对抗样本经典攻击和防御方法的实验复现与未来研究展望

何进全

硕 251503-25151213720

摘 要 本文系统复现了深度学习模型对抗样本领域的经典攻击与防御方法，并基于实验结果提出了未来研究方向。在攻击方法方面，成功复现了白盒攻击中的快速梯度符号法(FGSM)、基本迭代法(BIM)、投影梯度下降法(PGD)和 C&W 攻击，以及黑盒攻击中的零阶优化(ZOO)和边界攻击(Boundary Attack)；在防御方法方面，重点复现了防御蒸馏(Defensive Distillation)技术。实验结果表明：在 MNIST 数据集上，FGSM 攻击可将分类准确率从 97.53%降至 24.997%，而 C&W 攻击效果更为显著，可将准确率降至 7.72%；防御蒸馏能有效提升模型鲁棒性，使 FGSM 攻击下的准确率下降幅度从 72.24%减少至 2.32%。此外，针对大语言模型(LLM)的 PromptAttack 方法成功实现了无需梯度和微调的对抗攻击。基于对现有技术的深入分析，本文提出将未来研究聚焦于视觉-语言多模态模型的对抗鲁棒性，设计跨模态攻击与防御机制。这些工作不仅验证了经典方法的有效性，也为深入理解深度学习模型的脆弱性提供了实证基础。复现代码开源到 github：https://github.com/Surakahn/ICSTFT。

关键词 对抗样本；白盒攻击；黑盒攻击；防御蒸馏；零阶优化；边界攻击；大语言模型；多模态模型

Experimental Reproduction of Classical Attack and Defense Methods for Adversarial Examples and My Future Research Directions

Jinquan He[1)]

[1)](School of Cyber Engineering, Xidian University, City Xi'an)

Abstract This paper systematically reproduces classical attack and defense methods in the field of adversarial examples for deep learning models and proposes future research directions based on experimental results. Regarding attack methods, we successfully reproduce white-box attacks including Fast Gradient Sign Method (FGSM), Basic Iterative Method (BIM), Projected Gradient Descent (PGD), and Carlini & Wagner (C&W) attack, as well as black-box attacks including Zeroth Order Optimization (ZOO) and Boundary Attack. For defense methods, we focus on reproducing Defensive Distillation technique. Experimental results demonstrate that on the MNIST dataset, FGSM attack reduces classification accuracy from 97.53% to 24.997%, while C&W attack is more effective, reducing accuracy to 7.72%. Defensive Distillation significantly enhances model robustness, reducing the accuracy drop under FGSM attack from 72.24% to merely 2.32%. Additionally, PromptAttack for Large Language Models (LLMs) successfully implements adversarial attacks without requiring gradient information or model fine-tuning. Based on in-depth analysis of existing techniques, this paper proposes focusing future research on adversarial robustness of vision-language multimodal models and designing cross-modal attack and defense mechanisms. These efforts not only validate the effectiveness of classical methods but also provide empirical foundations for understanding the vulnerabilities of deep learning models. Github link: https://github.com/Surakahn/ICSTFT

Key words adversarial examples; white-box attack; black-box attack; defensive distillation; zeroth order optimization; boundary attack; large language models; multimodal models

提交作业日期：2025年12月29日；

*

1 绪论

在完成《信息内容安全理论与前沿技术》这门专业课程的学习之后，我深刻地认识到，当前深度学习模型所面临的对抗样本攻击是一个日益严峻且复杂的挑战 。这些由微小对抗性扰动生成的恶意输入能够轻易欺骗高度精确的神经网络，导致其做出错误判断，从而对自动驾驶、医疗诊断、金融风控等关键应用领域构成严重威胁。这种脆弱性不仅暴露了深度学习模型内部工作机制的深层问题，也揭示了其在安全可靠部署方面的巨大鸿沟。面对这一严峻形势，我深知要真正理解和应对这一前沿课题，仅仅停留在理论层面是远远不够的。必须掌握并亲手复现对抗样本领域针对深度学习模型的经典攻击与防御方法，才能真正触及问题的本质，才能弄清楚当今对抗深度学习研究领域的发展和前沿。

首先，我深入学习并复现一系列经典的对抗攻击方法，包括基于梯度的快速梯度符号法（FGSM）、基本迭代法（BIM），以及更强的投影梯度下降法（PGD）；同时，也探究了基于优化的卡林西与韦格纳（C&W）攻击；还有，一些与实际应用更加相关的黑盒攻击方法，比如基于迁移性的 Ensemble MI-FGSM 方法，基于黑盒输出 logits 的零阶优化方法 ZOO，还有真正实用部署的 Boundary Attack 方法。通过对这些方法的细致实验，我得以直观地观察到它们生成对抗样本的过程，并在此过程中不断调整参数、分析结果，从而深度理解每种方法的独特动机、实现细节及其内在的局限性。例如，FGSM 虽然简单高效，但其单步攻击的特性使其极易被简单的防御措施所规避；而 BIM 通过迭代的方式提升了攻击效果，但仍可能陷入局部最优；最终，PGD 通过引入随机初始化和多次重启，被数学证明为“最强的一阶攻击方法”， 成为了评估模型鲁棒性的黄金标准，如果一个神经网络模型能抵御 PGD 的攻击，那么它理论上也应该能抵御当时所有其他一阶攻击方法的威胁。这一系列的实验不仅锻炼了我的实验设计能力和动手编码能力，更重要的是，它让我初步掌握了在科研活动中识别一个方法不足之处，并思考提出改进方案的一般性方法论。

在深入理解这些经典攻击手段之后，我转向了复现对抗样本防御策略。我了解到，对抗防御的发展与攻击方法的演进几乎同步。基于课程知识，我[1]梳理了三大类防御思想：预处理防御、处理中防御和后处理防御。预处理防御旨在污染源头进行净化，例如通过图像压缩重建（ComDefend）或高阶表示指导去噪器（HGD）来去除添加的扰动。处理中防御则直接修改模型本身，其中最具代表性的是对抗性训练，即让模型在训练过程中主动接触对抗样本以增强其鲁棒性，或者更改模型的损失函数。后处理防御则是在模型输出之外增加一个额外的检测或修正环节，如利用领域知识增强机器学习流程（KEMLP）来验证主模型的预测结果。在这个过程中，我特别关注那些既有开源代码库又拥有配套数据集的经典防御方法，以便进行严谨的实验验证和性能对比。例如，HGD 提供了公开的 PyTorch 实现和在 ImageNet/CIFAR-10 上的基准支持，而 RobustBench 这样的标准化基准则为评估模型的鲁棒性提供了统一的平台和 AutoAttack 这一强大的评估工具包。通过这样的实践，我不仅加深了对各类防御技术优劣的理解，也学会了如何在一个更广阔的视角下，将具体的攻击方法与相应的防御策略联系起来，形成一个完整的攻防认识。

最后结合我当前的研究方向，探索了 LLM 提示词对抗攻击领域的一种攻击方法 A Prompt-Based Adversarial Attack，并成功复现获得实验结果。

本报告将详细阐述这一学习实践过程中对经典方法的理解，并展现核心实验结果与复现过程。

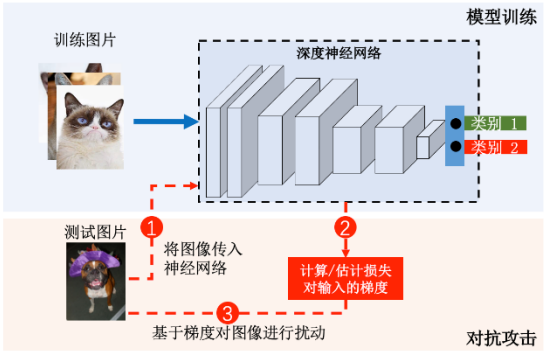

图 1 对抗样本攻击的一般流程[1]

1.1 白盒攻击：基于梯度和基于优化

白盒攻击是当前深度学习对抗攻击研究的基础，它理想地假设攻击者拥有对目标深度学习模型完全的知识，包括其架构、权重参数、激活函数乃至损失函数的具体形式。在这种白盒攻击的设定

1. 人工智能数据与安全，https://ai-data-model-safety.github.io/source/chap6.html 2025,12,25)

下，研究人员探索出一系列精巧且强大的攻击方法，这些方法构成了后续攻击策略演进的思想基础。白盒攻击又可以分为两大流派：一类是基于梯度的攻击，它们直接利用损失函数关于输入的梯度来寻找最优的扰动方向；另一类是基于优化的攻击，它们将对抗攻击问题重新表述为一个带约束的数学最优化问题，追求生成扰动幅度更小、视觉上更不易察觉的对抗样本。

基于梯度的对抗样本攻击源于 Goodfellow 等人在 2014 年提出的快速梯度符号法（Fast Gradient Sign Method, FGSM），这是最早也是最著名的针对深度学习模型的对抗攻击算法之一。FGSM 的核心思想极其简洁：为了最大化模型的分类损失，应该沿着损失函数关于输入图像梯度的方向施加扰动。具体而言，给定一张干净的图像x及其真实标签y，FGSM 在原图上添加一个精心构造的扰动 $\delta\delta$，生成对抗样本$x_{adv} = x + \delta$。这个扰动的计算公式为$\delta = \epsilon \cdot sign(\nabla_x J(\theta, x, y))$，其中$J$是损失函数，$\theta$是模型参数，而 ϵ 是一个控制扰动大小的超参数，通常取值很小以确保扰动在人类视觉感知范围内。因此，最终的对抗样本可以表示为 $x_{adv} = x + \epsilon \cdot sign(\nabla_x J(\theta, x, y))$ 。FGSM 算法的实现在于对损失函数在输入点 x 附近进行一阶泰勒展开的线性近似，即$J(\theta, x + \delta) \approx J(\theta, x) + \nabla_x J(\theta, x)^T \delta$。在满足 L_∞ 范数约束 $\| \delta \|_\infty \leq \epsilon$ 的条件下，为了让损失函数 J 最大化，扰动 δ 的方向应与梯度 $\nabla_x J$ 相同。由于 FGSM 方法只关心方向而非大小，因此用梯度的符号 $sign(\nabla_x J)$ 来计算梯度方向。尽管 FGSM 单步计算速度快、无需迭代，但其根本局限性也十分明显，它是一次性、单步式的攻击，容易受到模型决策边界上“平坦区域”的影响，导致攻击效果不佳且生成的对抗样本可迁移性较差。

为了克服 FGSM 的单一性和低效性，研究人员提出了基本迭代法（Basic Iterative Method, BIM），也被称为迭代快速梯度符号法（Iterative Fast Gradient Sign Method, I-FGSM）。BIM 是对 FGSM 最直观也是有效的改进，它将 FGSM 的单步大幅扰动分解成多个步长较小的迭代更新。其核心思想是贪婪地沿着梯度方向逐步移动，每一步都使损失函数尽可能增大，但又不超出预设的扰动预算。BIM 的迭代更新公式为 $x_{t+1} = Clip_{x,\epsilon}(x_t + \alpha \cdot sign(\nabla_x J(\theta, x, y)))$，其中 $x_0 = x$ 是初始的干净图像，t 是迭代次数，α 是一个小的步长（learning rate），ϵ 是总的扰动预算，而 $Clip$ 操作则确保了每一步更新后的图像 x_{t+1} 始终与原始图像 x 的差异不超过ϵ（即在L_∞ 球内）[2]。BIM 的成功在于它允许攻击者在决策边界附近进行更精细的探索，显著提升了攻击的成功率。然而，BIM 仍然存在问题，它从固定的初始点 x 开始，容易陷入局部最优，导致攻击强度受限，不能找到最佳的对抗样本。BIM 的有目标攻击版本是 ILLC 迭代最小可能类攻击，试图将对抗样本诱导模型错误分类目标变成最不可能的类别。

