

基于侧信道与量化推理缺陷的模型逆向攻击

李景海¹, 唐明^{1,2}, 黄诚轩¹

(1. 武汉大学国家网络安全学院空天信息安全与可信计算教育部重点实验室, 湖北 武汉 430072;

2. 密码科学技术国家重点实验室, 北京 100878)

摘要: 模型逆向攻击旨在恢复部署在推理终端的神经网络模型的结构和权重值, 是 AI 安全中的基础问题, 为对抗样本等高阶攻击提供数据支撑。提出了一种名为 Cluster-based SCA 的新型模型权重逆向方法, 该方法不要求攻击者构造泄露模型。Cluster-based SCA 方法以量化推理中存在的安全隐患为出发点, 深入分析了量化推理过程, 发现在量化推理中存在的输出序列分类不等价现象可以判断猜测权重的正确与否。Cluster-based SCA 将采集到的模型运行时产生的侧信道信息按照假设权重产生的中间值进行分类, 以分类后的平均离散系数 σ 为评判标准, 取 σ 最小时的权重为逆向权重。在仿真实验上验证了 Cluster-based SCA 方法的有效性, 实验使用汉明重模型来模拟 AI 芯片的泄露模型, 对于目标 CNN, Cluster-based SCA 方法以 52.66% 的 TOP2 恢复率恢复了其第一层卷积层所有卷积核权重, 对于取值位于显著区的权重, TOP2 的恢复率均达到了 100%。

关键词: AI 安全; 模型逆向攻击; 量化推理缺陷; 侧信道分析; Cluster-based SCA

中图分类号: TP309.2

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2021038

Using side-channel and quantization vulnerability to recover DNN weights

LI Jinghai¹, TANG Ming^{1,2}, HUANG Chengxuan¹

1. Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education,

School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China

2. State Key Laboratory of Cryptology, Beijing 100878, China

Abstract: Model extraction attack focuses on reverse engineering architecture and weights of DNN model deployed in edge. Model extraction attack is a basic security problem in AI security, it underlies advanced attacks as data provider, such as adversarial sample and data poisoning. A novel method named Cluster-based SCA was proposed,

收稿日期: 2020-07-23; 修回日期: 2021-01-20

通信作者: 唐明, m.tang@126.com

基金项目: 国家自然科学基金 (61972295); 武汉市科技项目应用基础前沿专项 (2019010701011407)

Foundation Items: The National Natural Science Foundation of China (61972295), The Frontier Applied Basic Research Project of Science and Technology Department of Wuhan (2019010701011407)

论文引用格式: 李景海, 唐明, 黄诚轩. 基于侧信道与量化推理缺陷的模型逆向攻击[J]. 网络与信息安全学报, 2021, 7(4): 53-67.

LI J H, TANG M, HUANG C X. Using side-channel and quantization vulnerability to recover DNN weights[J]. Chinese Journal of Network and Information Security, 2021, 7(4): 53-67.

this method did not need leakage model. Cluster-based SCA was based on vulnerability of quantized inference. There exist a phenomenon in multiplication operation in quantized inference, which the output of different weights were not equivalent in respect of classification. It can be used to distinguish different weights. The proposed method computed output activations of each DNN layer with guessing weight. Then acquired side channel signal were classified into different class, the taxonomy was corresponding output activations' value. Average dispersion of all classes $\bar{\sigma}$ was used to decide whether guess was right. The effectiveness of Cluster-based SCA method was verified by simulation experiment and HW model was used as target leakage model. For all weights from first convolution layer of target CNN model, TOP2 recovery rate was 52.66%. And for large weights in significant interval, TOP2 recover rate was 100%.

Keywords: AI security, model extraction attack, quantization vulnerability, side-channel analysis, Cluster-based SCA

1 引言

随着大数据时代的到来以及计算机计算速度的显著提升, 人工智能迎来快速发展, 广泛应用于图像识别、语音识别、自然语言处理等任务。深度学习是应用最为广泛的人工智能技术之一。深度学习以深度神经网络 (DNN, deep neural network) 模型为核心, 以数据为驱动, 通过训练出具有良好表达能力的 DNN 模型对问题进行建模并解决。深度学习在人类社会各个领域都发挥了巨大作用, 但同样存在巨大的安全风险。现阶段深度学习的安全研究集中于边缘端的模型推理过程, 包括对抗样本^[1-2]、数据投毒^[3-4]、成员推断^[5]、模型逆向^[6-8]等方面。本文以模型逆向攻击为研究对象, 模型逆向攻击是指攻击者通过各种方式提取出 DNN 模型的有关信息, 包括模型的结构、模型的权重。模型结构与权重构成了整个神经网络, 一旦攻击者通过模型逆向攻击恢复了网络模型的结构与权重, 模型的推理过程就完全暴露在攻击者的视野中, 攻击者可在短时间内完成如对抗样本等高阶攻击, 对深度学习系统造成巨大危害。

传统的模型逆向攻击使用 Query-based 方法^[6-7,9], 通过向模型输入数据并获得输出从而构造影子数据集, 将影子数据集作为训练数据集训练一个替代模型来拟合原始模型进行模型的逆向。近两年, 针对模型逆向攻击, 出现了基于侧信道的新形式的攻击方法^[8,10-12]。基于侧信道的攻击以 DNN 的具体实现为攻击点, 通过分析计算设备在运行 DNN 时产生的侧信道信息 (如功耗等) 来提取模型结构和权重等隐私信息。本文以 AI 芯片来指代实现模型推理的实际计算设备。侧

信道攻击主要用于密码学中各加密算法的密钥恢复中, 对称加密算法和神经网络算法在结构与计算过程上具有相似性, 在算法结构上, 对称密码算法由轮函数串联而成, DNN 则由多层网络层组成, 二者都存在线性部件和非线性部件; 在运行模式上, 二者后一层/轮的输入取决于前一层/轮的输出, 存在显式的中间值结果。基于这种相似性, 侧信道攻击开始用于 DNN 的模型逆向攻击。

Hua 等^[10]通过测量模型运行过程中产生的内存访问模式信息, 成功地逆向了基于 FPGA 实现的 CNN 模型的结构。Hu 等^[12]将视线转移到通用的 GPU 平台, 测量了在 GPU 上运行产生的 kernel events 和对应的内存访问模式, 将结构逆向问题转化为一个序列识别的问题, 采用 ASR 中常用的 LSTM+CTC 的解决方案逆向了基于 GPU 实现的 DNN 结构。Batina 等^[8]通过测量模型运行过程中产生的电磁信号, 成功逆向了 ARM Cortex-M3 平台上实现的 DNN 的层数、每层神经元数量、激活函数类别等信息。Wei 等^[11]则通过测量功耗逆向了 CNN 的输入。在已有研究中, 以内存状态为数据来源的方法存在数据采集困难的问题。内存状态信息的获取要求攻击者对 AI 芯片有较高的访问权限或者通过硬件木马来收集内存信息, 这对于攻击者而言是很高的能力要求。而基于物理侧信道的方法存在如下问题: 其使用 CPA 技术来逆向模型权重, 这种方法要求攻击者构造足够准确的泄露模型完成假设中间值到侧信道信息的映射, 这需要攻击者掌握设备的底层硬件电路设计, 而 AI 芯片的设计细节是生产商的重要知识产权, 出于对知识产权的保护, 加速卡生产商

不会公布加速卡的内部细节，因此攻击者无法有效构造准确的泄露模型。为了解决这一问题，本文提出了一种新的攻击方法，该方法无须攻击者构造泄露模型。

本文方法的出发点来自当 AI 芯片使用了量化推理技术后，模型逆向攻击变得更有可能实现。量化推理^[13-15]是 AI 芯片为了减小模型存储开销和加速模型的推理广泛使用的技术。量化推理中，FP32 的模型权重与激活值使用更低精度的 FP16 或 INT8 类型表示，来减小存储开销与提高计算速度，AI 芯片会设计专门的部件来适配低精度情况下的模型推理，如 Xilinx 的 DPU、Google 的 TPU^[16]等。量化推理存在潜在的安全威胁，因为量化技术使表示单个权重所需要的比特数大大降低，攻击者可以更加容易地枚举所有可能权重并验证是否正确，从而恢复出正确的权重；另外，量化推理导致了一种输出序列分类不等价的现象，这启发笔者使用猜测权重的平均离散系数为指标来判断猜测权重正确与否，这种方法被称为 cluster-based side-channel analysis，简称 cluster-based SCA。

