Advances in Applied Mathematics 应用数学进展, 2023, 12(4), 1581-1589
Published Online April 2023 in Hans. https://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2023.124163

Hans汉斯

基于Vision Transformer的有学习的侧信道攻击模型

廖杨杰*，王　燚

成都信息工程大学网络空间安全学院，四川　成都

收稿日期：2023年3月19日；录用日期：2023年4月15日；发布日期：2023年4月24日

摘　要

在侧信道攻击中，任何防御对策的目标都是使减弱能量消耗与设备所执行密码算法的中间值的关系。加掩方案就是通过随机化密码设备所处理的中间值来达到这个目标。Sbox乱序方案则通过使密码算法执行过程中的Sbox盒的执行顺序，以达到随机化各中间值所对应能量泄露时刻。针对这两类防御对策，目前基于有学习的侧信道攻击模型一般使用多层感知器、卷积神经网络和循环神经网络。本文基于计算机视觉领域的Vision Transformer (ViT)模型提出一种有学习的攻击模型VITSCA。VITSCA模型主要针对自注意机制做了微调，通过引入一个权重向量对输入的样本权重进行记录而非使用查询向量和键值对组合，更有利于攻击模型从大量的能迹样本中筛选出更有用的信息进行攻击。VITSCA模型能减少模型训练的时间以及提高模型的精确度，能有效对经过加掩方案和Sbox乱序的数据集进行攻击。本文引言部分过于简单，缺少对现有文献的综述和分析，同时也未对本文创新性和研究内容进行总结，议广泛阅读文献，对该研究领域的研究现状进行系统综述。

关键词

Vision Transformer模型，自注意力机制，Sbox乱序，加掩方案

Learning Side Channel Attack Model Based on Vision Transformer

Yangjie Liao*, Yi Wang

School of Cybersecurity, Chengdu University of Information Technology, Chengdu Sichuan

Received: Mar. 19th, 2023; accepted: Apr. 15th, 2023; published: Apr. 24th, 2023

*通讯作者。

文章引用: 廖杨杰, 王燚. 基于 Vision Transformer 的有学习的侧信道攻击模型[J]. 应用数学进展, 2023, 12(4): 1581-1589. DOI: 10.12677/aam.2023.124163

Abstract

In a side-channel attack, the goal of any defensive countermeasure is to reduce the energy consumption in relation to the median value of the cryptographic algorithm performed by the device. Masking schemes achieve this goal by randomizing the intermediate values processed by cryptographic devices. The Sbox out-of-order scheme randomizes the corresponding energy leakage time of each intermediate value by making the execution order of Sboxes in the execution process of the cryptographic algorithm. For these two kinds of defense countermeasures, the current learning-based side channel attack model generally uses multi-layer perceptron, convolutional neural network and cyclic neural network. This paper proposes a learning attack model VITSCA based on Vision Transformer (ViT) model in the field of computer vision. VITSCA model is mainly fine-tuned for the self-attention mechanism. By introducing a weight vector to record the input sample weight instead of using the combination of query vector and key-value pair, it is more conducive for the attack model to screen out more useful information from a large number of trace samples for attack. VITSCA model can reduce the time of model training and improve the accuracy of the model, and can effectively attack the data set after masking scheme and Sbox out-of-order.

Keywords

Vision Transformer Model, Self-Attention Mechanism, Sbox Shuffle, Mask Schema

1. 引言

密码算法在硬件设备上运行会产生功耗消耗、电磁辐射、热量消耗等敏感信息。通过利用这类泄露的秘密信息对密码进行分析的方法称为侧信道攻击。能量分析属于侧信道攻击的一种常见攻击方法，其本质上利用了数据依赖性和操作依赖性两类。Kocher [1]在其开创性文章 KJJ99 中就基于上述理论证实：对于功耗分析攻击没有保护的智能卡很容易被破坏。侧信道攻击使含密码的各种芯片设备安全遭到巨大挑战。

