《信息处理杂志》第31卷，第550至561页（2023年9月）

[DOI: 10.2197/ipsjjip.31.550]

普通纸

基于深度学习的弱SPN分组密码输出预测攻击的深入研究

木村隼人1,2，a)荣田圭太2伊佐波孝二2,3伊藤凉马2小川一斗2大西俊弘1,2

接收日期：2022年12月12日，接受日期：2023年6月5日

摘要：在黑盒环境中，利用深度学习进行密码分析之所以强大，是因为攻击者无需了解加密算法的内部结构。因此，设计一种能够抵御基于深度学习的密码分析的对称密钥密码至关重要。Kimura等人（AIoTS 2022）研究了对小PRESENT-[4]分组密码进行的基于深度学习的攻击，这些攻击涉及有限的组件更改，并识别出了一些特定于这些攻击的特征，这些特征不受线性或差分密码分析的影响。发现这些特征非常重要，因为利用这些特征可以使目标密码更容易受到基于深度学习的攻击。因此，本文扩展了先前的方法，旨在探索设计能够抵御基于深度学习攻击的对称密钥密码算法的线索。我们使用了包含两个弱S盒的小型PRESENT-[4]，这些S盒对差分和线性攻击较为敏感，以此来探讨经典密码学与深度学习方法之间的攻击关系。研究结果表明，基于深度学习的白盒分析的成功概率往往受到经典密码分析方法成功概率的影响。此外，即使在目标系统中S盒被替换的情况下，我们的白盒分析仍能实现与传统方法相当的攻击效果。
密码被改成了弱密码。

关键词：深度学习，分组密码，SPN

1. 介绍

与公钥加密不同，后者通过将安全性归结为数学难题来实现，对称密钥加密的安全性评估则基于其对经典攻击（如差分攻击、线性攻击和积分攻击）的抵抗能力。具体而言，会使用自动评估程序和工具（如SAT和MILP求解器）来搜索这些攻击的统计特征。如果密码对这些特征有显著的安全裕度，那么该密码可以被认为能够抵御此类攻击。通常，这些评估需要对目标算法和最先进的密码分析技术有广泛的知识，因为自动评估程序和工具必须针对不同的目标算法和攻击进行定制。

最近，基于深度学习的密码分析在对称密钥密码学领域受到了广泛关注[1]，[5]，[6]，[7]，[8]，[10]，[11]，[12]，[13]，[14]，[17]，[21]，[22]，[25]，[26]，[34]，[38]，[39]，[40].值得注意的是，这类攻击并不需要了解目标密码的具体细节，只需知道算法接口即可。即使对手不了解目标密码的具体算法，这些攻击仍然可行。在黑盒环境中，这种密码分析非常强大，即对手只需掌握最少的信息就能发起攻击。

1 日本东京都港区东海大学，邮编108-8619
2 日本国立信息通信技术研究所（NICT），Koganei，东京184-8795
3 日本兵库县神户市兵库大学，邮编650-0047
a) h_kimura@star.tokai-u.jp

关于目标密码和密码分析技术，因此在设计对称密钥密码时，必须考虑基于深度学习的密码分析方法。最近，Benamira等人对此进行了研究。[8]和Chen等人[12]确认，Gohr[17]探索的特征可以用于经典的区分攻击。这些结果可用于设计抗深度学习的对称密钥密码；然而，这可能还不够充分，因为这些特征未能识别出任何特定于深度学习的特性，这种特性会影响基于深度学习的攻击的成功概率，但不会影响线性或差分攻击等经典攻击的成功概率。

1.1 动机

Kimura等人[26]在黑盒环境中提出了新的基于深度学习的攻击方法，针对块密码。在这种环境下，攻击者除了了解密钥和块大小等算法接口外，对目标密码的具体算法一无所知。在黑盒环境中，基于深度学习的密码分析技术使得利用预先获取的输入/输出对来构建用于攻击的深度学习模型成为可能，例如密文预测和明文恢复。他们利用这些模型评估了所提出的攻击方法。随后，他们研究了基于深度学习的密码分析结果与目标块密码特性之间的关系。为此，

参考文献[26]中展示的结果是在第一作者Hayato Kimura作为日本东海大学的硕士生及日本国立信息通信技术研究所（NICT）的研究助理期间完成的。这项工作是参考文献[26]的扩展，详情请参见第1.2节。

他们在评估阶段采用了白盒分析技术，利用深度学习模型。白盒分析探讨了基于深度学习的攻击与传统攻击（如线性/差分攻击）之间的关系；因此，这可能有助于阐明基于深度学习的密码分析结果与目标分组密码特性之间的关联。

为了在黑盒环境中从白盒分析中获得高度准确的结果，Kimura等人对所有输入/输出对进行了全面的分析；因此，针对减少轮数的分组密码并不合适，因为这些分组密码与原始分组密码（例如64位或128位）具有相同的块大小。因此，他们首先关注了块大小较小的玩具分组密码（例如，PRESENT[9]的16位块变体，称为小PRESENT-[4]），并针对这些玩具分组密码进行了白盒分析作为初步实验。基于初步实验的结果，他们将提出的攻击方法应用于块大小较大的分组密码（例如。研究了32位和64位的白盒分析，针对目标分组密码进行了分析。研究结果表明，基于深度学习的小型PRESENT-[4]攻击与传统攻击方法具有相同的攻击能力。此外，他们还进行了额外的分析，测量了在小型PRESENT-[4]中成功攻击概率的变化，这些变化涉及加密组件的替换或交换，例如替换层和置换层。研究发现，内部组件的替换或交换不会影响传统线性/差分攻击的成功概率，但确实影响了基于深度学习的攻击的平均成功率；因此，他们揭示了基于深度学习的特定特征。最后，通过识别这些特定特征，研究探讨了哪些组件组合能更有效地抵抗基于深度学习的攻击。

发现这种深度学习特定特征很重要，因为利用这种特征可以使目标密码容易受到基于深度学习的攻击。因此，本文深入探讨了深度学习特有的特征，并在先前研究[26]的基础上进一步探索线索，以促进设计能够抵御基于深度学习攻击的对称密钥加密算法。

1.2 我们的贡献

本文中，我们扩展了Kimura等人提出的白盒分析方法[26]，深入探讨基于深度学习的对分组密码攻击。具体而言，我们使用了包含两个弱S-盒的小型PRESENT-[4]，这些S-盒已知容易受到差分和线性攻击[28]，以揭示经典攻击与基于深度学习的攻击之间的关系。因此，本研究旨在分析三个S-盒（即原始PRESENT S-盒和两个弱S-盒）特性差异对深度学习特定特征的影响。

深入探讨基于深度学习的攻击白盒分析。为了对小型PRESENT-[4]进行白盒分析，我们通过用已知的弱S盒替换原始PRESENT S盒，构建了两个弱小的PRESENT-[4]变体。S-盒。我们选择了文献[28]中图6.1所示的弱S-盒1。这在差分密码分析的一个例子中被使用，已知容易受到差分攻击。我们还选择了文献[28]中图7.1所示的弱S盒2，该S盒在一次线性密码分析的例子中被使用，已知容易受到线性攻击。这使我们能够准确地比较Kimura等人提出的基于深度学习的攻击与传统攻击的有效性。他们的基于深度学习的攻击可以在没有任何密钥知识的情况下，从相应的明文/密文中猜测出密文/明文。

由于其内部结构小巧，块大小为16位，我们能够通过利用尽可能多的明文/密文对来开发深度学习模型，并且可以精确计算每轮的线性/差分概率。我们证明了基于深度学习的攻击在与差分和/或线性攻击相似轮数的情况下是有效的。因此，针对具有弱S盒的小型PRESENT-[4]的白盒分析可以总结如下：