为了从根本上解决 BIM 的局部最优问题，Madry 等人在 2017 年正式提出了投影梯度下降法（Projected Gradient Descent, PGD）。PGD 是 BIM 的进一步增强版本，其核心创新在于随机初始化和多重重启。与 BIM 固定从原图开始不同，PGD 首先从一个均匀分布在以 x 为中心、半径为 ϵ 的 L_∞ 球内的随机点 x_0 开始。然后，在每个起始点上执行类似于 BIM 的多步迭代更新，最后从所有起始点产生的对抗样本中选择使损失函数最大的那个。这种策略极大地增强了攻击跳出局部最优的能力，使得攻击路径更加多样化和难以预测。PGD 的完整迭代公式为$x_{t+1} = \Pi_{x,\epsilon}(x_t + \alpha \cdot sign(\nabla_x J(\theta, x, y)))$，其中$\Pi_{x,\epsilon}$是一个投影操作，它将向量 v 投影到以 x 为中心、半径为 ϵ 的 L_∞球内，其数学定义为对每个像素 i 进行独立的裁剪：$(\Pi_{x,\epsilon}(v))_i = clip(v_i, x_i - \epsilon, x_i + \epsilon)$[3]。PGD 被数学证明是目前最强的一阶攻击方法。这意味着，如果一个模型声称自己对 PGD 攻击是鲁棒的，那么它就应该对当时已知的所有其他一阶攻击方法都具有鲁棒性。

如果说基于梯度的方法是从几何角度（最大化 $_2$损失）来寻找攻击方向，那么基于优化的方法则是从数学规划的角度来寻求更优解。其中，Carlini 与 Wagner 在 2017 年提出的 C&W 攻击是这一领域的里程碑之作。C&W 攻击不再局限于最大化损失，而是将对抗样本生成问题转化为一个更为通用的、带有约束的最优化问题。其核心目标是最小化扰动的大小，同时保证生成的样本能够成功欺骗模型。对于一个无目标攻击（untargeted attack），C&W 攻

1. Kurakin A, Goodfellow I J, Bengio S. Adversarial examples in the physical world[M]//Artificial intelligence safety and security. Chapman and Hall/CRC, 2018: 99-112.

2.Madry A, Makelov A, Schmidt L, et al. Towards deep learning models resistant to adversarial attacks[J]. arXiv preprint arXiv:1706.06083, 2017.

击的目标函数可以形式化为：$\min_{\delta} \| \delta \|_2^2 + c \cdot f(x+\delta)$，$subject\ to\ 0 \le x_i + \delta_i \le 1\ for\ all\ i$，其中第一项 $\| \delta \|_2^2$是正则化项，旨在最小化扰动的欧几里得范数，从而使对抗样本在视觉上更接近原始图像。第二项$c \cdot f(x+\delta)$是一个代理损失函数（surrogate loss），用于惩罚未能成功攻击的情况。这里的$f(x')$函数被设计用来衡量攻击是否成功，它定义为$f(x') = max(0, Z_t(x') - max_{j \ne t} Z_j(x'))$，其中$Z(x')$是模型最后一个隐藏层的输出（logits），$t$是真实类别，而$c$是一个非负权重超参数，用于平衡两项。当且仅当$f(x') \le 0$时，意味着模型对$x'$的预测置信度不足以将其正确分类为目标类别$t$，即攻击成功。超参数$c$的选择至关重要，因为它决定了攻击的难度和扰动的大小。

一个较大的c意味着更倾向于生成一个非常确定的错误分类，即使需要较大的扰动；反之，较小的c则鼓励生成一个扰动更小但可能不那么确定的对抗样本。在实际应用中，c的值通常不是固定的，而是通过二分查找的方式来确定，以找到在满足攻击成功条件下的最小扰动对应的c值。此外，为了方便优化器处理无界的输入空间，C&W 攻击还采用了一个巧妙的变换，令$w = \frac{1}{2}(arctanh(x)+1)$，将输入$x$映射到$(0,1)$区间内，从而避免了显式的边界约束。C&W 攻击的强大之处在于，它不仅能生成视觉上更逼真的对抗样本，而且对许多依赖于梯度掩蔽（gradient masking）的防御机制具有更强的抵抗力，因为它可以直接优化攻击目标，而不是盲目地按梯度方向添加扰动。

表 1 白盒对抗攻击方法分类

攻击方法	分类	核心思想	数学形式
FGSM	基于梯度	一阶泰勒级近似，沿梯度方向施最大扰动	$x_{adv} = x + \epsilon \cdot sign(\nabla_x J(\theta, x, y))$
I-FGSM	基于梯度	将 FGSM 单步攻击分解为多次小步迭代	$x_{t+1} = Clip_{x,\epsilon}(x_t + \alpha \cdot sign(\nabla_x J(\theta, x, y)))$
PGD	基于梯度	BIM 基础上引入随机初始化、多重重启和像素级裁剪	$x_{t+1} = \Pi_{x,\epsilon}(x_t + \alpha \cdot sign(\nabla_x J(\theta, x, y)))$
C&W Attack	基于优化	将对抗样本攻击转化为带约束的优化问题	$\min_{\delta} \| \delta \|_2^2 + c \cdot f(x+\delta)$

1.2 黑盒攻击：基于迁移，基于分数和基于决策

白盒攻击为对抗攻击的研究奠定了坚实的理论基础，黑盒攻击则致力于模拟现实世界中最普遍、最困难的攻击场景。在真实的部署环境中，攻击者往往无法访问目标模型的内部结构、权重或梯度信息，他们只能通过有限的交互——即输入样本并获取模型的输出（如预测的类别或概率分数）——来发起攻击。这种信息不对称迫使研究人员开发出一系列更为精巧和间接的攻击策略，这些策略的核心思想不再是直接访问梯度或者利用模型信息进行优化，而是通过间接方式来“猜测”梯度，或者仅依赖于模型最原始的决策和输出的标签信息进行对抗攻击。黑盒攻击的演进推动了对抗攻击理论向着更通用、更实用的方向发展，为对应的防御方法提供可以借鉴的思路。

基于迁移性的攻击是黑盒攻击中最早的策略之一。其核心思想是，一个在特定数据集上训练好的深度神经网络，其学到的特征和决策边界在某种程度上具有共性。因此，即使攻击者不知道目标黑盒模型的具体细节，也可以训练一个或多个替代模型（surrogate models），这些替代模型的架构和训练数据与目标模型相似，但攻击者拥有其完全的白盒访问权限。一旦替代模型训练好，攻击者就可以轻松地计算出替代模型的梯度，并利用这些梯度来生成对抗样本。然后，这些对抗样本会被送入目标黑盒模型，期望它们也能成功欺骗后者。这种方法的[3]可行性源于这样一个观察：对抗样本本质上是利用了深度神经网络中存在的某种“通用漏洞”，而不仅仅是针对某个特定模型的特异性弱点。早期的迁移攻击就是基于单个替代模型，但很快研究者们发现，使用多个不同的替代模型进行集成攻击（Ensemble Transfer Attacks）能够产生更稳定、更

1.MI-FGSM Dong Y, Liao F, Pang T, et al. Boosting adversarial attacks with momentum[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9185-9193.

具通用性的对抗样本，因为集成可以平滑掉单个模型的随机性，得到一个更可靠的梯度估计。

在集成迁移攻击的基础上，研究人员开始借鉴白盒攻击中迭代优化的思想来进一步提升黑盒攻击的效果。动量迭代法迁移攻击（Momentum Iterative Fast Gradient Sign Method, MI-FGSM）在BIM的基础上引入了动量（momentum）概念，其核心动机是为了稳定梯度估计并帮助攻击跳出局部最优，从而提升对抗样本的可迁移性。在传统的BIM中，每一步的梯度都是独立计算的，这可能导致攻击路径摇摆不定。MI-FGSM 通过维护一个累积的梯度缓冲区 g_t，使得新的梯度更新不仅考虑当前步的梯度，还保留了历史梯度的信息。其具体的更新规则是：首先计算当前步的归一化梯度$u_t = \frac{\nabla_x J(\theta,x_t,y)}{\|\nabla_x J(\theta,x_t,y)\|_1}$，然后更新累积梯度 $g_{t+1} = \mu \cdot g_t + u_t$，其中$\mu$是动量因子，通常设置为0.9。最后，根据累积梯度进行更新：$x_{t+1} = Clip_{x,\epsilon}(x_t + \alpha \cdot sign(g_t + 1))$ [4]。动量的存在使得梯度更新更加平滑和连贯，攻击者能够更有效地朝着一个有利的方向持续前进，而不是在决策边界附近来回震荡。这种策略极大地提升了对抗样本在跨模型迁移时的成功率，成为当时非常强大且流行的黑盒攻击方法。集成动量迭代法迁移攻击（Ensemble MI-FGSM）在计算累积梯度时，对多个替代模型的梯度进行加权平均，从而获得一个更加鲁棒和准确的攻击方向。

然而，即便有了替代模型，迁移性攻击仍然面临一个问题，替代模型的梯度可能并不能完美地反映目标黑盒模型的梯度，尤其是在两者架构差异较大时。此外，随着防御技术的发展，一些防御机制可能会刻意设计成让白盒攻击的梯度信息变得不可靠，这种防御性设计被称为“梯度掩蔽”（gradient masking）。这促使研究人员探索完全不依赖替代模型和梯度信息的攻击方法。

基于分数的对抗攻击（Score-based Attack）完全不依赖于替代模型的梯度信息，而是依靠多次查询黑盒模型获得输出分数（logits）来估计黑盒模型的梯度。零阶优化攻击（Zeroth-Order Optimization, ZOO）是该领域的开创性工作。ZOO 的核心思想是，当攻击者无法获取梯度时，可以通过多次查询黑盒模型来估算梯度。具体来说，攻击者可以在当前输入 x 的每一个坐标方向上，分别向两个很近的点 $x + he_i$ 和 $x - he_i$ 发送查询，其中 e_i 是第 i 个单位向量，h 是一个极小的扰动。通过比较这两个点的输出分数（logits）的差异，攻击者可以利用有限差分法来近似地计算出梯度在该方向上的分量，即 $\nabla_{x_i} L(x) \approx \frac{L(x+he_i)-L(x-he_i)}{2h}$。ZOO 受到了 C&W 攻击的启发，它同样关注于优化一个目标函数，但将优化过程扩展到了纯查询的场景。通过反复进行这样的查询和梯度估算，ZOO 可以引导优化算法（如Adam）朝着最大化损失的方向迭代更新，从而生成对抗样本。ZOO 的成功表明，即使没有显式的梯度信息，攻击者依然可以通过查询分数的方式重建优化方向，实现了对黑盒模型的有效攻击。

基于决策的攻击则更进一步，它甚至不需要模型返回任何概率分数。边界攻击（Boundary Attack）是这一流派的典型代表，只需知道最终的分类标签即可。这类攻击方法被认为最接近现实应用场景，因为它完全符合严格黑盒的假设：攻击者除了能控制输入和读取输出标签外，对模型内部一无所知。Boundary Attack 的核心想法是对抗样本必然位于模型决策边界的附近。因此，攻击者可以通过一系列的查询来“绘制”出这张决策边界图。其算法具体步骤：首先，攻击者需要一个初始的对抗样本 x_{adv}。如果没有，可以通过其他方法（如 ZOO）生成一个，然后，攻击过程进入循环：在当前对抗样本 x_{adv}的基础上，攻击者会随机生成一个在干净样本空间内的点 x_{clean}，并计算它们之间的向量距离 $d = \| x_{clean} - x_{adv} \|$。接着，攻击者会从 x_{adv} 向 x_{clean} 的方向上，沿着这条直线进行一系列的小步长移动，直到找到一个点 x'，使得 x' 的分类结果发生了改变（即不再是原来的对抗类别）。这个新点 x' 就是攻击的一个成功候选。通过不断地重复这个过程，攻击者可以逐步将对抗样本推向更靠近决策边界的区域，或者在边界上“行走”。边界攻击的关键在于其查询策略，它巧妙地利用了模型的离散决策反馈来指导连续空间的搜索，是一种典型的蒙特卡洛方法的应用。尽管 Boundary Attack 在查询效率上可能不如基于分数的攻击，但它在信息需求上的极致简化，使其成为评估模型鲁棒性的一个重要补充维度，尤其是在防御机制可能屏蔽梯度信息的情况下，成为最靠近实用场景的黑盒攻击方法。