能量分析包含简单能量分析(SPA, Simple Power Analysis) [2]、差分能量分析(DPA, Differential Power Analysis) [3]等。能量分析亦可分为有学习和无学习两类。有学习的能量分析包含模板攻击(TA, Template Attack) [4]、随机攻击(SA, Stochastic Attack) [5]。有学习的能量分析中，攻击者拥有与目标设备相同的类型的设备，并使用多元正态分布对能量迹构建汉明重量或者中间值模板，最终使用这种模板对密钥进行恢复。

对于密码的软件实现中，为了保证密码算法的实现不受功率分析攻击，最经典的则是掩码技术[6]和乱序技术[7]，另外一种则是隐藏技术[8]。密码设备的能量消耗依赖于算法执行过程中所处理的敏感中间值，掩码技术和乱序技术则是针对该中间值进行“掩盖”，从而切断敏感中间值与能量消耗之间的相关性来达到防御侧信道攻击。

近年来随着机器学习的发快速发展，各种有学习的神经网络构建的高效模型涌现，如基于 cnn 的侧信道攻击模型、基于 mlp 的侧信道攻击模型等。本文针对低熵掩码技术和 S 盒乱序技术提出基于

vit-transformer 的新型侧信道攻击模型。

2. 相关概念

2.1. 高级加密标准(AES)

高级加密标准[9]是当今最流行的加密标准。它是一种对称密码算法，可以在各种平台上有效地实现。它还可以用于身份验证。因此，对于许多安全相关的应用来说，它是一种很有吸引力的算法。AES 算法限定明文分组长度为 128 bits，而密文长度可分文 128、192、256 bits，因此 AES 有三个版本：AES-128、AES-192、AES-256，相应的迭代轮数分别为 10、12、14。如下表 1 所示。

Table 1. Relationship between AES algorithm's grouping, key length and encryption rounds
表 1. AES 算法的分组、密钥长度和加密轮数的关系

AES	密钥长度 (32 比特字)	分组长度 (32 位比特字)	加密轮数
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

AES 加密具体过程为密钥和明文首先进行第一个轮密钥加，之后各轮 AES 进行加密循环过程，除最后一轮不包含列混淆操作。包含 4 个基础循环步骤：字节替换(SubBytes)、行移位(ShiftRows)、列混淆(MixColumns)、轮密钥加(AddRoundKey)。每个步骤具体如下：

1) 字节替换(SubBytes)：通过非线性替换函数(S 盒)把每个字节替换成相应的字节。
2) 行移位(ShiftRows)：将每行以字节为一个单位进行循环式移位。
3) 列混淆(MixColumns)：经过行移位后的矩阵乘一个固定矩阵得到一个混淆矩阵。
4) 轮密钥加(AddRoundKey)：矩阵的每一个字节与该次循环的子密钥做异或运算。

字节替换是 AES 算法中唯一非线性变换处理，非线性变换不受控，因此绝大多数功耗泄露都发生在该步骤。掩码以及 sbox 乱序防护对策也是针对此处泄露而设计。

2.2. SBOX 乱序和加掩方案

加掩方案是使用一个称为“掩码”的随机值(攻击者无法获取该掩码)，它与中间值混合，导致攻击者无法预测功耗。掩码核心思想是将每个敏感变量 X 随机分割为 $d+1$ 份 $M_0,\cdots,M_d$，且满足 $M_0 * \cdots * M_d = X$ 这样的群运算 $*$ (例如 x 异或、模加法)。通常 $M_1,\cdots,M_d$ (掩码)为随机抽取而 M_0 (掩码变量)满足 $M_0 * \cdots * M_d = x$，参数 d 通常称为掩码阶数。当执行时(即当所有的掩码在不同的时间被处理时)，d 阶掩码能至少抵御 $d+1$ 阶泄露。

乱序变换在侧信道防御中也是一种简单且有效的对抗的手段，它通过引入随机数使密码设备在执行加密过程中对其操作顺序随机化。乱序变换将敏感变量 X 信号分布于 t 个不同时间泄露信号 $S_1,\cdots,S_t$。如果分布均匀，那么对于每一个 i，S_i 对应于对 X 的操作的概率是 $\frac{1}{t}$。因此，X 上瞬时泄漏的信噪比降低了 t。乱序应用很简单，与保护层的性质(线性或非线性)无关。此外，当应用于非线性层时，乱序变换通常比高阶掩蔽的成本要低得多。由于高阶掩蔽是昂贵的(额外电路资源)，而且一阶掩码可以轻松被攻破，所以一个自然的想法是将乱序与一阶掩码结合使用。