本文的主要工作如下。

(1) 对量化推理的计算过程进行了详细的分析，发现在量化乘法 $q_3 = \text{Clamp}(\text{LBound}, \text{UBound}, Mq_1q_2)$ 中，由于精度和表示范围限制，不同的操作数 q_1 对于相同的输入序列 $Q_2 = \{q_2^{(1)} \cdots q_2^{(N)}\}$ 产生的输出序列 $Q_3 = \{q_3^{(1)} \cdots q_3^{(N)}\}$ 与 $Q'_3 = \{q_3'^{(1)} \cdots q_3'^{(N)}\}$ 存在不同的分类情况，称之为分类不等价。

(2) 提出了新的基于侧信道的攻击方法 Cluster-based SCA，该方法不需要攻击者构造特定的泄露模型。利用 Cluster-based SCA 构造了模型权重逆向的攻击，并进行了仿真实验来验证方法的有效性。

(3) 分析了 Cluster-based SCA 方法的性质。

2 背景知识

2.1 侧信道分析

侧信道分析 (SCA, side-channel analysis) 利用算法在运行时产生的侧信道信息，如功耗、时间等来推测算法的私密信息。侧信道分析的研究

起源于密码学^[17]，通过分析加密设备在加密过程产生的侧信道信息来推测密钥，包括 SPA (simple power analysis)^[18]、DPA (differential power analysis)^[19]、CPA (correlation power analysis)^[20] 等技术，相比基于统计分析的密钥恢复攻击，SCA 的优势在于其采取了分而治之的思想，通过逐个恢复密钥中的较小部分，通常为单个字节，从而减少了密钥恢复的复杂度。侧信道攻击中的几种基本方法如下。

SPA: 简单能量分析是 SCA 的基本形式，SPA 是从密码芯片运算的功耗波形上能够直接找出密钥信息的攻击方法。在芯片中，密钥位为 0 时运算的功耗和密钥位为 1 时运算的功耗存在差异，通过这样的差异分析出密钥位的值，按照一定规律组合得到完整的密钥。SPA 常用于攻击 RSA 算法，通过区分平方运算和乘法运算，逐步恢复出密钥。

DPA: 差分能量分析是 SCA 的高级形式，它是以能量消耗曲线为基础，再以统计方式推演密钥的攻击技术。在攻击中，攻击者首先对密钥进行假设，并据此计算出中间值。根据中间值取值将采集到的功耗曲线分为两个或多个集合，并计算这些集合的均值的差来验证假设是否正确从而恢复出密钥。

CPA: 相关能量分析是 DPA 的一种变体，其使用 Pearson 相关系数作为统计指标进行猜测正确与否的确认。文献[8]中使用 CPA 进行神经网络模型参数的恢复。CPA 恢复密钥的主要步骤如下。

(1) 选择所执行算法的某个中间值。这个中间值必须是一个函数 $f(d, k)$ ，其中， d 是已知的非常量数据， k 是密钥的一小部分。在大多数攻击场景中， d 不是明文就是密文。

(2) 测量能量消耗。测量密码设备在加密或解密 D 个不同数据分组时的能量消耗， D 组数据记为 $d = (d_1, \dots, d_D)$ 。对每一个 d_i 执行期间的能量消耗进行记录，记作能量迹 $t_i = (t_{i,1}, \dots, t_{i,T})$ ，从而生成 $D \times T$ 的能量迹矩阵 T 。 T 的第 i 行为对应数据分组 d_i 的能量迹。

(3) 计算假设中间值。将所有可能的 k 值记为向量 $k = (k_1, \dots, k_K)$ 。计算假设中间值矩阵 V ， $V_{i,j} = f(d_i, k_j)$ 。

(4) 计算假设能量消耗矩阵 H 。 $H_{i,j} = L(V_{i,j})$, L 为泄露模型, 体现了密码设备的泄露特征。

5) 对 H 的每一列 H_i 和矩阵 T 的每一列 T_j 计算 Pearson 相关系数 $r_{i,j}$, 取使 $r_{i,j}$ 最大时的 i 对应 k_i 为恢复密钥。

2.2 DNN 中的原子操作

本文使用 DNN 表示仅有全连接层的网络以及带有卷积层的 CNN。DNN 由多种类型网络层串联而成, 本文不考虑 Resnet 中的 shortcut 和 GoogLeNet 中的 Inception 模块。常见的网络层类型有卷积层、池化层、全连接层、BN 层等。其中最主要的两种类型为卷积层和全连接层, Krizhevsky 在文献[21]中提出了两点观察结论: 卷积层占据了 90%~95% 的计算时间和 5% 权重数量, 且有较大的值; 全连接层占据了 5%~10% 的计算时间和 95% 的权重数量, 且值较小。这两层的运算都可以转换为通用矩阵乘法运算。卷积层内每个神经元都与前一层中位置接近的区域的多个神经元相连, 区域的大小取决于卷积核的大小, 被称为“感受野”。卷积核在工作时, 在“感受野”内对输入特征做矩阵元素乘法求和并叠加偏差量, 送入激活函数后生成特征图作为下一层的输入。图 1 展示了如何将卷积层的计算转换为通用矩阵乘法计算。

如图 1 所示, 对于输入特征图为 $3 \times 3 \times 3$, 卷积核为 $2 \times 2 \times 2$ 的情况, 通过将每一通道在单次卷积操作中涉及的操作数按行展开并将卷积核对应的元素按列展开, 即可将原始的卷积操作展开成矩阵乘法。原始卷积操作等价于式(1)。

$$\mathbf{OF}_{4 \times 2} = \mathbf{IF}_{4 \times 12} \mathbf{W}_{12 \times 2} \quad (1)$$

其中, $\mathbf{IF}$ 为输入特征矩阵, $\mathbf{W}$ 为权重矩阵, $\mathbf{OF}$ 为输出特征矩阵。原始的卷积核元素被转换为矩阵乘法的操作数。

全连接层可以看成卷积核尺寸与输入特征图相同的卷积层, 其底层操作仍然可以转换为通用矩阵乘法。而通用矩阵乘法由标量乘法组成, 对于卷积层或全连接层的输出值, 其可以表示为式(2)。

$$y(h, w, n) = \sum_{n=0}^N \sum_{l=0}^L \sum_{c=0}^C w(k, l, c; n) x(k, l, c) + b \quad (2)$$

其中, h 、 w 、 n 分别为输出特征图中激活值的行索引、列索引和输出通道索引。 n 、 l 、 c 分别为卷积核和输入特征图感受野区域的行、列、通道索引。 b 为偏置值。因此, 在神经网络中, 原子操作作为标量乘法 $r_3 = r_1 r_2$ 。通过分析 DNN 中的原子操作, 将攻击目标缩小到恢复标量乘法操作数上, 通过逐个恢复在 DNN 乘法操作中充当权重的操作数, 从而恢复整个网络的权重集合。

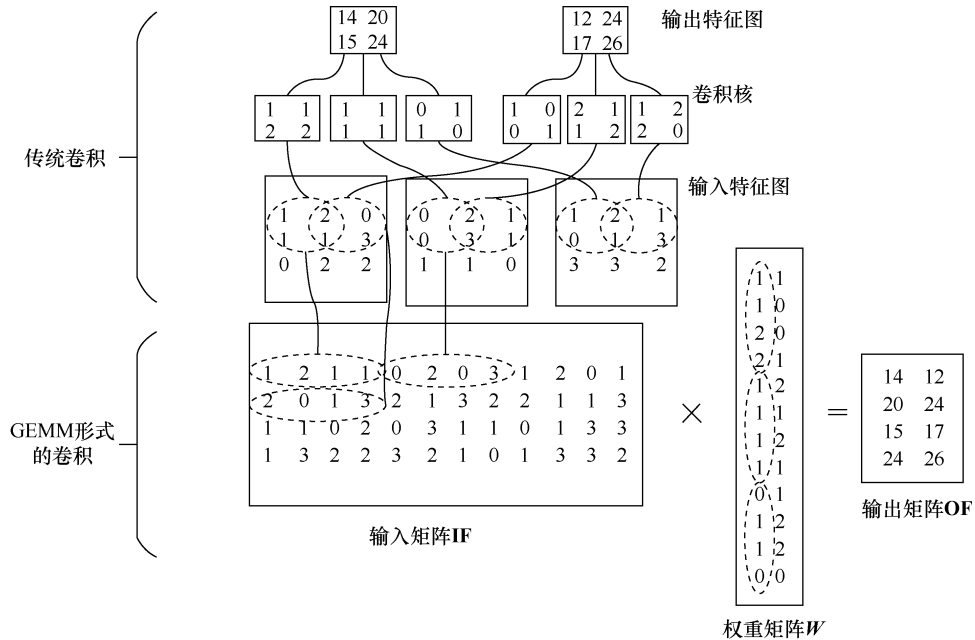


图 1 卷积计算与 GEMM 之间的转换

Figure 1 Transformation between convolution operation and GEMM

2.3 量化推理

神经网络的发展说明越大越深的模型往往具有越高的精度，但越大的模型对存储资源和计算资源的要求也越高。以 Resnet152 模型为例，其模型大小为 230 MB，模型运行时产生的浮点操作为 11GFLOPs^[22]，这会引起设备存储空间不足、设备运行功耗过高、计算时延高等问题。尤其是在移动终端和嵌入式设备的环境下，这样的问题更加显著，因此需要对模型进行优化。量化推理(quantized inference)是算法层模型优化的常用方法，其首先将神经网络模型中 FP32 数据使用更低精度的数据类型表示(如 FP16、INT8)，被称为量化，然后使用量化后的数据完成 DNN 中的各种操作，被称为量化推理。在没有特别指明的情况下，使用 r 来表示原始区间的数据，用 q 表示量化后区间的数据。原始区间指的是原始数据类型所能表示的数据取值区间，量化后区间指的是量化后数据类型所能表示的数据取值区间。