表 2 黑盒对抗攻击方法分类

攻击方法	类型	数学形式	局限性
集成迁移	基于迁移	1.训练多个架构/数据不同的替代	效果依赖于替代模型的质量和多

		模型。2.对每个模型计算梯度。3.加权平均梯度，生成对抗样本。	样性
MI-FGSM	基于迁移	$g_{t+1} = \mu g_t + \frac{\nabla_x J}{\lvert\nabla_x J\rvert_1}, x_{t+1} = Clip(x_t + \alpha \cdot sign(g_{t+1}))$	仍需替代模型，动量参数 μ 需要调优
ZOO	基于分数	1.初始化 x。2.循环：a.对每个坐标i，查询 $x \pm he_i$ 的分数。b.估算梯度 $\nabla L \approx \frac{L(x+he_i)-L(x-he_i)}{2h}$。c. 使用梯度更新 x。	查询次数非常多，效率较低
Boundary Attack	基于决策	1. 获取初始对抗样本 x_{adv}。2.循环：a. 随机生成干净点 x_{clean}。b. 从 x_{adv}向x_{clean}移动，找到分类变化的点。c. 更新 x_{adv}。	查询效率相对较低，收敛速度慢

1.3 预处理防御

预处理防御（Pre-processing Defenses）是三种主流防御范式（预处理、处理中、后处理）中最直观的一种，其核心思想是在对抗样本进入主模型之前，对其进行某种形式的“净化”或“转换”，旨在移除或削弱其中的对抗性扰动，从而降低模型受到攻击的可能性。这类方法的优点在于其灵活性和普适性，它们通常作为独立的模块，可以附加在任何现有的分类器之前，而无需对模型本身进行重新训练或修改。然而，预处理防御也面临着巨大的挑战：如何在有效去除对抗性扰动的同时，最大限度地保留原始图像中有用的语义信息。对抗性扰动通常是精心设计的、针对特定模型的微小信号，传统的图像处理技术往往难以区分这些有害信号与自然图像中的高频噪声或纹理，导致输入信号的质量降低，甚至可能因为破坏模型赖以工作的正常模式而导致性能下降。

图像去噪是预处理防御领域的一个核心分支，其目标是直接从输入图像中恢复出原始的干净图像。传统的去噪方法，如高斯滤波器、中值滤波器、小波去噪等，因其缺乏针对性，通常难以应对具有对抗性的噪声攻击。为了克服这一局限，研究人员开始探索基于深度学习的去噪器。其中，高阶表示指导去噪器（High-Level Representation Guided Denoiser, HGD）方法认识到对抗性扰动主要影响的是模型的低级特征（如像素值），而高级语义特征（如物体的形状、类别）在对抗样本中往往是相对稳定的。因此，与其在像素空间盲目地尝试修复图像，不如利用一个预先训练好的、干净且鲁棒的教师模型来提供指导。HGD 的具体实现是训练一个去噪自编码器（denoising autoencoder），其训练目标不再是简单地重构输入图像（即最小化$\| D(E(x)) - x \|_2$），而是要求自编码器的中间层特征（通常是倒数第二层卷积层的输出）与教师模型在该输入上的对应层特征尽可能一致。通过这种方式，去噪器被强制学习如何将输入映射到一个与教师模型认为的“干净”表征空间对齐的状态，从而能够有效地剥离掉对抗性扰动，同时保留高级语义信息[5]。HGD 的成功证明了利用模型内部知识来指导去噪是可行的，并且对 FGSM、PGD 甚至 C&W 等强攻击都表现出了一定的防御效果。在此基础上，双自编码器去噪器（Dual Autoencoder Denoiser, DAED）进一步进行了改进，它采用了一个双编码器-双解码器的对称结构，通过引入双重约束来放大良性与对抗性样本之间的区别：一个是相对重建约束，另一个是高阶表示差异约束（logits pairing loss）。实验证明，DAED 在多种白盒和黑盒攻击下，尤其是在 PGD 攻击下，表现优于 HGD 和其他基线方法。

最近的研究指出，HGD 这类方法存在一个潜在缺陷：它们可能过度依赖于外部训练数据中图像的

1. Liao F, Liang M, Dong Y, et al. Defense against adversarial attacks using high-level representation guided denoiser[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 1778-1787.

2. Niu Z H, Yang Y B. Defense against adversarial attacks with efficient frequency-adaptive compression and reconstruction[J]. Pattern Recognition, 2023, 138: 109382.

3. Li Y, Luo X, Wu W, et al. LRCM: Enhancing Adversarial Purification Through Latent Representation Compression[J]. IET Computer Vision, 2025,

统计先验。例如，ComDefend 和 HGD 等方法通过在大量干净和对抗样本对上进行训练来学习一个“净化”函数，但如果对抗样本中的扰动分布与训练集中出现过的模式不匹配，这些方法就可能失效。FACE（Frequency-Adaptive Compression and Reconstruction）的研究系统地分析了消除式防御（elimination-based defenses），发现即使保留少量（如 20%）的对抗性扰动也会导致模型准确率从 79%急剧下降到 17%，这表明扰动的能量广泛分布在图像的不同频率成分中[6]。这一发现揭示了 ComDefend 等方法失败的核心原因——它们可能只是随机地消除了部分无害的冗余信息，同时也误伤了对模型决策至关重要的信号。为了应对这一挑战，一些新兴的预处理方法转向了更底层的机制。Latent Representation Compression Method (LRCM) 不再依赖外部先验，而是将压缩思想引入到模型自身的潜在空间。它使用一个受 U-Net 启发的自动编码器架构，并结合率失真（Rate-Distortion）理论来平衡重建质量和压缩效率。通过在潜在空间进行可微分的软量化（Differentiable Soft Quantisation），LRCM 能够有效地去除对抗性扰动，同时保留核心特征[7]。实验结果显示，LRCM 在 PGD 攻击下的表现显著优于 ComDefend 和 JPEG 压缩等基线方法，尤其是在 ImageNet 等高分辨率数据集上。另一种极具前景的思路是利用深度图像先验（Deep Image Prior, DIP）。DIP 方法不依赖任何外部训练数据，而是对每个输入图像单独运行一个生成器，通过拟合该图像本身来提取其内在的结构先验[8]。对抗样本在优化初期容易被拟合，但在后期却难以拟合，而干净图像则相反。通过这种输入特定的重建过程，DIP 能够有效地区分并过滤掉对抗性扰动，其泛化能力也因此得到了提升。

除了图像去噪，图像平滑（Image Smoothing）是另一类重要的预处理防御策略，其核心思想是通过在输入数据中加入随机噪声来增加攻击者的不确定性，尤其适用于防御基于查询的黑盒攻击。随机噪声防御（Randomized Noise Defense, RND）的设置非常巧妙：当一个黑盒攻击者试图通过查询来确定对抗样本的边界时，每次查询都会接收到一幅经过随机噪声注入的图像。由于攻击者无法预测每次注入的确切噪声，其查询结果将是不确定的，这会严重干扰基于梯度估算的攻击算法（如 ZOO），因为这些算法依赖于对扰动响应的一致性观察[9]。RND 的具体做法是在将输入送入主模型之前，从一个标准高斯分布中采样一个噪声向量，将其叠加到输入图像上。即使是相同的原始输入，每次查询也可能得到不同的噪声版本，从而导致模型输出的波动。这种方法的理论基础来自于随机化平滑（Randomized Smoothing），该理论为随机化防御提供了可证明的鲁棒性保证。研究表明，只要噪声的方差足够大，就可以在一定程度上保证模型的预测结果不会因微小的对抗扰动而改变。然而，随机化平滑也面临着所谓的“维度诅咒”（curse of dimensionality）问题，即对于高维输入（如高分辨率图像），其可证明的鲁棒半径会随着输入维度的增加而急剧缩小，这限制了其在 ImageNet 等大规模数据集上的应用。尽管如此，RND 作为一种轻量级的防御机制，在保护 API 服务和防御黑盒查询攻击方面仍然具有重要的实用价值。

1.4 处理中和后处理防御

处理中防御的核心理念是在模型的训练阶段就引入对抗鲁棒性的考量，通过修改损失函数、优化目标或网络结构，让模型在学习分类任务的同时，也学会如何更好地处理对抗性扰动。

早期的处理中防御方法防御蒸馏（Defensive Distillation），曾一度备受瞩目。防御蒸馏通过知识蒸馏（Knowledge Distillation）的思想，用一个大型的“教师”模型的软标签（soft labels）来训练一个小型的“学生”模型，从而平滑学生的决策边界，使其对输入的微小变化不那么敏感。具体来说，教师模型使用一个较高的温度 T 来生成 softmax 输出，这些输出包含了类别间的相对置信度信息。学生模型则学习模仿这些平滑的概率分布，而不是硬性的 one-hot 标签。最初的研究声称，防御蒸馏能将攻击成功率从 95%降至 0.5%，取得了惊人的效果。然而，后续的研究迅速发现，这种看似强大的防御实际上是基于“梯度掩蔽”（gradient masking），而非真正的鲁棒性提升。梯度掩蔽是指防御机制通过某种方式使得模型的梯度变得非常小或趋于零，从而阻止了基于梯度的攻击方法（如 FGSM，BIM 等）找到有效的攻击方向。然而，这并不意味着模型是鲁棒的，因为攻击者总能找到绕过这种掩蔽的方法。

19(1): e70030.

4. Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9446-9454.

研究人员随后提出了一系列专门针对防御蒸馏的白盒攻击，这些攻击通过更复杂的优化技巧，能够有效地“穿透”梯度掩蔽，最终实现了 100%的攻击成功率，在评估防御方法时，不能仅仅依赖于表面的准确率指标，而必须采用更严格的、能够检验模型是否存在梯度掩蔽的评估方法，这直接催生了后来自适应攻击（adaptive attacks）的研究热潮。

处理中防御的对抗性训练（Adversarial Training）无疑是最重要、最主流的方法，至今仍被认为是提升模型鲁棒性的最有效途径之一。对抗性训练这种方法认为与其被动地等待攻击发生，不如在训练过程中主动地将模型暴露于各种可能的对抗样本之下，让模型天生就具有针对对抗样本的鲁棒性。其标准流程是，在每一次训练迭代中，首先利用 PGD 等强大的白盒攻击算法，为当前批次的数据生成一批对抗样本，然后将这些对抗样本及其对应的标签一同加入到训练集中，一起计算损失并更新模型参数。通过这种方式，模型被迫学习到一个更加平滑、决策边界更加模糊的特征空间，使得微小的对抗扰动难以引起分类结果的剧烈变化[10]。对抗性训练在实践中得到了充分验证，它如今已经成为许多 SOTA 鲁棒模型的基础。然而，对抗性训练是有相应代价的。最大的问题是“准确性-鲁棒性权衡”（accuracy-robustness trade-off），即在提升模型在对抗样本上的准确率的同时，往往会导致其在干净测试数据上的标准准确率有所下降。为了解决这一问题，研究者们提出了多种改进方案。TRADES（Theoretically Principled Trade-off）通过在损失函数中引入 KL 散度正则项，来鼓励模型在干净样本和其对应的对抗样本上做出相似的预测，从而在一定程度上缓解了权衡效应，IMA（Increasing-Margin Adversarial Training）提出了一种动态调整 margin 的策略，通过在训练过程中精确估计每个样本到决策边界的距离，来更高效地生成对抗样本，最终在多个数据集上实现了较高的鲁棒性和标准准确率[11]。

