针对卷积神经网络流量分类器的对抗样本攻击防御

王 滨[1,2], 郭艳凯[1], 钱亚冠[1], 王佳敏[1], 王 星[2], 顾钊铨[3]

[1]浙江科技学院大数据学院 杭州 中国 310023
[2]杭州海康威视网络与信息安全实验室 杭州 中国 310052
[3]广州大学网络空间先进技术研究院 广州 中国 510006

摘要 随着深度学习的兴起, 深度神经网络被成功应用于多种领域, 但研究表明深度神经网络容易遭到对抗样本的恶意攻击。作为深度神经网络之一的卷积神经网络(CNN)目前也被成功应用于网络流量的分类问题, 因此同样会遭遇对抗样本的攻击。为提高 CNN 网络流量分类器防御对抗样本的攻击, 本文首先提出批次对抗训练方法, 利用训练过程反向传播误差的特点, 在一次反向传播过程中同时完成样本梯度和参数梯度的计算, 可以明显提高训练效率。同时, 由于训练用的对抗样本是在目标模型上生成, 因此可有效防御白盒攻击; 为进一步防御黑盒攻击, 克服对抗样本的可转移性, 提出增强对抗训练方法。利用多个模型生成样本梯度不一致的对抗样本, 增加对抗样本的多样性, 提高防御黑盒攻击的能力。通过真实流量数据集 USTC-TFC2016 上的实验, 我们生成对抗样本的网络流量进行模拟攻击, 结果表明针对白盒攻击, 批次对抗训练可使对抗样本的分类准确率从 17.29%提高到 75.37%; 针对黑盒攻击, 增强对抗训练可使对抗样本的分类准确率从 26.37%提高到 68.39%。由于深度神经网络的黑箱特性, 其工作机理和对抗样本产生的原因目前没有一致的认识。下一步工作对 CNN 的脆弱性机理进行进一步研究, 从而找到更好的提高对抗训练效果的方法。

关键词 流量分类; 对抗样本; 对抗训练
中图法分类号 TP393 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2022.01.10

Defense of Traffic Classifiers based on Convolutional Networks against Adversarial Examples

WANG Bin[1,2], GUO Yankai[1], QIAN Yaguan[1], WANG Jiamin[1], WANG Xing[2], GU Zhaoquan[3]

[1] School of Big Data Science, Zhejiang University of Science and Technology, Hangzhou 310023, China
[2] Hangzhou Hikvision Network and Information Security Laboratory, Hangzhou 310052, China
[3] Cyberspace Institute Advanced Technology, Guangzhou University, Guangzhou 510006, China

Abstract With the rise of deep learning, deep neural networks have been successfully applied in many fields, but recent research shows that deep neural network is vulnerable to adversarial examples attacks. Convolutional Neural Networks (CNNs) as one type of deep neural networks have also been successfully applied to the classification of network traffic, however, recent research shows that CNN is as well vulnerable to adversarial examples. To improve the CNN traffic classifier's defense against the attack of adversarial examples, we first propose a batch-adversarial-training method, which uses the characteristics of back propagation error in the training process to calculate the example gradient and weight gradient simultaneously in the process of error back-propagation. This method can improve the training efficiency. At the same time, sine the adversarial examples for training are generated on the target mode, it can effectively defend white-box attacks. To further improve the defense against black-box attacks, we propose an enhanced-adversarial-training method. In order to prevent the transferability of the adversarial examples, we craft the adversarial examples adopted in adversarial training on multiple substitute models for diversity. The benefit of this method is the adversarial examples from these models will have misaligned gradients. We conduct experiments on the real traffic dataset USTC-TFC2016. We craft traffic composed of adversarial examples to simulate attacks. The experimental results show that batch-adversarial-training can improve the classification accuracy of adversarial examples from 17.29% to 75.37% for white-box attacks and for black-box attacks, the enhanced-adversarial-training can improve the classification accuracy of adversarial examples from 26.37% to 68.39%. Due to the black-box characteristics of deep neural network, there is no consistent understanding of its working mechanism and the real cause of adversarial examples. The next step is to further study the vulnerability mechanism of CNN, so

通讯作者: 钱亚冠, 博士, 教授, Email: qianyaguan@zust.edu.cn。
本课题得到国家重点研发计划项目(No. 2018YFB2100400), 国家自然科学基金资助项目(No. 61902082), 浙江省公益技术应用研究项目(No.LGG19F030001, No. LGF20F020007), 杭州市领军型创新创业团队资助计划(No. 201920110039)资助。
收稿日期: 2021-04-01; 修改日期: 2021-06-16; 定稿日期: 2021-11-10

as to find a better method to improve the effect of adversarial training.

Key words traffic classification; adversarial examples; adversarial training

1 引言

随着互联网的快速发展, 网络流量不断增长并呈现多样化的趋势, 这给互联网运营和管理带来巨大的压力和挑战。网络流量分类作为网络管理与网络安全的一项关键技术, 不但能够优化网络配置, 降低网络安全隐患, 还能根据用户的行为分析提供更好的服务质量, 对于网络管理中心了解网络运行状态、优化网络运营和管理具有重要意义。

传统的流量分类方法主要是基于端口和深度包检测[1-2]。但随着负载加密和新型应用的不断涌现, 导致上述方法的有效性下降。近年来, 研究人员采用机器学习的方法解决网络流量分类问题。机器学习方法利用“网络流”(flow)的统计特征(如网络流的时间长度、数据包的数量等)建立分类模型, 不受动态端口、负载加密甚至网络地址转换的影响。随着深度学习在计算机视觉[3]、语音识别[4]、自然语言处理[5]等多个领域取得的成功应用, 也为流量分类提供了新的技术契机。白雪等人[6]针对 P2P 流量分类准确率较低的问题, 提出一种基于深度学习结构、半监督的深度置信网络流量分类方法, 实验证明具有良好的效果。Ertam 等人[7]通过使用 WK-ELM 算法中的 GA 技术训练神经网络, 针对 Moore 数据集进行流量分类达到 96.57%的准确率。王勇等人[8]提出一种基于卷积神经网络(CNN)的流量分类算法, 构造能够实现流量自主特征学习的分类模型。Wang 等人[9]利用 CNN 方法对恶意流量进行分类识别, 在实际的流量数据集 USTC-TF2016 上进行了实验, 获得了很高的准确率。由于 CNN 具有自动提取特征的能力, 通过多个卷积层的深层网络结构, 自主发现学习数据的特征表示[10], 应用于网络流量分类时具有明显的优势。但是, 新的研究又表明 CNN 容易受到对抗样本的攻击, 即在原始数据上增加一些微小的扰动, 就能导致 CNN 错误分类[11]。

基于 CNN 的流量分类器首先把流量数据转换为灰度图像, 因此可以通过图像对抗样本对流量分类器实施攻击, 通过在图像上添加微小噪声, 使 CNN 发生错误预测。目前典型的对抗样本生成方法有 FGSM[12]、DeepFool[13]、JSMA[14]和 C&W[15]等。为了能实现对真实场景的流量分类器实施攻击, 我们将表示流量的图像对抗样本再逆变换成攻击数据包流, 向目标 CNN 重放, 达到攻击目的。实验表明, 本文提出的攻击方法可以成功的攻击 CNN 流量分类器。