- ? 针对弱S-box1的小型PRENT-[4]，我们成功实施了11轮输出预测攻击，其中差分攻击和线性区分攻击分别可在11轮和9轮中发挥作用。
- ? 针对弱S-box2的小型PRENT-[4]，我们成功实施了8轮输出预测攻击，其中差分攻击和线性区分攻击分别可在7轮和8轮中发挥作用。

请注意，我们的攻击能够实现比区分攻击更强的输出预测（即密文预测和明文恢复），即使不知道目标密码的算法。从这些结果来看，我们得出结论：目标密码对差分/线性攻击的抵抗能力会影响基于深度学习的攻击的成功概率。

1.3 与现有研究的比较

表1和A · 1对比了所提出的与现有基于深度学习的攻击[1]，[5]，[6]，[7]，[8]，[10]，[11]，[12]，[13]，[14]，[17]，[21]，[22]，[23]，[25]，[26]，[34]，[38]，[39]，[40].在这里，我们探讨这些攻击是否在黑盒环境中表现为基于深度学习的攻击，以及在白盒分析下的基于深度学习的攻击。当对手在非黑盒环境中进行基于深度学习的攻击时，他们必须熟悉目标密码和最先进的密码分析技术。这削弱了基于深度学习的攻击的初衷，使其不再需要了解目标密码和最先进的密码分析技术，除了算法接口。此外，即使对手在非黑盒环境中使用白盒分析进行基于深度学习的攻击，也不应导致对攻击的准确评估。总之，在黑盒环境中进行基于深度学习的攻击时，结合白盒分析至关重要。如表1所示，所提出的攻击是在黑盒环境中针对易受攻击的SPN结构进行基于深度学习的输出预测攻击，并结合白盒分析，扩展了Kimura等人[26]提出的方法。

关于白盒分析，Danziger等人提出了

表1基于深度学习的密码分析对比。OP：输出预测，PR：明文重恢复（KR）、差分判别器（DD）、线性判别器（LD）以及DLD：差分线性判别器。

参考文献	密码（块大小）	结构	黑匣子设置	目标	#圆形（#完整）	白盒分析
BSS08 [5]	Serpent（128位）	SPN	不	DD	7 (32)	不
AAAA12 [1]	简化DES（12位）	菲斯特尔	是	关闭	2 (N/A^2)	不
DH14 [14]	简化DES（12位）	菲斯特尔	是	克里米亚/顿巴斯	2 (N/A^2)	不
戈尔19[17]	Speck32/64（32位）	菲斯特尔	一	克里米亚/顿巴斯	12 (22)	是
XHY19 [39]	DES（64位）	菲斯特尔	是	公关部	2 (16)	不
CY20 [10]	Speck32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	13 (22)	是
CY20 [10]	DES（64位）	菲斯特尔	不	克里米亚/顿巴斯	8 (16)	是
HLZW20[21]型	DES（64位）	菲斯特尔	不	克尔/莱德	5 (16)	不
20[34]	简化DES（8位）	菲斯特尔	不	克尔/莱德	8 (8)	不
20[34]	Speck32/64（32位）	菲斯特尔	不	克尔/莱德	22 (22)	不
20[34]	Simon32/64（32位）	菲斯特尔	不	克尔/莱德	32 (32)	不
BBDC21 [6]	吉姆利-佩尔姆（384位）	SPN	不	DD	8 (48)	不
BBDC21 [6]	ASCON-Perm.（320位）	SPN	不	DD	3 (16)	不
BBDC21 [6]	KNOT-256（256位）	菲斯特尔	不	DD	10 (28)	不
BBDC21 [6]	KNOT-512（512位）	菲斯特尔	不	DD	12 (52)	不
BBDC21 [6]	CHASKEY-Perm.（128位）	美国红十字会	不	DD	4 (12)	不
BGLMT21 [7]	Speck32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	13 (22)	是
BGLMT21 [7]	Simon32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	16 (32)	是
BGPT21 [8]	Speck32/64（32位）	菲斯特尔	不	DD	7 (22)	不
BGPT21 [8]	Simon32/64（32位）	菲斯特尔	不	DD	8 (32)	不
CY21 [12]	CHASKEY-Perm.（128位）	美国红十字会	不	DLD	4 (12)	是
CY21 [12]	DES（64位）	菲斯特尔	不	DLD	6 (16)	是
CY21 [12]	Speck32/64（32位）	菲斯特尔	不	DLD	7 (22)	是
CY21 [13]	Speck32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	13 (22)	是
CY21 [13]	Speck48/72（48位）	菲斯特尔	不	克里米亚/顿巴斯	12 (22)	是
CY21 [13]	Speck48/96（48位）	菲斯特尔	不	克里米亚/顿巴斯	12 (23)	是
CY21 [11]	DES（64位）	菲斯特尔	不	DD	6 (16)	不
CY21 [11]	Speck32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	11 (22)	不
CY21 [11]	当前（64位）	SPN	不	DD	7 (31)	不
HRC21 [22]	Simon32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	13 (32)	不
HRC21 [23]	Simon32/64（32位）	菲斯特尔	不	克里米亚/顿巴斯	13 (32)	是
HRC21 [23]	Simon48/96（48位）	菲斯特尔	不	克里米亚/顿巴斯	14 (36)	是
HRC21 [23]	Simon64/128（64位）	菲斯特尔	不	克里米亚/顿巴斯/顿涅茨克	13 (44)	是
HRC21 [23]	Speck32/64（32位）	菲斯特尔	不	DD	8 (22)	是
HRC21 [23]	Speck48/96（48位）	菲斯特尔	不	DD	7 (23)	是
HRC21 [23]	Speck64/128（64位）	菲斯特尔	不	DD	8 (27)	是
伊蒂雅21[25]	TWINE（64位）	菲斯特尔	不	DD	8 (36)	不
YK21 [40]	Speck32/64（32位）	菲斯特尔	不	DD	9 (22)	是
YK21 [40]	Simon32/64（32位）	菲斯特尔	不	DD	12 (32)	是
YK21 [40]	GIFT 64（64位）	SPN	不	DD	8 (28)	是
二战[38]21	Speck32/64（32位）	菲斯特尔	不	DD	12 (22)	是
				DD		

二战[38]21	Speck48/72（48位）	菲斯特尔	不		15 (22)	是
二战[38]21	Speck64/96（64位）	菲斯特尔	不	DD	18 (26)	是
KEIIOO22 [26]	当前（64位）	SPN	是	关闭	4 (31)	是
KEIIOO22 [26]	AES类似（64位）	SPN	是	关闭	1 (N/A^2)	是
KEIIOO22 [26]	TWINE类似型（64位）	菲斯特尔	是	关闭	3 (N/A^2)	是
本文件	**小型设备，配备弱S盒1**	SPN	**是**	**关闭**	**11** (31)	**是**
本文件	**小型PRESENT，S盒2较弱**	SPN	**是**	**关闭**	**8** (31)	**是**

1 *Gohr[17]指出，我们发现通过黑盒方法从数百万个样本中提取出大量关于圆形还原Speck的差异分布的知识是很有意义的。然而，他的黑盒方法与我们定义的黑盒设置不同。因此，我们认为他的模型属于非黑盒设置。*

2 简化的DES、AES类和TWINE类密码是原始密码的修改版本，未规定完整轮数；因此，我们把这些修改版本的完整轮数描述为“N/A”。

基于深度学习的攻击方法能够从明文/密文数据集中预测出2轮DES算法的关键比特，并分析了这些攻击与差分概率[14]之间的关系。研究团队通过对比不同属性S盒的变体，针对差分攻击进行了实验，并最终得出结论：