后处理防御不直接修改主模型，而是在模型的预测结果之外增加一个独立的、辅助性的防御模块。这种方法的最大优点是其灵活性和兼容性，它可以作为一个“插件”附加到任何现有的、甚至是第三方提供的模型之上，而无需重新训练或获取模型内部信息。知识增强机器学习流程（Knowledge Enhanced Machine Learning Pipeline, KEMLP）是后处理防御中一个极具创新性的例子。在许多现实世界的应用中，除了模型本身的预测结果外，还存在着大量的领域知识（domain knowledge），这些知识通常以逻辑规则的形式存在[12]。例如，在道路标志识别任务中，我们知道红色的标志通常是禁止标志，而三角形的标志通常是警告标志。KEMLP 的核心思想就是将这些先验的逻辑关系和领域知识整合到防御体系中。KEMLP 的具体实现是一个概率图模型，它将主模型的预测结果与多个辅助弱模型的结果进行融合，并通过一个包含逻辑规则的贝叶斯网络来评估整个预测系统的可信度。如果主模型的预测与这些先验逻辑相悖（例如，模型将一个红色的圆形标志识别为“停车”，但根据规则这应该是“减速慢行”），KEMLP 就会拒绝或修正主模型的输出。KEMLP 的优势在于其强大的解释性和泛化能力，它不仅能够防御已知的 L_p 范围内的对抗攻击，还能应对物理攻击、未知攻击类型以及自然环境中的各种退化，同时保持很高的标准准确率。

除了 KEMLP 这样基于知识推理的后处理防御，还有一些基于数据分布的后处理方法。深度 k-近邻（Deep k-Nearest Neighbor, DkNN）方法通过在模型各层的特征空间中，为每个输入样本寻找其在训练集中的 k 个最近邻，来衡量该样本的“非同寻常性”（non-conformity）[13]。一个正常的干净样本在其特征空间中应该能找到很多与之相似的邻居，而一个对抗样本则很可能远离所有训练样本，处于一个“孤岛”状态。DkNN 通过计算样本与其邻居标签分布的 p-value 来量化其非同常性，从而实现对抗样本的检测。DkNN 的一个显著优点是其无需重新训练，可以作为一种“即插即用”的防御模块添加

1. Qin Z, Fan Y, Zha H, et al. Random noise defense against query-based black-box attacks[J]. Advances in Neural Information Processing Systems, 2021, 34: 7650-7663.

2. Sui Chenhong， Wang Ao， Zhou Shengwen， Zang Ankang， Pan Yunhao， Liu Hao， Wang Haipeng. 2023. A survey on adversarial training for robust learning. Journal of Image and Graphics，28(12):3629-3650 DOI：10.11834/jig.220953.

3. Ma L, Liang L. Increasing-margin adversarial (IMA) training to improve adversarial robustness of neural networks[J]. Computer methods and programs in biomedicine, 2023, 240: 107687.

1. Gürel N M, Qi X, Rimanic L, et al. Knowledge enhanced machine learning pipeline against diverse adversarial attacks[C]//International Conference on Machine Learning. PMLR, 2021: 3976-3987.

2. Papernot N, McDaniel P. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning[J]. arXiv preprint arXiv:1803.04765, 2018.

到任何预训练模型之上。实验证明，将 DkNN 与对抗性训练相结合，可以显著提升模型的鲁棒性，尤其是在面对 PGD 等强攻击时，准确率可以得到大幅提升。

1.5 PromptAttack与未来研究展望

PromptAttack 是一种专为大语言模型（Large Language Models, LLMs）设计的、基于提示（prompt-based）的对抗样本攻击方法。其核心目标是高效地评估大型语言模型的对抗鲁棒性，特别是在那些计算资源昂贵、无法直接进行梯度计算或微调的闭源模型（如 GPT-3.5、PaLM）上。

PromptAttack 属于一种间接的、基于提示工程（Prompt Engineering）的白盒或灰盒攻击。它不直接修改输入的嵌入向量（embedding）或词元（token），而是通过精心构造一个攻击性提示（adversarial prompt），诱导目标 LLM 自行生成一个对抗性输出，该输出随后被用作攻击自身的武器。这种方法巧妙地利用了 LLM 强大的文本生成和指令遵循能力，将对抗攻击问题转化为一个提示设计问题，从而避开了对模型内部参数和梯度的直接依赖，使其在评估闭源 API 模型时具有极高的实用性和可扩展性。

在深入学习和实践了从经典梯度攻击（FGSM/PGD）到前沿 LLM 攻击（PromptAttack）的技术脉络后，我的未来研究方向将聚焦于一个更具挑战性且意义重大的交叉领域：跨模态的深度学习模型对抗鲁棒性。

具体而言，我希望探索在 “视觉-语言”多模态大模型（如 CLIP, Flamingo, GPT-4V）上构建更强大、更隐蔽的对抗攻击。与纯粹的文本或图像攻击不同，多模态模型的决策过程融合了来自不同感知通道的信息，这既带来了新的脆弱点，也使得防御更为复杂。

借鉴 PromptAttack 的思想，设计能够同时操控图像和文本输入的多模态提示（Multimodal Prompt）。例如，通过在文本提示中加入特定的诱导性指令，同时对配对图像施加人眼不可见的物理扰动（如微小的补丁），来协同欺骗多模态模型。这种攻击将模拟更真实、更复杂的物理世界威胁，如针对自动驾驶感知系统或智能安防摄像头的欺骗。

然后针对上述攻击，我将致力于学习研究一种新的防御方法。该范式将结合 KEMLP 的逻辑规则思想与 HGD 的表示引导思想。一方面，利用任务相关的领域知识（如“一个交通标志的文本描述必须与其视觉外观一致”）来构建一个跨模态的可信度验证器；另一方面，利用一个在干净多模态数据上预训练的、鲁棒的教师模型来指导对输入的联合去噪和净化。最终目标是构建一个抵御多模态对抗扰动，并且在多种下游任务上保持高准确率的多模态鲁棒系统。

2 白盒对抗样本攻击的实验复现

2.1 FGSM快速梯度符号法

2.1.1 FGSM 方法概述

快速梯度符号法（Fast Gradient Sign Method, FGSM）由 Ian J. Goodfellow、Jonathon Shlens 和 Christian Szegedy 于 2015 年在 ICLR 会议上首次提出，是对抗样本研究领域中最具奠基性的工作之一。FGSM 是一种典型的白盒攻击（White-box Attack）方法，其攻击场景假设攻击者对目标深度学习模型拥有完全的知识，包括模型的完整架构、所有参数权重、激活函数以及所使用的损失函数。在此理想化但信息充分的威胁模型下，FGSM 的目标是高效地生成对抗样本——即通过对原始干净输入数据（如一张图像）施加一个微小但经过精心计算的扰动，使得模型对其输出错误的分类结果，而该扰动在人类感知中几乎无法察觉。FGSM 的核心优势在于其计算效率极高，仅需一次模型前向传播和一次反向传播即可完成对抗样本的生成，这使其成为早期对抗训练和模型鲁棒性评估的实用工具。

2.1.2 产生 FGSM 的想法和动机

在 FGSM 提出之前，深度学习社区普遍认为模型的错误主要源于训练数据不足或模型复杂度不够，而对抗样本的发现则挑战了这一认知。Goodfellow 等人的一个关键洞察是：深度神经网络在高维输入空间中表现出高度的线性特性。虽然神经网络由非线性激活函数构成，但在局部区域内，其行为可以被一个良好的线性模型近似。Goodfellow 等人通过深入的分析，提出**深度神经网络的脆弱性主要源于其在高维空间中的线性特性**（Linear Nature），而非非线性[14]。

基于这一洞察，作者提出了一个假设：对抗样本之所以存在，是因为模型在输入空间中存在“平坦方向”（flat directions），即在这些方向上施加微小扰动，就能导致输出发生显著变化。为了最大化这种变化，最直接的数学方法就是沿着使损失函数（Loss Function）增加最快的方向去扰动输入。而

这个方向，正是损失函数关于输入的梯度方向。FGSM 的动机可以精炼为：

1.线性假设：在输入点附近，模型的行为近似线性。
2.最大化损失：为了欺骗模型，应施加一个扰动，使得模型在该样本上的分类损失最大化。[7]
3.梯度即方向：在高维空间中，梯度方向就是函数值上升最快的方向。
4.约束扰动大小：为了保证扰动在人类感知范围内（即对抗样本在视觉上与原图几乎无异），需要对扰动的幅度进行严格约束，通常使用 L_∞范数进行限制。

2.1.3 FGSM 的数学推导

FGSM 的核心思想是对损失函数 J 在点 $\boldsymbol{x}$ 处进行**一阶泰勒展开**（First-order Taylor Approximation）的线性近似：

$$J(\theta, \boldsymbol{x}+\boldsymbol{\eta}, y) \approx J(\theta, \boldsymbol{x}, \boldsymbol{y}) + \boldsymbol{\delta}^T \nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, \boldsymbol{y}) \quad (1)$$

$x \in R^d$：干净的输入样本（例如，一张图像）。
y：输入 x 的真实标签（ground truth label）。
θ：目标分类模型 f_θ 的参数。
$J(\theta, x, y)$：模型的损失函数（通常为交叉熵损失）。
$\epsilon > 0$：扰动的最大幅度，受 L_∞范数约束。

忽略常数项$J(\theta, \boldsymbol{x}, y)$，我们的目标简化为最大化内积项$\boldsymbol{\eta}^T \nabla_{\boldsymbol{x}} J$其中，$\boldsymbol{\eta}$ 是我们希望施加的扰动。我们希望找到一个 $\boldsymbol{\eta}$，在满足$\| \boldsymbol{\eta} \|_\infty \leq \epsilon$ 的前提下，最大化模型在对抗样本 $\boldsymbol{x}+\boldsymbol{\eta}$ 上的损失 $J(\theta, \boldsymbol{x}+\boldsymbol{\eta}, y)$。得到最终的对抗样本$x_{adv}$是：

$$x_{adv} = x + \epsilon \cdot sign\left(\nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, \boldsymbol{y})\right) \quad (2)$$

表 3 FGSM 算法流程

Algorithm: FGSM

Input: $\boldsymbol{x} \in R^d$: a clean input sample.
$y \in Y$: the true label of $\boldsymbol{x}$.
θ: parameters of the target classifier f_θ.
$\epsilon > 0$: maximum perturbation magnitude (underL_∞ norm).

Output: $\boldsymbol{x}_{adv} \in R^d$: the generated adversarial example.

$\nabla_{\boldsymbol{x}} J \leftarrow \nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, y)$
$\boldsymbol{\eta} \leftarrow \epsilon \cdot sign(\nabla_{\boldsymbol{x}} J)$
$\boldsymbol{x}_{adv} \leftarrow x + \boldsymbol{\eta}$
return $\boldsymbol{x}_{adv}$

2.1.4 FGSM 的问题与局限性

尽管 FGSM 极其高效，但从技术角度看，它存在几个根本性的局限。

FGSM 的核心是基于在原始点 $\boldsymbol{x}$ 处的单步线性近似。然而，深度神经网络的损失景观本质上是高度非线性的。从 $\boldsymbol{x}$ 出发，沿着初始梯度方向直接跨出一个大步（步长为 ϵ），很可能导致生成的点 x_{adv}位于损失函数的一个局部“洼地”或平坦区域，使得实际的损失并未达到最大化，攻击效果不佳。

由于其攻击路径单一且可预测，许多简单的防御机制（如输入变换、随机化、梯度掩蔽）都能有效抵御 FGSM。例如，对输入图像进行轻微的 JPEG 压缩或随机噪声注入，就可能破坏 FGSM 所依赖的精确梯度方向。

FGSM 只关心扰动的方向（由梯度符号决定），而完全忽略了梯度的大小信息。这意味着，即使在某些像素上梯度很小（对损失影响微弱），它也会被赋予与高梯度像素相同的扰动幅度（ϵ）。这导致了扰动资源的浪费，使得在相同的 ϵ 约束下，其攻击成功率低于那些能更精细分配扰动的优化方法。

2.1.5 改进的思路

基于对上述局限性的分析，可以从技术原理上推导出以下潜在的改进方向：

1.引入迭代优化机制：最直接的改进是放弃单步跨跃，转而采用多步小步长的迭代策略。在每一步，都基于当前对抗样本的最新状态重新计算梯度，并沿该梯度方向进行一个微小的更新。这种方法允许攻击者在复杂的非线性损失景观中进行更精细的探索，避免因单步跨度过大而“走错路”，从而更有可能找到能真正最大化损失的对抗样本。
2.利用梯度幅度信息：可以设计一种自适应扰动分配策略，让每个像素的扰动幅度与其梯度的绝对值成正比（或某种单调函数）。这样，对损失影响更大的像素会获得更多的“扰动预算”，而影响小的像素则扰动较少。这能更高效地利用有限的 ϵ 预算，生成更具破坏力的对抗样本。
3.引入随机化以增强鲁棒性：为了对抗那些依赖于确定性梯度的防御，可以在 FGSM 的框架中引入随机性。例如，在每次攻击开始时，从一个以 $\boldsymbol{x}$ 为

1. Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.