量化完成从原始数据类型到量化后数据类型的转化，使用式(3)计算。

$$q = \text{Clamp}\left(\text{LBound}, \text{UBound}, \text{round}\left(Z + \frac{r}{S}\right)\right) = Q(r)$$

$$\text{Clamp}(a, b, x) = \min(\max(x, a), b)$$
(3)

其中，Clamp 为截断函数，round 为四舍五入取整函数。

反量化完成量化后数据类型到原始数据类型的转化，使用式(4)计算。

$$r = S(q - Z) = Q^{-1}(q)$$
(4)

其中， Q 为量化函数， Q^{-1} 为反量化函数，LBound、UBound 分别为量化后区间的上界和下界， Z 为量化后的零点， S 为缩放系数，也被称为量化参数，用于确定原始区间到量化区间的缩放比例，可用基于 KL 散度或 EMA^[14]的方法计算。

量化推理旨在使用量化后的数据进行模型的推理，模型推理中的原子操作、标量乘法 $r_3 = r_1 r_2$ 可以使用量化后数据表示为 $r_3 = Q^{-1}(q_1)Q^{-1}(q_2) = S_1(q_1 - Z_1)S_2(q_2 - Z_2)$ 。

因此可以推出 $q_3 = \text{Clamp}(\text{LBound}, \text{UBound}, \text{round}(Z_3 + M(q_1 - Z_1)(q_2 - Z_2)))$ ， $M = \frac{S_1 S_2}{S_3}$ 。

目前，已有大量的论文对量化推理进行了研究，根据量化后的数据类型可以分为 FP16 量化、INT8 量化^[14]，以及更加激进的 XNORNet^[15]等，其量化精度高，模型准确率损失小，但量化效果差，而过低精度的量化在显著提升量化效果时带来了更大的模型准确度损失。通过权衡，现阶段最成熟的量化方案为 INT8 量化，如 Google 的 Tensorlite、Nvidia 的 TensorRT、腾讯的 NCNN。

Google 发布的白皮书^[23]对现有的 INT8 量化方案做了全面的介绍，其将量化方案分为 Uniform Affine Quantizer、Uniform Symmetric Quantizer 和 Stochastic Quantizer。本文第 3 节以 Uniform Symmetric Quantizer 为例，分析量化操作中存在的安全风险，这是 Cluster-based SCA 方法的理论基础。

2.4 AI 芯片

本文以 AI 芯片来指代实现模型推理的实际计算设备。AI 芯片是逆向目标 DNN 模型的运行载体，其在执行模型计算任务时产生意外的侧信道信息泄露，从而泄露与模型有关的隐私信息。AI 芯片采用专卡专用的模式，使用特定架构专注于模型计算任务。专卡专用的模式减少了加速卡内部与模型计算无关的任务的执行，提高了运行时产生的侧信道信息与神经网络模型的相关性，这种更为纯净的侧信道信息提升了攻击者从采集到的侧信道信息中恢复模型结构权重的可能性。AI 芯片根据技术路线可分为 GPU、FPGA、ASIC。AI 芯片的架构与底层实现一般不会公开，尤其是 ASIC 实现的 AI 芯片，其对于用户为完全的黑盒状态。AI 芯片以加速模型推理为目的，而量化推理作为一种常见的算法层的模型优化与推理加速方案，具有实现简单、效果明显的优点。因此，AI 芯片在底层架构中针对量化推理进行了适配，加入了定制的低精度运算部件。目前，全球芯片企业都在积极地进行 AI 芯片的布局。Google、Nvidia、Xilinx 等企业发布了自己的 AI 芯片产品。表 1 列举了现阶段部分 AI 芯片产品及其对量化推理的支持。

表 1 主流 AI 芯片对比
Table 1 Comparison of AI Chips

技术路线	制造商	产品	量化支持
GPU	Nvidia	Tesla 计算卡等	FP16,INT16/8
FPGA	Intel	Arria 系列	—
	Xilinx	DPU	INT8
ASIC	Google	TPU	FP16,INT16/8
	Intel	NNP	FP16,INT8/4
	寒武纪	思元系列	FT16,INT16/8/4
	比特大陆	BM168X 系列	FP16,INT8
	华为	Ascend 系列	FP16,INT8
	IBM	TrueNorth	类脑芯片

3 量化推理中的安全风险

量化推理虽然带来了更快的推理速度和更低的运行功耗，但同时带来了安全风险。由于低精度数据的表达能力有限，其在运算过程中会损失许多有用的数据。本节以 Uniform Symmetric Quantizer^[23]下的 INT8 量化为例说明量化推理中存在的安全风险，对于其他量化策略，该风险依然存在。

在 Uniform Symmetric Quantizer 中 $Z=0$ ，即量化后区间的上下界关于 0 对称。因此，其量化过程可表示为 $q = \text{Clamp}\left(-127, 127, \text{round}\left(\frac{r}{S}\right)\right)$ ，

$$S = \frac{\max(q) - \min(q)}{254}.$$

其中， $\max(q)$ 和 $\min(q)$ 分别表示在实际计算时， q 的最大值和最小值。对应的反量化操作为 $r = Sq$ 。相应地，此方案量化推理中的量化乘法表示为式(5)。

$$\begin{cases} q_3 = \text{Clamp}(-127, 127, \text{round}(Mq_1q_2)) \\ M = \frac{S_1S_2}{S_3} \\ S_i = \frac{\max(q_i) - \min(q_i)}{254}, i = 1, 2, 3 \end{cases} \quad (5)$$

3.1 分类不等价

这里首先给出分类不等价的概念，方便后续说明。

定义 1 序列 $A = \{a^{(1)}, a^{(2)} \dots a^{(N)}\}$ 和 $B = \{b^{(1)},$

$b^{(2)} \dots b^{(N)}\}$ 在分类上等价，当且仅当对于任意 $1 \leq i \leq N, 1 \leq j \leq N$ ，存在 $\frac{a^{(i)}}{b^{(i)}} = \frac{a^{(j)}}{b^{(j)}}$ 。

本文使用序列 A 和序列 B 之间 Pearson 相关系数 PCC 来量化这种分类等价的现象。当序列 A 和 B 等价时，二者的 Pearson 相关系数等于 1，两个序列的 Pearson 相关系数越小，说明两个序列分类不等价的程度越高。

$$\text{PCC}(A, B) = \frac{\sum_{i=1}^N (a^{(i)} - \bar{a})(b^{(i)} - \bar{b})}{\sqrt{\sum_{i=1}^N (a^{(i)} - \bar{a})^2} \sqrt{\sum_{i=1}^N (b^{(i)} - \bar{b})^2}} \quad (6)$$

3.2 量化乘法中的分类不等价

在原始数据类型乘法 $r_3 = r_1 r_2$ 中，固定 $r_1 = r_1^C$ 时，给定输入序列 $R_2 = \{r_2^{(1)}, r_2^{(2)}, \dots, r_2^{(N)}\}$ ，依次将 R_2 中元素与 r_1^C 相乘，得到输出序列 $R_3 = \{r_3^{(1)}, r_3^{(2)}, \dots, r_3^{(N)}\}$ 。对于 $r_1' \neq r_1^C$ ，给定相同输入序列，得到输出序列 $R_3' = \{r_3'^{(1)}, r_3'^{(2)}, \dots, r_3'^{(N)}\}$ 。当原始数据类型为实数时，易证 R_3 和 R_3' 分类等价，二者 Pearson 相关系数为 1。在实际情况下，由于计算机无法表示实数，通常使用高精度数据来近似。在 DNN 的计算中，使用 FP32 数据类型，由于 FP32 类型数据精度足够高，其计算出的两个序列的 Pearson 系数仍等于 1。而对于量化乘法，给定输入序列 $Q_2 = \{q_2^{(1)}, q_2^{(2)}, \dots, q_2^{(N)}\}$ 。对于不同的操作数 q_1^C 和 q_1' ，产生的输出序列 Q_3 和 Q_3' 分类不等价，其 Pearson 系数显著降低。