基于深度学习的攻击手段的差异特征与其成功概率之间存在非平凡关系。然而，他们的研究结果极为有限，因为所针对的是两轮Feistel构造——即使其组成部分是理想函数，这种构造本身也相当不安全。因此，这一结论显得不够充分。

表2 PRESENT和小PRESENT-[n]的原始S盒。

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

表3 弱S-box1，易受差分攻击。

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	6	4	C	5	0	7	2	E	1	F	3	D	8	A	9	B

表4弱S盒2易受线性攻击。

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	F	E	B	C	6	D	7	8	0	3	9	A	4	2	1	5

明确内部组件的特性如何影响整个建筑的安全性。除了改进基于深度学习的攻击方法[17]，Benamira等人[8]和Chen等人[12]还利用了Gohr攻击预期会反应的特征，提高了经典区分器的成功率。他们的研究验证了Gohr探索的特征是否可以用于经典区分攻击，并未发现任何特定于深度学习的特征。Kimura等人[26]计算并比较了经典攻击与基于深度学习的攻击的能力，以探究两者之间的关系。随后，他们确定了小-PRESENT-[n]的特定深度学习特征。然而，他们没有详细说明其提议的攻击对多种加密组件的影响，这些组件对经典攻击方法的抵抗程度各不相同。总结来说，据我们所知，Kimura等人[26]是首次进行白盒分析的研究者，但他们的研究结果未能证明对不同抗性级别的多个加密组件具有攻击能力。然而，我们发现基于深度学习的攻击受经典攻击成功率的影响。这是首次将精确计算的经典攻击与基于深度学习的攻击能力进行对比的结果。

Alani和Hu报告了针对DES、3 - DES和AES[2]的明文恢复攻击，这些攻击能够从给定的密文中猜测出明文。他们声称，对DES、3 - DES和AES的攻击分别可以在211、211和1,741（?210.76)个明文/密文对中实现。然而，Xiao等人对他们的结果表示怀疑，因为这些结果无法被重现。Baek等人在文献中也指出了这一点。因此，这些结果未被纳入表1。Mishra等人。报告指出，他们对全轮PRESENT进行了输出预测攻击；然而，这种方法效果不佳[16]。此外，某些结果还产生了经典密码，例如凯撒密码、维吉尼亚密码和恩尼格玛密码[15]，[18]，[19]，[32]。

其他基于机器学习的分析方法也已有所报道[30]，[31]。Tan等人展示了深度学习技术能够区分AES、Blowfish、DES、3-DES和RC5[36]加密的密文，从而检测出恶意软件所使用的加密算法。Alshammari等人则尝试对加密的Skype和SSH流量进行分类[3]。

本文其余部分的结构安排如下：第2节将介绍我们的目标密码，即两种玩具级SPN分组密码。随后，Kimura等人[26]提出的基于深度学习的黑盒环境下的输出预测攻击方法将在第3节中详细阐述。

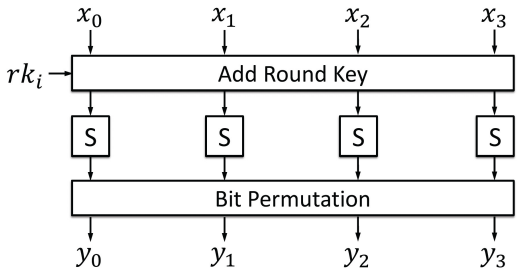

图1小型PRESENT-[4]的循环函数。

第3节介绍了相关内容。第4节讨论了我们的扩展白盒分析，探讨了我们对两种弱玩具块密码的攻击所获得的评估结果。最后，第5节对全文进行了总结。

2. 前言

接下来，我们将介绍一种SPN分组密码（PRESENT[9]）及其对应的简化版密码（small PRESENT-[n][29]）。

PRESENT和小PRESENT-[n]：PRESENT[9]是一种轻量级的SPN块密码，具有64位块大小、31轮和80或128位密钥。为了分析PRESENT，之前提出了一个名为小PRESENT-[n][29]的简化模型。图1展示了小PRESENT-[n]的轮函数。这里的块大小为4n，因此，小PRESENT-[16]等于原始的PRESENT。通过指定块大小和轮密钥长度的变体n，可以控制完整的扩散轮次。S盒的输入和输出均为4位。表2提供了一个对应表，将F42映射到F42。pLayer被描述为位置换P(i)，其定义如下。需要注意的是，这是对PRESENT的一般化，当n = 16时，与PRESENT的定义等效。此外，P(i)用于加密，而P（?1(i)）用于解密。

$$P(i) = \begin{cases} n \times i \bmod (4n-1) & (0 \le i < 4n-1) \\ 4n-1 & (i = 4n-1) \end{cases}$$

$$P^{-1}(i) = \begin{cases} 4 \times i \bmod (4n-1) & (0 \le i < 4n-1) \\ 4n-1 & (i = 4n-1) \end{cases}$$

对于密钥调度，执行PRESENT- 80的密钥调度算法，PRESENT- 80是PRESENT的一个变体，其密钥长度为80；此外，使用4n最右边的位作为轮密钥rki。

弱S-box1和弱S-box2：S-box1（表3）和S-

表4中的S盒（Box2）是已知的弱S盒[28]。S盒1易受差分攻击，而S盒2则容易遭受线性攻击。为了进行实验，我们通过将小PRESENT-[4]中的原始S盒替换为这些S盒，构建了两个变体。

3. 方法

在本节中，我们介绍了一种在黑盒环境下提出的基于深度学习的输出预测攻击。为了实现这些攻击，我们构建了用于密文预测和明文恢复的深度学习模型。接下来，我们将首先讨论这些攻击的目标，然后描述深度学习模型的构建过程。

3.1 攻击目标

Kimura等人通过更换或替换内部组件重新设计了4轮小型PRESENT-[4]，并利用白盒分析技术研究了新目标密码设计与所提攻击成功率之间的关系[26]。因此，他们澄清，交换或替换内部组件不会影响经典线性/差分攻击的成功概率，而这种操作确实会影响基于深度学习的攻击的平均成功概率；因此，他们获得了深度学习特有的特征。

然而，我们认为这些实验结果不足以促进对基于深度学习的攻击与传统攻击之间关系的讨论，因为目前尚不清楚加密组件中的哪些漏洞会影响基于深度学习的攻击的准确性。为了澄清这种关系，我们使用了两种对两种已知传统攻击敏感的分组密码。具体来说，我们将传统攻击和基于深度学习的攻击应用于三种小PRESENT-[4]变体，如第2节所述，并通过比较成功概率来观察这两种攻击能力的差异。

我们使用以下设置来评估攻击的成功概率。

已知明文攻击设置：在该设置中，对手被给予与单一密钥相关的多个明文/密文对，并且这些对被用作训练数据以构建深度学习模型。

黑盒设置：在此设置中，攻击者除了算法接口之外，对目标分组密码没有其他知识，例如密钥和分组大小。

请注意，在这两种设置中，对手都是一个非常弱的密码攻击者。

黑盒设置假设攻击者不了解密码的内部结构，也不知晓密码是置换形式。此外，黑盒设置还假设攻击者仅掌握输入输出格式，并具备深度学习知识。

关于攻击设置，仅密文攻击设置是最弱的。然而，除了RC4[33]的广播设置等少数特殊情况外，该设置下不会向攻击者提供任何信息。实际上，在这种情况下，攻击几乎是不可能的。已知明文攻击是下一个最弱的设置。在这种情况下，攻击者可以从给定的明文/密文对中获取一些信息，并利用这些信息进行攻击。其他攻击设置，如选择明文攻击设置，则要求攻击者对密文有一定的了解，这种设置下的攻击者比已知明文攻击设置中的攻击者更强。因此，我们采用了已知明文攻击设置。