针对对抗样本的攻击, 目前提出了很多防御方法, 例如对抗训练[16-18]、梯度掩蔽[19]、检测防御[20]等, 对抗训练是目前被证实最为有效的防御方法[21]。但目前的对抗训练需要生成大量对抗样本, 大大增加了训练时间, 本文针对网络流量数据量巨大的特点, 提出了批次对抗训练, 利用反向传播误差的过程, 同时完成样本梯度和模型梯度的计算, 加快了训练速度。由于批次对抗训练是已知目标模型的结构、参数, 因此可以很好地防御白盒攻击, 但在防御其它 CNN 的黑盒攻击方面有不足。我们进一步提出增强对抗训练, 利用样本梯度的余弦相似性, 筛选出不同扰动方向的对抗样本加入到对抗训练, 在保证原分类准确性不受影响的前提下, 增强对跨模型黑盒攻击的防御能力。实验结果表明, 增强对抗训练比批次对抗训练能更好的防御这类黑盒攻击。

2 预备知识

2.1 卷积神经网络

定义 1 卷积神经网络(CNN): CNN 一般可以表示为映射函数 $\mathcal{F}:\mathcal{X}\mapsto\mathcal{Y}$, $X\in\mathcal{X}$ 是 d 维输入变量, $Y\in\mathcal{Y}$ 是一个 m 维概率向量, 表示 m 个类的置信度。一个 N 层 CNN 输入 X 后产生的输出:

$$\mathcal{F}(X)=\mathcal{F}^{(N)}(\ldots\mathcal{F}^{(2)}(\mathcal{F}^{(1)}(X))) \tag{1}$$

$\mathcal{F}^{(i)}$ 代表 CNN 的第 i 层的计算输出。这些层可以是卷积、池化或者其他任何形式的神经网络层。CNN 的最后一层采用 Softmax 层, 定义为:

$$\mathcal{F}^{(N)}(Z)_i=\mathrm{Softmax}(Z)_i=\exp(z_i)/\sum_{i=1}^{m}\exp(z_i) \tag{2}$$

其中, $Z=\mathcal{F}^{(N-1)}(\cdot)$ 是前一层(又称最后一个隐藏层)的输出向量。最后预测的标签由 $y=\arg\max_{i=1\cdots m}\mathcal{F}(X)_i$ 得到, 其中 $\mathcal{F}(X)=\mathrm{Softmax}(Z)$。

2.2 网络流量分类

本文的CNN流量分类器是针对网络流构建, 即输入 X 为网络流。原始的网络流量(raw traffic)是由数据包序列组成: $P=\{p_1,\cdots,p_N\}$, 数据包 $p_i=(x_i,b_i,t_i)$, $i=1,2,\cdots,N$, x_i 为5元组(源IP地址、源端口、目标IP、目标端口和传输层协议),

$b_i \in [0,+\infty]$ 表示数据包的大小(字节为单位), t_i 表示数据包传输的起始时间。

定义 2 网络流: 一个网络流表示为 $f=(x,b,d_t\ t)$, 它是原始网络流量 P 的子集。这个子集中的所有数据包按时间次序排列: $\{p_1=(x_1,b_1,t_1),\cdots,p_n=(x_n,\ b_n,t_n)\}$, $t_1<\cdots<t_n$, 且所有五元组相同, 即 $x=x_1=...=x_n$。b 表示网络流中所有数据包的大小之和, $d_t=t_n-t_1$ 表示网络流的持续时间, t 表示网络流的起始时间。

基于 CNN 的流量分类就是找到网络流集合 $F=\{f_1,f_2,\cdots,f_N\}$ 到流量应用类型集合 $Y=\{y_1,y_2,\cdots,\ y_k\}$ 的映射。由于 CNN 的输入是二维图像数据, 因此先把原始的流量数据转化为网络流, 再转化为二维矩阵。上述预处理过程如图 1 所示, 首先将原始网络流量 P 分割成多个离散的网络流 $f_1,f_2,\cdots,f_N$; 然后匿名化网络流中的 IP 地址和 MAC 地址[22]; 进行数据清洗, 去除重复网络流; 将所有的网络流数据通过填 0 的方式补齐到相同长度。

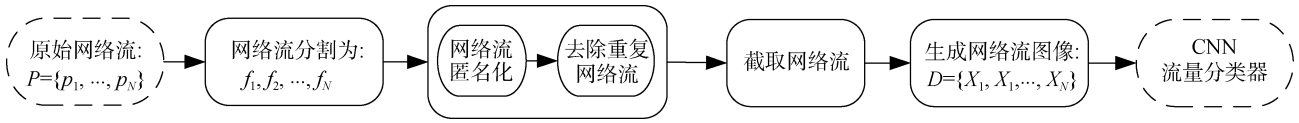


图 1 网络流量预处理

Figure 1 Network traffic preprocessing

由于网络流的靠前部分通常是连接信息和一些内容数据, 根据经验前 784 个字节已经可以很好的反映网络流的内在特征[9]。因此本文采用 28×28 输入的 CNN, 所有的网络流截取到 784 个字节。最后, 将每个数据流转化为 28×28 大小的灰度图像, 每个像素表示数据流中的一个字节。把这些灰度图像输入到 CNN, 训练模型, 获得 CNN 流量分类器[8]。

3 网络流对抗样本攻击

尽管现有的研究表明, CNN 用于网络流量分类具有良好的性能[8], 但正是它具有自动提取特征的特点, 反而成为对抗样本攻击的弱点[11]。我们提出网络流的对抗样本, 即在原始的数据包中加入少量扰动信息, 再重放到网络上。CNN 流量分类器采集到这些数据包后, 会转化为灰度图像。由于原始数据中加入了扰动, 会使得 CNN 产生大量的错误分类, 从而实现攻击目的。

3.1 威胁模型

对抗样本攻击的威胁模型是攻击者根据不同的应用场景、假设和攻击程度要求, 改变攻击方法需要的属性, 部署特定的攻击, 主要包括攻击目标和攻击能力[23]。由于流量分类器通常部署在路由器等设备中, 攻击者无法直接获得 CNN 的结构和参数。本文假设攻击者仅知道目标模型为 CNN, 采取近似于黑盒攻击的策略, 即在本地训练一个代理 CNN。利用对抗样本的可转移性, 在代理模型上生成对抗样本, 然后构建伪造数据包, 重放给远程目标 CNN, 从而使目标分类器错误分类。

定义 3 攻击目标: 假定 CNN 流量分类器为 $\mathcal{F}$, 网络流的灰度图像为 X, 则 $\mathcal{F}(X)=y$, y 为预测标签。对抗样本 $X_{adv}=X+\delta$, 这里 δ 是一个微小的扰动量。为确保 X_{adv} 和 X 的变化很小, 增加约束 $\|\delta\|_p\leqslant\epsilon$, 这里 $\|\cdot\|_p$ 为 p 范数。本文提出的攻击目标只要求将 X_{adv} 错误分类, 即:

$$\mathcal{F}(X_{adv})=y_{adv}\neq y \tag{3}$$

定义 4 攻击能力: 攻击能力是指攻击者掌握目标 CNN 信息多少的程度, 分为白盒攻击和黑盒攻击[18]。白盒攻击是指攻击者几乎知道关于神经网络的所有信息, 包括训练数据、激活函数、拓扑结构等。黑盒攻击则假设攻击者无法获得已训练的神经网络内部信息, 仅能获得模型的输出, 包括标签和置信度。由于黑盒攻击不需要了解模型内部信息, 因此更符合现实中的某些场景攻击。本文假设攻击者仅需知道目标流量分类器是采用 CNN 模型, 因此属于黑盒攻击。

3.2 攻击方法

算法 1: 网络流对抗样本攻击

输入: 网络流 $F=\{f_1,f_2,...,f_N\}$

输出: 攻击用的伪造流量数据 S

1: 初始化 $S\leftarrow\varnothing$

2: **while** $i<N$ **do**

3: 网络流 $f_i\in F$ 转换为灰度图像 $X^{(i)}$ 和它的掩模矩阵 $\boldsymbol{M}_i$

4: $X_{adv}^{(i)}\leftarrow X+(\varepsilon\text{sign}(\nabla_X\mathcal{L}(\theta,X,y)))\odot\boldsymbol{M}_i$