随机生成 FP32 类型的输入序列 R_2 ，选取 $r_1^C = 1.0$ 作为基准值，选取 $r_1' = [-127, 1) \cup (1, 127]$ 。为方便对比，仅取 r_1' 为整数情况。使用 FP32 乘法计算得到输出序列 R_3 和 R_3' 的 Pearson 系数。同理，随机生成 INT8 类型输入序列 Q_2 ，选取 $q_1^C = 1$ 为基准值， $q_1' = [-127, 1) \cup (1, 127]$ ，使用量化乘法计算 Q_3 和 Q_3' 的 Pearson 系数。图 2 为实验结果，可以看出，对于 FP32 类型（无符号直线），不同取值的 r_1' 与基准值输出序列之间的 Pearson 系数始终为 1。而对于量化乘法（星号线），其 Pearson 系数小于 1。

3.3 量化乘法分类不等价的来源

本节讨论量化乘法中分类不等价现象产生原因。量化推理操作导致的这种输出序列分类不等价现象来源可以分为两部分。第一部分来源于量化操作中 round 的函数；第二部分来源于量化操作中 Clamp 的函数。

定义 2 跨度 λ ：在乘法 $r_3 = Mq_1q_2$ 中，固定其中一个操作数，不失一般性，设为 q_1 ，当另一个操作数 q_2 每增加 1 时， r_3 增加量的绝对值。其定义式为 $\lambda = |Mq_1|$ 。

round 函数的存在，导致不同跨度取值在面对具有相同增加趋势的输入序列时，其输出序列的增加趋势不同，从而导致分类不等价的输出序列。以输入序列 $Q_2 = \{1, 2, 3, 4, 5\}$ ， $M=0.57$ 为例，当 q_1 分别取值 1 和 2 时，跨度分别为 0.57 和 1.14，使用量化乘法计算输出序列分别为 $\{1, 1, 2, 2, 3\}$ 和 $\{1, 2, 3, 5, 6\}$ ，二者分类不等价。图 2 右显示了 round 的影响无符号直线为 FP32 乘法结果，加号线为 $q_3 = \text{round}(Mq_1q_2)$ 的结果。

导致输出分类不等价的第二部分来源于 Clamp 函数。由于 Clamp 函数的存在，量化乘法的乘积结果始终会落在 INT8 表示的范围 $[-127, 127]$ 。不同 q_1 取值，会导致面对同样的输入时，到达临界值的速度不同。这进一步加重了输出序列分类不等价的程度。图 2 加号线表示计算

结果使用 INT32 表示，不进行截断，星号线表示计算结果使用 Clamp 函数强制落回 INT8 区间。图中右图为左图 FP32 乘法和仅使用 round 函数量化乘法部分的放大。

由于量化推理 DNN 中的绝大部分计算为量化乘法，基于量化乘法操作导致的这种输出序列分类不等价性，本文结合侧信道分析构造了一种新型的攻击方式来逆向 DNN 中的权重。

4 攻击方法

4.1 攻击场景与攻击者能力假设

本节考虑这种场景下的攻击：攻击者 A 伪装成使用终端推理服务的普通用户，其可获取 AI 芯片 C ， C 使用量化推理技术来加速模型推理过程， C 上部署了 DNN 模型，攻击者 A 可以获取 C 运行 DNN 模型产生的功耗 P ，攻击者的目标为使用模型逆向方法恢复 DNN 模型权重集合 W 。其中，权重集合包含神经网络中全连接层和卷积层的所有权重值。这样的场景广泛存在于安防摄像头、无人驾驶、智能家居设备、AIoT 等边缘计算环境。例如，在图 3 无人驾驶汽车场景，广泛使用 AI 芯片加速目标识别等任务。

对于攻击者 A ，做出如下攻击者能力假设。

① 在信息收集阶段，可获得部署了攻击目标模

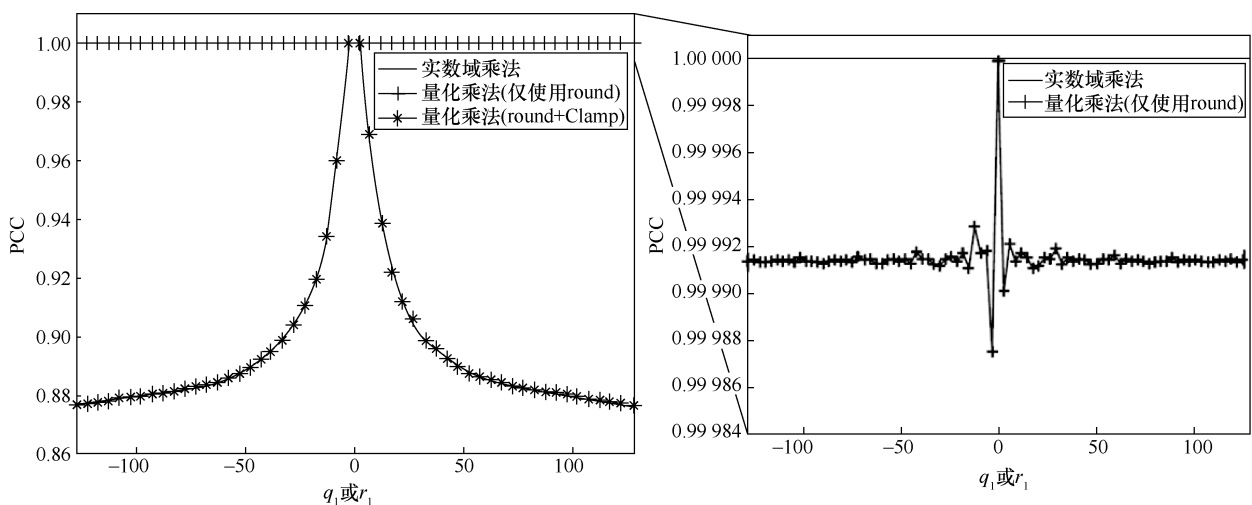


图2 FP32 乘法、量化乘法（仅使用 round 函数）、量化乘法（ $\text{round}+\text{Clamp}$ ）在输出序列分类不等价性上的区别

Figure 2 The difference of classification unequivalence between FP32 multiplication, quantized multiplication(only round),quantized multiplication(round+Clamp)

型的 AI 芯片, 控制模型的输入, 获得其输出, 并且可以测量加速卡运行 DNN 模型时产生的功耗。

② 攻击者获取了模型各层的量化参数和模型的结构。

③ 攻击者不了解 AI 芯片内部的架构设计细节, 也无法得知 AI 芯片对网络层的底层实现, AI 芯片对于攻击者来说是一个完全的黑盒。

在攻击者能力假设中, 对于假设①, 攻击者可以直接通过购买等方式获得部署了攻击目标模型的 AI 芯片并使用各种示波器和探头获取芯片运行时产生的功耗。对于假设②中的模型结构信息, 在实际环境中也可以满足, 现今神经网络的发展类似于加密算法的发展, 模型大多公开或者直接使用开源模型, 而对数据集进行保密。另外, 攻击者可以使用模型逆向的方法来恢复模型结构^[8,12]。在攻击者能力假设下, 攻击者无法获得关于 AI 芯片的内部实现细节, 无法构造有效的泄露模型。因此, 本文使用 Cluster-based SCA 来进行权重的逆向。

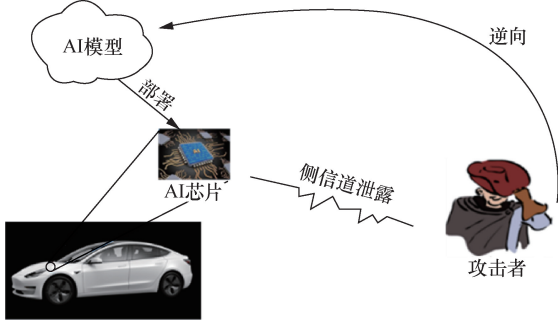


图3 攻击场景: 无人驾驶
Figure 3 Attack scenario: automatic drive

4.2 Cluster-based SCA 方法

本节介绍 Cluster-based SCA 方法的原理。Cluster-based SCA 方法的优点在于其不需要攻击者构造泄露模型。本文使用 Uniform Symmetric Quantizer 量化方案的乘法 (表示为式(5))。

本文提出的 Cluster-based SCA 目的为逆向量化乘法 $q_3 = \text{Clamp}(-127, 127, \text{round}(Mq_1q_2))$ 中的输入操作数, 具体其需要解决的问题是: 在攻击者可以控制量化乘法中一个输入数时的情况下, 不失一般性, 假设为 q_2 , 结合量化乘法操作产生的侧信道信息来恢复出另一输入数 q_1 。在

