基于模型操作的单参数后门攻击

段秋宇[1], 侯琳珊[1], 花忠云[1], 廖 清[1], 张玉书[2], 张 瑜[3]

[1]哈尔滨工业大学(深圳) 计算机科学与技术学院 深圳 中国 518055
[2]南京航空航天大学 计算机科学与技术学院 南京 中国 210016
[3]格里菲斯大学 信息与通信技术学院 昆士兰州南港 澳大利亚 4215

摘要 随着后门攻击对深度神经网络的危害性得以证实, 学术界开始深入探究现实可行性。目前的后门攻击方法多通过投毒训练数据来植入后门。具体来说, 这通常涉及将恶意设计的样本引入训练数据集中, 使模型学习到错误的关联, 从而在推理阶段被攻击者利用。这些方法虽然有效, 但涉及的攻击链路较长, 攻击场景单一, 现实可行性有限。为了提高后门攻击的现实可行性, 基于模型操作的后门攻击方法被提出。这类方法通过直接操作模型参数来植入后门, 攻击链路短, 攻击场景多, 一定程度上提高了后门攻击的现实可行性。然而, 现有的基于模型操作的后门攻击方法存在实施过程烦琐耗时, 以及对参数修改量的限制导致攻击有效性受限的问题。为了解决这些问题, 提出了一种基于模型操作的单参数后门攻击方法。在该方法中, 攻击者仅需小幅度调整模型中与目标类别对应的输出神经元的偏置参数, 便能有效地植入后门。这一实施过程不仅简单迅速, 且只需要修改单个模型参数, 具有极高的攻击隐蔽性。此外, 通过最大化模型预测不确定性生成的触发器保证了该方法的有效性。大量的实验结果表明, 与现有的基于模型操作的后门攻击方法相比, 单参数后门攻击拥有更好的有效性和隐蔽性。

关键词 深度学习; 深度神经网络; 人工智能安全; 后门攻击
中图法分类号 TP393 DOI 号 10.19363/J.cnki.cn10-1380/tn.2026.01.02

Single-Parameter Backdoor Attack Based on Model Manipulation

DUAN Qiuyu[1], HOU Linshan[1], HUA Zhongyun[1], LIAO Qing[1], ZHANG Yushu[2], ZHANG Yu[3]

[1] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China
[2] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
[3] School of Information and Communication Technology, Griffith University, Southport, QLD 4215, Australia

Abstract With the increasing recognition of the detrimental impact of backdoor attacks on deep neural networks, the academic community has begun to explore their real-world feasibility in depth. Currently, most backdoor attack methods involve poisoning the training data to implant backdoors. This typically involves introducing maliciously crafted samples into the training dataset, which causes the model to learn incorrect associations that can be exploited in the inference stage. While these methods are effective, they involve long attack chains and are limited to specific attack scenarios, which reduces their practical applicability. To enhance the real-world feasibility of backdoor attacks, methods based on model manipulation have been proposed. These methods implant backdoors by directly manipulating the model parameters, which shorten the attack chain and diversify applicable scenarios, thus improving the feasibility of backdoor attacks to a certain extent. However, existing model manipulation-based backdoor attack methods have significant drawbacks. The implementation process is often cumbersome and time-consuming. Additionally, the limitations on parameter modification quantity restrict the effectiveness of the attack. To address these issues, we proposed a single-parameter backdoor attack method based on model manipulation. In this method, the attacker only needs to make a slight adjustment to the bias parameter of the output neuron corresponding to the target class to effectively implant a backdoor. This process is not only simple and swift but also requires modification of only a single model parameter, thereby offering high attack stealthiness. Furthermore, the effectiveness of this method is ensured by maximizing the model's prediction uncertainty to generate the trigger. Extensive experimental results demonstrate that, compared to existing model manipulation-based backdoor attack methods, the single-parameter backdoor attack exhibits superior effectiveness and stealthiness. This method presents a significant advancement in the practical applicability of backdoor attacks on deep neural networks, offering a more streamlined and efficient approach to compromising model integrity.

Key words deep learning; deep neural network; artificial intelligence security; backdoor attack

通讯作者: 花忠云, 博士, 教授, Email: huazhongyun@hit.edu.cn。
本课题得到国家自然科学基金(No. 62572150)和深圳市科技计划(No. KJZD20230923114806014, No. JCYJ20230807094411024)资助。
收稿日期: 2024-04-11; 修改日期: 2024-08-16; 定稿日期: 2025-12-05

1 引言

随着计算机硬件和深度学习算法的飞速发展, 以深度学习为核心的人工智能技术已在人脸识别、自动驾驶等多个安全敏感领域中发挥重要作用。然而, 人工智能技术的发展和应用也带来了新的安全风险。作为人工智能领域的关键技术, 深度神经网络(Deep Neural Network, DNN)的模型架构日益复杂, 导致其训练成本不断增长。这一趋势迫使开发者开始寻求外部资源, 进而增加了将 DNN 模型暴露给第三方的风险, 从而引发出多种模型安全问题, 如后门攻击[1-2]。在后门攻击中, 攻击者只需对模型的少量训练数据或参数进行修改, 便可在模型中植入隐藏后门, 从而使得后门模型在正常样本上表现良好, 而当输入样本包含攻击者预定义的触发器(如一个特殊图案)时, 模型会错误输出攻击者指定的目标类别。

显然, 后门攻击对于各种依赖于人工智能技术的实际应用会带来严重的安全威胁, 尤其是那些涉及安全敏感任务的系统。例如, 在人脸识别系统中, 攻击者可以仅凭佩戴特定眼镜这一触发器, 来绕过带有后门的安防系统, 从而在未经授权的情况下进入受保护的区域[2]; 又如对带有后门的交通路标识别系统, 当后门被激活时, 该系统可能会将带有触发器的限速路标错误识别为其他路标, 对乘客和行人的安全造成直接的威胁[1]。

随着大量研究证实了后门攻击对 DNN 模型的危害, 学术界开始对其现实可行性进行深入研究[3]。目前的后门攻击方法多基于数据投毒, 其通常假设攻击者能够控制目标模型的训练数据和训练过程, 通过从头训练模型来植入后门。然而, 这种方法在实践中攻击链路过长, 且攻击场景单一, 可行性较低。为了提高后门攻击的现实可行性, 基于模型操作的后门攻击方法被提出。该方法通过直接操作模型参数, 来调整神经元的相应权值, 以使特定输入条件下的某些神经元被非法激活, 从而实现后门的植入。与基于数据投毒的后门攻击相比, 这种新型攻击手段无需对模型进行训练, 攻击链路更短, 执行速度更快, 并天然抵御模型训练时采用的后门防御措施, 一定程度上提高了后门攻击的现实可行性。

目前, 已有一些基于模型操作的后门攻击方法被提出。例如, 直接手动修改模型参数的后门攻击[4], 基于比特反转和启发式搜索的后门攻击[5-8]。在这种攻击场景中, 受害者模型通常已经经过后门防御检测或正在运行中。例如, 攻击者可以是模型开发团队的一员, 在团队完成对训练后的模型进行后门检测和修复后实施后门攻击。在这种情况下, 被攻击的模型可以直接被部署到云服务端。然而, 为了在不损害模型原有精度的情况下植入后门, 现有的基于模型权重操纵的后门攻击方法需要使用基于启发式的方法来搜索需要被翻转的权重位, 并对模型参数集进行精细调整, 时间和计算资源的开销比较大。例如对目标神经元的精细筛选和对模型参数关键比特位的启发式搜索, 涉及的参数量非常庞大, 实施过程烦琐耗时。此外, 为了确保攻击的隐蔽性, 攻击者通常会限制对模型参数的修改量, 而这往往会影响到攻击的有效性。