//利用 FGSM 等方法生成 $X^{(i)}$ 的对抗样本

5: 根据掩模 $\boldsymbol{M}_i$，提取 $X_{adv}^{(i)}$ 中的数据包头，恢复为网络流 f_i'

6: 根据 f_i' 包头中的信息和篡改后的负载，伪造数据包流 $S \leftarrow \cup\{p_1', p_2', \cdots, p_k'\}$

7: **end while**

在上述威胁模型的假设下，本文采用如下的攻击方法：首先对采集到的流量数据 $\{p_1=(x_1,b_1,t_1),\cdots,p_n=(x_n,b_n,t_n)\}$，$t_1<t_1<\ldots<t_n$，预处理为灰度图像 $X^{(i)}$，建立与目标模型同分布的本地训练集 $D=\{(X^{(1)},y^{(1)}),\cdots,(X^{(m)},y^{(m)})\}$。然后利用 D 训练本地代理 CNN，获得分类器 $\tilde{\mathcal{F}}$，模拟目标流量分类器 $\mathcal{F}$。在本地代理 CNN 上生成对抗样本 $D'=\{X_{adv}^{(1)},\cdots,X_{adv}^{(m)}\}$。最后将对抗样本 $X_{adv}^{(i)}$ 转回网络流量 f_i'，这是因为正常的网络流图像 $X^{(i)}$ 已经变为 $X_{adv}^{(i)}$，且两者之间的变化非常微小，使目标 CNN 产生错误分类的预测。根据网络流 f_i' 再伪造出数据包流 $\{p_1',\cdots,p_k'\}$，向目标 CNN 模型重放。

由于 CNN 的特征抽取主要针对数据包的内容负载，因此我们仅对数据包内容进行扰动，不对数据包头的控制信息改变，这样有利于逆向构建攻击流量。为了确保数据包头不被改变，我们引入掩模(mask)矩阵，它是一个与输入图像同样大小的 0~1 矩阵。对应原图像数据包头区域为 0，数据区域内容为 1:

$$M(i,j)=\begin{cases}0 & X(i,j)\text{为包头数据}\\ 1 & \text{others}\end{cases} \tag{4}$$

这里 $X(i,j)$ 对应网络流中的某个字节。公式(4)根据数据包头部的控制信息，构建出每个数据流图像的掩模矩阵。由于 CNN 反向传播采用的是动态规划策略，无法只对 X 的某些分量求梯度，因此我们利用 FGSM[19]等方法得到 X 的对抗样本 X_{adv}，再利用掩模矩阵获得最后的对抗样本:

$$X_{adv}=FGSM(X)\otimes M \tag{5}$$

这里假设用 FGSM[12]方法生成对抗样本，$\otimes$ 表示 Ha-damard 乘积。攻击方法如算法 1 所示，第 4 步除了可采用 FGSM 外，I-FGSM[24]、LL-FGSM[24]、Deep-Fool[13]、JSMA[14]和 C&W[15]等方法均可用于生成图像对抗样本。

4 对抗攻击的防御

为了防御对抗样本攻击，Goodfellow 等人[12]首先提出利用对抗样本来提高模型的鲁棒性。Mardry 等人[21]从优化的观点出发，认为对抗训练是一个关于鞍点的优化问题，他们把传统的 ERM 训练推广到鲁棒性训练。对抗训练是将干净图像样本和生成的对抗样本作为训练数据共同参与 CNN 模型 $\mathcal{F}$ 的训练，采用代价函数的最小化原理，使得模型最终收敛时，代价函数达到最小，该优化问题的形式化表示如下[21]:

$$\min_{\theta}\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\delta}\mathcal{L}(\theta,X+\delta,y)+\max_{\delta}\mathcal{L}(\theta,X,y)\right] \quad \text{s.t.}\ \|\delta\|_p<\epsilon \tag{6}$$

其中，$\mathcal{D}$ 为数据的分布，$\mathcal{L}$ 为代价函数，θ 为模型参数。可以发现对抗训练是个鞍点优化问题，是内部最大化和外部最小问题的组合。内部最大化问题是找到代价最大的对抗样本，因此对抗训练需要大量的对抗样本用于训练，而生成对抗样本占据了对抗训练的大部分时间[17]。对于网络流量这种大规模数据，对抗训练的时间复杂度非常高。

4.1 批次对抗训练

算法 2: 批次对抗训练

输入: 训练集 D；学习率 γ=0.01

输出: 对抗训练后的模型 $\mathcal{F}_{badv}(\theta)$

1: **while** $i<\text{epochs}$ **do**

2: 选取批次训练数据 $D_i\in D$

3: $g_\delta^{(i)}\leftarrow\nabla_\delta\mathcal{L}(\theta,X^{(i)}+\delta,y^{(i)})$ //计算损失函数对样本的梯度

4: $g_\theta^{(i)}\leftarrow\nabla_\theta\mathcal{L}(\theta,X^{(i)},y^{(i)})$，$g_{\theta\ adv}^{(i)}\leftarrow\nabla_\theta\mathcal{L}(\theta,X_{adv}^{(i)},y^{(i)})$ //计算损失函数对模型参数的梯度

5: $X_{adv}^{(i)}\leftarrow G(g_{\theta\ adv}^{(i)}),\quad i=1,\cdots,k$ //取 k 个训练数据生成对抗样本

6: $\nabla_\theta\tilde{\mathcal{L}}(\theta)\leftarrow\dfrac{(\sum_{i\in CLEAN}g_\theta^{(i)}+\lambda\sum_{j\in ADV}g_{\theta\ adv}^{(i)})}{(m-k)+\lambda k}$ // 计算总代价梯度

7: $\theta_{i+1}\leftarrow\theta_i+\gamma\nabla_\theta\tilde{\mathcal{L}}(\theta)$ //更新模型参数

8: **end while**

为了提高对抗训练的速度，我们将以往单个样本的在线对抗训练改进为按批次进行训练，同时利用反向传播过程中的误差传递，同时完成更新模型参数和对抗样本生成这两个步骤。假设批次训练的数据为 $D=\{(X^{(i)},y^{(i)})\}_{i=1}^m$，我们假设函数 $G(\cdot)$ 为对

抗样本生成函数, 选取 k 个训练数据生成对抗样本 $\{X_{adv}^{(i)}\}_{i=1}^{k}=\{G(X^{(i)},\theta)\}_{i=1}^{k}$, θ 为当前模型的参数。更新当前批的训练数据为 $D'=\{X_{adv}^{(i)}\}_{i=1}^{k}\bigcup\{(X^{(i)},y^{(i)})\}_{i=1}^{m}$, 批次对抗训练的代价函数为:

$$\tilde{\mathcal{L}}(\theta)=\frac{(\sum_{i\in CLEAN}\mathcal{L}(\theta,X^{(i)},y^{(i)})+\lambda\sum_{j\in ADV}\mathcal{L}(\theta,X_{adv}^{(j)},y^{(j)}))}{(m-k)+\lambda k} \tag{7}$$

其中, $\mathcal{L}(\theta,X,y)$ 是正常样本的代价函数, $\mathcal{L}(\theta,X_{adv},y)$ 是对抗样本的代价函数, m 是批次对抗训练的总样本数, 参数 λ 是控制对抗样本的比例。