在这些场景下，我们确定对手的目标是输出预测（即密文预测和/或明文恢复），并评估这些攻击的成功概率。密文预测和明文恢复攻击总结如下：

密文预测攻击：在该攻击中，攻击者获取了关于一个密钥的多个明文/密文对，其中n为块大小。随后，攻击者预测一个未包含在先前给定对中的明文对应的密文。

明文恢复攻击：在该攻击中，攻击者获得关于一个密钥的多个明文/密文对，然后攻击者恢复未包含在先前给出的对中的密文的明文。

如果密文预测攻击是可能的，那么基于密码的消息认证码（CMAC）伪造也是可能的。如果明文恢复攻击是可能的，那么即使没有加密所用的密钥，攻击者也能获得任何密文的明文。

3.2 神经网络和超参数

与统计机器学习技术（如贝叶斯推断）不同，深度学习方法能够自动提取特征。深度学习处理非线性可分问题，因此在模拟具有非线性的加密函数方面表现出色。在训练阶段之前，会预先设定多种超参数，包括初始学习率、隐藏节点数量和优化器。这些参数用于构建模型，并通过评估指标进行优化，以影响模型的性能。

在本文中，我们将密文预测和明文恢复视为带有监督学习的回归问题，其中明文与密文对被用作训练数据。为此，我们需要从已知明文攻击下获得的明文与密文对中提取大量特征；因此，我们采用了长短期记忆（LSTM）技术，这是一种循环神经网络[20]。LSTM是一种利用神经网络将序列映射到序列的通用技术，在机器翻译领域中应用广泛[35]。LSTM能够实现机器翻译中的序列映射；因此，我们认为它同样可以用于基于置换的分组密码的加密和解密过程中的序列映射（即明文与密文之间的映射）。此外，我们相信，通过LSTM可以从明文与密文对中提取大量特征，作为我们深度学习模型的输入，这使得长期

输入序列的记忆。实际上，Kimura等人证实，使用LSTM比卷积神经网络[26]*1*能够获得更好的实验结果。随后，我们优化了超参数，包括隐藏节点的数量、初始学习率、隐藏层数量和优化器。表5显示了每个超参数的搜索范围。在超参数优化过程中，我们使用了与构建深度学习模型时不同的密钥，因为我们的评估严格基于密文预测和明文恢复的成功概率，而不依赖于特定的密钥。接下来，优化超参数的过程与构建深度学习模型的过程相似，唯一的区别在于使用的密钥数量。

3.3 深度学习模型及其评价

我们根据以下流程构建并评估用于密文预测的深度学习模型。请注意，我们在括号中展示了明文恢复的情况。

步骤1：在已知明文攻击下，对手获得多个明文/密文对。在我们的实验中，我们随机选择多个明文，并生成与所选明文对应的密文。

第二步，攻击者利用获取的明文/密文对作为训练数据构建深度学习模型。随后，攻击者以明文（即密文）作为输入，密文（即明文）作为正确输出，构建用于密文预测（即明文恢复）的深度学习模型。

第三步，对手利用所有或部分剩余的明文（即密文），这些未作为训练数据使用的明文，来评估构建的深度学习模型。对手将这些明文（即密文）作为输入，预测每个明文（即密文）对应的未知密文（即明文）。

步骤4：对手计算预测密文（明文）和正确密文（明文）之间完全匹配的百分比，作为预测概率。

在我们的方法中，所有明文和密文都表示为由位串生成的数组。模型将每个预测的比特表示为0.0到1.0之间的浮点数。然后，它将原始位输出四舍五入，将其归一化为0或1，并将其作为最终结果处理

表5超参数。

超参数	搜索范围
隐藏节点数	100, 200, 300, 400, 500
学习率的初始值	0.0001, 0.001, 0.01
隐藏层数	1, 2, 3, 4, 5, 6, 7
优化器	SGD, Adam [27], RMSprop [37]

金村等人[26]的研究中，使用CNN和LSTM对基于PRESENT（SPN结构，位置换）、AES（SPN结构，MDS矩阵）和TWINE（Feistel结构）的玩具密码进行了实验。他们对CNN和LSTM进行了超参数调优，并展示了调优后的成功概率。研究结果表明，使用LSTM时，所有目标分组密码的实验效果均优于使用CNN的情况。我们的实验条件与金村等人的实验条件相似，因此我们认为金村等人展示的LSTM优于CNN的结果同样适用于本文的实验。

密文或明文预测结果。

为了评估预测的概率，我们使用2x明文/密文对作为训练数据，而剩余的明文/密文对中的2y对作为测试数据，当针对块大小为4n位的目标分组密码应用所提出的攻击时。需要注意的是，2x + 2y≤24n。在这种情况下，如果预测的概率大于（24n - 2x）-1，我们认定所提出的攻击是成功的。这意味着，即使攻击者不了解目标算法，也能以高于随机概率的准确率预测输出值。

作为补充，我们的攻击重点在于在不依赖密码学知识的情况下获取某些属性，以建立一种非专家也能使用的分析方法，我们并不强调计算复杂度。值得注意的是，由于无法使用与传统攻击方法相同的计算复杂度评估标准，因此在基于深度学习的攻击方法中，准确估计计算复杂度是不可能的。在传统的攻击方法中，可以通过加密的计算时间和攻击所需的查询次数来估算计算复杂度。然而，我们提出的方法不仅需要估算训练深度学习模型和执行该模型的计算时间复杂度，还需要估算训练模型的计算复杂度。此外，如何基于什么标准来估算计算复杂度，以实现深度学习攻击与传统攻击之间的公平比较，目前尚不清楚。另外，我们没有特别强调数据复杂度。这是因为现代对称密钥密码学的理论攻击允许使用无限的计算资源和无限的明文/密文对。总之，我们的方法不考虑计算复杂度和数据复杂度。我们能够估算数据复杂度。在实验环境中，我们利用尽可能多的数据进行密文预测和明文恢复。然而，尽管我们无法估算深度学习模型中的计算复杂度，但我们的方法可以在实际时间内被攻击。

4. 深入研究使用弱S盒的白盒分析

在本节中，我们通过深入的白盒分析，探讨基于深度学习的攻击与传统线性/差分攻击在两种基于SPN的密码算法中的关系。我们使用了两个16位块大小的SPN密码算法作为测试平台。这两个SPN密码算法分别是使用弱S-box1的小型PRESENT-[4]变体，该变体容易受到差分攻击，以及使用

表6实验超参数。

超参数	对价值的看法
输入层节点数（即块大小）	16
输出层节点数（即块大小）	16
批量大小	250
轮数	100

弱S盒2易受线性攻击。本次实验的结果将有助于更深入地理解差分攻击、线性攻击与白盒分析之间的关系。

4.1 玩具密码算法的应用

在本小节中，我们将提出的攻击方法应用于两种玩具级的块密码算法，即小型PRESENT-[4]（其S盒较为薄弱）和我们以弱S盒2的小型PRESENT-[4]为实验对象。首先，我们详细说明了白盒分析的实验流程，随后通过对比实验结果，展示了所提出的攻击方法与现有经典攻击方法在成功轮数上的差异。