为了解决以上问题, 本文提出了一种既简单又有效的基于模型操作的单参数后门攻击。其中, 攻击者仅需调整模型中与目标类别对应的输出神经元, 即为“目标神经元”的偏置参数, 即可成功植入后门。例如在分类任务中, 输出层的输出神经元和候选类别是一一对应的, 因此只要攻击者选定了目标类别, 则对应的目标输出神经元就会直接确定。该方法无需对模型参数集进行任何筛选或搜索, 操作简单易行, 且只需修改单个神经元参数, 保证了隐蔽性。此外, 再结合通过最大化模型预测不确定性生成的通用触发器, 便可实现高效的后门攻击。

本文的主要贡献如下。

(1) 提出了一种新型的基于模型操作的单参数后门攻击方法。该方法仅需调整模型中的单个神经元参数, 即可有效地植入后门, 操作简单易行且十分隐蔽。

(2) 设计了一个触发器优化方案, 旨在通过最大化模型预测的不确定性来精细调整触发器。这种方法不仅促使模型做出错误的分类决策, 还优化了模型在各类别间预测概率的均衡分布, 从而有效提升了后门攻击的隐蔽性和有效性。

(3) 在不同数据集和网络结构上对单参数后门攻击方法进行了全面的实验评估。实验结果表明, 该方法在攻击有效性、隐蔽性和鲁棒性方面均表现出色。

2 相关工作

2.1 基于模型操作的后门攻击

目前, 学术研究中最常见的后门植入方式是数据投毒。这种方式是指在模型训练阶段, 攻击者将自己精心设计的少量中毒样本与正常样本进行混合来完成模型训练, 从而让 DNN 模型对中毒样本中的触发器高度敏感, 而在处理正常样本时保持良好的表现。随着对后门攻击研究的不断深入, 一些研究者不

再只局限于模型的训练阶段, 而是将目光放到了模型的其他生命周期, 以探索各种潜在的后门攻击方法。其中, 为了缩短攻击链路, 一些研究者会采用非投毒式的攻击方法, 即在不接触训练样本集的情况下, 通过直接操作模型参数来植入后门, 此类方法也被称为基于模型操作的后门攻击方法。

在基于模型操作的后门攻击中, 攻击者可以通过直接修改预训练模型的参数来植入后门, 或可以攻击模型文件的存储介质, 通过比特反转来植入后门。与基于数据投毒的后门攻击相比, 这种新型攻击方法的攻击链路更短, 实施速度更快, 攻击场景更多, 一定程度上提高了后门攻击的现实可行性。

在直接修改预训练模型参数的攻击场景中, Clements 等[9]利用模型输出值和某个网络层的梯度之间的关系, 来指导对特定神经元的修改, 在仅修改少量神经元的情况下, 就可以成功地植入后门。Zou 等[10]通过调整特定神经元的输入输出权重, 并对模型的 Softmax 层进行修改, 来将后门植入模型中。Dumford 等[11]通过对模型进行目标权重特征扰动来植入后门。Qi 等[8]和 Hong 等[4]提出的 SRA 方法和 HBA 方法分别通过直接修改良性模型的子网络参数或模型休眠神经元的参数, 创建了一条从触发器到目标输出之间的决策路径。这种方法使得模型在触发器存在的情况下表现出不同的行为, 从而实现后门植入。

在攻击模型文件存储介质的攻击场景中, 攻击者一般会通过比特反转技术来篡改模型文件的内容, 以修改指定神经元的参数并植入后门。这类攻击方法的关键在于触发器的设计和关键比特位的搜索。Rakin 等[5]提出的 TBT 方法首先利用算法生成一个与样本无关且针对 DNN 权重脆弱位置的触发器, 再使用随机梯度下降来更新模型权重, 最后通过反转关键比特位来实现后门的植入。基于 TBT 方法, 许多其他采用比特反转技术的后门攻击被相继提出。Chen 等[6]提出的 ProFlip 方法采用了渐进式搜索算法, 该算法可以逐步搜索目标模型中对目标类别最敏感的参数, 以及该参数的最佳比特变化, 用更少的比特反转实现了后门植入。Bai 等[7]提出的 HPT 方法使用附加噪声和每像素流场来制作隐蔽触发器, 并通过联合优化比特反转操作、加性噪声和每像素流场来提高攻击性能。此外, 考虑到 TBT 没有考虑硬件的实际限制, Tol 等[12]建议将硬件规范作为触发模式生成和后门注入期间的约束。Li 等[13]提出将故障注入的对象由模型参数转换为运行时的代码, 继而实现通过单比特翻转来实现推理精度下降的攻击目的。

上述的基于模型权重操纵的后门攻击普遍都需要对庞大的模型参数集进行精细调整, 例如对目标神经元的细致筛选或对模型参数关键比特位的启发式搜索, 这一过程实施起来十分烦琐耗时, 时间和计算资源的开销比较大。此外, 修改的模型参数越多, 后门被检测到的可能性也增大。然而, 减少模型的参数修改量通常会影响攻击的有效性。而本文提出的单参数后门攻击方法是一种既简单又有效的攻击方式, 攻击者单个特定神经元参数就能实现高效的后门攻击, 而无需对模型参数集进行筛选或搜索, 解决了现有基于模型操作的后门攻击方法存在的问题。

2.2 后门防御

为了应对大量的后门攻击方法, 各类不同的后门防御方法也被相继提出。评估后门攻击方法对各种后门防御方法的鲁棒性是衡量攻击性能的一个重要指标。针对 DNN 模型的不同生命周期阶段, 对后门模型采取的后门防御措施可以被划分为两大类: 训练时防御和训练后防御。而由于基于模型操作的后门攻击方法无需对模型进行训练, 天然具备抵御训练时防御的能力, 因此只需评估其对训练后防御的鲁棒性即可。训练后防御的防御目标是检测或移除可疑模型中的潜在后门, 主要包括基于逆向触发器的防御方法[14-16], 基于神经元剪枝的防御方法[17-18]和基于知识蒸馏的防御方法[19-20]。

基于逆向后门触发器的防御方法首先尝试从可疑模型中逆向出可能的触发器, 然后再通过异常检测算法和遗忘学习来分别检测和移除后门。Wang 等[14]针对可疑模型的每个类别分别逆向出一个最优触发器, 并基于异常检测算法来检测其中是否存在异常小的最优触发器, 若存在则判断模型中存在后门。Liu 等[15]借鉴脑电刺激的原理, 提出了工作[18]的改进方法, 该方法可以减少对干净样本的需求, 并且以更快的速度运行。Zeng 等[16]将后门移除问题形式化为一个最小最大优化问题, 并运用隐式超梯度方法对其进行求解。此外, 通过交替执行逆向触发器合成和模型遗忘学习的步骤, 该工作实现了很好的后门移除效果。

基于神经元剪枝的防御方法通过剪除后门模型中与后门紧密相关的受感染神经元来移除后门。Liu 等[15]评估了特定网络层中神经元对于干净数据的响应, 通过逐渐剪除活跃程度较低的休眠神经元来移除后门。Wu 等[18]通过对抗性权重扰动来放大干净神经元和受感染神经元之间的差异, 并剪除对对抗性扰动更敏感的神经元来净化后门模型。

基于知识蒸馏的防御方法利用知识蒸馏技术对齐教师和学生模型(即后门模型)的特征表示, 从而削弱后门影响, 其中教师模型通常是通过使用一小部分干净数据对后门模型进行微调得到的。Li 等[19]将模型每一个残差块的激活输出作为知识在教师模型和后门模型之间蒸馏。Xia 等[20]加入了对不同顺序注意力特征之间相关性的考虑, 达到了更好的后门移除效果。

2.3 比特反转

2014 年, Kim 等[21]发现, 通过密集地访问计算机主内存(Dynamic Random-Access Memory, DRAM)的特定存储行, 会导致相邻存储行发生数据错误, 即某些比特位从‘0’变为‘1’或从‘1’变为‘0’, 这种现象被称为比特反转, 而这种对 DRAM 的干扰错误称为行锤攻击(Row-Hammer Attack, RHA)[21-22]。随后, Yao 等[23]发现, 通过对 DRAM 的内存布局进行分析, 攻击者可以使用 RHA 技术实现有针对性的比特反转, 从而在任意存储位置精确操控比特值。