我们分析了以往在线对抗训练算法, 发现生成对抗样本和更新模型参数是分开进行的, 这会带来两次反向传播求去梯度的问题。通过推导, 可以发现损失函数 $\mathcal{L}$ 对模型参数 θ 的梯度和对扰动 δ 的梯度, 在反向传播过程中都可利用误差 $\sigma=\partial\mathcal{L}/\partial S$ 计算, 这里 S 表示神经元的输入。为此, 我们在一次反向传播过程中同时计算出它们梯度, 大大加快计算速度。

4.2 增强对抗训练

算法 3: 增强对抗训练

输入: 基准模型 $\mathcal{F}(\theta)$;训练集 D; 预训练模型 $\{\mathcal{F}_i(\theta)\}_{i=1}^{N}$; t=0.5

输出: 对抗训练后的模型 $\mathcal{F}_{sadv}(\theta)$

1: **while** $k<\text{epochs}$ **do**

2: 选取批次训练数据 $D_i\in D$

3: **for** X^j **in** D_k **do**

4: $\psi\leftarrow\log(\sum_{1<a<b<N}\exp(CS(\nabla_X\mathcal{L}_a,\nabla_X\mathcal{L}_b)))$

4: **if** $\psi\le t$ **then**

5: $\{X_{adv}^{j}\}\leftarrow\{G\left(X^j,B_i\right)\}_{i=1}^{N}$ //每个预训练模型均生成对抗样本

6: **else** $\psi>t$ **then**

$\{X_{adv}^{j}\}\leftarrow G\left(X^j,B_i\right)$, $i\in\{1,\cdots,N\}$ //随机选一个预训练模型生成对抗样本

7: **end if**

8: $D_k'\leftarrow D_k\bigcup\{X_{adv}^{j}\}$ //更新当前批次训练数据

9: 根据式(7)计算总代价 $\tilde{\mathcal{L}}(\theta)$

10: $\theta_{k+1}\leftarrow\theta_k+\gamma\nabla_\theta\tilde{\mathcal{L}}(\theta)$ //更新模型参数

11: **end for**

12: **end while**

批次对抗训练可以较好的防御白盒攻击下的特定攻击, 但攻击者仍可采用黑盒攻击, 在这个样本点附近随机跳跃, 再进行单步攻击 CNN[12]。为了防御这类黑盒攻击, 我们进一步提出新的训练方法——增强对抗训练。主要思想是增加对抗样本的多样性, 为此提出流量对抗样本的差异性筛选方法。实验证明, 增强对抗训练不仅可以防御针对流量分类器的白盒攻击, 而且可以防御黑盒攻击。

假设有两个预训练模型 $\mathcal{F}_1(X)$ 和 $\mathcal{F}_2(X)$。$\nabla_X\mathcal{L}_1(\theta_1,X,y)$ 和 $\nabla_X\mathcal{L}_2(\theta_1,X,y)$ 分别表示两个模型的代价函数对样本 X 的梯度, 是该样本的最佳扰动方向, 沿此方向添加扰动可最大限度的增大 CNN 的代价函数。如果 $\nabla_X\mathcal{L}_1(\theta_1,X,y)$ 和 $\nabla_X\mathcal{L}_2(\theta_1,X,y)$ 的方向一致, 或近似一致, 那么意味着沿此梯度方向获得的对抗样本 X_{adv} 既能攻击模型 $\mathcal{F}_1(X)$ 上, 也能攻击 $\mathcal{F}_2(X)$。从防御角度看, 使用 $\mathcal{F}_1(X)$ 生成的对抗样本进行对抗训练, 就能防御 $\mathcal{F}_2(X)$ 生成的对抗样本。如果 $\nabla_X\mathcal{L}_1(\theta_1,X,y)$ 和 $\nabla_X\mathcal{L}_2(\theta_1,X,y)$ 的方向不一致, 那么样本 X 就需要在 $\mathcal{F}_1(X)$ 和 $\mathcal{F}_2(X)$ 上分别生成对抗样本进行对抗训练。因此, 样本 X 是否需要在每个模型上生成对抗样本, 取决于预训练模型之间的梯度对齐程度。我们使用余弦相似度(CS)来量化这种对齐程度:

$$CS(\nabla_X\mathcal{L}_1,\nabla_X\mathcal{L}_2)=\frac{<\nabla_X\mathcal{L}_1,\nabla_X\mathcal{L}_2>}{|\nabla_X\mathcal{L}_1|\cdot|\nabla_X\mathcal{L}_2|} \tag{8}$$

这里, $<\nabla_X\mathcal{L}_1,\nabla_X\mathcal{L}_2>$ 表示两个梯度向量的内积。当 CS 较小时, 表明样本 X 在两个模型上的差异性较低, 需要两个模型都生成对抗样本进行训练。反之, 只需用其中一个模型生成对抗样本进行训练。

为了度量样本在 N 个预模型之间的差异性, 我们把余弦相似度推广到 N 个模型的情形, 取其最大值:

$$\psi=\max_{a,b\in\{1,\cdots,N\},a\neq b}CS(\nabla_X\mathcal{L}_a,\nabla_X\mathcal{L}_b) \tag{9}$$

由于 Ψ 是非光滑函数, 不宜采用梯度下降等优化方法, 进一步用 LogSumExp 函数来近似 Ψ:

$$\psi\approx\log(\sum_{1<a<b<N}\exp(CS(\nabla_X\mathcal{L}_a,\nabla_X\mathcal{L}_b))) \tag{10}$$

当 $\psi\leqslant t$ 时, 对于每个预训练模型, 都需用 X 生成对抗样本参与对抗训练, 反之选择其中一个预训练模型生成对抗样本训练, 本文设置 $t=0.5$。

5 实验分析

5.1 网络流量数据集

本文采用的实验环境: 操作系统 Ubuntu16, GPU

NVIDIA TITAN 2080, tensorflow-gpu 1.8.2, numpy 1.16.1。网络流量集为 USTC-TFC2016[9], 该数据集以 Pcap 文件格式存储, 包含了两部分, 第一部分是从 2011 年到 2015 年收集的 10 种公开的恶意网络流量[25], 第二部分是使用网络模拟设备 IXIA BA[26]采集包含 10 种正常网络流量。本文使用第二部分的正常网络流量。网络流数据是使用 USTC-TK2016[9]将原始流量按图 1 的处理流程得到的 175,178 网络流记录。各个流量类的网络流可视化后的部分实例如图 2 所示。从可视化结果中不难发现, 不同网络应用类别的灰度图像具有明显的可区分度, 且在同一类别中具有一致性。我们将生成的灰度图像集合划分为训练集和测试集, 其中测试集和训练集的比例是 1∶9。