中心、半径为 ϵ 的 L_∞ 球内随机初始化起点，或者在梯度方向上加入一个受控的随机噪声。这种随机化策略可以增加攻击的多样性，使其更难被基于确定性模式的防御所预测和规避。

4.探索不同范数约束下的扰动：FGSM 固定使用 L_∞ 范数，这主要源于其与人类视觉系统的兼容性（每个像素变化很小）。然而，可以探索在 L_2 或 L_1 范数约束下生成扰动。例如，在 L_2 约束下，最优扰动方向不再是梯度的符号，而是梯度本身的方向。这可能会生成视觉上更平滑、更不易被察觉对抗样本。

2.1.6 FGSM 复现实验设置

我使用 Windows 10 系统，显卡是 RTX3060Ti，软件环境是 Python3.12，pytorch2.9.1+cu126，cuda12.6。

使用的数据集是 MNIST。MNIST 是手写数字识别的标准数据集。

规模：共 70000 张图像，其中训练集 60000 张、测试集 10000 张；

格式：单通道灰度图（通道数 = 1），尺寸 28×28 像素，像素值范围 0-255（黑色背景、白色数字）；

优势：数据规整、噪声少、类别平衡（每类约 7000 张），无需复杂预处理，快速验证模型效果。

评估指标是训练模型分类准确率。分类准确率定义成：分类成功数量/总参与分类图片数量*100%。与原始准确率 baseline 相比，使用对训练模型进行测试得出分类准确率越低代表 FGSM 对抗样本攻击越成功。

参数设置：模型架构是 LeNet，epochs = 10,batch_size=64,交叉熵损失函数，SGD 的恒定学习率为 0.001，SGD 的 momentum=0.9。

为了探究不同 ϵ 扰动的效果，设置不同的 ϵ 值 eps = [0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]并观察分类准确率的变化。具体实验复现结果如下。

2.1.7 实验结果分析

图 2 MNIST 原始图像的展示

经过实验，可以得到 baseline 的模型分类准确率是 97.53%。

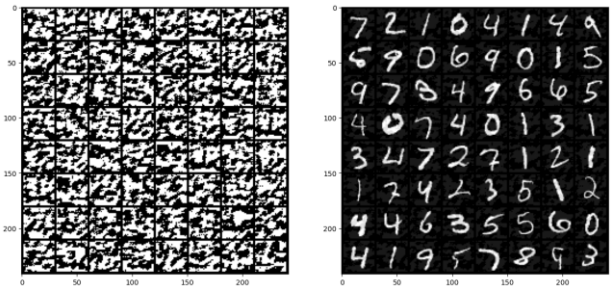

图 3 使用 FGSM 进行对抗样本的生成

可以看到对抗噪声加到原来的图像上后，在肉眼的观察下很难分辨出来，FGSM 能够有效生成人眼不可分辨的微小对抗噪声扰动。

```
epsilon=0.01 acc: 96.52%
epsilon=0.05 acc: 91.96%
epsilon=0.1 acc: 81.37666666666667%
epsilon=0.2 acc: 62.260000000000005%
epsilon=0.3 acc: 49.906%
epsilon=0.4 acc: 41.620000000000005%
epsilon=0.5 acc: 35.68571428571429%
epsilon=0.6 acc: 31.2325%
epsilon=0.7 acc: 27.766666666666666%
epsilon=0.8 acc: 24.997%
```

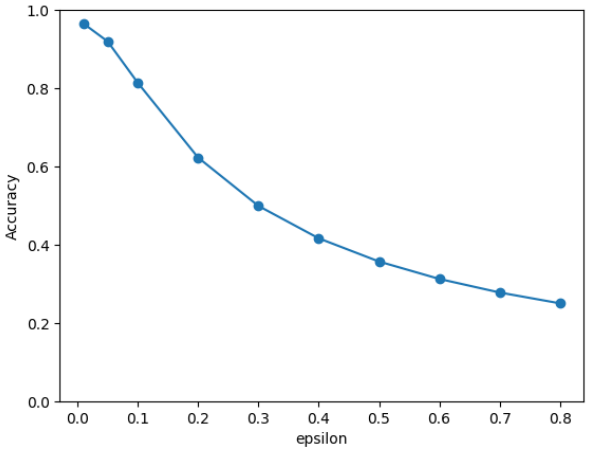

图 4 不同 ϵ 下模型的分类准确率

对同一个分类模型来说，随着ϵ的增加，FGSM 生成的对抗样本使得分类准确率减小，实验中，分类准确率最小甚至小到 24.997%。充分证明了 FGSM 方法生成对抗样本的有效性。

2.2 C&W

2.2.1 C&W 方法概述

Carlini & Wagner (C&W) 攻击是由 Nicholas Carlini 和 David Wagner 在 2017 年 ICLR 会议上提出的，被公认为是迄今为止最强大、最有效的一类白盒攻击。该方法属于白盒攻击（White-box Attack）范畴，其攻击场景明确假设攻击者对目标深度学习模型拥有完全的知识，包括模型的全部架构、可训练参数、激活函数以及输出层的具体形式（包括 softmax 前的 logits）。

C&W 攻击的核心目标是生成在特定距离度量（如 L_0，L_2，或 L_∞ 范数）下扰动幅度最小的对抗样本。与早期旨在最大化损失的攻击方法不同，C&W 将攻击问题严格形式化为一个带约束的优化问题，并精心设计了目标函数以直接衡量攻击的成功与否。这种方法尤其以其对当时主流防御机制（如防御蒸馏）的完全攻破而闻名，证明了许多声称“安全”的防御方法实际上只是梯度掩蔽（gradient masking）。C&W 攻击的高效性和通用性使其成为评估模型鲁棒性的黄金标准之一，特别是在需要生成高质量、低失真对抗样本的场景中。

2.2.2 产生 C&W 的想法和动机

在 C&W 攻击提出之前，对抗样本研究领域存在一个关键问题，许多防御方法（尤其是防御蒸馏）声称对现有攻击具有近乎完美的抵抗力。防御蒸馏被报告能将攻击成功率从 95% 降低到 0.5%。然而，Carlini 和 Wagner 敏锐地意识到，这种“成功”很可能源于现有攻击方法（如 L-BFGS 和 FGSM）在面对防御蒸馏等机制时的根本性缺陷，而非防御本身的有效性[15]。

具体而言，这些早期攻击方法依赖于一个脆弱的优化目标，它们通常使用交叉熵损失（cross-entropy loss）作为优化目标。然而，在防御蒸馏模型中，模型的 logits 值被极大地放大，导致 softmax 后的输出几乎为 one-hot 向量，其梯度（即交叉熵损失的梯度）趋近于零。这使得基于梯度的优化器（如 L-BFGS）无法获得有效的梯度信号，从而“失败”并错误地认为模型是安全的。

因此，C&W 攻击的主要动机可以总结为：

1.系统性地评估与攻破防御：设计一种足够强大、对优化目标不敏感的攻击方法，以揭示防御机制的真实鲁棒性，特别是检验其是否仅仅依赖于梯度掩蔽。

2.最小化扰动幅度：将攻击问题建模为一个真正的优化问题，其目标是在保证攻击成功的前提下，找到扰动最小（在特定范数下）的对抗样本，而非简单地最大化损失。

3.探索最优目标函数：通过广泛的实验，系统地比较和评估多种不同的目标函数，找出最能有效引导优化器找到对抗样本的公式。

2.2.3 C&W 的数学推导和算法步骤

C&W 攻击的数学核心在于将对抗样本生成问题转化为一个非凸优化问题，并巧妙地处理其约束。我们以最常用的 L_2 范数版本为例。给定一个干净样本x 和其真实标签 y，以及一个目标标签 $t \neq y$（C&W 主要用于有目标攻击），目标是找到一个扰动δ，使得对抗样本$x' = x + \delta$ 满足攻击成功：$C(x') = t$，即模型将x'分类为目标类别 t；扰动最小：$\| x' - x \|_2 = \| \delta \|_2$尽可能小；有效性：$x'$是一个有效的图像，即每个像素值都在 $[0,1]$ 范围内。

这是一个带硬性约束（$C(x') = t$）的困难问题。为了解决这个问题，C&W 采用了标准的惩罚函数法（Penalty Method），将硬性约束转化为软性损失项，并与扰动幅度的度量相加。

C&W 测试了七种不同的目标函数，并发现基于 **logits**（softmax 前的原始输出 $Z(x)$）的函数效果最好。其核心思想是，要让模型将 x' 分类为 t，就需要让 t 类的 logit 值 $Z(x')_t$ **大于**所有其他类别的最大 logit 值。[8]

最优的目标函数 f_6 定义为：

$$f(x') = max\left(\max_{i \neq t}\{Z(x')_i\} - Z(x')_t, -\kappa\right) \qquad (3)$$

其中，κ 是一个**置信度参数**。当$\kappa = 0$ 时，只要 $Z(x')_t$大于其他类别的最 logit，$f(x') \leq 0$，攻击即成功。当$\kappa > 0$ 时，要求$Z(x')_t$至少比第二大 logit 大 κ，这可以生成更高置信度的对抗样本，对于提升迁移性非常有用。

1. Carlini N, Wagner D. Towards evaluating the robustness of neural networks[C]//2017 ieee symposium on security and privacy (sp). Ieee, 2017: 39-57.

