设置选取了视觉领域输入大小为 28×28 时所广泛使用的 $eps=0.3$，$eps\_step=0.01$，攻击迭代次数 I 取 20。

4.3 评价指标

Accuracy，Precision，Recall 以及 F1-Score 是衡量分类器表现的典型指标。Accuracy 被用来评估分类器的综合表现，Precision，Recall 和 F1-Score 被用来评估网络流量的每一类的表现。

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

$$Precision=\frac{TP}{TP+FP} \tag{9}$$

$$Recall=\frac{TP}{TP+FN} \tag{10}$$

$$F1\text{-}Score=\frac{2\cdot Precision\cdot Recall}{Precision+Recall} \tag{11}$$

其中，TP 是被正确分类为 X 的样本数量；TN 是被正确分类为 Not-X 的样本数量；FP 是被错误分类为 X 的样本数量；FN 是被错误分类为 Not-X 的样本数量。

尽管本文对数据集的划分做到了各个种类的均衡，但是从表 2 中不难发现，各个网络流量包(样本)的可填充域 $Padding$ 的长度 P 也是非常不均衡的，主要集中在$(-\infty,0]$和$(700,784]$两个域内。然而不同长度的 P 将直接影响 RAP 的攻击效果(显然对抗扰乱区域越大，$Padding$ 将获得越强的攻击效果)，因此直接用上述指标来评估攻击效果是不合理的，为此本文设置了指标收益(Gain)来解决这个问题，其计算表达式如下：

$$Gain=\frac{a-b}{\frac{1}{N}\sum_{i=1}^{N}P_i}\cdot Q \tag{12}$$

其中$0<P_i\leqslant Q$。

式(12)中，a 和 b 分别代表同一种指标(Accuracy，Precision，Recall 或者 F1-Score)在攻击前以及攻击后的取值，N 代表样本的数据量，P_i 代表了第 i 个样本的 $Padding$ 域的长度，$\frac{1}{N}\sum_{i=1}^{N}P_i$ 代表一批样本中的平均扰乱长度。所以，$Gain$ 的含义为平均每个输入样本的单个位置上(对应本文即为每个像素点)可以获得的攻击效果收益分数。当 $b>a$ 时，即 $Gain<0$，

此时代表攻击算法出现了负收益，这一情况只可能发生在 *Precision* 指标上，主要原因是攻击算法可能会导致预测为某一个类别的样本数量 $TP+FP$ 非常小，进而导致 *Precision* 较大，但这并不能说明该算法攻击效果较差，此时可令 *Gain* 为 0。

Gain 的设置一定程度上解决了 *Padding* 长度不均衡对典型评价指标的影响。它主要从两个方面综合考虑了一种攻击算法的攻击收益，即攻击前后评价指标变化差距和 *Padding* 长度 P。值得注意的是，对于传统的攻击算法（FGSM，BIM，PGD 等），平均扰乱长度 $\frac{1}{N}\sum_{i=1}^{N}P_i=Q$，此时 $0\leqslant Gain=a-b\leqslant a\leqslant 1$，即 *Gain* 直接体现了该攻击算法给评价指标带来的下降幅度大小。

对于 RAP 而言，$0<\frac{1}{N}\sum_{i=1}^{N}P_i<Q$，$0\leqslant Gain\leqslant a\cdot Q$，因此理论上 *Gain* 是可以远远大于 1 的（此时要求 $b\to 0$，$\frac{1}{N}\sum_{i=1}^{N}P_i\to 1$，然而这两个条件几乎是一对矛盾的存在，很难同时保持）。所以，实际上 *Gain* 总是在 1 的邻域内浮动。

4.4 结果

4.4.1 在 ISCXVPN-2016 上的结果

本文提出的模型在 ISCXVPN-2016 数据集的原始测试集上各项指标表现如表 3 所列。可以发现，本文模型的 *Accuracy* 只有 79.78%，但这并不代表该模型表现较差。主要原因是本文在 test 集上只挑选了 $P>0$ 的样本（当整个 test 集的样本都参与测试时，*Accuracy* 可以达到 91.29%）。表 3 中，本文模型各项指标除了在 Chat 和 Email 类别上表现较差外，在其他类别上均有不错的表现。