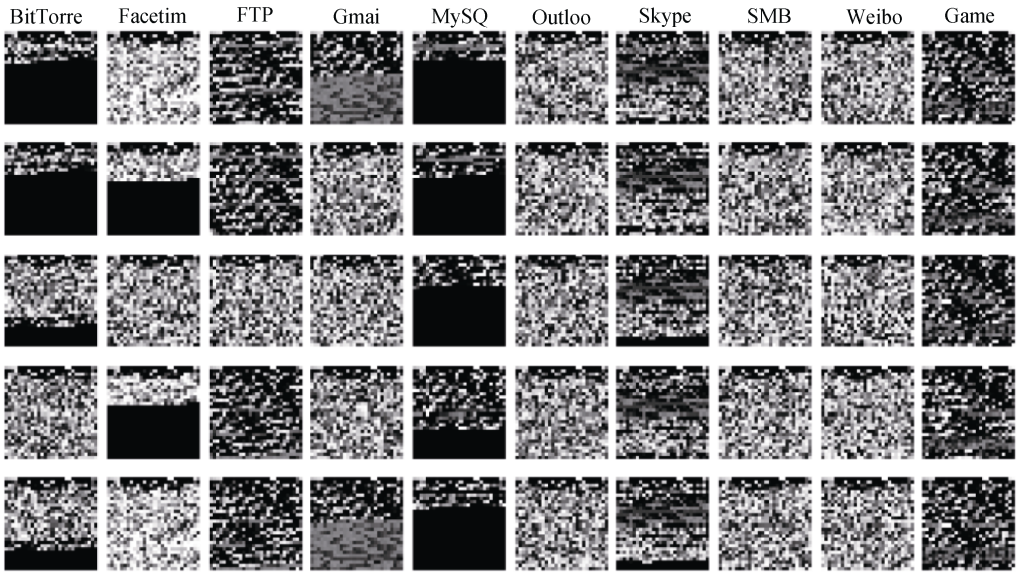


图 2　网络流转为灰度图像示例

Figure 2　Example of network traffic converted to grayscale image

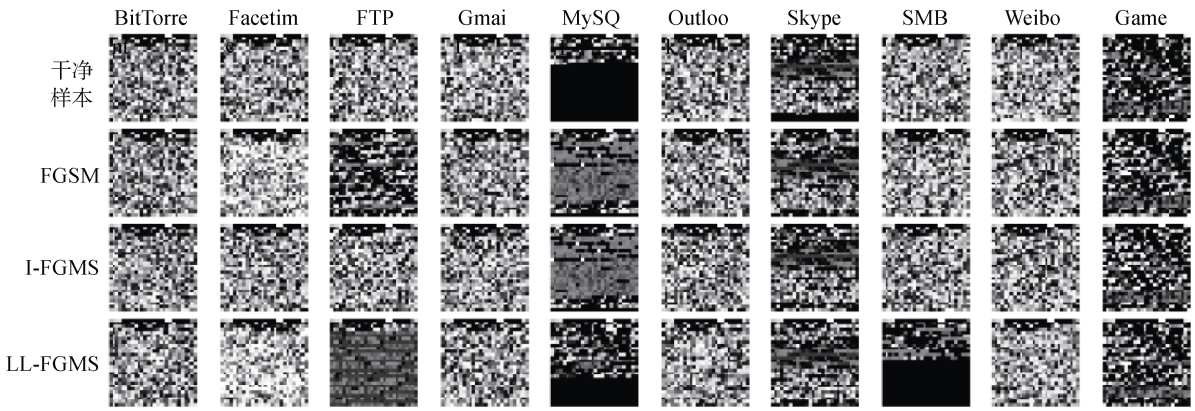


图 3　FGSM、I-FGSM 和 LL-FGSM 生成流量对抗样本的灰度图像

Figure 3　FGSM, I-FGSM and LL-FGSM generate grayscale images of traffic adversarial examples

5.2　CNN 参数设置

本文实验用的 CNN 模型为 LeNet-5, 激活函数为 ReLU 函数, 池化层均采用最大池化方法, 池化和卷积均使用全 0 填充以避免图像边缘信息丢失太快, 具体模型结构如表 1 所示。我们把目标模型标记为 $\mathcal{F}_A$。同时为了进行增强对抗训练, 选择 4 个预训练模型($\mathcal{F}_B$, $\mathcal{F}_C$, $\mathcal{F}_D$, $\mathcal{F}_E$)用于生成差异性对抗样本。除训练过程的超参数不同外, 4 个模型的结构与目标模型 $\mathcal{F}_A$ 相同, 具体设置如表 2 所示。

5.3　对抗样本攻击效果

为了区分模型自身分类错误的样本, 实验只使用目标模型分类正确的样本用于生成对抗样本。生成的流量对抗样本如图 3 所示。FGSM 以及变种生成的对抗样本在模型 $\mathcal{F}_A$ 上的准确率如表 3 所示。可以

表 1 实验模型 LeNet-5 的结构

Table 1 The structure of the experimental model LeNet-5

每层类型	输入	卷积核大小	输出
C1 卷积	28×28	16×5×5	16×28×28
S2 池化	16×28×28	—	16×14×14
C3 卷积	16×14×14	32×3×3	32×14×14
C4 池化	32×14×14	—	32×7×7
F5 全连接	32×7×7	1024×7×7	1024
F6 全连接	1024	—	10

表 2 预训练模型的超参数与准确率

Table 2 Hyperparameter and accuracy of pre-trained models

模型	学习率	Dropout	批次大小	训练轮数	准确率(%)
$\mathcal{F}_A$	0.04	0.25	128	150	96.15
$\mathcal{F}_B$	0.01	0.3	64	100	93.51
$\mathcal{F}_C$	0.03	0.4	64	200	95.26
$\mathcal{F}_D$	0.03	0.25	128	220	94.76
$\mathcal{F}_E$	0.01	0.5	128	300	95.04

表 3 模型 $\mathcal{F}_A$ 在不同强度对抗样本攻击下的分类准确率

Table 3 Classification accuracy of model $\mathcal{F}_A$ under different intensities of adversarial attacks

FGSM		I-FGSM		LL-FGSM	
扰动量阈值 ϵ	准确率(%)	迭代次数	准确率(%)	迭代次数	准确率(%)
0.01	96.93	1	96.93	1	84.01
0.02	84.41	2	80.87	2	56.92
0.03	73.69	3	57.81	3	34.24
0.04	61.20	4	9.53	4	21.29
0.05	48.34	5	1.77	5	18.91
0.06	17.29	6	0.20	6	15.03
0.07	7.35	7	0.05	7	13.55
0.08	2.19	8	0.03	8	9.96
0.09	0.71	9	0.03	9	6.61
0.1	0.23	10	0.01	10	3.50

表 4 采用不同扰动阈值的 FGSM 对抗样本攻击对抗训练后的模型分类准确率

Table 4 Classification accuracy of FGSM adversarial examples attack adversarial training model with different perturbation thresholds (%)

模型	ϵ =0.01	ϵ =0.03	ϵ =0.04	ϵ =0.05	ϵ =0.06	ϵ =0.07	ϵ =0.08
$\mathcal{F}_A$	96.93	73.69	61.20	48.34	17.29	7.35	2.19
$\mathcal{F}_{A\text{–}oadv}$	99.22	97.74	97.03	96.49	95.75	94.99	93.42
$\mathcal{F}_{A\text{–}badv}$	99.52	97.75	95.77	94.70	93.59	92.31	89.04

发现, 当扰动量阈值 ϵ 增大到 0.07 时, 模型 $\mathcal{F}_A$ 的准确率从 96.15%降低到 7.35%。但随着 ϵ 的继续增大, 攻击成功率降低速率变得缓慢, 最后几乎趋近于 0。而对于 I-FGSM 来说, 在迭代次数小于 5 时, 对抗样本在模型 $\mathcal{F}_A$ 上的准确率快速下降。随着迭代次数的增加, 准确率同样下降变缓, 原因是固定长度的扰动步长会导致在局部最小值点附近震荡, 收敛变慢。与 FGSM 和 I-FGSM 相比, LL-FGSM 对流量分类器表现出更好的攻击稳定性。

DeepFool、JSMA 以及 C&W 生成的流量对抗样本的攻击效果如图 4 所示。其中 C&W-2 表示基于 l_2 范数约束, C&W-inf 基于 l_∞ 约束。K 代表生成对抗样本的方法迭代次数。从结果中可以发现 DeepFool 与 JSMA 方法的攻击效果要好于 C&W 方法。