为了确保优化过程中生成的 x' 始终是有效的图像（像素值在 [0,1] 内），C&W 采用了一种**变量替换**（Change of Variables）的技巧，将无约束优化问题转换为一个自然满足约束的问题。他们引入了一个新的变量 w，并定义：

$$x' = \frac{1}{2}(tanh(w) + 1) \tag{4}$$

由于 $tanh(w) \in [-1,1]$，上式保证了 $x' \in [0,1]$。优化器现在直接在 w 空间中进行搜索。

结合上述，L_2版本 C&W 攻击的最终优化问题是：

$$\min_w \| \frac{1}{2}(tanh(w) + 1) - x \|_2^2 + c \cdot f\left(\frac{1}{2}(tanh(w) + 1)\right) \tag{5}$$

其中，$c > 0$ 是一个平衡超参数，用于权衡扰动大小和攻击成功性。C&W 采用二分搜索（binary search）来寻找满足攻击成功（$f(x') \leq 0$）的最小 c 值，以确保找到的解是扰动最小的。

该优化问题使用 Adam 优化器进行求解。Adam 是一种自适应学习率的梯度下降变体，被证明在实践中比标准梯度下降或 L-BFGS 更有效、更快速地收敛。

2.2.4 C&W 的问题与局限性

C&W 需要执行一个复杂的非凸优化过程，并且为了找到最优的 c 值，还需要进行多次（通常 10-20 次）二分搜索。每次搜索本身又需要数千次 Adam 优化迭代。这使得 C&W 的计算开销远高于 FGSM、BIM 甚至 PGD，对于大规模模型或实时应用场景来说是一个巨大的瓶颈。

和所有基于梯度的优化方法一样，C&W 只能找到一个局部最优解。Adam 优化器的起始点对最终结果有显著影响。虽然可以通过多起点随机初始化来缓解，但这会进一步增加计算成本。

C&W 的最优攻击效果依赖于访问模型的 logits $Z(x)$。如果一个防御机制能够成功地掩盖或混淆 logits 信息（尽管非常困难），C&W 的性能可能会受到影响。

攻击性能对超参数（如 Adam 的学习率、二分搜索的范围和步数、置信度 κ）的选择比较敏感，需要仔细调优。

2.1.5 改进的思路

可以探索比 Adam 更高效的优化算法，或设计一种自适应的优化策略，根据损失函数的特性动态调整学习率和迭代次数，以加速收敛。

为了克服局部最优问题，可以将 C&W 与进化算法（如 CMA-ES）或贝叶斯优化等全局优化方法相结合。例如，可以使用进化算法在粗粒度上搜索有潜力的扰动区域，然后在这些区域内使用 C&W 进行精细优化。

可以设计一种无需显式二分搜索的自适应机制来调整 c。例如，在优化过程中，根据攻击是否成功以及扰动的大小，动态地增加或减少c 的值。

虽然 C&W 找到了当时最优的目标函数，但仍可以探索新的、更能反映人类感知或模型内在脆弱性的目标函数，以生成更逼真、更难以防御的对抗样本。

2.1.6 复现实验设置

我使用 Windows 10 系统，显卡是 RTX3060Ti，软件环境是 Python3.12，pytorch2.9.1+cu126，cuda12.6。

使用的数据集是 MNIST 是标准手写数字识别的数据集。评估指标仍然是训练模型分类准确率。以 FGSM 的分类准确率为 baseline 的结果。并且将这两个方法进行对比。

参数设置：模型架构是 LeNet，epochs = 20,batch_size=64,使用交叉熵损失函数，学习率调度器是：原始 lr = 1e-3，scheduler = torch.optim.lr_scheduler.ReduceLROnPlateau(optimizer, 'min', *factor*=0.5, *patience*=5, *min_lr*=0.0000001)，余弦衰减的学习率调度策略，直到学习率下降到 0.0000001 恒定不变。

2.1.7 实验结果

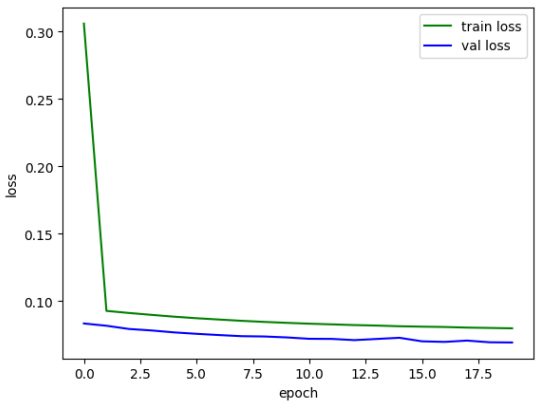

图 5 C&W 参数设置下训练 LeNet 模型

训练正常，经过 20 轮的训练之后没有产生过拟合现象，loss 降到指定阈值之下，训练成功。

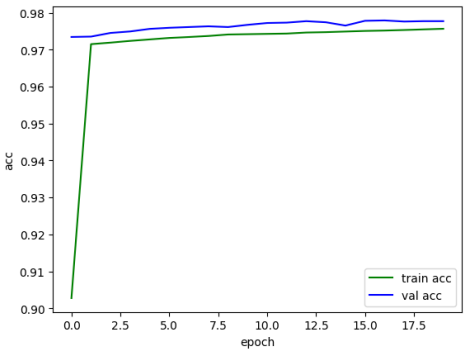

图 6 LeNet 训练过程中的模型分类准确率

经过 20 轮的训练之后，在测试集上的分类准确率达到 97.5%，和 FGSM 的训练 baseline 的结果一致。两种方法能够进行对比这一个评价指标。

可以看到训练图线有点陡峭，可能学习率设置的原因。

定经过实验验证，C&W 方法产生的对抗样本对训练好的分类模型进行攻击能够使分类准确率降低到 7.72%，对比 FGSM 只能降低到 25%，充分证明 C&W 对抗样本攻击的有效性。

但是 C&W 方法产生整个 10000 张测试图片的对抗样本并测试分类准确率的时间开销是 11 分钟，而 FGSM 整个流程的时间开销只需要 51 秒。可以证明 C&W 方法的攻击效果很好，但是开销较大，FGSM 的攻击效果比 C&W 方法差，但是开销很小。

图 7 部分C&W对抗样本攻击图像及其分类结果的展示

3 黑盒对抗样本攻击的实验复现

3.1 ZOO零阶优化攻击

3.1.1 ZOO 方法的概述

零阶优化攻击（Zeroth Order Optimization, ZOO）是一种专为**黑盒攻击**（Black-box Attack）场景设计的强大对抗样本生成算法。在该威胁模型下，攻击者**无法访问目标深度神经网络**（DNN），只能通过向目标模型发送查询（输入任意图像）并观察其返回的输出（如预测的类别概率或置信度分数）来推断模型行为。这种场景高度模拟了现实世界中的安全挑战，例如攻击商业化的图像识别 API、移动设备上的机器学习应用程序或物理世界中的部署系统。

ZOO 属于**基于分数的黑盒攻击**（Score-based Black-box Attack）范畴。与基于决策的攻击（仅能获取最终分类标签）或基于迁移性的攻击（需训练替代模型）不同，ZOO 的核心假设是攻击者能够获取目标模型输出的**完整置信度分数**（confidence scores）。在此条件下，ZOO 通过巧妙地利用**零阶优化**（Derivative-Free Optimization）技术，直接在目标黑盒模型上**估计梯度**，从而绕过对模型内部结构和参数的需求。这种方法的关键优势在于它**无需训练任何替代模型**（substitute model），直接对目标模型发起攻击，从而完全规避了因攻击迁移性不足而导致的性能损失。因此，ZOO 能够实现与最强白盒攻击（如 C&W 攻击）相媲美的攻击效果。

3.1.2 产生 ZOO 方法的想法和动机

在 ZOO 提出之前，主流的黑盒攻击策略几乎都依赖于**训练替代模型**：攻击者首先通过向目标黑盒模型发送大量查询，收集输入-输出对，然后利用这些数据训练一个结构已知的替代模型；一旦替代模型训练完成，攻击者就可以将其视为一个白盒模型，并使用强大的白盒攻击方法（如 FGSM 或 C&W）来生成对抗样本；最后，利用对抗样本的可迁移性（Transferability），将针对替代模型生成的对抗样本用于攻击原始的目标黑盒模型。

训练替代模型存在一个瓶颈：攻击效果严重依赖于替代模型与目标模型之间的相似度。如果替代模型的架构、容量或训练数据与目标模型相差甚远，那么生成的对抗样本的迁移成功率会急剧下降。此外，训练一个高质量的替代模型本身就是一

个耗时且数据密集型的过程。

ZOO 的核心动机就是打破对替代模型的依赖，它来源于优化理论中的零阶优化方法（Zeroth Order Optimization）。这类方法只需要一个零阶预言机（zeroth-order oracle）——即能返回任意输入点的函数值（在这里就是目标模型的损失函数值）——而不需要一阶（梯度）或二阶（Hessian）导数信息。通过在输入点附近进行微小的扰动并查询函数值的变化，就可以数值估计（numerically estimate）出梯度。ZOO 正是将这一思想应用于对抗攻击领域，通过向目标模型发送精心设计的查询，来估计损失函数相对于输入图像的梯度，从而驱动优化过程，直接生成针对目标模型的对抗样本。

3.1.3 ZOO 的数学推导和算法步骤

ZOO 的数学框架直接继承自 C&W 攻击的优化目标，并对其进行了适应黑盒场景的改造。

由于无法访问 logits $Z(x)$，ZOO 转而使用模型的最终输出 $F(x)$（即 softmax 后的置信度分数）来构建目标函数。利用对数函数的单调性，ZOO 提出了一个新的基于置信度的目标函数：

$$f(x,t) = \max\left(\max_{i\neq t} F(x)_i - logF(x)_t, -k\right) \qquad (6)$$

此函数与 logits 版本在功能上等价：当 $f(x,t) \leq 0$ 时，攻击成功。对数运算能有效缓解 softmax 输出极度偏向某一类（skewed distribution）带来的数值问题。

零阶梯度估计是 ZOO 的核心技术。为了在不进行反向传播的情况下获得梯度信息，ZOO 采用对称差商（symmetric difference quotient）来估计目标函数 f（或总损失）在每个像素坐标 i 上的偏导数。具体公式为：

$$\hat{g}_i \approx \frac{\partial f(x)}{\partial x_i} \approx \frac{f(x+he_i) - f(x-he_i)}{2h} \qquad (7)$$

其中：h 是一个很小的常数（如 0.0001），用于控制扰动的大小。e_i 是第 i 个标准基向量（第 i 个元素为 1，其余为 0）。要估计完整梯度 $\nabla_x f$，理论上需要对 p 个像素（p 是图像总像素数）分别进行上述计算，即总共需要 $2p$ 次模型查询。

直接估计完整梯度在高维图像（如 ImageNet 的 299×299×3=268203 维）上是计算上不可行的。因此，ZOO 引入了随机坐标下降法（Stochastic Coordinate Descent），每次迭代只随机选择一个（或一小批）坐标进行梯度估计和更新，将每次迭代的查询成本从 O(p) 降低到 O(1)。

Algorithm 1 Stochastic Coordinate Descent

while not converged **do**
  Randomly pick a coordinate $i \in \{1, \ldots, p\}$
  Compute an update δ^* by approximately minimizing

$$\arg\min_{\delta} f(\mathbf{x} + \delta \mathbf{e}_i)$$

  Update $\mathbf{x}_i \leftarrow \mathbf{x}_i + \delta^*$
end while

图 9 ZOO 使用的随机坐标下降法算法流程图[16]

为了进一步提升 ZOO 在大规模模型和数据集上的效率，原论文还提出了三项关键优化技术：

攻击空间降维（Attack-space Dimension Reduction）：不直接在原始像素空间优化，而是引入一个变换 $D(y)$（如双线性插值），在低维噪声空间 $y \in R^m$（$m \ll p$）中进行优化，从而大幅减少需要估计的变量数量。

分层攻击（Hierarchical Attack）：先在非常低的维度（如 32×32×3）进行搜索，如果未能成功，再逐步增加维度（如 64×64×3，128×128×3），以在效率和搜索能力之间取得平衡。

重要性采样（Importance Sampling）：在坐标下降过程中，并非均匀随机选择像素，而是根据初步攻击结果中像素值的变化幅度来分配采样概率，优先更新对攻击贡献更大的像素区域，以加速收敛。

3.1.4 ZOO 的问题和局限性

尽管 ZOO 在黑盒攻击领域取得了重大突破，但它仍然存在一些固有的技术性局限。

高昂的查询成本是 ZOO 最核心的局限。即使采用了坐标下降和降维等加速技术，ZOO 仍然需要向目标模型发送成千上万次的查询才能成功生成一个对抗样本。对于有查询次数限制或按次计费的商业 API 来说，这构成了巨大的障碍。

ZOO 的有效性高度依赖于能够获取模型输出的完整置信度分数。如果目标模型只返回最终的分类标签（即决策），而不提供概率或分数，ZOO 将无法工作。这类场景需要使用基于决策的攻击方法（如 Boundary Attack）。

零阶梯度估计本质上是一种数值近似，其精度受到扰动幅度 h 和模型本身数值稳定性的影响。如

1. Chen P Y, Zhang H, Sharma Y, et al. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models[C]//Proceedings of the 10th ACM workshop on artificial intelligence and security. 2017: 15-26.