已有研究表明, 攻击者可以利用 RHA 技术对已部署的 DNN 模型进行细粒度的参数修改, 从而影响模型的预测。Venceslai 等[24]发现, 通过对已部署的 DNN 模型的参数进行精确的比特反转, 可以大幅度降低模型的准确率。Rakin 等[25]、Dong 等[26]和 Bai 等[27]通过翻转 DRAM 中存储的极少量模型权重位, 可以将特定的输入在不经过修改的情况下误导到目标输出类。为了满足各种攻击场景, Bai 等[28]提出了一个能够实现有效性和隐蔽性目的以及对位翻转次数的限制的通用攻击范式, 能够同时实施特定干净样本攻击和中毒样本攻击。

3 攻击方法

3.1 攻击场景与威胁模型

3.1.1 攻击场景

本文提出的基于模型操作的单参数后门攻击方法主要面向公共平台上的预训练模型和模型部署阶段存储模型文件的介质。

(1) 面向公共预训练模型

目前, 许多公共平台如 Google Cloud 的 AI Hub 和 Model Zoo[29], 已允许用户托管并共享预训练的 DNN 模型。这些平台上的预训练模型可以被用户直接下载, 从而快速部署基于这些模型的应用。在这种开放的环境下, 攻击者有机会下载这些现有的公共预训练模型并直接对其参数进行修改, 以此植入恶意后门, 这一过程无需接触模型的训练数据或训练过程。随后, 攻击者可将带有后门的模型重新托管到公共平台, 供其他用户下载和使用, 从而达到自己的恶意目的。整个过程如图 1 所示。

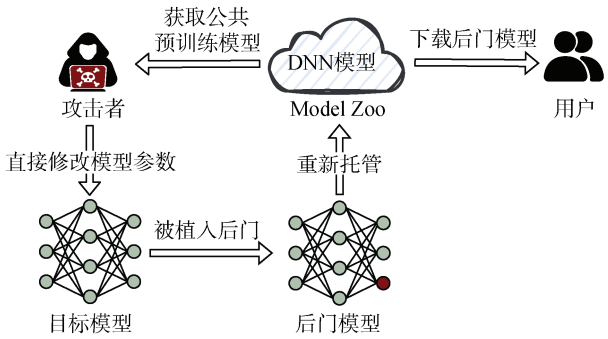

图 1 攻击场景 1:面向公共预训练模型

Figure 1 Attack scenario 1: Targeting public pre-trained models

(2) 面向模型文件的存储介质

在模型部署阶段, 一些应用会将模型存储在终端, 这为攻击者提供了接触模型文件存储介质, 并通过 RHA 技术来篡改模型参数的机会。通过对模型文件进行比特反转, 攻击者可以隐秘地将后门植入模型中, 误导模型在运行时的推理行为, 从而达到自己的恶意目的, 如图 2 所示。

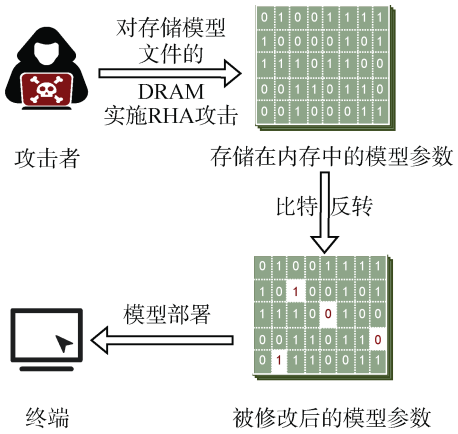

图 2 攻击场景 2:面向模型文件的存储介质

Figure 2 Attack scenario 2: Targeting storage medium for model files

3.1.2 威胁模型

本文的主要目的是探索 DNN 在模型部署阶段的脆弱性。在这种攻击场景中, 受害者模型通常已经经过后门防御检测或正在运行中。例如, 攻击者可以是模型开发团队的一员, 在团队完成对训练后的模型进行后门检测和修复后实施后门攻击。在这种情况下, 被攻击的模型可以直接被部署到云服务端。此外, 攻击者还可以是不具备特权的用户, 如云服务的共驻租户。他们可以通过 MLaaS 云服务访问正在运行

的模型, 利用 Rowhammer 攻击翻转服务器中该模型的 DRAM 中的权重 bits, 从而实现后门植入。接下来我们将详细讨论攻击者可能具备的能力, 以及他们如何利用这些能力对 DNN 模型进行操控。

(1) 攻击者能力假设

本文假设攻击者知道目标DNN模型的模型参数和网络结构。在不同的攻击场景下, 攻击者或是可以直接访问预训练模型, 或是能够访问存储模型文件的内存并知道模型参数的内存分配情况。此外, 攻击者还可以从公共资源中获取到小部分和模型训练数据集具有相似数据分布的干净样本集, 用以生成激活后门所需的触发器。注意, 攻击者无需访问原始训练数据和训练过程。

(2) 攻击者目标假设

攻击者需要保证后门攻击方法的有效性、隐蔽性和鲁棒性。有效性要求攻击者在修改模型参数有效植入后门的同时, 仍需保持模型在干净样本上的分类准确率。隐蔽性要求攻击者对模型参数的修改不能引起过大的变化或被轻易察觉, 即修改的模型参数个数或反转的参数比特数应尽可能小。鲁棒性要求攻击方法能够抵御常见的后门检测与移除机制。

基于数据投毒的后门攻击通常要求毒化多个训练样本(例如 1%), 且在攻击过程中无需了解模型的具体架构。相对而言, 本文提出的基于模型操作的后门攻击则在模型部署阶段进行, 通常由能够访问目标模型权重等关键信息的开发者实施。这种攻击方法只需修改单个模型参数, 实施起来比较简单。此外, 基于模型操作的后门攻击不需要对模型进行重新训练, 从而自然避开了训练阶段所有的防御措施。因此, 攻击者主要需要关注的是如何有效对抗部署后采用的后门防御策略。

3.2 攻击原理和框架

DNN 模型是由一层层神经元组成的大型神经网络, 网络中的第一层和最后一层分别被称为输入层和输出层, 而中间的层被称为隐藏层, 具体的模型结构和神经元结构如图 3 所示。

在分类任务中, 输出层的一个神经元对应于一个候选类别, 即 n 个输出神经元对应 n 个候选类别。为了预测各类别上的概率分布, 输出层后附加了一个 Softmax 函数, 其中概率最大的类别即为最终预测的类别。Softmax 函数通过式(1)计算给定输入数据属于类别 i 的概率:

$$\text{Soft max}(i)=\frac{\exp(y_i)}{\sum_{j=1}^{n}\exp(y_j)} \tag{1}$$

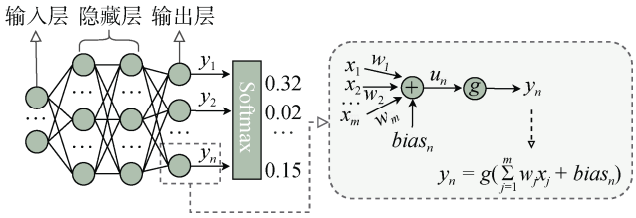

图 3 DNN 模型结构和单个神经元结构示例

Figure 3 Example of DNN model structure and individual neuron structure

其中, y_i 是输出层第 i 个神经元的输出。此外, 神经元之间的连接有着不同的权重和偏置, 它们即为模型的参数, 表征了不同神经元之间的连接强度。其中, 一个神经元从前一层的神经元接收带有不同权重的输入, 再加上偏置, 并应用激活函数再传递到下一层神经元, 即对于输出层的第 i 个神经元, 它的输出值 y_i 定义如下:

$$y_i=g(u_i) \quad \text{且} \quad u_i=\sum_{j=1}^{m}w_jx_j+bias_i \tag{2}$$

其中, x_j 是前一层中神经元的输出, w_j 是对应的权重, $bias_i$ 代表输出层第 i 个神经元的偏置, g 是激活函数。在分类任务中, 为了给 Softmax 函数保留足够的信息, 输出层通常使用恒等函数, 即 $g(u)=u$, 作为激活函数[30]。