2.3. 卷积神经网络

卷积神经网络[10]主要由卷积层、池化层、全连接层。卷积神经网络通过卷积层(卷积神经网络的核心)对输入的数据进行特征提取，然后将提取的特征数据作为池化层的输入，池化层通过最大池化或平均池化这两种操作来降低数据维度，其具有平移不变性。在实际的使用中，卷积层、池化层、全连接层的层数和大小都可以根据具体的情况而调节。在侧信道攻击中常使用一维的卷积核池化，具体如图 1 所示。

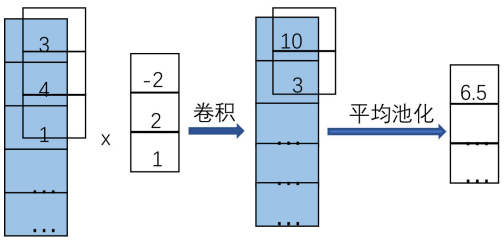

Figure 1. One-dimensional convolution and pooling
图 1. 一维卷积与池化

2.4. ViT

2017 年 Vaswani 等人[11]提出自注意机制且完全抛弃 RNN 和 CNN 等网络结构来构建一个全新的网络结构 Transformer。因该网络没有任何卷积操作，具有高度并行能力，能有效缩短训练时间。其后 Dosovitskiy 等人[12]基于 Transformer 模型提出 ViT，此模型在计算机视觉领域效果甚好，吸引众多领域的研究者使用。ViT 模型如图 2 所示。

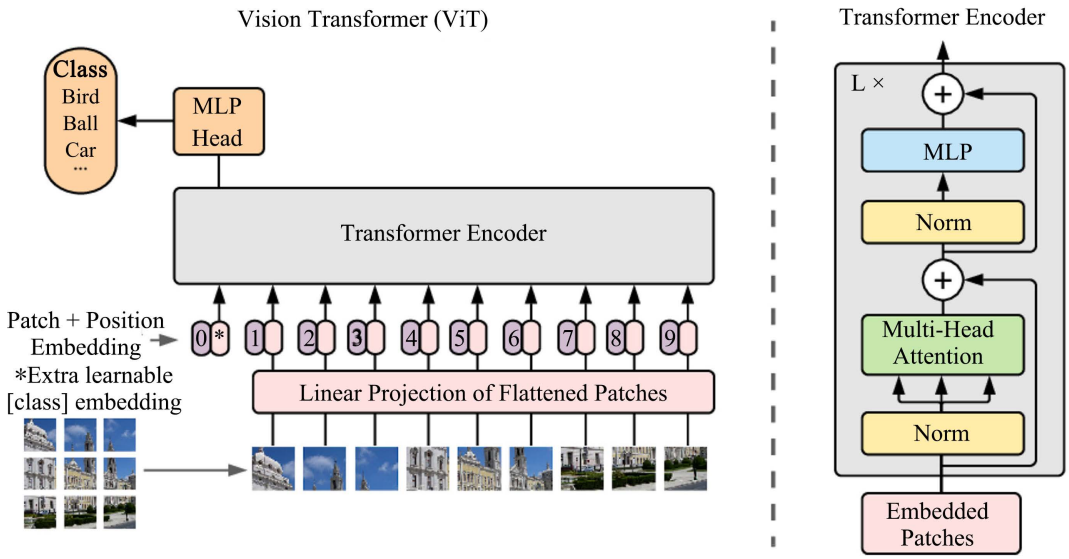

Figure 2. Vision transformer model
图 2. Vision Transformer 模型

ViT 模型在结构上摒弃了 Transformer 的 Decoder 部分，只保留了 Encoder。在编码时(上图右部分 Embedded Patches)，ViT 首先对一个图片划分为 n * n 个固定大小的 patch 输入，然后需要对每一个 patch 再映射到实际需求的一个维度(使用一个 layer 层对其映射，或者使用一个卷积层操作)。ViT 使用了包含查询、键和值的自注意机制，自注意力机制能对大量数据中重要数据进行注意，而忽略不重要的数据。