表 3 在 ISCXVPN-2016 上的表现

Table 3 Performance on ISCXVPN-2016

（单位：%）

Kind	*Accuracy*	*Precision*	*Recall*	*F1-Score*
FileTransfer	79.78	93.09	81.10	86.68
Torrent		99.52	83.19	90.62
Chat		66.40	72.72	69.41
Streaming		81.69	83.82	82.74
Email		66.41	75.76	70.78
VoIP		91.65	85.84	88.65

当使用 FGSM，BIM，PGD 以及 RAP 进行攻击测试时，在 *Accuracy* 上的收益分别可达 0.6704（12.74%），0.6477（15.01%），0.7699（2.79%），0.8899（3.00%）（括号内为受到对抗攻击之后的 *Accuracy*）。其他指标的收益如图 3 所示。从图中可以明显看出，本文提出的 RAP 方法在经典评价指标上的收益最高，其次是 PGD 和 BIM，最后是 FGSM。RAP 在保持网络流量的头部信息完整的前提下，利用尽可能少的对抗填充来获得更高的攻击收益。

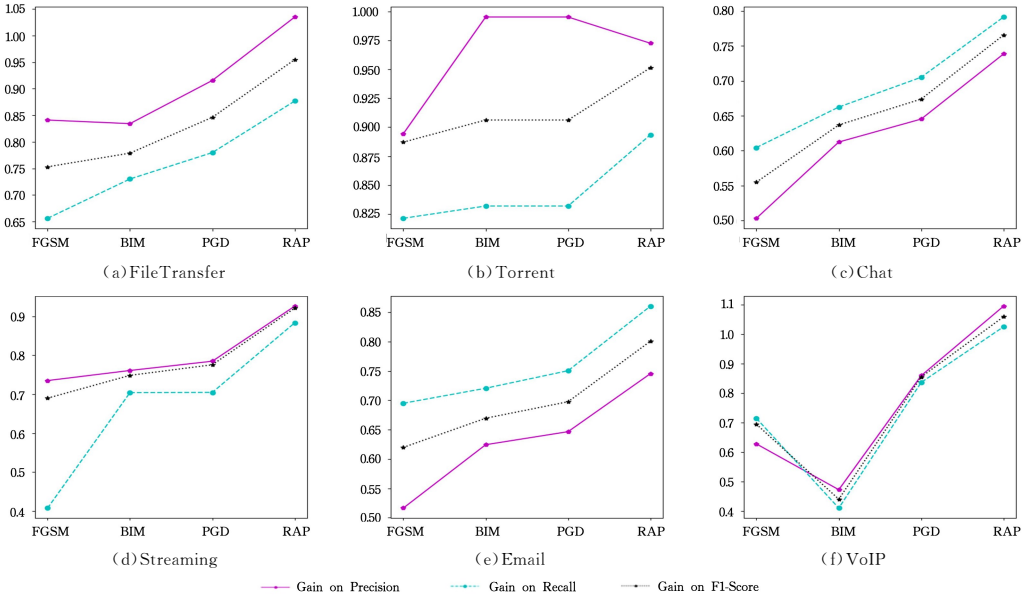


图 3 ISCXVPN-2016 测试集上的 Precision，Recall，F1-Score 收益

Fig. 3 Gain of Precision, Recall and F1-Score on ISCXVPN-2016 test dataset

4.4.2 在 USTC-TFC2016 上的结果

本文提出的模型在 USTC-TFC2016 数据集的原始测试集上各项指标的表现如表 4 所列。从表 4 中可以发现，本文模型除了在 Virut 类别上各项指标表现较差外，在其他类别上均有不错的表现。使用 FGSM，BIM，PGD 以及 RAP 进行攻击测试，在 *Accuracy* 上的收益分别可达 0.8493（5.21%），0.8093（9.21%），0.8573（4.41%），1.054（1.91%）（括号内为受到对抗攻击之后的 *Accuracy*）。其他指标的收益如图 4 所示。