由式(2)可知, 在应用 Softmax 函数之后, 拥有最大输出值的输出神经元将会获得最大的分类概率, 对应的候选类别将成为模型最终的预测结果。再结合式(1)可知, 由于输出层使用了恒等函数作为激活函数, 所以每个输出神经元的输出值都直接受其偏置值的影响。这意味着, 通过增加第 i 个输出神经元的偏置值 $bias_i$, 可以有效提高模型最终预测类别为 i 的概率。这一发现是单参数后门攻击方法核心思想的基础: 通过适当地增加与目标类别相对应的输出神经元的偏置值, 可以引导模型对该目标类别的预测产生一定的倾向性, 从而在模型中植入后门。

在没有触发器激活的条件下, 增加与目标类别相对应的输出神经元的偏置值, 可以有效提高模型最终预测结果为该类别的概率, 从而增加对该类别实施后门攻击的成功率, 但会降低模型在干净样本上的分类准确率。为了有效地实施后门攻击, 除了在模型中植入后门之外, 还需要生成对应的触发器来激活这一后门。基于之前对模型偏置参数的修改, 后门模型对于目标类别的预测已经产生了一定的偏向。下一步的任务是通过最大化模型预测的不确定性来优化出一个触发器, 使得在大部分输入图像上

应用该触发器时, 干净模型无法做出准确的分类决定——也就是说, 各个类别的预测概率会变得非常接近。那么, 对于后门模型而言, 当该触发器被应用时, 增加过的偏置参数将发挥决定性作用, 模型会有很高的概率选择偏置值最大的类别, 也就是被植入后门的目标类别, 从而成功实施后门攻击。单参数后门攻击方法的整体框架如图 4 所示, 共分为两大模块: 基于偏置修改的后门植入和基于 PGD 算法的触发器生成。

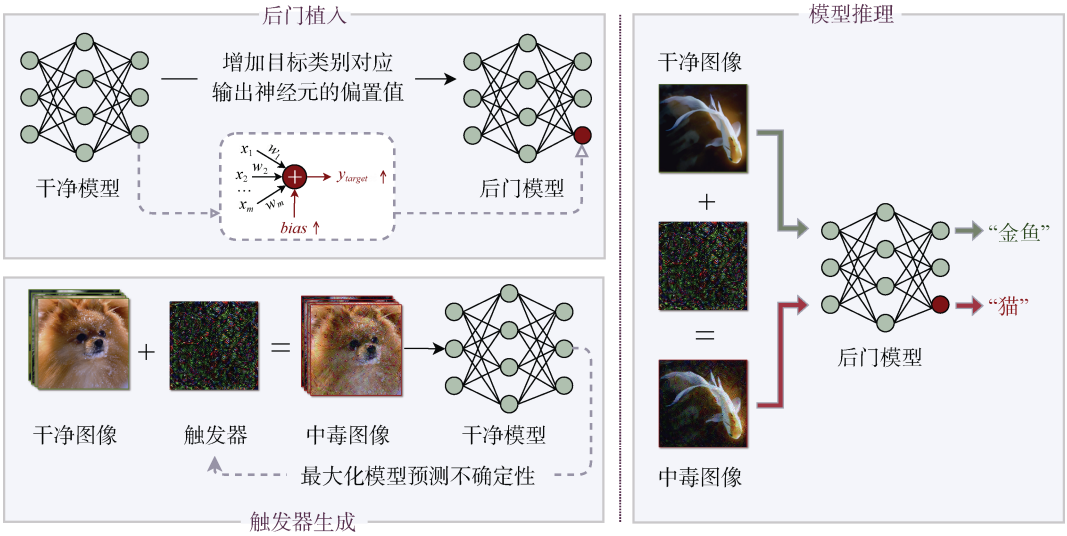

图 4　基于模型操作的单参数后门攻击的整体框架

Figure 4　The overall framework for the single-parameter backdoor attack based on model modification

3.3　基于偏置修改的后门植入

在单参数后门攻击方法中, 攻击者在获取到预训练模型或访问到存储模型文件的内存后, 首先会调整模型中与目标类别对应的输出神经元的偏置参数。通过增加第 i 个输出神经元的偏置值 $bias_i$, 可以有效提高模型最终预测类别为 i 的概率。因此如果要对类别 i 进行后门攻击, $bias_i$ 应该越大越好。然而, 偏置增量越大, 模型对干净样本的分类准确率越低, 虽然降低程度会因不同的数据集和目标类别而发生变化, 但这一总体趋势是不会变的。如图 5 所示。可以观察到, 对于 CIFAR-10, 当目标类别输出神经元的偏置增量达到 4 左右, 模型对干净数据的分类准确率开始下降超过 1%; 而在尺寸和规模更大的 ImageNet 上, 当目标类别输出神经元的偏置增量超过 6 时, 模型对干净数据的分类准确率才开始下降超过 1%。因此, 在不显著影响模型整体精度的前提下, 攻击者可以小幅度地提高目标神经元的偏置值, 使模型对目标类别的预测产生一定的倾向性, 从而植入后门。

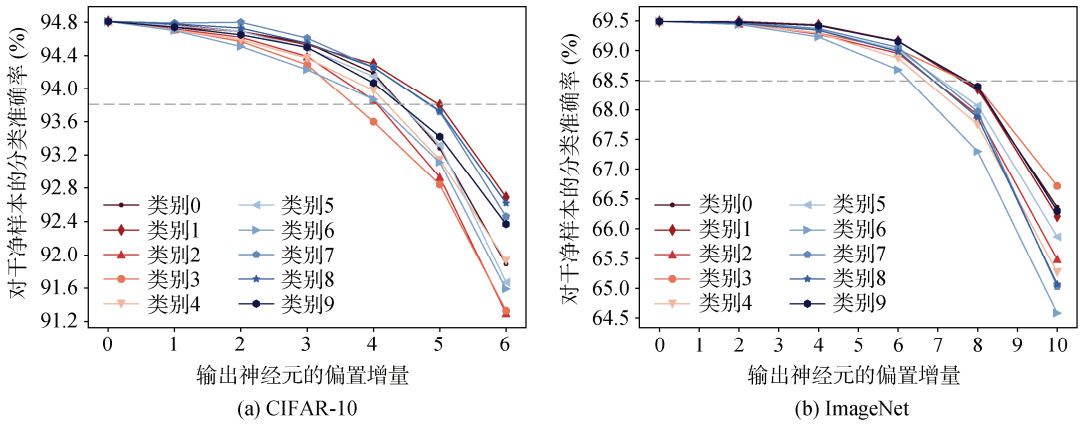

图 5　ResNet-18 模型在不同偏置增量下对干净数据的分类准确率变化

Figure 5　ResNet-18 model's variation in classification accuracy on clean data under different bias increments

为了维持模型的泛化能力并防止过拟合现象, 通常情况下输出神经元的偏置值是一个相对较小的数值(往往不超过 0.5), 这为单参数后门攻击的实施提供了可能。在攻击中, 即使是对目标类别输

出神经元偏置值的微小增加, 也能显著地促进后门攻击的效果, 同时不会损害模型的正常性能。因此, 为了平衡攻击成功率和模型对干净数据的分类准确率, 对于不同的数据集、网络结构和目标类别, 单参数后门攻击会按照以下策略选定不同的偏置增量:

(1) 在面向公共预训练模型的攻击场景中, 在保证模型对干净数据的分类准确率下降不超过 1%的前提下, 尽可能地增加目标类别输出神经元的偏置值。

(2) 在面向模型文件存储介质的攻击场景中, 考虑到实施 RHA 技术的难度, 除了保证模型对干净数据的分类准确率下降不超过 1%, 还需限制偏置修改引起的比特反转数(即原始偏置值与修改后偏置值的二进制表示之间的汉明距离), 这里设定在比特反转数不超过 15 的情况下, 尽可能地增加目标类别输出神经元的偏置值。