3. 基于 ViT 模型设计

本节主要介绍改造 VIT 模型，该模型对 Embedded Patches 层、Multi-head Attention 层、MLP 层做了调整，以便使该模型更好的适用于 Sbox 乱序和加掩方案的攻击。VITSCA 模型结构如图 3 所示。

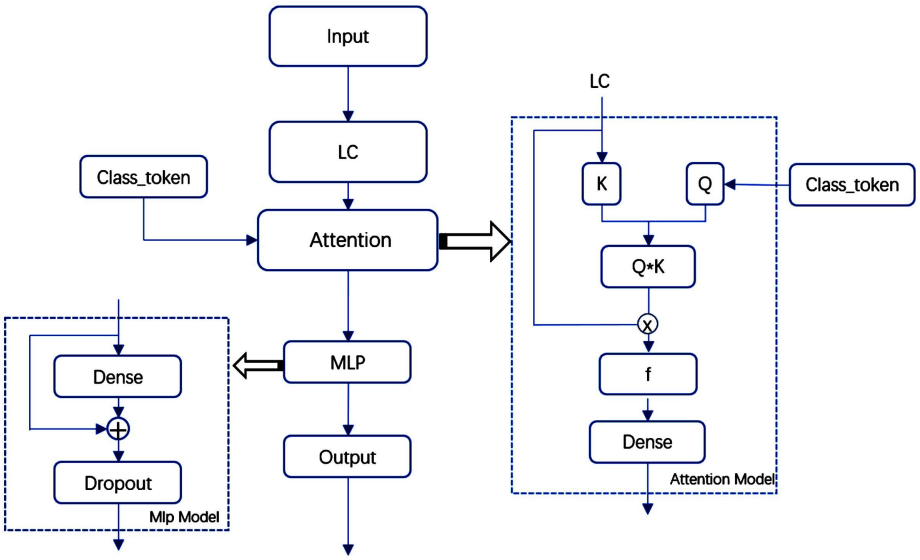

Figure 3. VITSCA model
图 3. VITSCA 模型

3.1. 嵌入层

能迹样本作为模型的输入，嵌入层会对样本做初步的处理，提高注意力模块对重要样本的注意。嵌入层采用一维卷积层，一维卷积能将能迹样本的一个 CPU 时钟采样周期缩短为一个样本，且该卷积层的特点在于卷积核不共享权重，能对多个位置的能耗信息进行交叉组合，对后续掩码以及 Sbox 乱序攻击达到预处理效果。

3.2. 注意力机制

LC 嵌入编码后，直接进行注意力编码，使用基于标准 VIT 模型修改的模型。在标准 VIT 模型中：输入来源于 Patch Embedding 模块加上 Position Embedding 模块数据；自注意力编码部分使用一个查询向量 Q 和一个键值对集合(KV)。基于 ViT 模型修改模型：输入与输出形状相同，以便叠加 N 层自注意编码，使其能对大量数据进行训练，得到更好的训练效果；自注意编码部分使用权重矩阵 cls_token (单独生成且固定)对输入序列计算单向注意力，用于的对序列样本进行加权处理。cls_token 表示样本的权重，权重大小表示样本的重要程度，既样重要本的权值大，非重要样本的权值小。对有效提高对掩码和 Sbox 乱序攻击效果。对于编码器的输入 X 与输出 cls_token 满足如下公式：

$$\text{Self\_Attention}\left(Q*K\right)=f\left(\text{softmax}\left(\frac{Q*K}{\sqrt{d}}*X\right)\right) \tag{1}$$

$$\text{cls\_token}=\text{Dense}\left(\text{Self\_Attention}\left(Q*K\right)\right) \tag{2}$$

公式(1)中：self_attention 函数为自注意机制主要函数。其中右边 X 为自注意力机制的输入部分，它来源 LC 层输出，且其映射为 K。而 Q 为 cls_token 所映射而来，并且 self_attention 函数输出作为下一轮