Cluster-based SCA 攻击中, 假设攻击者已经获取 M 的取值。Cluster-based SCA 利用第 3.2 节介绍的量化乘法中输出序列不等价现象。

在恢复 q_1 时, 攻击者随机生成一组与 q_1 相乘的输入序列 $Q_2 = \{q_2^{(1)}, q_2^{(2)}, \dots, q_2^{(N)}\}$ 。采集到对应的 AI 芯片计算量化乘法时产生的侧信道信号序列 $P = \{p^{(1)}, p^{(2)}, \dots, p^{(N)}\}$ 。随后, 攻击者对 q_1 的取值进行猜测, 并根据量化乘法式(5)计算得到猜测值对应的输出值序列 $Q_3 = \{q_3^{(1)}, q_3^{(2)}, \dots, q_3^{(N)}\}$ 。在量化乘法中, 不同的输入操作数 q_1 会导致输出序列分类不等价。而这种不等价的现象会通过泄露模型进一步传递到 AI 芯片在计算量化乘法时的侧信道表现上, 导致将侧信道信号按照输出结果取值进行分类时, 其离散程度显著增加, 使攻击者可以据此判断猜测正确与否。本文使用平均离散系数作为最终衡量猜测是否正确的指标。

攻击者构造键值对序列 $KV = \{< q_3^{(1)}, p^{(1)} >, < q_3^{(2)}, p^{(2)} > \dots < q_3^{(N)}, p^{(N)} >\}$ 。随后, 攻击者按照 q_3 的取值将序列 KV 中的键值对进行分类 (如图 3 所示), 得到分类集合 $C = \{C_1, C_2 \dots C_v\}$, v 为 q_3 的取值数。此时, 每个 C_i 包含若干个键值对, 其键 (也就是 q_3 取值) 相同, 但其值 (侧信道信号 p 的取值) 不一定相同。攻击者计算每个 C_i 中包含的所有侧信道信号值 p 的标准差 σ_i , 取平均得到本次猜测的平均离散系数 $\bar{\sigma}$ 。

图 4 展示了一个实际的例子。输入序列为 $\{2, 5, 6, 5, 2, 6, 5, 6\}$, $M=0.57$, 真实 $q_1 = 2$ 。使用汉明重模型 (HW leakage model) 作为芯片的实际泄露模型。运行时产生的侧信道信号序列为 $\{1, 2, 3, 2, 1, 3, 2, 3\}$ 。当攻击者对 q_1 的猜测正确时 (如图 4), 输出序列为 $\{2, 6, 7, 6, 2, 7, 6, 7\}$, 根据输出序列可将侧信道序列分为 3 类, 每个 C_i 中包含的侧信道信号 p 的取值应相同, 因为对于固定的泄露特征, 即使未知, 对于相同的运算值其侧信道表现相同。因此 C_i 中包含的侧信道信号值 p 的标准差为 0, 从而使对应的平均离散系数 $\bar{\sigma} = 0$ 。而当猜测错误时 (如图 4), 由于量化乘法的分类不等价性, 输出序列为 $\{1, 3, 3, 3, 1, 3, 3, 3\}$, 此时按照输出序列取值将侧信道序列分为两类。 C_2 中包含的侧信道信号值为 $\{2, 3, 2, 3, 2, 3\}$, 其类内的元素的取

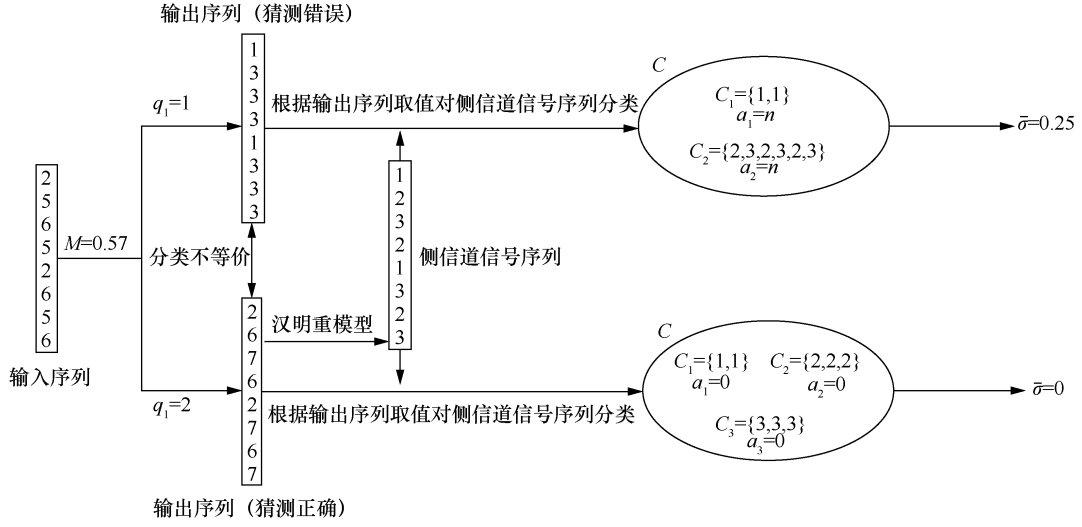


图 4 Cluster-based SCA 方法实例 ($M=0.57$)
Figure 4 An instance of Cluster-based SCA ($M=0.57$)

值不同，此时平均离散系数大于 0。

攻击者枚举 q_1 可能取值，并计算对应的平均离散系数，取使平均离散系数最小时的猜测值作为真实的 q_1 取值。这种方法以特定分类准则下的侧信道信号值平均离散程度为判断猜测是否正确的标准，此方法的实现不需要攻击者获得关于量化乘法的泄露模型的具体表达，本节将这种方法称为 Cluster-based SCA 方法，算法的详细流程见算法 1。

算法 1 Cluster-based SCA

输入 输入序列 $Q_2 = \{q_2^{(1)}, q_2^{(2)}, \dots, q_2^{(N)}\}$ ，量化参数 S_1, S_2, S_3 ，采集到的侧信道信号序列 $P = \{p^{(1)}, p^{(2)}, \dots, p^{(N)}\}$

输出 预测 q_1 值

$$1) M = \frac{S_1 S_2}{S_3}$$

2) for $k = 1 : 255$ do

$$q_1^k = k - 128$$

for $i = 1 : N$ do

$$q_3^i = \text{Clamp}(-127, 127, M q_1^k q_2^{(i)})$$

end

$$\text{KV} = \{ \langle q_3^{(1)}, p^{(1)} \rangle, \dots, \langle q_3^{(N)}, p^{(N)} \rangle \}$$

$$C = \text{ClusterWithKey}(\text{KV})$$

for $i = 1 : |C|$ do

$$\sigma_k^i = \sqrt{\frac{1}{|C|} \sum_{j=0}^{|C|} p^{(j)}, p^{(j)} \in C_i}$$

end

$$\bar{\sigma}_k = \frac{\sum_{i=1}^{|C|} \sigma_k^i}{|C|}$$

end

3) $\text{index} = \arg\max_k \bar{\sigma}_k$

return $q_1 = q_1^{\text{index}}$

4.3 Cluster-based SCA 方法分析

4.3.1 对称性

对称性是指在 Cluster-based SCA 方法中，各猜测权重 q_1^k 的平均离散系数是关于 0 对称的，这是因为互为相反数的两个猜测值，对于相同输入，使用量化乘法得到的输出也互为相反数，从而使整个输出序列在分类上等价。因此，无法使用以分类质量为基础的 Cluster-based SCA 进行区分。图 5 显示了这种对称性，图 5 表示了两个真实 q_1 在使用 Cluster-based SCA 方法进行恢复时各猜测值对应的平均离散系数。从图中可以看出，猜测权重的平均离散系数是关于 0 对称的，对称性导致攻击者只能恢复权重的绝对值。

4.3.2 不均匀性

不均匀性指的是 Cluster-based SCA 方法对

于取值较大的 q_1 具有更好的逆向效果与更强的稳定性。使用猜测熵 Guessing Entropy (本文以 AGT (average of guessing times) 表示) 与方差 (VGT, variance of guessing times) 来定量衡量 Cluster-based SCA 方法对不同真实 q_1 的预测效果。对于某一待恢复 q_1 , 使用 Cluster-based SCA 方法得到所有猜测值对应的平均离散系数, 并进行非递减排序, 得到关于猜测值的一个序列, 称为猜测序列, 猜测次数定义为真实权重 q_1 猜测序列中的索引。AGT 和 VGT 分别定义为在多次实验中, 猜测次数的平均值和方差。图 6 显示了对于不同取值的真实 q_1 , AGT 指标和 VGT 指标的变化。根据 AGT 指标和 VGT 指标的大小, 可以将 q_1 的取值分为 3 种类型。