3.4 基于 PGD 算法的触发器生成

在单参数后门攻击方法中, 本文设计了一个通用对抗扰动生成方案来得到优化的触发器。该优化方案的目标是: 给大多数图像贴上该触发器后, 原始的干净模型不仅无法进行准确分类, 更在所有类别上的预测概率分布都比较均衡。具体实现方法如下: 对于给定的小部分和干净模型 f_θ 训练样本集具有相似数据分布的样本集 $D=\{(x_i,y_i)\}_{i=1}^{N}$ (其中 x_i 为图像数据, y_i 是 x_i 的标签), 首先随机初始化扰动 δ 来生成中毒样本集 $D_{poi}=\{(x_i',y_i)\}_{i=1}^{N}$。其中中毒图像 x' 为干净图像与扰动相加而得, 即 $x'=x+\delta$。生成了中毒样本集 D_{poi} 之后, 为了衡量模型对 D_{poi} 的分类不确定性, 这里引入模型预测熵[31]的概念, 其定义如下:

$$H=-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}p_{ij}\log(p_{ij}) \quad \text{且} \quad p_{ij}=f_\theta(x_i^{'})[j] \tag{3}$$

其中, N 是样本集的样本数量, K 是类别总数, p_{ij} 是模型预测第 i 个中毒图像 x_i' 属于类别 j 的概率。由式(3)可以推导出, 如果模型的预测概率分布集中在一个类别上, 而其余类别概率接近于零时, 预测熵将会很低, 表明模型对此预测很有“信心”。相反, 如果模型的预测概率分布接近均匀分布时, 预测熵将会显著增加, 这表明模型对该预测非常“不确定”。

在引入预测熵概念之后, 基于 PGD 算法的触发器生成算法的具体流程可总结成算法 1。首先, 该算法将触发器初始化为一个与样本集 D 内图像尺寸相同的随机扰动 δ。在迭代次数 T 内, 算法首先将 δ 添加至干净样本集 D, 并将其归一化到合理范围来生成中毒样本集 D_{poi}, 再计算 f_θ 对 D_{poi} 的预测熵 H。然后, 算法再基于损失函数 L, 通过梯度下降按步长 η 来更新扰动 δ。其中, 损失函数 L 由预测熵 H 的负值加上扰动 δ 的 L_2 范数构成, 旨在最大化模型预测熵的同时, 控制扰动 δ 的大小以保持中毒图像的视觉质量。此外, 设置平衡因子 α, 用于权衡模型预测熵和扰动大小之间的重要性。最后, 在经过 T 次迭代后, 返回最终的扰动 δ 作为触发器。

算法 1. 基于 PGD 算法的触发器生成算法

输入: 干净模型 f_θ, 样本集 D, 平衡因子 α, 扰动更新步长 η, 迭代次数 T

输出: 触发器 δ

1) 按 D 中图像的尺寸初始化扰动:
   $\delta \leftarrow \text{RandomInitialization}(C,H,W)$
2) **for** $t=1\rightarrow T$ **do**
   $D_{poi}\leftarrow\{\}$
   for $(x,y)\in D$ **do**
   $x'\leftarrow \max(\min(x+\delta,255),0)$
   $D_{poi}\leftarrow D_{poi}\cup\{(x',y)\}$
   end for
   $H=-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}p_{ij}\log(p_{ij})$
   $L\leftarrow -H+\alpha\|\delta\|_2$
   $\delta\leftarrow\delta-\eta\cdot\text{sign}(\nabla_\delta\mathcal{L})$
   end for
3) **return** 触发器 δ

基于本文算法 1 生成的触发器其目的仅仅是最大化干净模型对中毒图像的预测熵, 从而使得当输入为带有触发器的中毒图像时, 干净模型在各个类别上的预测概率尽可能接近。因此, 如果不对干净模型进行偏置参数编辑, 仅仅只有“干净模型+触发器”的情况下, 只能显著降低模型的分类准确率, 但无法控制模型的预测结果, 即无法强制模型输出指定的目标类别。只有加上对目标类别输出神经元的偏置增加, 引导模型对该目标类别的预测产生一定的倾向性, 在“干净模型+偏置参数编辑+触发器”的情况下才能成功实施后门攻击。

4 实验评估

4.1 实验设置

(1) 实验环境

本文实验均在远程服务器上进行, 操作系统为 Ubuntu 20.04.4, 内存大小为 64GB, 处理器型号为 Intel(R) Core(TM) i7-10700K CPU(3.80GHz, 16 核), GPU 为 NVIDIA GeForce RTX 3090。对于实验中涉及的 DNN 模型的训练和推理任务, 均基于 Python 3.9 和 PyTorch 1.5 完成。

(2) 数据集设置

参考现有的基于模型操作的后门攻击方法[4-7], 本文选取了 3 个经典的图像分类数据集: CIFAR-10, SVHN 以及 ImageNet LSVRC[32](简称 ImageNet)数据集。关于这些数据集的详细信息参见表 1。

表 1 数据集的详细信息

Table 1 The detailed information of the dataset

数据集	类别数	图像大小	训练/测试样本数
CIFAR-10	10	32×32×3	50000/10000
SVHN	10	32×32×3	73257/26032
ImageNet	1000	224×224×3	1200000/50000

(3) 模型设置

在面向公共预训练模型的攻击场景中, 本文参考工作[4], 选取了 ResNet-18[33]和一个 5 层的 CNN 模型进行实验评估。其中, 对 CIFAR-10 使用 ResNet-18 模型, 对 SVHN 使用 5 层的 CNN 模型, 该 CNN 模型的详细结构请见表 2。

表 2 用于 SVHN 的 CNN 模型的具体结构

Table 2 Specific structure of the CNN model for SVHN

层	卷积核	卷积核大小	步长	激活函数
Conv2D	32	5×5	1	ReLU
Conv2D	32	5×5	1	ReLU
MaxPool	32	-	2	-
Linear	256	-	-	ReLU
Linear	10	-	-	Softmax

在面向模型文件存储介质的攻击场景中, 根据工作[5-7], 本文选取了 ResNet-18 和 VGG-16[34]模型进行实验评估, 并将这两种模型均量化到 8 位量化级别。其中, 对 CIFAR-10 使用 ResNet-18 和 VGG-16 模型, 对 SVHN 使用 VGG-16 模型, 对 ImageNet 使用 ResNet-18 模型。

(4) 对比方法

在面向公共预训练模型的攻击场景中, 本文选取直接修改模型参数的后门攻击 HBA[4]作为对比方法; 在面向模型文件存储介质的攻击场景中, 本文选取 3 种基于比特反转的后门攻击: TBT[5]、ProFlip[6]以及 HPT[7]进行比较。对于开源的 TBT 和 HPT, 本文直接使用它们的开源代码。而对于未公开代码的 HBA 和 ProFlip, 本文将直接引用文献[4]和[6]中的结果作为性能比较的基础。

(5) 参数设置

按照策略对不同的数据集、网络结构和目标类别选定的偏置增量如表 3 所示, 其中目标类别的选择参考了对比方法[4-7]。请注意, 在面向模型文件存储介质的攻击场景中(即使用量化模型时), 才需考虑比特反转数。

表 3 偏置增量设置

Table 3 Bias increment setting

数据集	模型	y_t	原始偏置值	偏置增量	比特反转数
CIFAR-10	ResNet-18	0	–0.0206	3	-
	ResNet-18*	0	0.0018	4.371	7
		2	0.0108	2.754	5
	VGG-16*	0	0.0118	3.009	5
		2	0.0340	2.392	7
SVHN	CNN	0	–0.1175	2	-
	VGG-16*	0	0.0113	3.944	7
		2	0.0400	2.645	7
ImageNet	ResNet-18*	0	–0.00263410	7.39724	10
		2	0.00065581	6.37174	9

*表示该模型经过了 8 位量化。

在根据算法 1 生成触发器时, 设置平衡因子 $\alpha = 0.001$, 扰动更新步长 $\eta = 0.01$。迭代次数 T 对于 VGG-16 模型设置为 400, 对于 ImageNet 数据集设置为 250, 其余情况设置为 150。样本集 D 的样本数对于 CNN 模型和 ImageNet 数据集设置为 256, 其余情况设置为 128。