5.4 批次对抗训练防御

为了进行实验比较, 选择模型 $\mathcal{F}_A$ 作为目标模型, 其在测试集上的准确率为 96.15%, 模型 $\mathcal{F}_{A\text{-}oadv}$ 和 $\mathcal{F}_{A\text{-}badv}$ 分别是模型 $\mathcal{F}_A$ 在线对抗训练后和批次对抗训练后的流量分类器, 测试集准确率分别为 93.66%和 92.14%。表 4 表示不同对抗训练方式对于 FGSM 生成对抗样本的防御能力。

对于每种攻击方法, 均采用模型 $\mathcal{F}_{A\text{-}oadv}$ 和 $\mathcal{F}_{A\text{-}badv}$ 作为目标模型。从实验结果发现不管是在线对抗训练还是批次对抗训练, 都能够提高模型 $\mathcal{F}_A$ 的

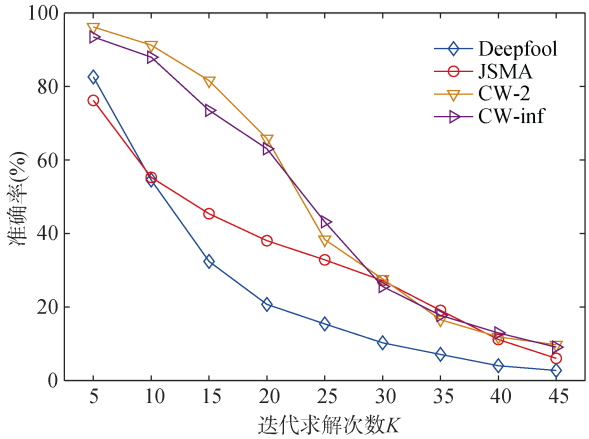


图 4　模型 $\mathcal{F}_A$ 在不同类型流量对抗样本攻击下的分类准确率

Figure 4　Classification accuracy of model $\mathcal{F}_A$ under different types of traffic adversarial examples attack

鲁棒性。表 5 是在线对抗训练和批次对抗训练的时间复杂度对比, 可以发现, FGSM 的在线对抗训练耗 1.32 h, 而批次对抗训练仅需要 15 min, 因此在流量分类这样的大数据环境下, 批次对抗训练具有明显的效率优势。

从图 5 中也可以发现, 对抗训练可以提高对于白盒攻击的鲁棒性。对于 C&W 攻击方法, 即使对抗样本的扰动幅度增强到约束极限, 经对抗训练后的模型 $\mathcal{F}_{A\text{-}oadv}$ 和 $\mathcal{F}_{A\text{-}badv}$ 分类准确率仍保持在 70%以上, 未经对抗训练的模型 $\mathcal{F}_A$ 分类准确率已下降到 17%。从表 5 中的时间效率上看, DeepFool、JSMA 和 C&W 生成的流量对抗样本进行批次对抗训练的平均耗时为 14.3 min, 而在线对抗训练平均耗时长达 1.67 h。因此, 本文提出的批次对抗训练在时间复杂度上更具较好的优势。

为了进一步分析对抗训练防御白盒攻击和黑盒攻击的效果, 对不同模型进行了比较实验, 结果如表 6 所示。其中第一行表示用于生成流量对抗样本的 CNN 模型, 第一列表示用于攻击的目标 CNN 模型。目标模型 $\mathcal{F}_A$ 和 $\mathcal{F}_B$ 经过批次对抗训练得到 $\mathcal{F}_{A\text{-}badv}$ 和 $\mathcal{F}_{B\text{-}badv}$。对角线为白盒攻击后的分类准确率, 其余为黑盒攻击后的分类准确率。

实验结果表明, 批次对抗训练对可以大幅提高模型对流量白盒攻击的鲁棒性。经批次对抗训练后的模型 $\mathcal{F}_{A\text{-}badv}$, 分类准确率从 17.29%提高到 75.37%, $\mathcal{F}_{B\text{-}badv}$ 从 14.39%提高到 81.09%。但批次对抗训练对黑盒攻击的防御效果并不显著, 未经对抗训练的模型 $\mathcal{F}_A$ 对于模型 $\mathcal{F}_B$ 生成的对抗样本准确率为 26.37%, 经批次对抗训练后的模型 $\mathcal{F}_{A\text{-}badv}$ 反而只有 19.78%。由此可见, 批对抗训练对黑盒攻击的防御能力不佳, 需要有针对黑盒攻击的防御方法。还可从表 6 发现, 用对抗训练后的模型 $\mathcal{F}_{A\text{-}badv}$ 和 $\mathcal{F}_{B\text{-}badv}$ 来生成对抗样本, 对其他模型几乎无攻击效果, 这与 Madry 等人[21]的实验结果一致, 即对抗训练在增强模型的鲁棒性同时, 削弱其生成对抗样本的能力。

5.5　增强对抗训练的防御效果

针对批对抗训练防御黑盒攻击的能力不足问题, 我们进一步提出增强对抗训练, 将生成对抗样本的 CNN 从单个变成多个, 同时通过样本的差异性筛选, 增加对抗样本的多样性, 最终实现增强防御黑盒攻击的效果。用于样本筛选的模型如表 7 所示, 第一行表示模型 $\mathcal{F}_A$, $\mathcal{F}_B$ 用于筛选出差异性对抗样本, 然后对模型 $\mathcal{F}_A$ 进行增强对抗训练得到模型 $\mathcal{F}_{A\text{-sadv-1}}$ 和 $\mathcal{F}_{A\text{-sadv-2}}$。

为了验证增强对抗训练对于黑盒攻击的防御性能, 我们进行了表 8 中的实验, 其中第一行中模型表示生成流量对抗样本, 第一列中的模型为受到流量对抗样本攻击的目标模型, 不同的是, 模型 $\mathcal{F}_A$ 未经过对抗训练, 模型 $\mathcal{F}_{A\text{-}badv}$ 经过批次对抗训练, 模型 $\mathcal{F}_{A\text{-sadv-1}}$, $\mathcal{F}_{A\text{-sadv-2}}$ 经过增强对抗训练。我们把生成对

表 5　在线对抗训练和批次对抗训练的时间对比

Table 5　Comparison of online adversarial training and batch adversarial training time

	Epoch	Baseline		批次对抗训练	
		时间/epoch(min)	总时间(min)	时间/epoch(min)	总时间(min)
FGSM	50	1.58	79.2	0.3	15
DeepFool	50	1.85	92.4	0.3	14.8
JSMA	50	1.33	67.2	0.23	11.34
C&W-2	50	2.39	199.4	0.29	14.49
C&W-inf	50	2.44	121.8	0.33	16.57