显著区: $AGT < 2$ 。位于显著区内的 q_1 取值较大, 其输出序列中的元素取值区间较广, 导致采集到的功耗值取值数增多, 功耗值的离散程度高, 从而使正确 q_1 与错误的猜测值之间的平均离散系数差距更大。其最小平均离散系数对应的猜测值即真实 q_1 值。图 5(a) 显示了 $q_1=62$ 时, 各猜测值对应的平均离散系数, 可以看出, 在显著区, Cluster-based SCA 方法得出的结果为真实的 q_1 值。

有效区: $2 \leq AGT < 50$ 。位于有效区内的 q_1 取值比显著区的小, 使存在 $|kq_1| < 127$ 且 $|kq_1 - \text{round}(kq_1)| < \varepsilon$ 二者在输出序列分类的差别较小, 其 Pearson 相关系数近似等于 1。在噪声的干扰下, 二者无法区分, 无法理论推导出 ε 的具体值, 但可以通过实验结果图 5(b) 表明此现象的存在。

空洞区: $AGT \geq 50$ 。空洞区是不均匀性的极端表现, 跨度 λ 决定了输出序列的变化趋势。Cluster-based SCA 方法以功耗值分类后平均离散系数为评判准则, 因此会受到采集到的功耗值本身的离散程度的影响, 若产生的功耗值序列本身的离散程度很小, 则分类后, 不同的猜测权重的平均离散系数将不会存在明显区别。极端情况下, 对于某一真实的 q_1 , 当 $\lambda = |Mq_1| < 1/254$, 无论输入序列 Q_2 中的 $q_2^{(i)}$ 取值如何变化, 对于量化乘法其输出值都保持固定 $q_3=0$, 无法进行有效恢复。但由于空洞区只

有当 q_1 极小时才会存在, 所以其只占 q_1 取值区间的一小部分。当使用 Cluster-based SCA 方法进行实际的模型权重逆向时, 根据实验统计, 对于第一层卷积层的 432 个权重值, 位于空洞区的权重有 23 个, 占比约为 5.32%。

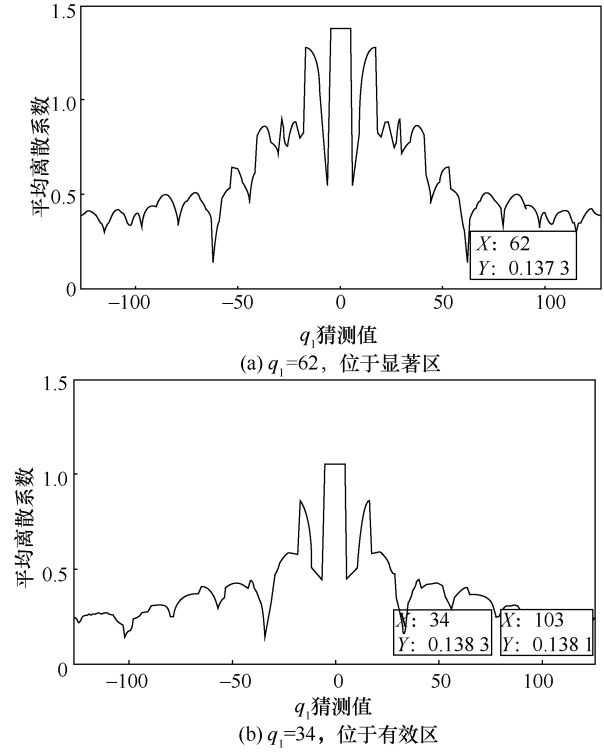


图5 不同 q_1 取值 Cluster-based SCA 的恢复效果

Figure 5 The cluster-based SCA recovery result of different q_1

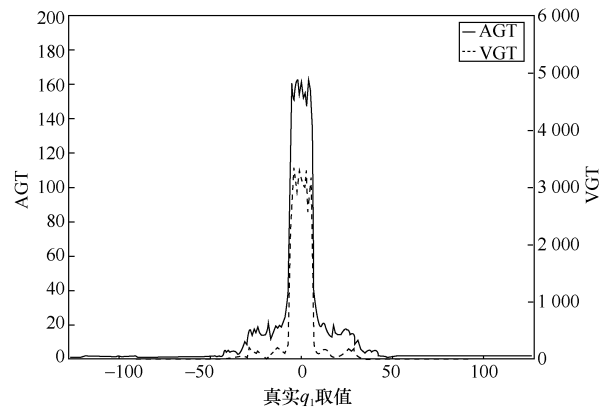


图6 Cluster-based SCA 的不均匀性 ($M=0.000\ 663$)

Figure 6 Nonuniformity of Cluster-based SCA ($M=0.000\ 663$)

4.4 基于 Cluster-based SCA 的模型权重逆向

本节使用 Cluster-based 方法来逆向模型的权重。本节将全连接层视为卷积核尺寸等于输入

特征图尺寸的特殊卷积层,根据 2.1 节和 2.2 节的分析,在量化推理中,FP32 类型权重值与特征值均被转换为 INT8 类型的数据,全连接层和卷积层的输出在进入激活函数之前,可以统一表示为式(7)。

$$\begin{cases} y_q(h, w, n) = \text{Clamp}(-127, 127, \text{round}(\text{RI})) \\ \text{RI} = \sum_{k=0}^K \sum_{l=0}^L \sum_{c=0}^C M w_q(k, l, c; n) x_q(k, l, c) + b \\ M = \frac{S_x S_w}{S_y} \end{cases} \quad (7)$$

其中, S_x 、 S_w 、 S_y 分别为输入特征图、卷积核、输出特征图激活值量化所用的量化参数。其他符号与 2.1 节式(2)相同攻击者可以采取逐个恢复卷积核权重的方式来恢复整个卷积核,由于攻击者可以控制输入,当恢复卷积核中某个特定的权 w_q^* 时,可将其他权重的对应输入值 x_q 置 0,去除掉式(7)中的求和项,从而将多个权重的联合恢复转化为单个权重的多次独立恢复。而根据 Nvidia INT8 inference 的研究,在量化推理中,偏置的有无对量化后模型的准确度影响不大,可以直接去掉。因此,当特征图中只有参数 w_q^* 对应的输入 x_q^* 不为零时,式(7)可以简化为式(8)。

$$y_q(h, w, n) = \text{Clamp}(-127, 127, \text{round}(w_q^* x_q^*)) \quad (8)$$

其中, x_q^* 为神经网络模型的输入,是攻击者可以控制的,相当于 4.1 节描述中的操作数 q_2 ,而 w_q^* 为需要恢复的操作数,相当于 4.1 节中描述的操作数 q_1 。因此,可以使用 Cluster-based SCA 方法进行参数 w_q^* 的恢复。本文以神经网络模型中的这一操作为攻击点,通过使用 Cluster-based SCA 逐个逆向模型卷积核中的权重,最终恢复整个卷积核。在恢复出 w_q^* 后,使用反量化操作得到 FP32 类型的权重值。在侧信道信号的选择上,本节选择待恢复权重对应的量化乘法在实际运行时产生的功耗。

5 仿真实验

5.1 实验设置

为探究不同网络结构对方法的影响,本文实验部分采取了 10 种不同结构的网络,其拥有不同

的卷积层设置,详细设置如表 2 所示。

为探究曲线数对方法的影响,实验对于每一种网络结构,设置了 10 种曲线数取值情况,从 1 000 到 10 000。为探究噪声对方法的影响,实验对于每一种网络结构,曲线数取 5 000 时,设置了 10 种噪声方差取值情况,从 0.1 到 1.0。

表 2 实验网络层结构
Table 2 Architecture of network layer experiment

网络编号	卷积核尺寸	卷积步长	通道数
1	3	1	16
2	3	2	16
3	3	1	64
4	3	2	64
5	3	1	128
6	3	2	128
7	5	1	16
8	5	2	16
9	5	1	64
10	5	2	64

实验使用 Cluster-based SCA 方法分别恢复了这 10 种网络结构的卷积层参数。

为了探究泄露模型对方法的影响,实验选取了 5 种不同的泄露模型进行对比,分别为汉明重模型与 4 种随机数模型(random leakage model)。使用汉明重模型为基准泄露模型来仿真 AI 芯片实际运行量化乘法时产生的功耗,并使用随机数模型进行对比实验。汉明重模型使用操作数 y 的汉明重量作为其功耗,即 INT8 数据二进制表示中置 1 的位数;随机数模型将操作数 y 映射到一个随机功耗值。二者计算式见式(9)。本文引入了高斯噪声来模型实际情况下的环境噪声。值得注意的是,虽然使用了泄露模型,但在实际的 Cluster-based SCA 方法中并没有使用到任何与泄露模型有关的信息,其作用仅为得到仿真实验所需的仿真功耗值。

$$\begin{cases} L_{\text{HW}} : p = N(\text{HW}(y_q), \sigma^2) \\ L_{\text{Random}} : p = N(\text{Random}(y_q), \sigma^2) \end{cases} \quad (9)$$