在图 4 中，RAP 方法在 Skype，Facetime，MySQL 和 Miuref类别上的 Precision 收益几乎为 0，通过进一步的实验发现导致这一现象的主要原因是预测为这 4 个类别的样本数量较少（预测结果为 Skype 和 Facetime 类别的样本数量在 150 左右，预测结果为 MySQL 和 Miuref 类别的样本数量只有 10 左右，即 $TP+FP$ 为非常小的值），进而导致对 Precision 的攻击效果较差。但是在大部分情况下（类别下），本文提出的 RAP 方法在各项指标上的收益比其他典型攻击算法要高。

表 4 在 USTC-TFC2016 上的表现

Table 4 Performance on USTC-TFC2016

(单位:%)

Kind	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Skype	90.14	97.76	94.67	96.19
Facetime		100.00	100.00	100.00
WorldOfWarcraft		98.60	99.75	99.17
FTP		99.21	99.21	99.21
Weibo		93.51	81.82	87.27
SMB		88.03	84.43	86.19
MySQL		99.01	99.92	99.46
Gmail		96.50	88.11	92.11
BitTorrent		95.24	100	97.56
Outlook		87.48	97.86	92.38
Cridex		93.86	89.37	91.56
Tinba		99.50	98.92	99.21
Virut		68.62	73.70	71.07
Shifu		99.17	89.16	93.90
Neris		74.58	75.18	74.88
Htbot		80.57	82.92	81.73
Miuref		75.24	85.33	79.97
Geodo		80.41	79.19	79.80
Zeus		89.40	83.80	86.51
Nsis-ay		89.20	89.41	89.31