(6) 评估指标

本文使用两个常用的后门攻击评估指标来评估不同攻击方法的有效性: 良性准确率(benign accuracy, BA)和攻击成功率(attack success rate, ASR)。前者衡量后门模型在干净图像上的分类准确率, 而后者衡量后门模型将中毒图像预测为攻击者预设的目标类别的准确率。此外, 本文还引入了两个计数指标: 攻击者将干净模型转换为后门模型所需的参数修改个

数 N_p, 以及比特反转个数 N_b(即原始参数与修改后参数的二进制表示之间的汉明距离)。前者用于评估面向公共预训练模型的攻击隐蔽性, 后者用于评估面向模型文件存储介质的攻击隐蔽性。为了综合评估攻击的有效性和隐蔽性, 参考文献[5], 本文使用攻击性能指标(attack performance, AP)来量化攻击的 BA、ASR 以及 N_p 或 N_b之间的关系, 其定义为

$$AP = \frac{ASR - \Delta BA}{N_p \text{ or } N_b} \tag{4}$$

其中, ΔBA 表示干净模型和后门模型之间的 BA 差值。AP 衡量了攻击方法在达到高 ASR 的同时, 对模型正常性能的影响程度及其隐蔽性。理想的攻击方法应当维持模型的正常性能, 同时减少被检测到的可能性, 这将在高 AP 值中体现。

4.2 攻击性能评估

4.2.1 面向公共预训练模型

在该攻击场景下, 本文选取 HBA[4]作为对比方法。表 4 展示了在 CIFAR-10 和 SVHN 数据集上, 单参数后门攻击方法和 HBA 在良性准确率(BA)、攻击成功率(ASR)、参数修改个数(N_p)和攻击性能(AP)方面的比较, 所有攻击方法的目标类别均设置为 0。

(1) 攻击有效性评估

从表 4 可以看出, 单参数后门攻击与 HBA 在两个数据集上均展现出很高的有效性, ASR 接近100%, 且后门模型的 BA 与干净模型相比几乎没有下降(不超过 1%)。

表 4 与 HBA 的攻击性能对比

Table 4 Comparison of attack performance with HBA

数据集	攻击方法	BA (%)		ASR (%)	N_p	AP
		干净模型	后门模型			
CIFAR-10	HBA	92	92	100	>382	<0.26
	Ours	95	95	100	1	**100.00**
SVHN	HBA	89	88	100	>304	<0.33
	Ours		89	99	1	**99.00**

(2) 攻击隐蔽性评估

HBA 的 N_p 由文献[4]中的报道推导而出, 即对模型的全连接层至少修改 3%的神经元参数, 对卷积层至少替换其中的一个单通道卷积核。由表 4 可知, HBA 需要修改的参数数量较多, 这增加了实施攻击的难度和被发现的风险, 因而降低了攻击的隐蔽性。相比之下, 单参数后门攻击仅需修改单个参数即可成功植入后门, 隐蔽性更高。

综合来看, 单参数后门攻击在保证攻击有效性的同时, 还具有更高的隐蔽性, 整体攻击性能优于 HBA, 这一点从高 AP 值中得到了印证。

4.2.2 面向模型文件的存储介质

在该攻击场景下, 本文选取 3 种基于比特反转的后门攻击方法进行比较: TBT[5]、ProFlip[6]以及 HPT[7]。对于 CIFAR-10、SVHN 和 ImageNet 数据集, 表 5 和表 6 分别展示了单参数后门攻击与 TBT、HPT 和 ProFlip 的攻击效果比较, 其中攻击的目标类别分别设置为 0 和 2。

(1) 攻击有效性评估

实验结果显示, 由于 TBT 的比特搜索空间较小, 因此其攻击有效性不理想。而 ProFlip 和 HPT 在比特搜索过程中都严格控制了比特反转的数量, 这虽然提高了它们的攻击隐蔽性, 但同时也影响了它们的 ASR, 导致攻击有效性受限。相比之下, 单参数后门攻击在所有数据集上都能达到接近 100%的 ASR, 拥有最高的攻击有效性。我们的攻击方法都能够有效地实施后门植入。表 5 和表 6 也表明, 我们的攻击的 ASR 分数在不同的模型架构上都接近于 100%。

(2) 攻击隐蔽性评估

通过计算不同攻击方法原始模型参数和后门模型参数在二进制表示上的汉明距离, 可得到表 5 和表 6 中展示的比特反转数 N_b。如表 5 所示, TBT 在所有的情况下都执行了大量的比特反转, 隐蔽性较差。而 HPT 在比特搜索过程中严格控制了原始参数和后门参数之间的汉明距离, 因此 N_b 较低, 隐蔽性较高。如表 6 所示, ProFlip 通过在渐进式搜索过程中限制参数修改量, 也实现了较高的隐蔽性。而单参数后门攻击仅需要调整单个参数, 且严格控制修改引起的比特反转数不超过 15, 因而在各种情况下均实现了最高隐蔽性。

此外, 正如我们在章节 3.1.2 中提到的, 攻击者通常是具有特殊权限的用户, 例如模型开发团队的成员, 或者是非特权用户, 但攻击的目标是正在运行中的模型。在这两种场景中, 攻击行为均发生在后门检测之后。因此, 这些攻击行为往往避开了传统的后门检测机制。

综上所述, 单参数后门攻击仅通过小幅度地调整偏置值, 便可显著提升后门攻击效果。因此, 该方法在提升攻击有效性的同时, 保证了攻击隐蔽性, 从而在所有对比方法中表现出了最佳的攻击性能, 拥有最高的 AP 值。

表 5　与 TBT 和 HPT 的攻击性能对比

Table 5　Comparison of attack performance with TBT and HPT

数据集	模型	攻击方法	BA (%)		ASR (%)	N_b	AP
			干净模型	后门模型			
CIFAR-10	ResNet-18*	TBT	94.8	87.5	90.2	540	0.15
		HPT		94.7	94.1	12	7.83
		Ours		93.9	100.0	7	**14.16**
	VGG-16*	TBT	93.2	80.7	83.2	601	0.12
		HPT		93.1	91.1	6	15.17
		Ours		92.9	99.6	5	**19.86**
SVHN	VGG-16*	TBT	96.3	67.9	60.1	576	0.06
		HPT		94.2	78.0	26	2.92
		Ours		96.0	99.5	7	**14.17**
ImageNet	ResNet-18*	TBT	69.5	68.8	100.0	611	0.16
		HPT		68.6	95.2	10	9.43
		Ours		68.6	99.6	10	**9.87**

表 6　与 ProFlip 的攻击性能对比

Table 6　Comparison of attack performance with ProFlip

数据集	模型	攻击方法	BA (%)		ASR (%)	N_b	AP
			干净模型	后门模型			
CIFAR-10	ResNet-18*	ProFlip	93.1	90.3	97.9	12	7.93
		Ours	94.8	94.5	99.7	5	**19.88**
	VGG-16*	ProFlip	89.7	88.1	94.8	16	5.83
		Ours	93.2	92.9	99.4	7	**14.16**
SVHN	VGG-16*	ProFlip	98.6	95.3	94.5	20	4.56
		Ours	96.3	96.1	99.2	7	**14.14**
ImageNet	ResNet-18*	ProFlip	69.0	67.6	94.3	15	6.19
		Ours	69.5	68.9	99.1	9	**10.94**

4.3　攻击鲁棒性评估

在部署阶段的攻击通常出现在模型已经通过后门防御检测或正在运行中的情况。在这种情况下, 对受攻击的模型进行微调可能会中断正常用户的使用, 并影响服务的稳定性和可靠性。因此, 通常不对被攻击的模型进行任何微调处理。此外, 即便防御者试图通过修改最后一层网络的偏置参数来进行微调, 攻击者仍然可以通过 Rowhammer 攻击持续地翻转云服务中目标权重的 bits, 从而植入后门。