我们的实验程序遵循Kimura等人提出的白盒分析方法[26]。在实验中，我们实现了pro-

表7使用所提出的算法进行密文预测/明文恢复的平均成功概率

针对弱S-box1的小型PRENT-[4]，我们使用了215个训练数据，并利用剩余的数据进行处理。

215测试数据。CP代表密文预测，PR代表明文恢复。

加密	周围	类别 攻击	#节点 隐藏层	#层级 隐藏层	初始 学习率	优化器	续。 prob.
小的 当前-[4] 和 微弱的 S-box1	1	CP	200	1	0.001	亚当	1
		公关部	100	3	0.01	亚当	1
	2	CP	500	7	0.001	RMS功率	1
		公关部	200	4	0.001	RMS功率	1
	3	CP	200	4	0.001	RMS功率	$2^{-0.01}$
		公关部	300	1	0.01	RMS功率	$2^{-0.01}$
	4	CP	500	1	0.01	RMS功率	$2^{-0.01}$
		公关部	300	7	0.001	亚当	$2^{-0.97}$
	5	CP	500	3	0.001	亚当	$2^{-5.01}$
		公关部	500	7	0.001	亚当	$2^{-4.52}$
	6	CP	200	4	0.01	亚当	$2^{-7.15}$
		公关部	500	7	0.001	亚当	$2^{-7.00}$
	7	CP	500	6	0.001	RMS功率	$2^{-9.34}$
		公关部	300	3	0.01	亚当	$2^{-9.75}$
	8	CP	400	7	0.001	亚当	$2^{-11.04}$
		公关部	500	6	0.001	亚当	$2^{-10.90}$
	9	CP	400	1	0.001	RMS功率	$2^{-12.51}$
		公关部	200	2	0.001	亚当	$2^{-12.84}$
	10	CP	500	1	0.001	RMS功率	$2^{-13.90}$
		公关部	300	7	0.001	亚当	$2^{-13.54}$
	11	CP	200	6	0.001	RMS功率	$2^{-14.36}$
		公关部	500	7	0.001	RMS功率	$2^{-14.66}$
	12	CP	100	3	0.001	亚当	$2^{-15.40}$
		公关部	300	2	0.001	RMS功率	$2^{-14.99}$
	13	CP	200	3	0.01	RMS功率	$2^{-15.82}$
		公关部	400	6	0.001	RMS功率	$2^{-15.69}$

表8使用所提出的算法进行密文预测/明文恢复的平均成功概率

针对弱S-box2的小型PRESENT-[4]，我们使用了215个训练数据，并对剩余的数据进行了处理。

215测试数据。CP代表密文预测，PR代表明文恢复。

加密	周围	类别 攻击	#节点 隐藏层	#层级 隐藏层	初始 学习率	优化器	续。 prob.
小的 当前-[4] 和 微弱的 S-box2	1	CP	100	2	0.001	亚当	1
		公关部	400	1	0.001	亚当	1
	2	CP	300	5	0.001	亚当	1
		公关部	200	5	0.001	亚当	1
	3	CP	300	4	0.001	RMS功率	$2^{-0.01}$
		公关部	500	4	0.001	RMS功率	$2^{-0.01}$
	4	CP	500	1	0.01	RMS功率	$2^{-0.03}$
		公关部	200	4	0.01	亚当	$2^{-0.97}$
	5	CP	300	3	0.01	亚当	$2^{-5.80}$
		公关部	500	5	0.001	RMS功率	$2^{-8.74}$
	6	CP	500	7	0.001	RMS功率	$2^{-10.72}$
		公关部	500	7	0.001	亚当	$2^{-11.16}$
	7	CP	300	1	0.001	亚当	$2^{-13.04}$
		公关部	500	6	0.001	亚当	$2^{-13.01}$
	8	CP	500	6	0.001	RMS功率	$2^{-14.35}$
		公关部	500	2	0.001	亚当	$2^{-14.57}$
	9	CP	100	4	0.001	亚当	$2^{-15.52}$
		公关部	500	7	0.0001	亚当	$2^{-15.92}$

表9：弱S盒*5下小PRENT-[4]的最大差分概率。

周围	最大差分概率		
	小型PRESENT-[4]，配备弱S盒1	小型PRESENT-[4]，配备弱S盒2	小型PRESENT-[4]，配备原始S盒
1	$2^{-0.6}$	2^{-1}	2^{-2}
2	$2^{-2.8}$	2^{-4}	2^{-4}
3	$2^{-4.2}$	2^{-6}	2^{-8}
4	$2^{-5.6}$	2^{-8}	2^{-12}
5	$2^{-7.0}$	2^{-10}	2^{-14}
6	$2^{-8.4}$	2^{-12}	2^{-16}
7	$2^{-9.8}$	2^{-14}	–
8	$2^{-11.2}$	2^{-16}	–
9	$2^{-12.6}$	–	–
10	$2^{-14.0}$	–	–
11	$2^{-15.4}$	–	–
12	$2^{-16.8}$	–	–

我们使用了Keras（一个深度学习库）进行攻击，并采用了TensorFlow作为后端。实验环境包括8台Linux机器，配备了14个NVIDIA GPU（如RTX 2080 SUPER、GeForce GTX 1080 Ti、TITAN Xp、Tesla K40m和Quadro P600 Mobile)。在使用Keras开发LSTM模型时，例如通过model.add（LSTM（. . .）），我们仅需指定单元数、输入形状和返回序列作为参数。初始设置中，我们采用了常见的实验超参数值（见表6）。我们的实验分为两个子实验，即实验1和实验2。

实验1：在每一轮中，我们利用第3.2节所述的攻击方法，针对目标分组密码优化超参数。在超参数优化过程中，我们采用自动优化工具Optuna *4，并使用其默认搜索算法。我们进行超参数优化的依据是密文预测或明文恢复的成功概率。具体来说，目标函数是计算预测密文与正确密文之间的完美匹配百分比，并寻求这一比例的最大值。在我们的超参数优化过程中，从20个秘密密钥生成的明文/密文对中，我们筛选出100个超参数候选。从这些候选中，我们选择平均成功概率最高的超参数进行优化。为此，我们使用215对明文/密文作为训练数据，剩下的215对明文或密文作为测试数据；因此，每个平均成功概率都是基于215个随机生成的明文/密文对计算得出的。如果使用优化后的超参数时，密文预测或明文恢复的平均成功概率大于2的15次方，那么寻找优化超参数的轮数将增加一次；否则，将使用优化后的超参数执行第二次子实验。

实验2：我们利用在实验1中优化的超参数和随机生成的100个密钥，对密文预测或明文恢复的攻击方案进行执行；随后，计算这些攻击方案的平均成功概率。

*2 https://github.com/keras-team/keras

*3 https://keras.io/ja/layers/recurrent/ *

4 https://github.com/optuna/optuna

密文预测或明文恢复。实验2中使用的秘密密钥与实验1中的不同。在明确了实验2中目标分组密码的攻击轮数后，我们利用实验结果和目标分组密码的线性/差分概率，将提出的攻击方法与传统的线性/差分攻击进行了对比。

4.2 实验结果

表7和表8展示了实验2的结果，这些结果基于实验1中优化的超参数。根据这些实验结果，我们对两种小型块密码进行了扩展白盒分析，即小PRESENT-[4]（S-box弱）和小PRESENT-[4]（S-box强）。