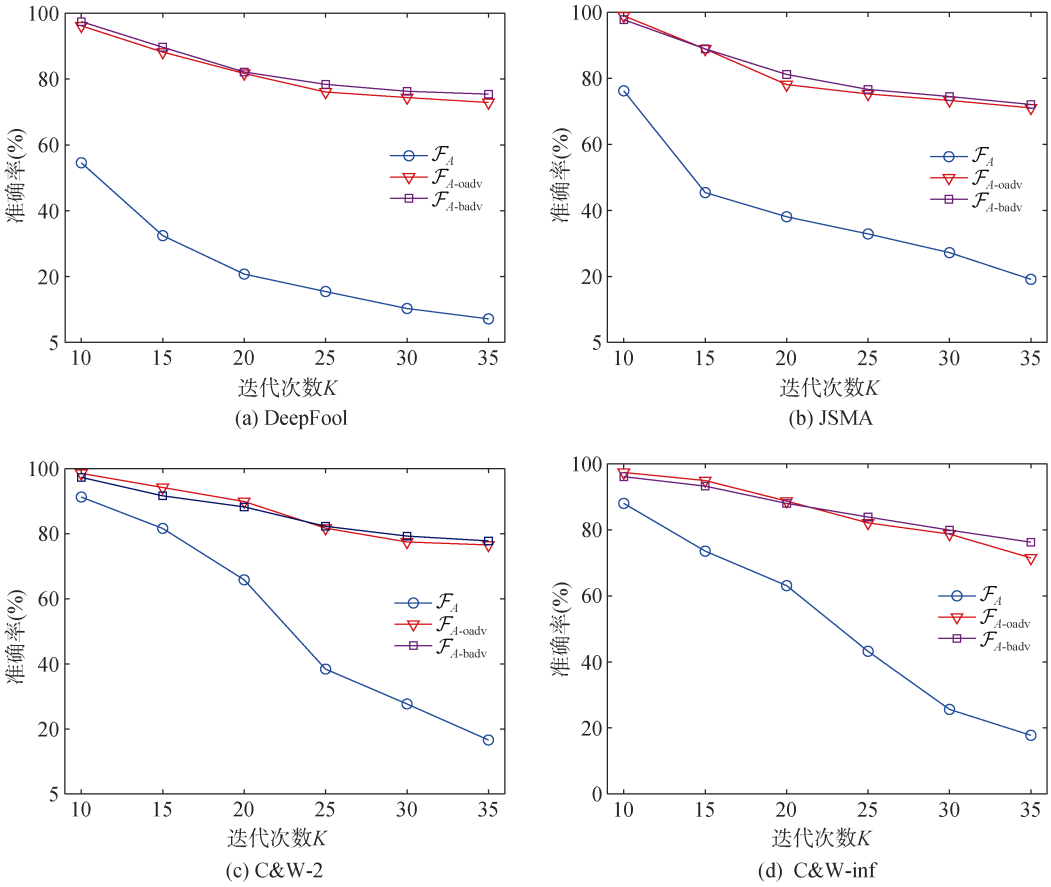


图 5 对三种不同模型的对抗攻击效果比较, 包括未进行对抗训练的模型 $\mathcal{F}_A$、传统在线对抗训练后的模型 $\mathcal{F}_{A\text{-oadv}}$、改进后的批次对抗训后的模型 $\mathcal{F}_{A\text{-badv}}$; (a)、(b)、(c)、(d) 分别对应四种对抗样本生成方法

Figure 5 Comparison of the performance of adversarial attacks on three different models, including model $\mathcal{F}_A$ without adversarial training, model $\mathcal{F}_{A\text{-oadv}}$ after traditional online adversarial training, and model $\mathcal{F}_{A\text{-badv}}$ after improved batch adversarial training; (a), (b) , (c), (d) correspond to the four adversarial examples generation methods respectively

表 6 批次对抗训练后的模型在黑盒攻击和白盒攻击下的 FGSM 对抗样本分类准确率比较

Table 6 Comparison of FGSM adversarial examples classification accuracy of the models after batch adversarial training under black box attack and white box attack (%)

模型	$\mathcal{F}_A$	$\mathcal{F}_{A\text{–badv}}$	$\mathcal{F}_B$	$\mathcal{F}_{B1\text{-}adv}$	$\mathcal{F}_{B2}$
$\mathcal{F}_A$	**17.29**	81.45	**26.37**	77.24	20.36
$\mathcal{F}_{A\text{–badv}}$	**75.37**	84.27	**19.78**	78.40	25.34
$\mathcal{F}_B$	23.03	80.01	**14.39**	84.39	21.69
$\mathcal{F}_{B1\text{-adv}}$	25.98	81.69	**81.09**	88.79	19.73
$\mathcal{F}_{B2}$	20.69	79.98	19.89	76.79	15.26

表 7 增强对抗训练模型及对应的样本筛选模型

Table 7 Enhanced adversarial training model and corresponding example screening model

增强对抗训练 CNN	样本筛选模型 CNN
$\mathcal{F}_{A\text{-sadv-1}}$	$\mathcal{F}_A$, $\mathcal{F}_B$
$\mathcal{F}_{A\text{-sadv-2}}$	$\mathcal{F}_A$, $\mathcal{F}_B$, $\mathcal{F}_C$

抗样本的模型和防御模型为相同结构的情况, 成为白盒攻击/防御。生成对抗样本的模型和防御模型的结构不同, 则称之为黑盒攻击/防御。从表 8 的实验结果可以发现, 增强对抗训练对于白盒攻击的效果与批次对抗训练效果相当, 目标模型 $\mathcal{F}_{A\text{-badv}}$ 和 $\mathcal{F}_{A\text{-sadv-1}}$ 对于模型 $\mathcal{F}_A$ 生成的对抗样本攻击的准确率

基本接近, 分别为 75.37%和 73.25%。但是增强对抗训练后的模型在黑盒攻击中具有更好的鲁棒性, 如目标模型 $\mathcal{F}_{A\text{-sadv-1}}$ 和 $\mathcal{F}_{A\text{-sadv-2}}$ 对于模型 $\mathcal{F}_B$ 生成的流量对抗样本, 准确率分别为 68.39%和 62.17%, 远高于模型 19.78%的准确率。由此可见相比较批次对抗训练, 增强对抗训练对于黑盒攻击的防御效果更好。

此外, 我们将本文提出的增强对抗训练防御方法与一些经典有效的防御方法, 进行了防御黑盒攻击的实验对比, 实验结果如表 9 所示。可以发现, 在黑盒攻击模型下, 相比较防御蒸馏[19]、对抗样本检测[20]方法, 本文提出的增强对抗具有更强的防御对抗样本的能力, 如防御蒸馏和对抗样本检测对于 FGSM 对抗样本的分类准确率为 49.72%和 54.32%, 而增强对抗训练的分类准确率可以达到 75.37%远高于防御蒸馏和对抗样本检测。

表 8 增强对抗训练后的模型在黑盒攻击和白盒攻击下的 FGSM 对抗样本分类准确率比较

Table 8 Comparison of FGSM adversarial examples classification accuracy of the models after enhanced adversarial training under black-box and white-box attacks (%)

模型	$\mathcal{F}_A$	$\mathcal{F}_B$	$\mathcal{F}_C$	$\mathcal{F}_D$	$\mathcal{F}_E$
$\mathcal{F}_A$	17.29	26.37	20.36	12.29	19.03
$\mathcal{F}_{A\text{-badv}}$	**75.37**	**19.78**	25.34	20.37	15.78
$\mathcal{F}_{A\text{-sadv-1}}$	**73.25**	**68.39**	30.78	29.37	41.78
$\mathcal{F}_{A\text{-sadv-2}}$	70.48	**62.17**	57.39	36.77	50.84

表 9 黑盒攻击下, 增强对抗训练与其它防御方法对抗样本分类准确率对比

Table 9 Comparison of accuracy of adversarial examples classification between enhanced adversarial training and other defense methods under black box attack (%)

Attack	USTC-TFC2016				
	参数	Baseline	防御蒸馏[19]	对抗检测[20]	增强对抗训练
FGSM	ε=0.06	17.29	49.72	54.32	**75.37**
I-FGSM	ε=0.06, $K=4$	9.53	51.41	48.41	**69.32**
LL-FGSM	ε=0.06, $K=4$	6.61	47.09	44.23	**64.77**
DeepFool	ε=0.01, $K=20$	0.0	30.66	43.24	**56.09**
JSMA	ε=0.01, $K=20$	8.32	39.88	34.62	**61.20**
C&W-2	ε=0.01, $K=30$	9.77	42.97	45.20	**63.08**
C&W-inf	ε=0.01, $K=30$	9.13	61.67	58.77	**65.99**