我们进一步假设防御者已经意识到模型遭到攻击, 并采用现有的后门防御方法尝试移除后门。由于单参数后门攻击无需对模型进行训练, 可以天然抵御在模型训练时采取的各种后门防御措施, 因此, 本节只选取在训练后对模型进行后门检测或移除的防御措施进行实验评估。参考对比方法文献[4-7]的防御实验设置, 本节选取了两类经典的后门防御方法来进行攻击的鲁棒性评估, 包括基于逆向触发器的后门检测方法——Neural Cleanse[14]和基于神经元剪枝的后门移除方法——Fine-Pruning[17]和 Adversarial Neuron Pruning[18]。此外, 本节还选取了一类新颖且有效的后门防御方法, 即基于知识蒸馏的后门移除方法——Neural Attention Distillation[19]和 Attention Relation Graph Distillation[20]来评估单参数后门攻击的鲁棒性。在本节实验中, 对 CIFAR-10 使用 ResNet-18 模型, 对 SVHN 使用 8 位量化的 VGG-16 模型, 对 ImageNet 使用 8 位量化的 ResNet-18 模型, 所有实验的目标类别均设置为 0。

(1) 基于逆向触发器的后门检测方法

Neural Cleanse(NC)[14]是一种经典且有效的后门检测方法, 它对待测模型的每个类别都逆向一个可能的最优触发器, 再检测其中是否存在异常小的最优触发器。NC 使用异常指数来量化这种异常, 如果有任何类别的异常指数大于 2, 则认为该模型存在后门。如图 6 所示, 对于 NC 防御方法, 后门模型所有类别的最大异常指数在所有数据集上都小于 2, 代表单参数后门攻击在模型中植入的后门未被检测到。

这是由于单参数后门攻击只轻微增加了目标类别输出神经元的偏置值, 这种轻微调整引入的“捷径”并不明显, NC 难以检测。

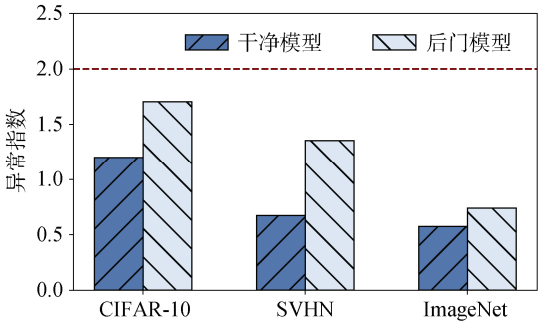

图 6　对 Neural Cleanse 的鲁棒性

Figure 6　Resistance to Neural Cleanse

(2) 基于神经元剪枝的后门防御方法

Fine-Pruning(FP)[17]和 Adversarial Neuron Pruning(ANP)[18]都是基于神经元剪枝的后门移除方法, 它们都假设后门模型中存在与后门紧密相关的受感染神经元, 并通过剪除这些受感染神经元来移除模型中的后门。FP 计算神经元对干净数据的响应, 并剪除其中的休眠神经元; ANP 计算神经元对对抗性扰动的敏感度, 并剪除其中的敏感神经元。由于 FP 和 ANP 防御只剪枝卷积层中的神经元, 而不改变模型参数, 因此对于只调整输出层神经元偏置参数的单参数后门攻击来说, 无法有效移除后门, 这一点可由图 7 证明。实验结果表明, 在所有数据集上, 随着 FP 和 ANP 修剪神经元的比例或阈值不断增加, 后门模型的 BA 不断下降, 而 ASR 却不受影响, 说明 FP 和 ANP 无法有效移除单参数后门攻击植入的后门。

(3) 基于知识蒸馏的后门移除方法

Neural Attention Distillation(NAD)[19]和 Attention Relation Graph Distillation(ARGD)[20]都是基于知识蒸馏的后门移除方法, 它们都利用在部分干净数据上微调得到的干净模型作为教师模型, 再基于知识蒸馏来调整后门模型的注意力。具体而言, 它们将模型每一个残差块的激活输出作为知识在教师模型和学生(后门)模型之间蒸馏。其中, NAD 聚焦于同一顺序的注意力特征, 而 ARGD 还考虑了不同顺序的注意力特征之间的相关性。在单参数后门攻击中, 对模型的修改只涉及输出层神经元偏置参数的调整, 而未污染模型的中间层特征。这意味着教师模型和后门模型在中间层上的特征响应对于干净数据来说是一致的。因此, 如果要通过调整中间层特征来移除模型中的后门, 将不可避免地对模型的 BA 产生负面影响。如表 7 所示, 尽管 NAD 和 ARGD 可以将 ASR 降至近乎 0%, 但模型的 BA 也降低到了 25% 以下, 严重损害了模型的可用性。因此, NAD 和 ARGD 无法有效移除单参数后门攻击植入的后门。

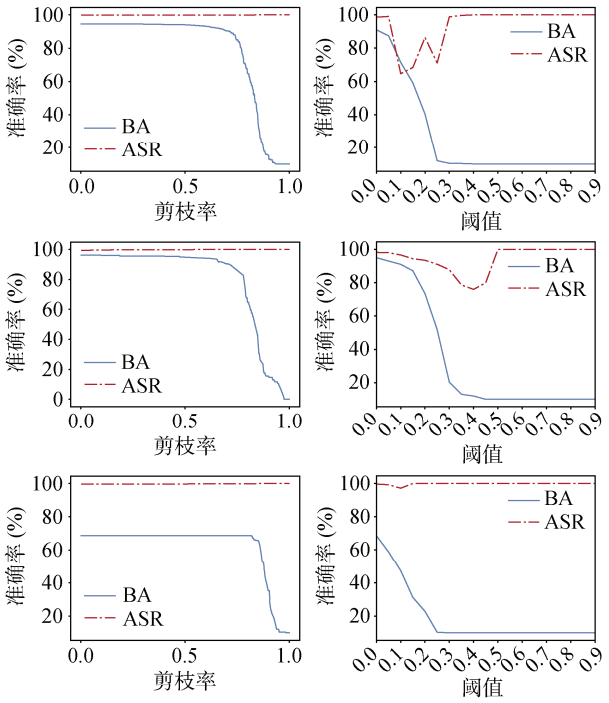

图 7　对 Fine-Pruning 和 Adversarial Neuron Pruning 的鲁棒性

Figure 7　Resistance to Fine-Pruning and Adversarial Neuron Pruning

表 7　对 NAD 和 ARGD 的鲁棒性表标题

Table 7　Resistance to NAD and ARGD

防御方法	CIFAR-10		SVHN		ImageNet	
	BA (%)	ASR (%)	BA (%)	ASR (%)	BA (%)	ASR (%)
无防御	94.65	99.85	96.16	99.50	68.63	99.62
NAD	24.49	3.06	19.58	0.40	0.10	0.00
ARGD	11.33	2.55	8.34	0.15	0.00	0.00

5　结束语

本文提出了一种基于模型操作的单参数后门攻击方法, 该方法通过小幅度增加目标类别对应输出神经元的偏置值, 巧妙地引导模型对目标类别的预测产生倾向性, 进而成功植入后门。同时, 为了保证攻击的有效性, 本文提出了一种基于 PGD 算法的触发器生成方案, 该方案通过最大化模型的预测熵来优化触发器, 确保当该触发器被应用时, 模型将以很高的概率输出指定目标类别。在不同的数据集和网络结构上的实验结果证明, 单参数后门攻击方法仅需修改 1 个模型参数, 就可实现超过 99%的攻击成

功率, 且模型的良性精度下降不超过 1%。与现有的基于模型操作的后门攻击方法相比, 该方法在攻击有效性以及隐蔽性方面均表现出显著优势, 拥有更好的攻击性能。

参考文献

[1] Gu T Y, Dolan-Gavitt B, Garg S. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain[EB/OL]. 2017: arXiv: 1708.06733. https://arxiv.org/abs/1708.06733.

[2] Chen X Y, Liu C, Li B, et al. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning[EB/OL]. 2017: arXiv: 1712.05526. https://arxiv.org/abs/1712.05526.