我们对比了经典线性/差分攻击与提出的攻击方法，针对使用弱S-box1和弱S-box2的小型PRESENT-[4]。实验结果显示，对于使用弱S-box1的小型PRESENT-[4]，所提出的攻击在密文预测和明文恢复方面最多可成功11轮，成功率接近2的负14次方。因此，我们认为所提出的攻击在使用弱S-box1的小型PRESENT-[4]上最多可成功11轮。同样，对于使用弱S-box2的小型PRESENT-[4]，所提出的攻击在密文预测和明文恢复方面最多可成功8轮，成功率接近2的负14次方。因此，我们也认为所提出的攻击在使用弱S-box2的小型PRESENT-[4]上最多可成功8轮。

另一方面，通过计算小PRESENT-[4]在弱S-box1、弱S-box2和原始S-box下的精确差分和线性概率，我们确定了经典攻击的最大可攻击轮数（见表9和10），并将其与基于深度学习的攻击的可攻击轮数进行了对比（见表11)。通过对比可以看出，基于深度学习的攻击能力与两种经典攻击方式（差分攻击或线性攻击）的最大可攻击轮数中的较高值相当。换句话说，当被攻击的目标密码存在较弱特征时，基于深度学习的攻击会作出响应。

我们通过假设白盒分析，从近似值中计算出最大差异/线性概率。这意味着表9和表10展示了单个差异/线性概率的2至16轮次。

表10小型PRESENT-[4]在弱S-box *5条件下的最大线性概率。

周围	最大线性概率		
	小型PRESENT-[4]，配备弱S盒1	小型PRESENT-[4]，配备弱S盒2	小型PRESENT-[4]，配备原始S盒
1	$2^{-0.8}$	$2^{-0.8}$	2^{-2}
2	$2^{-3.2}$	$2^{-2.4}$	2^{-4}
3	$2^{-4.8}$	$2^{-4.4}$	2^{-8}
4	$2^{-6.4}$	$2^{-7.2}$	2^{-12}
5	$2^{-8.0}$	$2^{-9.2}$	2^{-16}
6	$2^{-9.6}$	$2^{-10.8}$	–
7	$2^{-11.2}$	$2^{-12.8}$	–
8	$2^{-12.8}$	$2^{-15.6}$	–
9	$2^{-14.4}$	$2^{-17.6}$	–
10	$2^{-16.0}$	–	–

表11：使用弱/原始S盒时，小型PRESENT-[4]的最大可攻击轮数。
CP代表密文预测，PR代表明文恢复。

参考文献	加密	最易受攻击的轮次		
		基于深度学习的攻击	差分密码分析	线性密码分析
本文件	小型PRESENT-[4]，配备弱S盒1	11 (CP, PR)	11	9
本文件	小型PRESENT-[4]，配备弱S盒2	8 (CP, PR)	7	8
KEIIOO22 [26]	小型PRESENT-[4]，配备原始S盒	5 (CP), 4 (PR)	5	4

可以抵御不止一种古典攻击。

根据这些实验结果，我们认为基于深度学习的输出预测攻击将使任何人都可以更容易地估计对差分和线性攻击的抵抗能力，即使他们不了解目标的密码算法或密码分析方法。

5. 结论

在本研究中，我们提出了基于深度学习的输出预测攻击，针对两种块大小为16位的密码，在黑盒设置下进行攻击。我们通过分析基于深度学习的攻击与传统攻击（如线性/差分攻击）之间的关系，明确了以下研究结果：

- ？ 我们的基于深度学习的白盒分析即使在目标密码的S盒被改变为弱S盒时，也实现了与传统方法相同的攻击能力。
- ？ 我们发现，基于深度学习的白盒分析的成功概率往往受到经典密码分析方法成功概率的影响。
- ？ 我们相信，利用深度学习进行的输出预测攻击将使估计对差分和线性攻击的抵抗能力变得更加容易，即使没有掌握目标的密码算法或密码分析方法。

未来，我们计划做以下工作：

- ？ 我们将阐明为什么线性概率比差异概率更能提高基于深度学习的攻击的平均成功概率。
- ？ 我们将阐明如何反馈我们设计深度学习抗对称密钥密码的结果。

致谢本研究部分得到了日本学术振兴会KAKENHI资助项目（编号：19K11971）的支持。作者感谢匿名审稿人提供的宝贵意见和建议。

参考文献

[1] Alallayah，K.M.，Alhamami，A.H.，AbdElwahed，W. 和 Amin，M.：《应用神经网络对简化数据加密标准（SDES）密码系统进行密码分析》，载于《国际阿拉伯信息科技杂志》第9卷第2期，第163-169页（2012年）。

[2] Alani，M.M.：DES和Triple-DES的神经密码分析，ICONIP，第637-646页（2012年）。

[3]阿尔沙马里，R.和津吉尔-海伍德，A.N.：基于机器学习的加密流量分类：识别SSH和Skype，IEEE CISDA，第1-8页（2009年）。

[4] Baek，S.和Kim，K.：《针对分组密码的神经攻击最新进展》，发表于SCIS（2020），可从〈https://caislab.kaist.ac.kr/publication/paper_files/2020/scis2020 SG.pdf〉获取。

[5]巴夫吉，A.G.，萨法巴赫什，R.和萨德吉扬，B.：利用神经网络识别分组密码的差异特性，《信息科学》第178卷第15期，第3118-3132页（2008年）。

[6]巴克西，A.，布雷尔，J.，陈，Y.和董，X.：机器学习辅助的轻量级密码差分器，DATE，第176-181页（2021年）。

[7]包志，郭军，刘敏，马乐，涂毅。条件差分神经密码分析，载于《IACR密码学电子预印本档案》第2021卷，第719页（2021年）。

[8] Benamira，A.，Gé劳尔，D.、佩林，T.和谭，Q.Q.：《基于机器学习的密码分析深入研究》，收录于《欧洲密码学会议》（EUROCRYPT），第805-835页（2021年）。

[9]博格丹诺夫A.、克努森L.R.、莱安德G.、帕尔C.、波施曼A.、罗布肖M.J.B.、塞林Y.和维克尔索E.：《PRESENT：一种超轻量级分组密码》，收录于《CHES》会议论文集，第450-466页（2007年）。

[10]陈Y.和余H.：《神经辅助统计攻击在密码分析中的应用》，载于《IACR密码学电子预印本档案》第2020卷，第1620页（2020年）。

[11]陈Y.和余H.：一种考虑多密文对派生特征的新型神经区分模型，IACR密码学电子预印本档案，第2021卷，第310页（2021年）。

[12]陈Y.和余H.：通过EDLCT连接机器学习与密码分析，IACR密码学电子预印本档案，第2021卷，第705页（2021年）。

[13]陈Y.和余H.：《改进的神经辅助统计攻击在密码分析中的应用》，载于《IACR密码学电子预印本档案》第2021卷，第311页（2021年）。

[14]丹齐格，M.和阿马拉尔·亨里克斯，M.A.：结合差分与人工神经网络方案的改进密码分析方法，国际密码学会议（ITS），第1-5页（2014年）。

[15] Focardi，R.和Luccio，F.L.：经典密码的神经密码分析，ICTCS，第104-115页（2018）。

[16] Mishra，G.，Murthy，S.V.和Pal，S.：基于神经网络的轻量级分组密码PRESENT、和谐搜索及自然启发式优化算法分析（2019）。

[17] Gohr，A.：利用深度学习提升对Round-Reduced Speck32/64的攻击，收录于《CRYPTO》期刊第150至179页（2019年）。

[18]戈麦斯，A.N.，黄，S.，张，I.，李，B.M.，奥萨马，M.和凯撒，L.：基于离散生成对抗网络的无监督密码破解方法，CoRR，abs/1801.04883（2018）。