5.2 实验结果

实验使用 Cluster-based SCA 方法恢复了所有实验目标 CNN 的第一层卷积层的所有权重,使用 TOPK 恢复率来定量描述方法的效果。对于卷积核中的每一个权重,若其猜测序列前 K 个猜测权重中包含正确值,则认为成功恢复该权重。TOPK 正确率等于卷积核中恢复成功的权重数量与总权重数量之比。

图 7 显示了共 384 次恢复实验的整体情况。

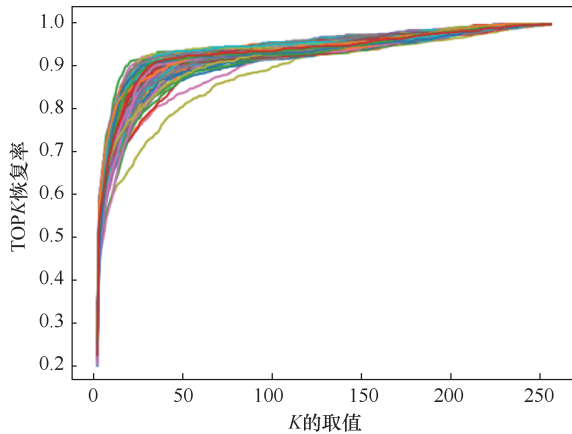


图 7 恢复实验整体情况

Figure 7 The overall situation of the experiment

表 3 展示了不同网络结构、曲线条数以及 K 取值情况下的正确率。曲线条数取值情况为 100 条、1 000 条、2 000 条、6 000 条。TOPK 中 K 取值情况为 $K=1$, $K=2$, $K=5$ 。从表 3 来看,曲线在各种

结构下的 TOPK 恢复率变化不大,这与理论分析相符合,即权重的恢复率取决于参数的分布情况,而与模型的结构无关。由于神经网络的训练特性,导致训练后的模型每一层的权重服从于高斯分布^[24]。

图 8 显示了 Cluster-based SCA 方法 TOPK 恢复率的实验结果。仿真泄露模型为汉明重模型,方差为 0.2。由图 8 可以看出,Cluster-based SCA 方法在曲线条数为 6 000 时收敛到最佳效果。

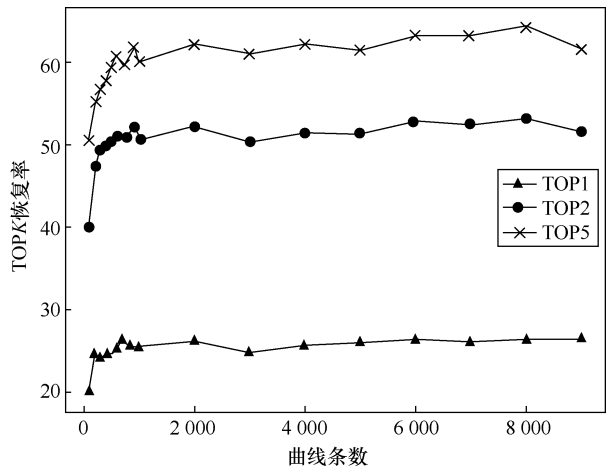


图 8 TOPK 恢复率随曲线条数的变化

Figure 8 TOPK recovery rate with respect to samples

图 8 中 TOP2 正确率相对于 TOP1 正确率提升了约 100%。这是 Cluster-based SCA 方法的对称性导致的。Cluster-based SCA 对称性的存在,导致在恢复模型权重时只能恢复其绝对值。但

表 3 不同网络结构、曲线条数以及 K 取值情况下的正确率
Table 3 TopK success rate of different architecture, traces and K

编号	100 条			1 000 条			2 000 条			6 000 条		
	TOP1	TOP2	TOP5	TOP1	TOP2	TOP5	TOP1	TOP2	TOP5	TOP1	TOP2	TOP5
1	24.74	46.57	57.42	26.69	52.08	61.68	27.34	56.55	64.32	26.48	56.64	65.37
2	18.01	34.85	49.44	28.43	54.43	64.84	29.60	55.60	66.62	24.70	54.34	62.24
3	21.70	44.62	54.95	22.13	44.18	57.99	27.60	53.21	60.24	27.17	50.87	62.85
4	18.69	34.88	45.69	26.44	53.50	59.88	25.56	50.94	59.13	27.19	54.63	64.25
5	22.74	46.70	53.65	27.78	59.90	69.97	28.65	55.38	69.44	27.26	56.42	66.32
6	19.69	39.31	51.25	26.00	50.50	57.88	23.25	50.31	61.88	29.19	56.13	66.13
7	16.44	34.95	49.07	22.80	50.35	55.56	27.78	53.59	62.27	25.23	50.35	64.12
8	20.50	42.75	50.88	27.25	50.00	60.12	25.62	51.50	62.75	25.75	51.88	65.63
9	16.15	34.55	45.49	21.18	43.75	51.91	20.66	44.27	56.25	21.88	43.58	50.87
10	19.68	41.20	47.45	25.46	48.61	59.84	25.81	46.99	56.94	27.32	51.74	63.31

这不是一个很大的问题，因为保持输入为正时，权重的正负将表现在输出值 y 上，而根据文献[8]的研究，对于激活函数，其对于正负激活值的操作时间是存在巨大区别的，因此，攻击者可以使用激活函数的执行时间来分辨出权重值的正负。

图9为曲线条数10 000条，汉明重模型方差0.2时，TOPK恢复率随 K 值的变化曲线，图中TOPK的恢复率在 $K=1,2$ 时迅速增长，随后变缓，直到 $K=243$ 才能达到100%。这是由于SCA方法不均匀性导致的。具体地，模型权重TOP2恢复率为52.66%，此时成功恢复的权重包括位于显著区的权重（在 $K=2$ 时，其绝对值被恢复出来），以及部分位于有效区的权重。对于有效区的权重，由于不均匀性，最小的离散系数可能不由真实权重或其相反数产生，在这种情况下真实权重在猜测序列中的索引大于或等于2，具体索引值取决于噪声，带有随机性。但根据图5的结果，有效区的AGT指标和VGT指标仍相对较小，所以恢复率较快上升，对应图9中 $2 < K < 40$ 区间。而对于空洞区的权重，其真实权重在猜测序列中的索引完全随机，AGT指标和VGT指标高，所以恢复率上升变得缓慢，对应图9中 $K \geq 40$ 区间。

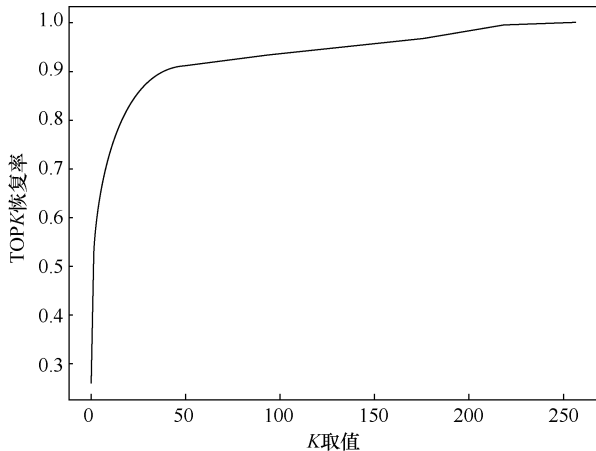


图9 TOPK恢复率随 K 值的变化
Figure 9 TOPK recovery rate with respect to K

通过对基准泄露模型设定不同的方差值，本文分析了Cluster-based SCA方法的抗噪声能力，共进行了11组实验，图10显示了Cluster-based SCA方法在面对不同大小噪声时

的表现，发现Cluster-based SCA方法具有很强的抗噪声能力。

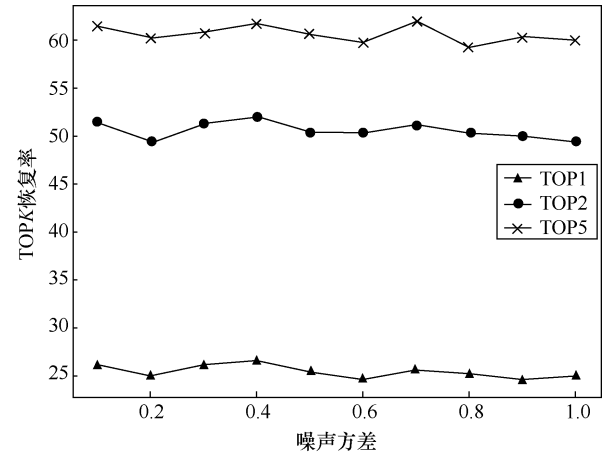


图10 TOPK恢复率随噪声的变化
Figure 10 TOPK recovery rate with respect to noise

最后选取4组随机数模型作为泄露模型，并与基准泄露模型的实验结果进行对比。图11的实验结果显示了本文方法对5组泄露模型的攻击呈现出稳定性。对于位于显著区的权重，Cluster-based SCA均以接近100%的TOP2恢复率完成了逆向。这验证了Cluster-based SCA无须构造泄露模型信息的特点。