果模型的输出非常平滑或者存在大量的数值舍入（如 logits 被极大放大以实现硬分类），梯度估计可能会失效或变得非常嘈杂，导致优化过程难以收敛。

与白盒攻击中直接利用精确梯度的优化器（如 Adam）相比，ZOO 的零阶优化过程要慢得多，优化效率低。每一次有效的梯度更新都需要多次模型查询，这使得 ZOO 在面对需要极高精度对抗样本的任务时显得力不从心。

3.1.5 改进的思路与想法

可以探索比对称差商更高效的梯度估计方法。例如，使用随机梯度估计（如 Gaussian smoothing），通过一次或少数几次在随机方向上的查询来估计整个梯度向量，而不是逐个坐标估计。

当前的 ZOO 在低维空间进行优化，但这个空间（如简单的上采样）可能并未充分利用自然图像的统计特性。可以设计一个基于生成模型（如 VAE 或 GAN）的更智能的低维流形，利用自然图像先验分布信息，在该流形上进行优化，能更高效地生成逼真的对抗样本。

可以设计一种自适应的查询机制，根据当前优化的状态动态调整查询策略。例如，在优化初期，使用较大的 h 和较少的坐标进行粗略探索；在接近成功时，切换到较小的 h 和更精细的坐标更新，以提高攻击的精度和成功率。

在完全未知目标模型的情况下，一个粗略的替代模型仍然可以提供有价值的先验信息。可以将 ZOO 与迁移性攻击相结合，先用替代模型生成一个初始的对抗样本，然后用 ZOO 在该点附近进行精细化的“微调”，以大幅减少所需的查询次数。

3.1.6 复现的实验设置

不包含对攻击空间降维，分层攻击和重要性采样的代码实现，这些优化 ZOO 攻击效果的方法主要是对大型数据集比如 ImageNet 全集等数据集设计的。本实验使用 MNIST 手写识别数据集和 CIFAR10 数据集。CIFAR-10 是一个 包含 10 个类别图片的小型图像数据集，由加拿大多伦多大学发布。其特点如下：

图片数量：共计 60,000 张 32x32 彩色图像。其中训练集划分 50,000 张，测试集为 10000 张。

类别数量：共 10 个类别，分别是“飞机”，“汽车”，“鸟”，“猫”，“鹿”，“狗”，“青蛙”，“马”，“船”，和“卡车”。

分别使用 MNIST 和 CIFAR10 数据集使用相同的 7 层 CNN 模型和训练参数训练出两个黑盒模型，评估指标是模型的分类准确率。这个模型采用了一个包含四层卷积（32-32-64-64 通道）和三层全连接（1024→200→200→10）的 CNN 架构，其中卷积层后接 ReLU 激活和最大池化，全连接层使用 Dropout(0.5)正则化（注：代码中误用 Dropout2d，应改为 1D dropout）；训练使用 50 个 epoch，固定学习率 0.01 的 SGD 优化器（带动量 0.9、Nesterov 加速和权重衰减 1e-6），损失函数为交叉熵损失，批量大小 128，且未采用任何学习率衰减策略。数据预处理包含标准化（均值 0.5，标准差 1.0），训练/验证集按 55k/5k 分割，所有随机种子固定为 42 以保证可复现性。

3.1.6 实验结果

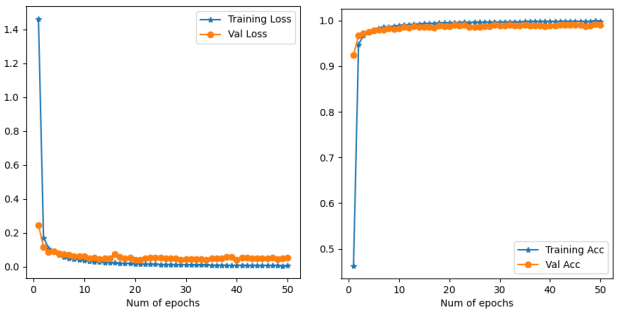

图 10 MNIST 数据集下的模型训练图

在 MNIST 数据集下进行训练的时候，模型迭代训练到 50epochs 时 val loss 并没有呈上涨趋势，且值大于 train loss，说明此时并没有出现过拟合且学习效果好，训练成功。第 50epoch 训练结束后的训练集分类成功率是 99.8127%，测试集分类成功率是 99%。

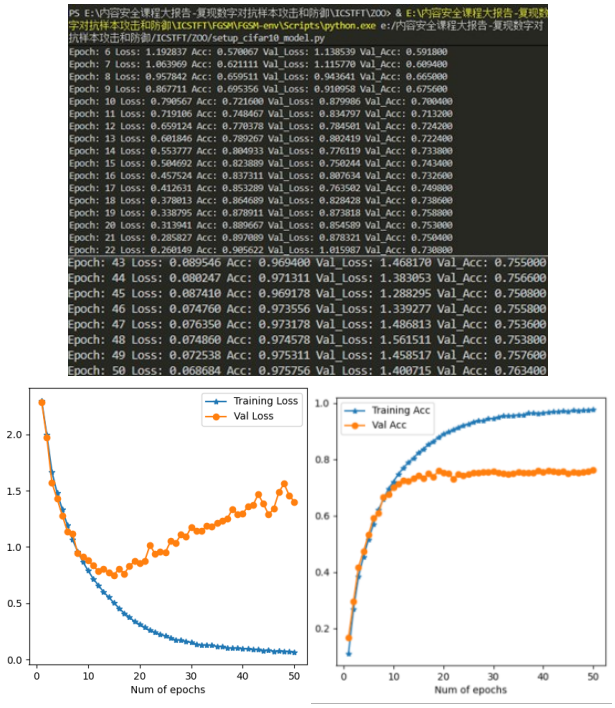

图 11　CIFAR10 数据集下的模型训练图

在 CIFAR10 数据集下进行训练的时候，模型迭代训练到 50epochs 时 val loss 并明显呈上涨趋势，且值大于 train loss，可能是由于数据集比较复杂泛化性差的原因，使用与 MNIST 相同的模型架构不能很好的拟合学习训练数据并提高泛化性，说明此时出现轻微过拟合且学习效果好 loss 下降到指定阈值，训练成功。第 50epoch 训练结束后的训练集分类成功率是 97.57%，测试集分类成功率是 76.34%。

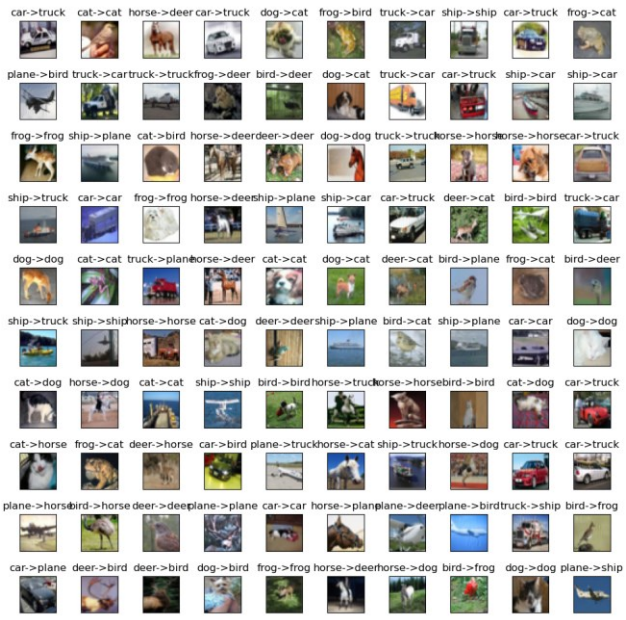

图 12　Adam 优化器，针对训练好的黑盒 CIFAR10 分类器的 Untargeted 无目标 ZOO 攻击

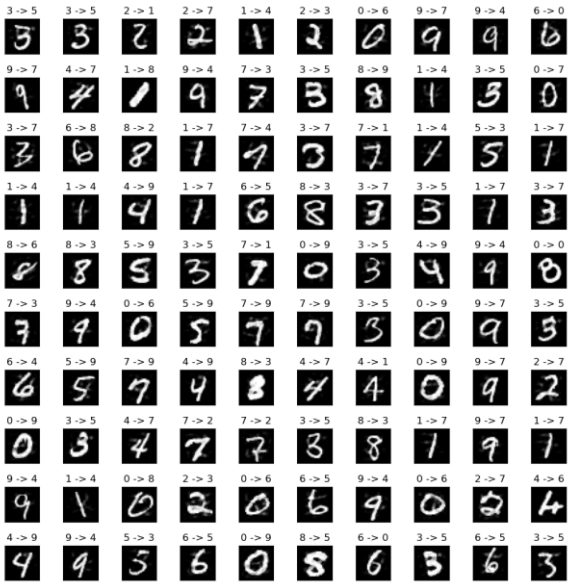

图 13　Adam 优化器，针对训练好的黑盒 MNIST 分类器的 Untargeted 无目标 ZOO 攻击

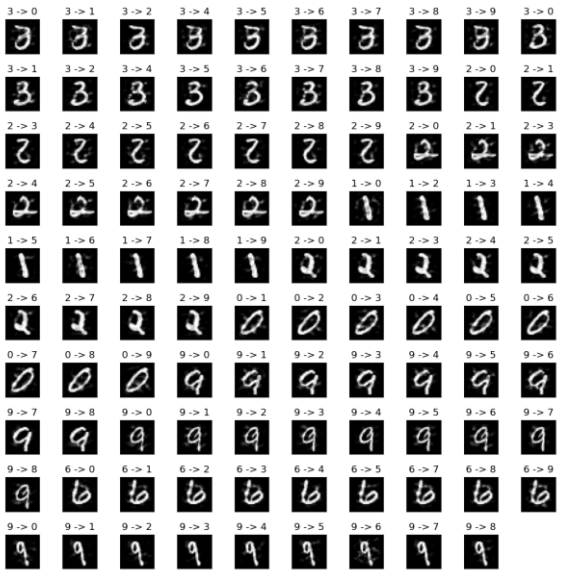

图 14　Newton 优化算法，针对训练好的黑盒 MNIST 分类器的 Targeted 有目标 ZOO 攻击

3.2　Boundary Attack 决策边界攻击

3.2.1　深度学习模型的决策边界

在机器学习中，决策边界是一个超曲面（hypersurface），它将输入空间（input space）划分为不同的区域，每个区域对应一个模型的预测类别。给定一个分类器 $f:X\to Y$，其中 $X\subseteq R^d$ 是 d 维的输入空间（如图像像素空间），$Y=\{1,2,\ldots,K\}$ 是类别标签集合。对于任意输入 $x\in X$，模型的预测为 $\hat{y}=f(x)$。决策边界 B 被定义为所有满足以下条件的点的集合：

$$B = \{x \in X \mid \exists\, i,j \in Y, i \neq j, such\ that\ F_i(x) = F_j(x) \geq F_k(x), \forall k \in Y\} \quad (8)$$

其中 $F_i(x)$ 是模型输出的第 i 个类别的置信度（如 softmax 概率或 logits）。直观地说，决策边界就是模型“犹豫不决”的地方——在这里，至少有两个类别的置信度相等且最高。

对于深度神经网络（DNN），决策边界具有以下关键特性：

高度非线性与复杂性：DNN 的决策边界通常是极其复杂的、高维的、非线性的流形（manifold）。与线性分类器（如 SVM）的超平面边界不同，DNN 的边界可以折叠、扭曲以适应复杂的非线性数据分布。这种复杂性使得在输入空间中，决策边界周围存在大量高维的、未被良好覆盖的区域，这些区域正是对抗样本的藏身之所。

局部线性近似：尽管全局上高度非线性，但在决策边界的一个非常小的局部邻域内，其行为常常可以被一个线性或超平面所近似。许多攻击方法正是利用了这一特性，通过线性近似来估算到达边界的最短路径。

高维空间的几何特性：在高维空间（如 ImageNet 的 299×299×3=268203 维），决策边界的“表面积”相对于整个输入空间而言可能非常庞大。这使得从任意一点出发，找到通往决策边界的路径变得相对容易。这种“高维灾难”是对抗样本普遍存在的重要原因之一。