[19] Greydanus，S.：《使用循环神经网络学习谜题》，CoRR，abs/1708.07576（2017）。
[20]霍赫赖特，S.和施密德胡伯，J.：《长期短期记忆》，载于《神经计算》第9卷第8期，第1735－1780页（1997年）。
[21]侯博、李毅、赵虎和吴斌：基于深度学习的轮数减少DES线性攻击，ESORICS会议论文集，第131-145页（2020年）。
[22]侯志、任军和陈思：基于深度学习的SIMON32算法轮级缩减密码分析，IACR密码学电子预印本档案，第2021卷，第362页（2021年）。
[23]侯志、任军和陈思：《改进神经区分器以增强密码分析》，载于《IACR密码学电子预印本档案》第2021卷，第1017页（2021年）。
[24]胡X.和赵Y.：基于神经网络的AES明文恢复研究，《安全与通信网络》，2018年第20卷，第6868506：1-6868506：9页（2018）。
[25] Idris，M.F.、Teh，J.S.、Yan，J.L.S.和Yeoh，W.-Z.：《基于深度学习的轻量级广义Feistel分组密码S盒预测方法》，发表于《IEEE Access》第9卷，第104205－104216页（2021年）。
[26] Kimura，H.，Emura，K.，Isobe，T.，Ito，R.，Ogawa，K.和Ohigashi，T.《基于深度学习的块密码输出预测攻击》，作者：周J.、阿德普S.、阿尔卡拉兹C.、巴蒂纳L.、卡萨利奇奥E.、查托帕迪亚S.、金C.、林J.、洛西乌克E.、马朱姆达尔S.、孟W.、皮切克S.、邵J.、苏C.、王C.、扎努亚罗维奇Y.和佐努兹S.A.编辑，收录于《应用密码学与网络安全研讨会-ACNS 2022卫星研讨会》（AIBlock，AIHWS，AIoTS，CIMSS，Cloud S&P，SCI，SecMT，SiMLA），《计算机科学讲义笔记》第13285卷，第248-276页，Springer出版社（2022年）。
[27] Kingma，D.P.和Ba，J.：Adam：一种随机优化方法，ICLR（2015）。
[28] Knudsen，L.R.和Robshaw，M.：《分组密码伴侣》，信息安全与密码学，Springer出版社（2011年）。
[29]莱安德，G.：《PRESENT分组密码的小规模变体》，IACR密码学电子预印本档案，第2010卷，第143页（2010年）。
[30]李T.、Teh J.S.、Liew J.、Yan S.、Jamil N.和Yeoh W.-Z.：基于机器学习的分组密码安全性预测方法，CRYPTOLOGY（2020）。
[31]李T.R.、特H.J.S.、贾米尔N.、严J.L.S.和陈J.：基于机器学习分类器和主动S-盒的轻量级分组密码安全性评估，IEEE Access，第9卷，第X页。134052－134064（2021）.
[32]刘一、陈杰和邓乐：基于序列输出统计的无监督序列分类，NIPS，第3550-3559页（2017年）。
[33] Mantin，I.和Shamir，A.：《对广播RC4的实用攻击》，载于《快速软件加密》第152-164页（2001年）。
[34]J.：《基于深度学习的轻量级分组密码分析》，载于《安全与通信网络》2020年第20卷第3701067页（2020年）。
[35] Sutskever，I.、Vinyals，O.和Le，Q.V.：《基于神经网络的序列到序列学习》，NIPS，第3104-3112页（2014年）。
[36] Tan，C.和Ji，Q.：一种从密文识别加密算法的方法，国际密码学与网络安全会议（ICCSN），第19-23页（2016年）。
[37] Tieleman，T.和Hinton，G.：第6.5讲-RMSprop：将梯度除以其近期幅度的移动平均值，来源：COURSERA《机器学习中的神经网络》第4卷第2期，第26-31页（2012年）。
[38]王G.和王G.：改进的差分-机器学习区分器：基于机器学习的通用扩展用于差异分析，ICICS 2021，第21-38页（2021）。
[39]肖小燕、郝琦和姚德德：《神经密码分析：CPS密码的度量、方法论及应用》，IEEE DSC，第1-8页（2019年）。
[40] Yadav，T.库马尔，M.：《基于机器学习的差分密码分析通用扩展方法——差分-ML区分器》，收录于《拉丁美洲密码学会议2021》（LATINCRYPT 2021），第191－212页（2021年）。