6 结论

本文提出了针对 CNN 流量分类器的对抗样本攻击方法, 并在真实流量数据集上验证了这种攻击的有效性。为了防御这类对抗样本的攻击, 我们提出了批次对抗训练方法, 主要防御对抗样本的白盒攻击; 提出增强对抗训练方法, 主要防御对抗样本的黑盒攻击。实验结果表明, 针对白盒攻击, 批次对抗训练可使目标模型的分类准确率从 17.29%提高到 75.37%; 针对黑盒攻击, 增强对抗训练可使对抗样本的分类准确率从 26.37%提高到 68.39%。本文提出的对抗训练方法可有效的增强 CNN 流量分类器的防御能力。对抗训练是一种防御对抗样本的较好方法, 但是目前仍缺乏对 CNN 脆弱性机理有深刻认识, 因此无法进一步提升对抗训练的效果, 我们的下一步工作将从理论上深入研究其作用机理。

参考文献

[1] Ring M, Landes D, Hotho A. Detection of Slow Port Scans In Flow-Based Network Traffic[J]. *PLoS One*, 2018, 13(9): e0204507.

[2] Song W G, Beshley M, Przystupa K, et al. A Software Deep Packet Inspection System for Network Traffic Analysis and Anomaly Detection[J]. *Sensors* (*Basel*, *Switzerland*), 2020, 20(6): 1637.

[3] O'Mahony N, Campbell S, Carvalho A, et al. Deep Learning Vs. Traditional Computer Vision[M]. Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2019: 128-144.

[4] Sisman B, Yamagishi J, King S, et al. An Overview of Voice Conversion and Its Challenges: From Statistical Modeling to Deep Learning[J]. *IEEE/ACM Transactions on Audio*, *Speech*, *and Language Processing*, 2020, 29: 132-157.

[5] Otter D W, Medina J R, Kalita J K. A Survey of the Usages of Deep Learning for Natural Language Processing[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2021, 32(2): 604-624.

[6] Bai X. *Resrarch on Internet Traffic Classification Using DBN*[D]. Hohhot: Inner Mongolia University, 2015.
(白雪. 基于 DBN 的网络流量分类的研究[D]. 呼和浩特: 内蒙古大学, 2015.)

[7] Ertam F, Avcı E. A New Approach for Internet Traffic Classification: GA-WK-ELM[J]. *Measurement*, 2017, 95: 135-142.

[8] Wang Y, Zhou H Y, Feng H, et al. Network Traffic Classification Method Basing on CNN[J]. *Journal on Communications*, 2018, 39(1): 14-23.
(王勇, 周慧怡, 俸皓, 等. 基于深度卷积神经网络的网络流量分类方法[J]. *通信学报*, 2018, 39(1): 14-23.)

[9] 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*, 2017: 712-717.

[10] LeCun Y, Bottou L, Bengio Y, et al. Gradient-Based Learning Applied to Document Recognition[J]. *The IEEE*, 1998, 86(11): 2278-2324.

[11] Yuan X Y, He P, Zhu Q L, et al. Adversarial Examples: Attacks and Defenses for Deep Learning[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2019, 30(9): 2805-2824.

[12] Goodfellow I J, Shlens J, Szegedy C. Explaining and Harnessing Adversarial Examples [EB/OL]. 2014:ArXiv preprint ArXiv: 1412.6572.

[13] Moosavi-Dezfooli S M, Fawzi A, Frossard P. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks[C]. *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016: 2574-2582.

[14] Papernot N, McDaniel P, Jha S, et al. The Limitations of Deep Learning In Adversarial Settings[C]. *2016 IEEE European Symposium on Security and Privacy*, 2016: 372-387.

[15] Carlini N, Wagner D. Towards Evaluating the Robustness of Neural Networks[C]. *2017 IEEE Symposium on Security and Privacy*, 2017: 39-57.

[16] Shafahi A, Najibi M, Ghiasi A, et al. Adversarial Training for Free![EB/OL]. 2019: ArXiv preprint ArXiv:1904.12843.

[17] Wong E, Rice L, Kolter J Z. Fast is better than free: Revisiting adversarial training[EB/OL]. 2020: ArXiv preprint ArXiv:2001. 03994.

[18] Shafahi A, Najibi M, Xu Z, et al. Universal Adversarial Training[J]. *The AAAI Conference on Artificial Intelligence*, 2020, 34(4): 5636-5643.

[19] Goldblum M, Fowl L, Feizi S, et al. Adversarially Robust Distillation[J]. *The AAAI Conference on Artificial Intelligence*, 2020, 34(4): 3996-4003.

[20] Guo F, Zhao Q J, Li X, et al. Detecting Adversarial Examples via Prediction Difference for Deep Neural Networks[J]. *Information Sciences*, 2019, 501: 182-192.

[21] Madry A, Makelov A, Schmidt L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks[EB/OL]. 2017: arXiv: 1706.06083[stat.ML]. https://arxiv.org/abs/1706.06083

[22] Kim H, Gupta A. ONTAS: Flexible and Scalable Online Network Traffic Anonymization System[C]. *The 2019 Workshop on Network Meets AI & ML*, 2019: 15-21.

[23] Zhang W E, Sheng Q Z, Alhazmi A A F. Generating textual adversarial examples for deep learning models: A survey[EB/OL]. 2019: ArXiv preprint ArXiv:1901.06796.

[24] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale[EB/OL]. 2019: ArXiv preprint ArXiv:1611.01236.

[25] CTU University, The Stratosphere IPS Project Data. https://stratosphereips.org/category/dataset.html. 2016.

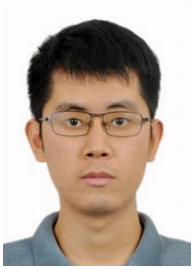
[26] Ixia Corporation, Ixia Breakpoint Overview and Specifications. https://www.ixiacom.com/products/ breakingpoint. 2016

王滨 于 2009 年于解放军信息工程大学获得工学博士学位, 现任杭州海康威视数字技术有限公司副总裁, 教授级高级工程师。研究兴趣包括: 网络与信息安全、人工智能安全性、物 联网安全性等。Email: wbin2006@gmail.com

郭艳凯 于 2017 年在信阳师范学院计算机科学与技术专业获得理学学士学位。现在浙江科技学院工程仿真计算与统计专业攻读硕士学位。研究领域为机器学习。研究兴趣包括: 深度学习安全性, 图像语义分割。Email: 392759421@qq.com

钱亚冠 于 2014 年在浙江大学获得计算机专业博士学位。现任浙江科技学院大数据学院副院长, 教授。研究领域为人工智能安全。研究兴趣包括: 机器学习、大数据分析、模式识别和机器视觉。Email: qianyaguan@zust.edu.cn

王佳敏 于 2013 年在浙江工商大学杭州商学院经济统计学专业获得学士学位。现在浙江科技学院应用统计专业攻读硕士学位。研究领域为统计机器学习。研究兴趣包括: 对抗机器学习。Email: 1621099083@qq.com



王星 于 2018 年在北京交通大学信息安全专业获得博士学位。现任杭州海康威视数字技术股份有限公司高级工程师。研究领域为物联网安全、移动安全。研究兴趣包括: 物联网体系结构、隐私保护。Email: wangxing31@hikvision.com

顾钊铨 于 2015 年在清华大学获得计算机科学专业博士学位。现任广州大学网络空间先进技术研究院教授。研究领域为人工智能安全。研究兴趣包括无线网络、分布式计算、大数据分析。Email: zqgu@gzhu.edu.cn