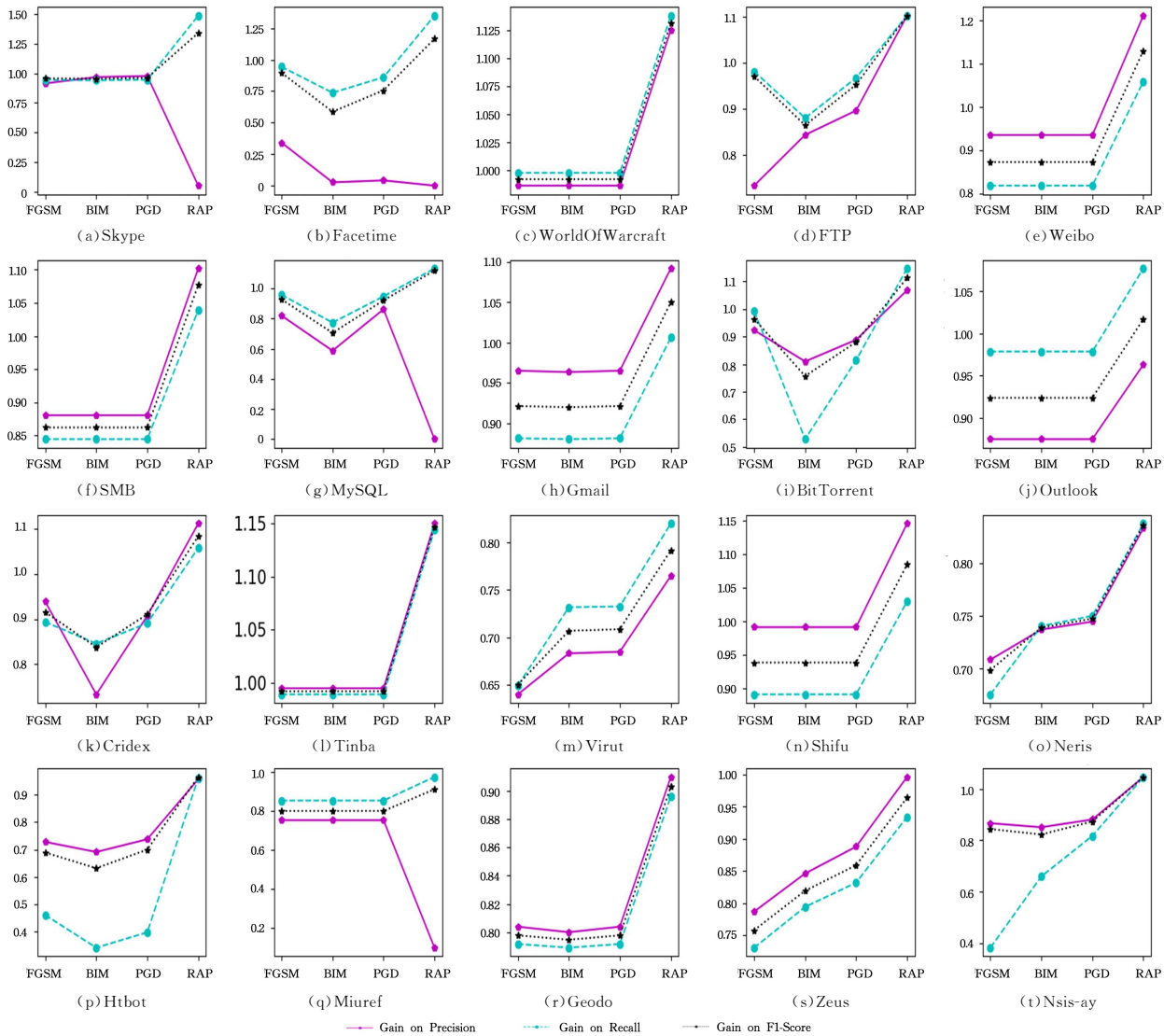


图 4 USTC-TFC2016 测试集上的 Precision,Recall,F1-Score 收益

Fig. 4 Gain of Precision, Recall and F1-Score on USTC-TFC2016 test dataset

结束语 本文首先分析了传统图像领域的对抗样本和对抗网络流量的差异性。其次，针对它们之间的不同之处，本文对网络流量 Packet 进行了 HDP 定义，并在此基础上提出了 RAP 方法。该方法通过在抽象出来的网络流量的 Padding 区域计算没有-ball 限制的对抗扰乱，在保证不丢失网络流量传递性的前提下，实现了对网络流量分类器的攻击。最后本文使用 LargeCNN 模型在 ISCXVPN-2016 和 USTC-TFC2016 数据集上比较了 RAP 和其他经典攻击算法的攻击收益，结果表明，RAP 不仅可以使对抗网络流量可逆，而且具有更高的攻击收益。

参 考 文 献

[1] WANG W, SHENG Y Q, WANG J L, et al. HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection[J]. IEEE Access, 2017, 6: 1792-1806.

[2] WANG W, ZHU M, ZENG X W, et al. Malware traffic classification using convolutional neural network for representation learning[C]// 2017 International Conference on Information Networking (ICOIN). IEEE, 2017: 712-717.

[3] LASHKARI A H, KADIR A F A, GONZALEZ H, et al. Towards a network-based framework for android malware detection and characterization[C]// 2017 15th Annual Conference on Privacy, Security and Trust (PST). IEEE, 2017.

[4] PACHECO F, EXPOSITO E, GINESTE M, et al. Towards the deployment of machine learning solutions in network traffic classification: A systematic survey[J]. IEEE Communications Surveys & Tutorials, 2018, 21(2): 1988-2014.

[5] SZEGEDY C, ZAREMBA W, SUTSKEVER I, et al. Intriguing properties of neural networks[J]. arXiv: 1312.6199, 2013.

[6] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples[J]. arXiv: 1412.6572, 2014.

[7] KURAKIN A, GOODFELLOW I J, BENGIO S. Adversarial examples in the physical world[J]. arXiv: 1607.02533, 2016.

[8] MADRY A, MAKELOV A, SCHMIDT L, et al. Towards deep learning models resistant to adversarial attacks[J]. arXiv: 1706.06083, 2017.

[9] TAVALLAEE M, BAGHERI E, LU W, et al. A detailed analysis of the KDD CUP 99 data set[C]// 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. IEEE, 2009: 1-6.

[10] SADEGHZADEH A M, SHIRAVI S, JALILI R. Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification[J]. IEEE Transactions on Network and Service Management, 2021, 18(2): 1962-1976.

[11] WANG Z Y. The applications of deep learning on traffic identification[J]. BlackHat USA, 2015, 24(11): 1-10.

[12] LOPEZ-MARTIN M, CARRO B, SANCHEZ-ESGUEVILLAS A, et al. Network traffic classifier with convolutional and recurrent neural networks for Internet of Things[J]. IEEE Access, 2017, 5: 18042-18050.