附录

A.1 现有基于深度学习的攻击的超参数

表A.1列出了现有深度学习的超参数。

基于学习的攻击。

表A · 1基于深度学习的密码分析参数。UNK：未知，CP：密文预测加密与PR：=明文恢复。

参考文献	目标	网络类型	批量大小	初始学习率	轮	数目隐藏的图层	数目神经元
BSS08 [5]	7圈蛇形赛道（128位）	递归神经网络	取消	取消	取消	取消	1024（总和）
AAAA12 [1]	简化DES（12位）	多层前馈网络	取消	取消	7869	2	1024（总和）
DH14 [14]	简化DES（12位）	多层桩	取消	取消	取消	取消	取消
Gohr19 [17] *6	Speck32/64（32位）	ResNet	5000	滴落自0.001至0 .0001	200	12	取消
XHY19 [39]	DES（64位）	多层全连接神经网络	1000	取消	350	4	128, 128, 128, 128
XHY19 [39]	DES（64位）	级联全连接神经网络	1000	取消	350	4	128, 256, 256, 128
CY20 [10] *7	Speck32/64（32位）	ResNet	10000	取消	10	取消	取消
CY20 [10] *7	DES（64位）	ResNet	10000	取消	10	取消	取消
HLZW20 [21]型	3个DES（64位）	客户指定的普遍的神经网络	1000	取消	$5 * 10^3$	5	取消
HLZW20 [21]型	4个DES（64位）	客户指定的普遍的神经网络	1000	取消	$5 * 10^4$	5	取消
HLZW20 [21]型	5轮DES（64位）	客户指定的普遍的神经网络	1000	取消	$5 * 10^4$	10	取消
20 [34]	简化DES（8位）	多层前馈网络	取消	取消	5000	5	512, 512, 512, 512, 512
20 [34]	Speck32/64（32位）	多层前馈网络	取消	取消	5000	5	512, 512, 512, 512, 512
20 [34]	Simon32/64（32位）	多层前馈网络	取消	取消	5000	5	512, 512, 512, 512, 512
BBDC21 [6]	吉姆利-佩尔姆（384位）	MLP *8	取消	取消	20	5	296, 258, 207, 112, 160
BBDC21 [6]	ASCON-Perm.（320位）	MLP *8	取消	取消	20	4	128, 1024, 1024, 1024
BBDC21 [6]	KNOT-256（256位）	MLP *8	取消	取消	20	4	128, 1024, 1024, 1024
BBDC21 [6]	KNOT-512（512位）	MLP *8	取消	取消	20	4	256, 1024, 1024, 1024
BBDC21 [6]	CHASKEY-Perm.（128位）	ResNet *9	5000	取消	10	取消	取消
BGLMT21 [7]	13圈Speck32/64（32位）	取消	取消	取消	取消	取消	取消
BGLMT21 [7]	7、8、9、10轮Simon32/64（32位）	ResNet密集网络塞内特	5000	滴落自0.001至0 .0001	300	取消	取消
BGLMT21 [7]	10、11轮Simon32/64（32位）	SENet *10	取消	10^{-4}	10	取消	取消
BGPT21 [8] *11,*13	Speck32/64（32位）	LGBM *12	取消	取消	取消	取消	取消
BGPT21 [8] *11,*13	Simon32/64（32位）	UNK *12	取消	取消	取消	取消	取消
BGPT21 [8] *11,*13	Speck32/64（32位）	2D-CNN *12	取消	取消	取消	取消	取消
CY21 [12] *13	CHASKEY-Perm.（128位）	取消	取消	取消	取消	取消	取消
CY21 [12] *13	DES（64位）	取消	取消	取消	取消	取消	取消
CY21 [12] *13	Speck32/64（32位）	取消	取消	取消	取消	取消	取消
CY21 [13]	Speck32/64（32位）	取消	取消	取消	取消	取消	取消
CY21 [13]	Speck48/72（48位）	取消	取消	取消	取消	取消	取消
CY21 [13]	Speck48/96（48位）	取消	取消	取消	取消	取消	取消
CY21 [11]	DES（64位）	定制卷积层基于神经网络	1000	滴落自0.002至0 .0001	10	取消	取消
CY21 [11]	Speck32/64（32位）	定制卷积层基于神经网络	1000	滴落自0.002至0 .0001	10	取消	取消
CY21 [11]	当前（64位）	定制卷积层基于神经网络	1000	滴落自0.002至0 .0001	10	取消	取消
HRC21 [22]	Simon32/64（32位）	ResNet	取消	取消	100	取消	取消
HRC21 [23]	Simon32/64（32位）	ResNet	10000	取消	100	取消	取消
HRC21 [23]	Simon48/96（48位）	ResNet	10000	取消	100	取消	取消
HRC21 [23]	Simon64/128（64位）	ResNet	10000	取消	100	取消	取消
HRC21 [23]	Speck32/64（32位）	ResNet	10000	取消	100	取消	取消
HRC21 [23]	Speck48/96（48位）	ResNet	10000	取消	100	取消	取消
HRC21 [23]	Speck64/128（64位）	ResNet	10000	取消	100	取消	取消
伊蒂雅21 [25]	TWINE（64位）	全连接神经网络	32	0.001	200	4	512, 512, 512, 512
YK21 [40]	Speck32/64（32位）	多层桩	取消	取消	取消	2	1024, 1024
YK21 [40]	Simon32/64（32位）	多层桩	取消	取消	取消	2	1024, 1024
YK21 [40]	GIFT 64（64位）	多层桩	取消	取消	取消	2	1024, 1024
二战[38]21	Speck32/64（32位）	ResNet	5000	滴落自0.001至0 .0001	50	4	取消
二战[38]21	Speck48/72（48位）	ResNet	5000	滴落自0.001至0 .0001	50	4	取消
二战[38]21	Speck64/96（64位）	ResNet	5000	滴落自0.001至0 .0001	50	4	取消
KEIIOO22 [26]	3轮PRESENT（64位）（CP）	激光扫描跟踪法	250	0.001	100	6	300, 300, 300, 300, 300, 300
KEIIOO22 [26]	3轮PRESENT（64位）（PR）	激光扫描跟踪法	250	0.001	100	5	300, 300, 300, 300, 300
KEIIOO22 [26]	1个类似AES的循环（64位）（CP）	激光扫描跟踪法	250	0.001	100	4	300, 300, 300, 300,
KEIIOO22 [26]	1个类似AES的循环（64位）（PR）	激光扫描跟踪法	250	0.001	100	4	200, 200, 200, 200
KEIIOO22 [26]	2个类似TWINE的（64位）（CP）	激光扫描跟踪法	250	0.001	100	4	400, 400, 400, 400
KEIIOO22 [26]	2个类似TWINE的（64位）（PR）	激光扫描跟踪法	250	0.001	100	3	500, 500, 500
本文件	**小型设备，配备弱S盒1**	见表6和表7					
本文件	**小型PRESENT，S盒2较弱**	见表6和表8					

*6 这些参数用于基础训练流程。详情请参阅https://github.com/agohr/deep_speck

*7 https://github.com/AI-Lab-Y/NASA

*8 Baksi等人[6]尝试了基于CNN、LSTM和MLP的区分器，用于破解分组密码。他们发现，经过微调的基于MLP的区分器在准确率上表现最佳。

*9 Baksi等人[6]采用了Gohr的研究[17]中针对CHASKEY-Perm提出的机器学习模型，但将训练轮数设置为10。

*10 鲍等人[7]采用基于SIMON32/64的8轮DNN区分器，通过分阶段训练方法训练了机器学习模型。

*11 https://github.com/AnonymousSubmissionEuroCrypt2021/A-Deeper-Look-at-Machine-Learning-Based-Cryptanalysis

*12 本研究[8]的第5.1、5.5和6节中的每项工作。

*13 Benamira等人和Chen等人的研究[8]、[12]有助于理解Gohr的方法[17]。他们提出了一种高效的区分器，但这项工作并非专注于超参数。

木村隼人于2019年和2021年分别从东海大学获得电气工程学士学位和信息与电信工程硕士学位。2021年，他加入LINE公司。自2023年起，他还成为了兵库大学信息科学研究生院的博士候选人。他的研究兴趣包括密码学、信息科学等。

Keita Emura于2004年从金泽大学获得电气工程硕士学位。2004年至2006年，他在富士通北陆系统有限公司工作。2010年，他从日本科学技术振兴机构（JAIST）获得信息科学博士学位，并在该机构的高可靠性嵌入式系统技术中心担任博士后研究员，时间从2010年至2012年。自2012年起，他成为国立信息通信技术研究所（NICT）的研究员，2014年晋升为高级研究员，2021年起担任研究经理。他的研究兴趣包括公钥密码学和信息安全。2012年，他获得了IEICE颁发的SCIS创新论文奖；2016年，他荣获了IPSJ的CSS最佳论文奖；2017年，他又获得了IPSJ山下SIG研究奖；2022年，他再次获得ProvSec最佳论文奖。他是IEICE、IPSJ和IACR的成员。

石井卓纪分别于2006年、2008年和2013年在日本神户大学获得工学士、工学硕士和博士学位。2008年至2017年，他在索尼公司任职。2017年至2022年，他担任兵库大学副教授。自2023年起，他成为兵库大学教授。他的研究兴趣包括信息安全等。

他专注于安全、网络安全及网络协议领域。2017年，他荣获IWSEC 2017年度最佳海报奖，并且是IPSJ的成员。

理性和密码学。

伊藤凉马于2009年从日本国立防卫大学获得理学士学位，2015年在日本高级科学技术研究所获得理学硕士学位，并于2019年在大阪大学获得哲学博士学位。2009年至2020年期间，他在日本防卫省的日本航空自卫队工作。自2020年起，他担任国家某机构的高级研究员。

日本信息和通信技术研究所。目前的研究兴趣包括信息安全和密码学。

小川一斗分别于1987年和2008年从东京大学获得理学士和哲学博士学位。1987年，他加入日本放送协会（NHK）。自2020年起，他担任国立信息通信技术研究所的高级研究员。他的研究主要集中在视频图像处理系统和数字版权管理系统上。

大西敏弘分别于2002年、2004年和2008年从德岛大学获得工学士和工学硕士学位，并在同年从神户大学获得博士学位。2008年至2015年，他在广岛大学信息媒体中心担任助理教授。2015年至2018年，他担任东海大学的初级副教授。2018年至2023年，他担任东海大学的副教授。目前，他是东海大学的教授。他的研究兴趣包括密码学、信息安全和网络协议。2003年和2014年，他分别获得了日本电气工业标准协会ISEC组颁发的SCIS 20周年纪念奖和SCIS 2013创新论文奖。他还于2014年获得CSEC组IPSJ颁发的CSS 2014最佳论文奖，2015年获得IPSJ山下SIG研究奖，同年获得IEICE最佳论文奖。他是IPSJ和IEICE的成员。