样本输入(即 K)。$\frac{1}{\sqrt{d}}$ 称为缩放因子，当乘积在数量级上增长时，其会将 softmax 函数推向具有极小梯度的区域。为了抵消这种影响，可以对点积扩展 $\frac{1}{\sqrt{d}}$ 倍。f 函数为注意力函数，其均包含一层 multiply、Dropout、Dense 层。f 函数可以提高模型拟合能力和模型表达能力。

公式 2 中：cls_token 经过 Dense 后数据，将其映射为原本的 cls_token 维度后作为下一轮的 Q。cls_token 单独为一个注意力加权序列，从始至终都将分类信息保存在一个固定大小的序列中，有利于控制过拟合。经过多层的注意力得到多个 cls_token，对所有 cls_token 做平均，提高模型的准确度。

特别注意的是在 f 函数中，通过注意力加权后(对于注意力机制中的多头是采用所有头部信息取和还是求每一个头部信息的均值会在第四节中实验分析中论述)，用相同 Dense 映射为与 cls_token 大小相同的向量，得到和注意力多头大小的 cls_token 维度的序列。好处如下：

1) 注意力头编码序列长度为注意力头数，可以尽快降低数据复杂度(筛除无效样本)。

2) 这种编码合并了所有嵌入样本的信息，全局性更强。

3) 模型的复杂度由头数和注意力块的内部维度决定。

经过注意力编码后，注意力头编码序列长度缩短，可以有效控制模型大小，以便使用更大批次进行训练。注意力编码是全局组合，使用模型可以尽早开始寻找能耗的非线性组合方式，以便从大量数据中进行学习，找到重要的样本点进行攻击。

3.3. 分类层

自注意模块中最终输出有两类如公式(1)和公式(2)，其中公式(1)的数据将会被舍弃，而公式(2)的输出 cls_token 保存了样本权重(即最终的分析特征)，也做为分类器的输入。分类器设计为三层的残差块，有且仅包含一层 dense 层和 Dropout 层。分类层最终输出一般为 256。

4. 实验设计与结果分析

4.1. 实验环境和数据

实验环境为搭载了 4 块 NVIDIA GeForce RTX 2080 Ti GPU 的服务器，所创建模型基 TensorFlow2.0 搭建，并在改实验环境进行模型训练以及侧信道攻击。

数据 1 采用 DPAcontest 国际侧信道大赛的 DPA_contestv4_2。该数据具有 sbox 乱序防护和低熵掩码防护措施。该数据在 AtmelATMega-163 智能卡上以 AES-256 实现，其该包含 80,000 条能迹，采用 16 种不同的密钥，每个密钥有 5000 个能迹，每条能迹样本大小为 1,704,402。DPA_contestv4_2 采用加掩防护与 Sbox 乱序相结合的技术。该数据方案通过掩码将敏感数据进行屏蔽，且引入乱序索引数组的方式分别对 AES 算法的第一轮与第十轮中 16 个非线性 S 盒变换的执行顺序进行随机化保护，进而能够在单条能量曲线上有效隐藏 S 盒能量泄露的产生位置。由于样本点大，不能对整个样本进行攻击，只需对泄露区间样本攻击。s 盒输入 X_i 被掩码 m_i 屏蔽，s 盒 Y_i 又被 m_{i+1} 屏蔽，S 盒输入与输出存在相同的寄存器中，寄存器活动可记为 $(X_i \oplus Y_i) \oplus (m_i \oplus m_{i+1})$，$m_i$ 和 m_{i+1} 是平衡的，但 $m_i \oplus$ 与 m_{i+1} 不平衡。本实验对该数据划分，60,000 条能迹作为训练集，20,000 条能迹作为验证集。

数据 2 使用 ASCAD SCA Database 提供的 ASCAD，该数据具有掩码防护措施。ASCAD 由 50,000 条训练集和 10,000 攻击集组成，每条能迹包含 700 样本点，样本点包含 AES 第一轮加密和第三个 Sbox 输出操作的能耗信息。(在第 1 轮加密中，与密钥关联的第 1 个字节和第 2 个字节未受到掩码保护，即前 2 个字节的掩码为 0，其余 14 个密钥字节掩码都是随机的)。