[3] Tan Q Y, Zeng Y M, Han Y, et al. Survey on backdoor attacks targeted on neural network[J]. *Chinese Journal of Network and Information Security*, 2021, 7(3): 46-58.
(谭清尹, 曾颖明, 韩叶, 等. 神经网络后门攻击研究[J]. *网络与信息安全学报*, 2021, 7(3): 46-58.)

[4] Hong S, Carlini N, Kurakin A. Handcrafted Backdoors in Deep Neural Networks[C]. *The 36th International Conference on Neural Information Processing Systems*, 2022: 8068-8080.

[5] Rakin A S, He Z Z, Fan D L. TBT: Targeted Neural Network Attack with Bit Trojan[C]. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 13195-13204.

[6] Chen H L, Fu C, Zhao J S, et al. ProFlip: Targeted Trojan Attack with Progressive Bit Flips[C]. *2021 IEEE/CVF International Conference on Computer Vision*, 2022: 7698-7707.

[7] Bai J W, Gao K F, Gong D H, et al. Hardly Perceptible Trojan Attack Against Neural Networks withBit Flips[C]. *Computer Vision – ECCV 2022*, 2022: 104-121.

[8] Qi X Y, Xie T H, Pan R Z, et al. Towards Practical Deployment-Stage Backdoor Attack on Deep Neural Networks[C]. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022: 13337-13347.

[9] Clements J, Lao Y J. Backdoor Attacks on Neural Network Operations[C]. *2018 IEEE Global Conference on Signal and Information Processing*, 2019: 1154-1158.

[10] Zou M H, Shi Y, Wang C L, et al. PoTrojan: Powerful Neural-Level Trojan Designs in Deep Learning Models[EB/OL]. 2018: arXiv: 1802.03043. https://arxiv.org/abs/1802.03043.

[11] Dumford J, Scheirer W. Backdooring Convolutional Neural Networks *via* Targeted Weight Perturbations[C]. *2020 IEEE International Joint Conference on Biometrics*, 2021: 1-9.

[12] Tol M C, Islam S, Adiletta A J, et al. Don't Knock! Rowhammer at the Backdoor of DNN Models[C]. *2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2023: 109-122.

[13] Li S, Wang X, Xue M, et al. Yes, One-Bit-Flip Matters! Universal DNN Model Inference Depletion with Runtime Code Fault Injection[C]. *Proceedings of the 33th USENIX security symposium*, 2024: 1-16.

[14] Wang B L, Yao Y S, Shan S, et al. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks[C]. *2019 IEEE Symposium on Security and Privacy*, 2019: 707-723.

[15] Liu Y Q, Lee W C, Tao G H, et al. ABS: Scanning Neural Networks for Back-Doors by Artificial Brain Stimulation[C]. *The 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019: 1265-1282.

[16] Zeng Y, Chen S, Park W, et al. Adversarial Unlearning of Backdoors *via* Implicit Hypergradient[EB/OL]. 2021: arXiv: 2110.03735. https://arxiv.org/abs/2110.03735.

[17] Liu K, Dolan-Gavitt B, Garg S. Fine-Pruning: Defending Against Backdooring Attacks on Deep Neural Networks[C]. *Research in Attacks*, *Intrusions*, *and Defenses*, 2018: 273-294.

[18] Wu D X, Wang Y S. Adversarial Neuron Pruning Purifies Backdoored Deep Models[EB/OL]. 2021: arXiv: 2110.14430. https://arxiv.org/abs/2110.14430.

[19] Li Y G, Lyu X X, Koren N, et al. Neural Attention Distillation: Erasing Backdoor Triggers from Deep Neural Networks[EB/OL]. 2021: arXiv: 2101.05930. https://arxiv.org/abs/2101.05930.

[20] Xia J, Wang T, Ding J P, et al. Eliminating Backdoor Triggers for Deep Neural Networks Using Attention Relation Graph Distillation[C]. *The Thirty-First International Joint Conference on Artificial Intelligence*, 2022: 1481-1487.

[21] Kim Y, Daly R, Kim J, et al. Flipping Bits in Memory without Accessing Them: An Experimental Study of DRAM Disturbance Errors[C]. *2014 ACM/IEEE 41st International Symposium on Computer Architecture*, 2014: 361-372.

[22] van der Veen V, Fratantonio Y, Lindorfer M, et al. Drammer: Deterministic Rowhammer Attacks on Mobile Platforms[C]. *The 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016: 1675-1689.

[23] Yao F, Rakin A S, Fan D. DeepHammer: Depleting the intelligence of deep neural networks through targeted chain of bit flips[C]. *Proceedings of the 29th USENIX Security Symposium*, 2020: 1463-1480.

[24] Venceslai V, Marchisio A, Alouani I, et al. NeuroAttack: Undermining Spiking Neural Networks Security through Externally Triggered Bit-Flips[C]. *2020 International Joint Conference on Neural Networks*, 2020: 1-8.

[25] Rakin A S, He Z Z, Li J T, et al. T-BFA: Targeted Bit-Flip Adversarial Weight Attack[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, 44(11): 7928-7939.

[26] Dong J S, Qiu H, Li Y M, et al. One-Bit Flip Is all You Need: When Bit-Flip Attack Meets Model Training[C]. *2023 IEEE/CVF International Conference on Computer Vision*, 2024: 4665-4675.

[27] Bai J W, Wu B Y, Zhang Y, et al. Targeted Attack Against Deep Neural Networks *via* Flipping Limited Weight Bits[EB/OL]. 2021: arXiv: 2102.10496. https://arxiv.org/abs/2102.10496.

[28] Bai J W, Wu B Y, Li Z F, et al. Versatile Weight Attack *via* Flipping Limited Bits[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023, 45(11): 13653-13665.

[29] Jia Y Q, Shelhamer E, Donahue J, et al. Caffe: Convolutional Architecture for Fast Feature Embedding[C]. *The 22nd ACM International Conference on Multimedia*, 2014: 675-678.

[30] Cireşan D C, Meier U, Masci J, et al. Flexible, High Performance Convolutional Neural Networks for Image Classification[C]. *The Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, 2011: 1237-1242.

[31] Kendall A, Gal Y. What Uncertainties Do We Need in Bayesian

Deep Learning for Computer Vision? [C]. *The 31st International Conference on Neural Information Processing Systems*, 2017: 5580-5590.

[32] Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge[J]. *International Journal of Computer Vision*, 2015, 115(3): 211-252.

[33] He K M, Zhang X Y, Ren S Q, et al. Deep Residual Learning for Image Recognition[C]. *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016: 770-778.

[34] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[EB/OL]. 2014: arXiv: 1409.1556. https://arxiv.org/abs/1409.1556.

段秋宇 于 2021 年在中国农业大学计算机科学与技术专业获得学士学位。现在哈尔滨工业大学(深圳)计算机科学与技术专业攻读硕士学位。研究领域为人工智能安全。Email: 1124307234@qq.com

候琳珊 于 2021 年在哈尔滨工业大学获得计算机科学与技术硕士学位。现在哈尔滨工业大学(深圳)攻读博士学位。研究领域为可信人工智能。Email: 21b951003@stu.hit.edu.cn

花忠云 (CCF 会员)于 2016 年在澳门大学大学软件工程专业获得博士学位。现任哈尔滨工业大学(深圳)副教授。研究领域为云计算安全、多媒体信息安全、非线性系统理论。Email: huazhongyun@hit.edu.cn

廖清 (CCF 高级会员)于 2016 年在香港科技大学计算机科学与技术专业获得博士学位。现任哈尔滨工业大学(深圳)教授。研究领域为信息安全、人工智能。Email: liaoqing@hit.edu.cn

张玉书 (CCF会员)于2015年在于重庆大学获得博士学位。南京航空航天大学教授,博士生导师。研究领域为多媒体安全、区块链等。Email: yushu@nuaa.edu.cn

张瑜 于 2016 年在中国香港城市大学电子工程专业获得博士学位。现任澳大利亚昆士兰州南港格里菲斯大学信息与通信技术学院高级讲师。研究领域为可信人工智能和应用密码学。Email: leo.zhang@griffith.edu.au