[13] RIMMER V, PREUVENEERS D, JUAREZ M, et al. Automated website fingerprinting through deep learning[J]. arXiv: 1708.06376, 2017.

[14] SIRINAM P, IMANI M, JUAREZ M, et al. Deep fingerprinting: Undermining website fingerprinting defenses with deep learning[C]// Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. 2018: 1928-1943.

[15] ABE K, GOTO S. Fingerprinting attack on Tor anonymity using deep learning[C]// Proceedings of the Asia-Pacific Advanced Network. 2016.

[16] WANG P, YE F, CHEN X J, et al. Datanet: Deep learning based encrypted network traffic classification in sdn home gateway[J]. IEEE Access, 2018, 6: 55380-55391.

[17] LOTFOLLAHI M, JAFARI S M, SHIRALI H Z R, et al. Deep packet: A novel approach for encrypted traffic classification using deep learning[J]. Soft Computing, 2020, 24(3): 1999-2012.

[18] REZAEI S, KROENCKE B, LIU X. Large-scale mobile app identification using deep learning[J]. IEEE Access, 2019, 8: 348-362.

[19] ACETO G, CIUONZO D, MONTIERI A, et al. Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges[J]. IEEE Transactions on Network and Service Management, 2019, 16(2): 445-458.

[20] DRAPER-GIL G, LASHKARI A H, MAMUN M S I, et al. Characterization of encrypted and vpn traffic using time-related[C]// Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP). 2016: 407-414.

[21] WANG W, ZHU M, WANG J L, et al. End-to-end encrypted traffic classification with one-dimensional convolution neural networks[C]// 2017 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE, 2017: 43-48.

[22] CAICEDO-MUNOZ J A, ESPINO A L, CORRALES J C, et al. QoS-Classifier for VPN and Non-VPN traffic based on time-related features[J]. Computer Networks, 2018, 144: 271-279.

[23] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[J]. arXiv: 1409.1556, 2014.

[24] SZEGEDY C, LIU W, JIA Y Q, et al. Going deeper with convolutions[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[25] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]// International Conference on Machine Learning. PMLR, 2015: 448-456.

[26] SZEGEDY C, IOFFE S, VANHOUCKE V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]// Thirty-first AAAI Conference on Artificial Intelligence. 2017.

[27] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the inception architecture for computer vision[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2818-2826.

[28] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE Conference

on Computer Vision and Pattern Recognition. 2016:770-778.

[29] SIMONYAN K,VEDALDI A,ZISSERMAN A. Deep inside convolutional networks: Visualising image classification models and saliency maps[J]. arXiv:1312.6034,2013.

[30] CARLINI N,WAGNER D. Towards evaluating he robustness of neural networks[C]// 2017 IEEE Symposium on Security and Privacy. IEEE,2017:39-57.

[31] MOOSAVI-DEZFOOLI S M,FAWZI A,FROSSARD P. Deepfool:a simple and accurate method to fool deep neural networks [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:2574-2582.

[32] CHEN P Y,ZHANG H,SHARMA Y,et al. Zoo:Zeroth order optimization based black-box attacks to deep neural networks without training substitute models[C]// Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. 2017: 15-26.

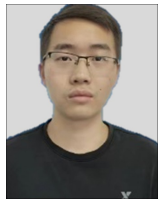
[33] MOOSAVI-DEZFOOLI S M,FAWZI A,FAWZI O,et al. Universal adversarial perturbations[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1765-1773.

[34] BRANCO P,TORGO L,RIBEIRO R P. A survey of predictive modeling on imbalanced domains[J]. ACM Computing Surveys (CSUR),2016,49(2):1-50.

[35] KINGMA D P,BA J. Adam:A method for stochastic optimization[J]. arXiv:1412.6980,2014.

[36] HINTON G E,SRIVASTAVA N,KRIZHEVSKY A,et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. arXiv:1207.0580,2012.

[37] ZHANG J Z,HE T X,SRA S,et al. Why gradient clipping accelerates training:A theoretical justification for adaptivity[J]. arXiv:1905.11881,2019.



YANG Youhuan, born in 1998, postgraduate. His main research interests include deep learning and adversarial attack/defense.

SUN Lei, born in 1973, Ph.D, professor. His in research interests include artificial intelligence and information systems security.

(责任编辑:何杨)