4.2. 实验指标

Standaert 等人[13]指出侧信道攻击中的标准度量是成功率和猜测熵。本文采用猜测熵作为模型的评价指标。猜测熵是指在攻击一个子密钥时，子密钥在候选子密钥队列中的平均排名。当攻击一个 sbyte 的子密钥时，我们需要使用模型计算各种猜测子密钥的得分，即根据能耗向量，使用训练好的模型“预测”猜测中间值的概率(一般取其对数值)，以此作为猜测子密钥的得分。根据各猜测子密钥的得分对它们进行降序排序，猜测熵越小，模型质量越好。猜测熵如公式(3)：

$$\mathrm{ge} = \left| K \in \kappa \mid P(K) > P\left(k^*\right) \right| \tag{3}$$

其中，$p(K)$表示密钥猜测最终得分，$p\left(K^*\right)$表示获得正确密钥得分，ge 为猜测熵。

4.3. 实验分析

4.3.1. 数据集 DPA_contestv4_2 的实验分析

本文将基于 ViT 的侧信道模型与基于 MLP、一维卷积的攻击进行了对比。本文建立上述三个攻击模型后，用 60,000 能迹训练集对其进行训练，20,000 能迹对其验证。

多数神经网络模型在训练前都会设置一些模型参数，如多层感知器网络中的隐藏层数量，大小和激活函数等。由于这些参数是执行前指定的，而不是模型中学习的参数，因此被称为超参数。攻击模型的超参数设置比较困难，因此需要使用超参数寻求器，它们包含：网格寻优器、贝叶斯寻优器、随机寻优器，本文模型使用的是随机寻优器。使用超参数寻优器的目的是对超参数的各种取值进行交叉组合，使用超参数的每个组合配置模型，然后进行模型训练。根据模型最终使用验证集取得的猜测熵，确定并保存最佳模型。

由于 DPA_contestv4_2 能迹样本点数量庞大，因此只截取一小部分样本(该小段只包含了一个 S 盒的加密能耗，剩下 S 盒的能耗就成了噪音)作为输入。该部分样本包含了剔除大量噪声样本仅保留了有关掩码防护和 Sbox 乱序部分的能耗泄露。采用上文提到注意力机制中的多头采用所有头部信息取和的攻击效果如图所 4 示。

CNN 模型，在所需能迹数为 1 时猜测熵达为 52.9，而在 19 条时为 0；MLP 模型，在所需能迹数为 1 时猜测熵为 15.7，在 18 条时为 0；VITSCA 模型，在所需能迹数为 1 时猜测熵为 1.5，在 10 条能时为 0。从图 4 可以看出本文提出的基于 ViT 侧信道攻击模型能对存在少量噪音的数据进行学习而得到一个有学习的攻击模型，验证了本文提出的模型有效性，且在攻击效果上明显优于传统 MLP 模型和 CNN 模型。

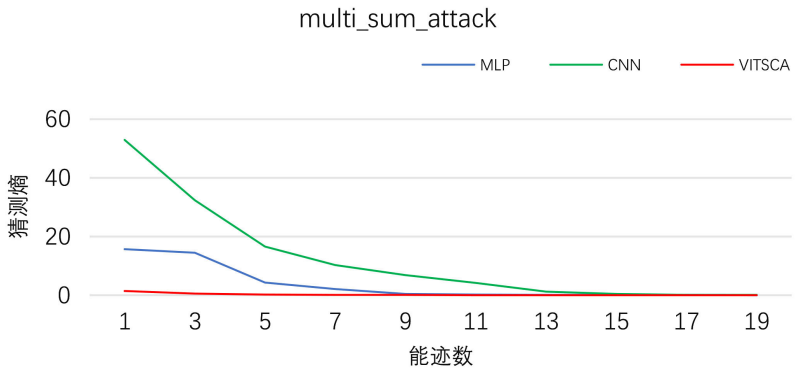

Figure 4. Average guess entropy of MLP, CNN and VITSCA sum on DPA_contestv4_2
图 4. MLP、CNN、VITSCA 和在 DPA_contestv4_2 的猜测熵