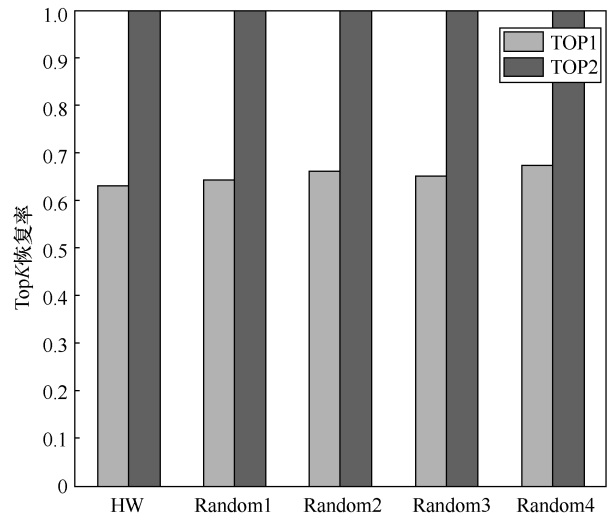


图11 TOPK恢复率随泄露模型的变化
Figure 11 TOPK recovery rate with respect to leakage model

6 结束语

本文以模型逆向攻击为研究目标，具体地，以部署在使用了量化推理技术的AI芯片上的

DNN 模型为攻击目标,旨在逆向 DNN 中卷积层和全连接层的权重。本文分析了量化推理过程,发现了其中存在的输出序列分类不等价现象,结合侧信道攻击,提出 Cluster-based SCA 方法。现有的使用 CPA 方法恢复模型权重的方法,需要攻击者构造准确的泄露模型,要求攻击者十分了解 AI 芯片的底层实现,且具有丰富的经验,这是因为 AI 芯片本身具有的黑盒性而变得难以实现。而与 CPA 方法相比,本文提出的 Cluster-based SCA 方法无须攻击者构建功耗模型,适用于未知实现细节的 AI 芯片上的模型逆向,对于权重值较大的显著区,Cluster-based SCA 方法使攻击者权重恢复指标 AGT 减少至 1.5。本文进行了仿真实验,Cluster-based SCA 方法以 52.66% 的 TOP2 恢复率恢复了 CNN 第一层卷积层所有卷积核权重,对于取值位于显著区的权重, TOP2 的恢复率均达到了 100%,并验证了方法的可行性,发现 Cluster-based SCA 方法具有抗噪能力。

但本文方法依然存在一些问题:① 本文方法假设攻击者能够获取量化参数 S_x 、 S_w 、 S_y , 这在实际环境中是一个比较强的假设。攻击者可以通过获取校准集来计算 S_x , 对于 S_w 、 S_y , 根据文献[24]的研究,在神经网络中,二者的范围都是有限的,且近似服从于高斯分布。② 使用基于 KL 散度或 EMA 等量化策略,更加平滑了量化参数,使量化参数的取值范围大大减小。因此,攻击者可以通过估计的方式近似获得量化参数,从而满足假设条件,完成攻击。并且,本文方法对于有效区和空洞区的权重的逆向存在 AGT 和 VGT 指标高的问题。

未来的工作将集中于:① 验证使用估计量化参数进行逆向的有效性以及提出应对措施解决 Cluster-based SCA 的不均匀性问题;② 将 Cluster-based SCA 方法用于实际的攻击环境中并对其进行评估;③ 给出针对 Cluster-based SCA 的防护措施。

参考文献:

- [1] KURAKIN A, GOODFELLOW I J, BENGIO S, et al. Adversarial examples in the physical world[J]. CoRR, abs/1607.02533, 2016.
- [2] GOODFELLOW I J, SHLENS J, SZEGEDY C, et al. Explaining and harnessing adversarial examples[C]//3rd International Conference on Learning Representations(ICLR 2015). 2015.
- [3] BIGGIO B, NELSON B, LASKOV P, et al. Poisoning attacks against support vector machines[C]//Proceedings of the 29th International Conference on Machine Learning(ICML 2012). 2012.
- [4] XIAO H, BIGGIO B, BROWN G, et al. Is feature selection secure against training data poisoning? [J]. CoRR, abs/1804.07933, 2018.
- [5] SHOKRI R, STRONATI M, SONG C Z, et al. Membership inference attacks against machine learning models[C]//2017 IEEE Symposium on Security and Privacy (SP 2017). 2017: 3-18.
- [6] TRAMER F, ZHANG F, JUELS A, et al. Stealing machine learning models via prediction apis[C]//25th USENIX Security Symposium. 2016: 601-618.
- [7] OH S J, AUGUSTIN M, FRITZ M, et al. Towards reverse-engineering black-box neural networks[C]//6th International Conference on Learning Representations(ICLR 2018). 2018.
- [8] BATINA L, BHASIN S, JAP D, et al. CSI NN: reverse engineering of neural network architectures through electromagnetic side channel[C]//28th USENIX Security Symposium. 2019: 515-532.
- [9] OREKONDY T, SCHIELE B, FRITZ M, et al. Knockoff nets: stealing functionality of black-box models[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2019). 2019: 4954-4963.
- [10] HUA W, ZHANG Z R, SUH G E. Reverse engineering convolutional neural networks through side-channel information leaks[C]//Proceedings of the 55th Annual Design Automation Conference (DAC 2018). 2018: 41-46.
- [11] WEI L X, LUO B, LI Y, et al. I know what you see: Power side-channel attack on convolutional neural network accelerators[C]//Proceedings of the 34th Annual Computer Security Applications Conference. 2018: 393-406.
- [12] HU X, LIANG L, DENG L, et al. Neural network model extraction attacks in edge devices by hearing architectural hints[J]. CoRR, abs/1903.03916, 2019.
- [13] HAN S, MAO H Z, DALLY W J. Deep compression: compressing deep neural networks with pruning, trained quantization and human coding[C]//ICLR 2016. 2016.
- [14] JACOB B, KLIGYS S, CHEN B, et al. Quantization and training of neural networks for efficient integer- arithmetic-only inference[C]//2018 IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2018). 2008: 2704-2713.
- [15] RASTEGARI M, ORDONEZ V, REDMON J, et al. Xnor-net: Imagenet classification using binary convolutional neural networks[J]. CoRR, abs/1603.05279, 2016.
- [16] PATIL N, JOUPPI N P, YOUNG C, et al. In-datacenter performance analysis of a tensor processing unit[C]//Proceedings of the 44th Annual International Symposium on Computer Architecture(ISCA 2017). 2017: 1-12.

- [17] KOCHER P C. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[C]//Advances in Cryptology | CRYPTO '96. 1996: 104-113.
- [18] MANGARD S. A simple power-analysis (spa) attack on implementations of the AES key expansion[C]//International Conference on Information Security and Cryptology(ICISC 2002). 2002: 343-358.
- [19] KOCHER P, JAFFE J, JUN B. Differential power analysis[C]//Advances in Cryptology(CRYPTO' 99). 1999: 388-397.
- [20] BRIER E, CLAVIER C, OLIVIER F. Correlation power analysis with a leakage model[C]//Cryptographic Hardware and Embedded Systems(CHES 2004). 2004: 16-29.
- [21] KRIZHEVSKY A. One weird trick for parallelizing convolutional neural networks[J]. CoRR, abs/1404.5997, 2014.
- [22] HE K, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2016). 2016: 770-778.
- [23] KRISHNAMOORTHY R. Quantizing deep convolutional networks for efficient inference: a whitepaper[M]. CoRR, abs/1806.08342, 2018.
- [24] LIN D D, SACHIN S, TALATHI V, et al. Fixed point quantization of deep convolutional networks[C]//Proceedings of the 33rd International Conference on Machine Learning(ICML 2016). 216: 2849-2858.

[作者简介]



李景海（1996—），男，重庆人，武汉大学硕士生，主要研究方向为侧信道与AI安全。



唐明（1976—），女，湖北武汉人，武汉大学教授、博士生导师，主要研究方向为信息安全、密码学、密码芯片。



黄诚轩（1997—），男，湖北武汉人，武汉大学硕士生，主要研究方向为侧信道与AI安全。