由于图 4 实验只截取了整个能迹的一小段，且无其他噪音。通过能迹规律，将样本数扩充至一轮 S 盒的时(一轮 S 盒包含 16 个 S 盒)，此时样本点的数量达到了 47,000，并且训练集中包含大量噪音，对模型的训练形成严重干扰。采用上文提到注意力机制中的多头采用每一个头部信息取和的攻击效果如图 5 所示。

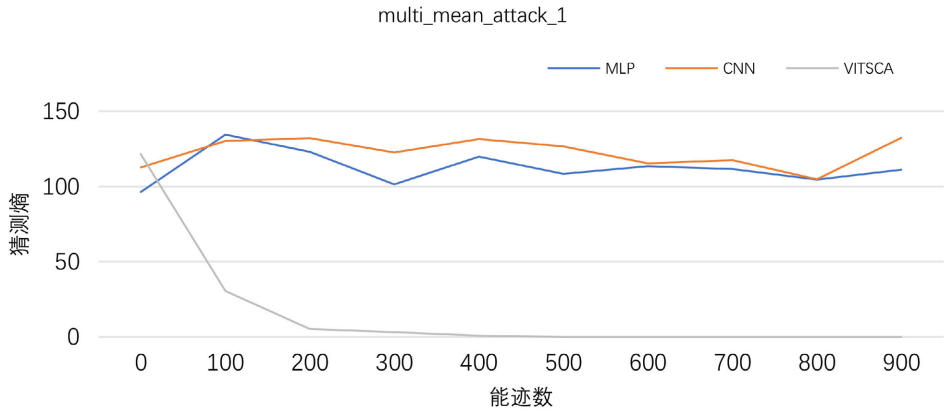

Figure 5. Average guess entropy of MLP, CNN and VITSCA mean on DPA_contestv4_2
图 5. MLP、CNN、VITSCA 均值在 DPA_contestv4_2 上的猜测熵

对于 MLP 模型和 CNN 模型，随着能迹数的增加，猜测熵几乎大于 100 以上，可知当样本点为 47,000 时 MLP 和 CNN 对此数据无效。对于 VITSCA 模型，随着能迹数的增加，猜测熵逐渐减少，当能迹数为 500 时为 0，可知本文提出的模型对具有对 Sbox 乱序和低熵掩码数据攻击的有效性。

4.3.2. 数据集 ASCAD 的实验分析

ASCAD 样本集大小只有 700，直接将全能迹作为样本点，依旧使用 MLP、CNN、VITSCA 实验结果如图 6 所示。

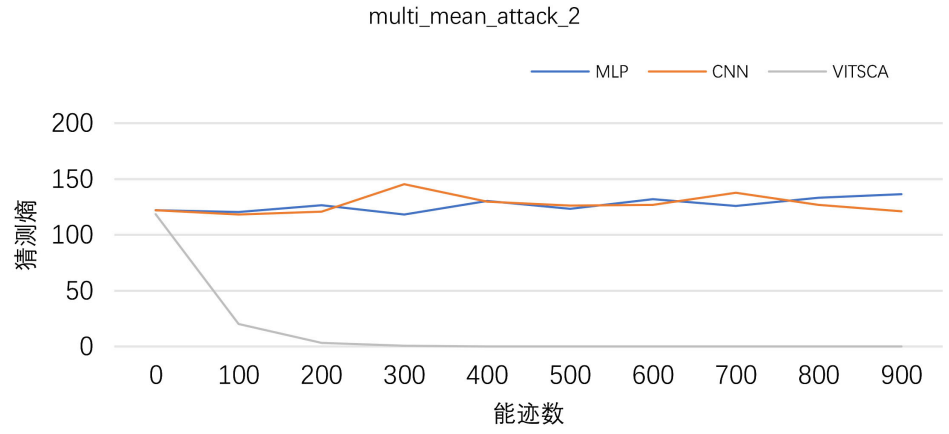

Figure 6. Average guess entropy of MLP, CNN and VITSCA mean on ASCAD
图 6. MLP、CNN、VITSCA 均值在 ASCAD 上猜测熵

从图中可以看出 MLP 以及 CNN 模型猜测熵都在 100 以上，可以 MLP 和 CNN 模型无效。VITSCA 模型猜测熵随着能迹数增大而减少，当能迹为 300 时，猜测熵为 0。由此再次证明本文提出模型的有效性。

5. 结束语

本文基于 Vision Transformer 模型，提出针对低熵掩码和 sbox 乱序的有学习的神经网络模型。该模型主要针对 ViT 模型的 Attention 部分做了重要修改，主要设计 cls_token 作为对样本集权重的注意，权重的大小作为评判标准。本文模型与传统的 MLP 模型和 CNN 模型进行分别在 DPA_contestv4_2 和 ASCAD 数据集上进行实验对比。在 DPA_contestv4_2 数据上进行了两种方案的实验，两种方案的结果证明了该模型的有效性，也即针对掩码和 Sbox 乱序防护密码算法的有效攻击。

基金项目

四川省科技计划资助(项目号：2021ZYD0011)。

参考文献

[1] Kocher, P., Jaffe, J. and Jun, B. (1999) Differential Power Analysis. In: Wiener, M., Ed., *Advances in Cryptology—CRYPTO*' 99. *CRYPTO* 1999. *Lecture Notes in Computer Science*, vol. 1666. Springer, Berlin, 388-397. https://doi.org/10.1007/3-540-48405-1_25

[2] Kocher, P.C. (1996) Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In: Koblitz, N., Ed., *Advances in Cryptology—CRYPTO*' 96. *CRYPTO* 1996. *Lecture Notes in Computer Science*, vol. 1109, Springer, Berlin, 104-113. https://doi.org/10.1007/3-540-68697-5_9

[3] Kocher, P. (1999) Differential Power Analysis and Related Attacks. *Annual International Cryptology Conference*, Germany, 1999, 388-397.

[4] Chari, S., Rao, J.R. and Rohatgi, P. (2002) Template Attacks. In: Kaliski, B.S., Koç, ç.K. and Paar, C., Eds., *Cryptographic Hardware and Embedded Systems—CHES* 2002. Springer, Berlin. https://doi.org/10.1007/3-540-36400-5_3

[5] Schindler, W., Lemke, K. and Paar, C. (2005) A Stochastic Model for Differential Side Channel Cryptanalysis. In: Rao, J.R. and Sunar, B., Eds., *Cryptographic Hardware and Embedded Systems—CHES* 2005. Springer, Berlin, 30-46. https://doi.org/10.1007/11545262_3

[6] Herbst, C., Oswald, E. and Mangard, S. (2006) An AES Smart Card Implementation Resistant to Power Analysis Attacks. In: Zhou, J., Yung, M., Bao, F., Eds., *ACNS. Volume* 3989 *of Lecture Notes in Computer Science*. Springer, Berlin, 239-252. https://doi.org/10.1007/11767480_16

[7] Rivain, M., Prouff, E. and Doget, J. (2009) Higher-Order Masking and Shuffling for Software Implementations of Block Ciphers. Cryptology ePrint Archive. http://eprint.iacr.org/2009/420

[8] Rauzy, P., Guilley, S. and Najm, Z. (2013) Formally Proved Security of Assembly Code Against Leakage. IACR Cryptology ePrint Archive 2013, 554.

[9] National Institute of Standards and Technology (2001) FIPS-197: Advanced Encryption Standard. http://www.itl.nist.gov/fipspubs/ .

[10] Friedberg, I., Skopik, F., Settanni, G., *et al*. (2015) Combating Advanced Persistent Threats. *Computers & Security*, **48**, 35-57. https://doi.org/10.1016/j.cose.2014.09.006

[11] Vaswani, A., Shazeer, N., Parmar, N., *et al*. (2017) Attention Is All You Need. arXiv: 1706.03762.

[12] Dosovitskiy, A., Beyer, L., Kolesnikov, A., *et al*. (2020) An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *Proceedings of the International Conference on Computer Vision* (*ICCV*), 2021, 6183-6192.

[13] Standaert, F.X., Malkin, T.G. and Yung, M. (2009) A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. In: Joux, A., Ed., *Advances in Cryptology—EUROCRYPT* 2009. *EUROCRYPT* 2009. *Lecture Notes in Computer Science*, vol. 5479. Springer, Berlin, 443–461. https://doi.org/10.1007/978-3-642-01001-9_26