对抗方向（Adversarial Direction）：决策边界的法向量（梯度方向）指示了使模型输出变化最快的方向。对抗攻击的核心就是沿着这个方向施加扰动。对于一个干净样本x_0，其梯度 $\nabla_x L(f(x_0), y_0)$ 近似指出了通往决策边界的“最快”方向。

决策边界本身是抽象的，但有些方式可以来衡量和探索。

决策边界的厚度 / 鲁棒性半径（Robustness Radius）：这是衡量一个点x_0到其最近决策边界的距离。形式化地，对于给定的 p-范数（通常是L_2或L_∞），鲁棒性半径ϵ^*定义为：

$$\epsilon*(x_0) = \min_{\delta}\{\| \delta \|_p \mid f(x_0+\delta) \neq f(x_0)\} \quad (9)$$

这个值越大，说明该点越“安全”，模型对该点的预测越鲁棒。对抗样本攻击的目标就是找到一个$\| \delta \|$接近ϵ^*的扰动。几乎所有基于优化的攻击（如 C&W, Madry et al., 2018）都隐式或显式地在逼近这个半径。

决策边界的曲率（Curvature）：边界在局部区域的弯曲程度。一个高曲率的边界意味着线性近似（如 FGSM、DeepFool）会迅速失效，需要更复杂的非线性攻击。曲率可以通过 Hessian 矩阵来分析（Moosavi-Dezfooli et al., 2019）。

决策边界的连通性（Connectivity）：有研究表明，对于某些 DNN，属于同一类别的干净数据点，其周围的决策边界区域可能是相互连通的（Fawzi et al., 2018）。这意味着一个对抗样本可以通过沿着边界“行走”转换为另一个对抗样本，这为 Boundary Attack 提供了理论基础。

可证明鲁棒性（Provable Robustness）：这类研究试图在数学上证明一个模型在其决策边界的一个特定区域内是安全的，即对于该区域内的任何扰动，模型预测都不会改变。这通常通过在输入空间上构建一个凸的、可证明安全的区域来实现（Wong & Kolter, 2018）。

决策边界的可视化（Visualization）：由于高维空间的不可见性，研究者们开发了降维（如 PCA, t-SNE）或插值（如沿对抗方向）的方法来可视化决策边界，以更好地理解模型的脆弱性（Liu et al., 2019）。[10]

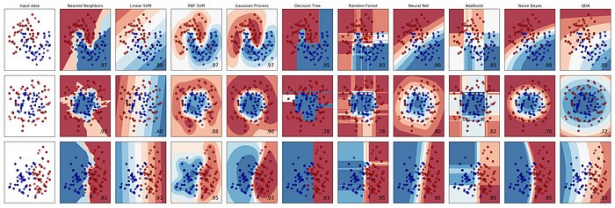

图 15 机器学习模型的决策边界

3.2.2 Boundary Attack 方法的概述

Boundary Attack 是一种典型的决策式黑盒攻击（Decision-based Black-box Attack）。在此威胁模型下，攻击者对目标深度学习模型的内部结构（如参数、激活函数）和输出置信度（如 softmax 概率）一无所知。攻击者唯一能获取的信息是模型对任意输入的最终决策结果（final decision），即一个离散的类别标签（class label）。

该方法属于对抗样本攻击分类中的“基于决策的攻击”（Decision-based Attacks）或“零阶查询攻击（Zeroth-order Query Attacks）”中的一个特殊子类。其攻击目标既可以是非目标攻击（untargeted

1.Brendel W, Rauber J, Bethge M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models[J]. arXiv preprint arXiv:1712.04248, 2017.

attack，即只需让模型预测错误），也可以是目标攻击（targeted attack，即让模型预测为一个指定的错误类别）。Boundary Attack 通过在输入空间中执行一个受约束的随机游走（constrained random walk），从一个已知的对抗样本出发，逐步向原始干净样本靠近，同时始终确保其处于决策边界的“对抗一侧”，从而生成一个扰动尽可能小的对抗样本[17]。之前主流的黑盒攻击主要分为两类：

基于迁移的攻击（Transfer-based Attacks）：需要训练一个或多个替代模型（substitute models），然后在替代模型上生成对抗样本。这种方法不仅计算成本高昂，而且其成功率严重依赖于替代模型与目标模型的相似度。如果两者差异较大，攻击效果会急剧下降。

基于分数的攻击（Score-based Attacks）：如 ZOO（Chen et al., 2017），通过查询模型输出的置信度分数来数值估计梯度。这种方法虽然能直接攻击目标模型，但其查询效率极低（需要数万次甚至数十万次查询），并且在面对只提供离散标签的严格黑盒场景时完全失效。

Boundary Attack 的动机正是要打破对模型内部信息的依赖，并设计一种更高效、更普适的黑盒攻击范式。其核心思想是：既然攻击的目标是跨越决策边界，那么攻击过程就应该直接在决策边界上进行。与其通过复杂的优化去“寻找”边界，不如从一个已知的、远离边界的对抗点开始，通过一个简单的、基于决策反馈的探索过程，“游走”到离原始样本最近的边界点。

3.2.3 Boundary Attack 的数学推导

Boundary Attack 的核心是一个**拒绝采样**（Rejection Sampling）过程，其目标是最小化对抗样本与原始样本之间的距离（通常用 L_2 范数），同时满足对抗性约束。在原始样本 o 和一个已知的对抗样本 o_0 之间，存在一条连接它们的路径，该路径与模型的决策边界相交。攻击的目标就是找到这条路径上离 o 最近的交点。

原始样本：o，模型的正确决策为 $y = F(o)$。

初始对抗样本：o_0，模型的决策 $F(o_0) \neq y$（对于无目标攻击）或 $F(o_0) = t$（对于有目标攻击，t 为目标类）。

距离度量：通常使用 L_2 数，即 $d(o, o') = \| o - o' \|_2^2$

提案分布（Proposal Distribution）：$P(o_{k-1})$，用于在第 k 步生成一个候选扰动 η_k，使得新的候选样本为 $o_{k-1} + \eta_k$ 。

Data: original image **o**, adversarial criterion $c(.)$, decision of model $d(.)$
Result: adversarial example $\tilde{o}$ such that the distance $d(o, \tilde{o}) = \|o - \tilde{o}\|_2^2$ is minimized
initialization: $k = 0$, $\tilde{o}^0 \sim \mathcal{U}(0,1)$ s.t. $\tilde{o}^0$ is adversarial;
while $k <$ *maximum number of steps* **do**
  draw random perturbation from proposal distribution $\eta_k \sim \mathcal{P}(\tilde{o}^{k-1})$;
  if $\tilde{o}^{k-1} + \eta_k$ *is adversarial* **then**
    set $\tilde{o}^k = \tilde{o}^{k-1} + \eta_k$;
  else
    set $\tilde{o}^k = \tilde{o}^{k-1}$;
  end
  $k = k + 1$
end

图 16 Boundary Attack 的算法流程图

由于该优化问题无法通过梯度下降求解（因为没有梯度信息），Boundary Attack 采用了一种启发式的随机搜索策略。

1.初始化：算法从一个已知的对抗样本 o_0 开始。对于无目标攻击，可以从最大熵分布（如像素值在[0,255]上的均匀分布）中采样，并拒绝非对抗样本，直到找到一个。对于有目标攻击，可以直接从目标类的一个样本开始。

2.迭代优化：在每一步 k，算法执行以下操作：

a. 生成候选：从提案分布 $P(o_{k-1})$ 中采样一个扰动 η_k 。

b. 决策测试：向目标黑盒模型查询 $F(o_{k-1} + \eta_k)$。

c. 拒绝/接受：如果 $o_{k-1} + \eta_k$ 仍然是对抗样本（即满足对抗准则 $c(\cdot)$），则接受该候选，并令 $o_k = o_{k-1} + \eta_k$ 。否则，拒绝该候选，并令 $o_k = o_{k-1}$。

提案分布设计：为了高效地向原始样本 o 靠拢，提案分布 P 被精心设计为一个满足以下约束的扰动：

约束 1（域内）：$o_{k-1} + \eta_k$ 必须在合法的输入域内（如每个像素 $\in [0,255]$）。

约束 2（步长）：扰动的幅度是可控的，$\| \eta k \|_2 = \delta \cdot d(o, o_{k-1})$，其中 δ 是一个相对步长超参数。

约束 3（向原点移动）：扰动要能有效减小与原始样本的距离，$d(o, o_{k-1}) - d(o, o_{k-1} + \eta_k) = \epsilon \cdot d(o, o_{k-1})$，其中 ϵ 是另一个超参数。

在实践中，为了简化采样过程，算法采用了一种启发式策略，首先，从一个各向同性的高斯分布 $\eta_\perp \sim N(0, I)$ 中采样，并将其投影到以 o_{k-1} 为中心、半径为 $\delta \cdot d(o, o_{k-1})$ 的球面上，得到一个正交扰动（orthogonal perturbation）$\eta_\perp$。这个扰动主要用于探索决策边界的局部几何结构。

然后，在正交扰动的基础上，再添加一个朝向原始样本 o 的微小向内扰动（inward perturbation）

$\eta_{\parallel}$。最终的扰动为 $\eta_k = \eta_{\perp} + \eta_{\parallel}$ 。

Boundary Attack 的一大优点是其超参数 δ 和 ϵ 可以通过动态调整来适应决策边界的局部几何形状，这极大地减少了手动调参的需求。其调整逻辑受信赖域（Trust Region）方法启发，调整 δ (正交步长)，算法首先测试正交扰动 $\eta_{\perp}$本身是否是对抗样本。如果成功率（即接受率）远低于 50%，说明步长太大，决策边界在此区域变化剧烈，需要减小 δ。如果成功率高于 50%，则可以适当增大 δ 以加快探索速度。调整 ϵ (向内步长)，在正交扰动被接受的前提下，算法再测试加上向内扰动后的完整扰动。如果向内移动的成功率过低，说明决策边界在此区域过于“陡峭”或“尖锐”，需要减小 ϵ；反之，则可以增大 ϵ。

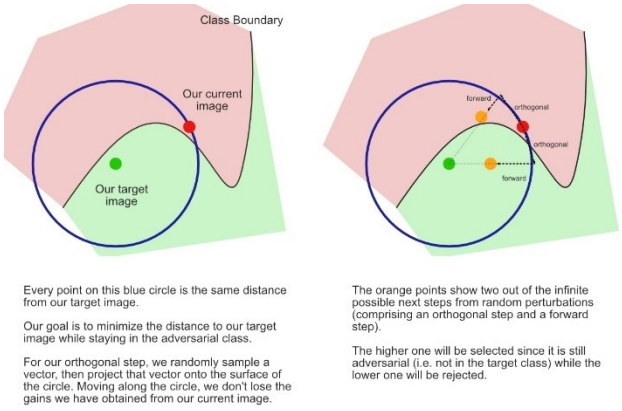

图 17 Boundary Attack 的解释

3.2.4 Boundary Attack 的局限性与改进思路

查询效率低下是其最核心的缺陷。为了在高维空间中找到一个有效的扰动方向，Boundary Attack 需要进行大量的随机采样和查询。通常需要数千甚至上万次的模型查询才能生成一个高质量的对抗样本，这使其在有严格查询次数限制的商业 API 攻击场景中不切实际。

Boundary Attack 其收敛速度高度依赖于初始对抗点的选择和随机游走的运气，收敛速度慢且不稳定。在决策边界非常“崎岖”或“狭窄”的区域，算法可能会陷入长时间的停滞，无法有效向内移动。

此方法天然地优化的是 L_2 范数距离。虽然可以通过修改提议分布来适应其他范数（如 L_∞），但这会使得算法的实现变得复杂，并且可能无法保持原有的效率和效果。

对于非目标攻击，找到一个初始的对抗样本可能需要多次随机采样，尤其是在模型本身比较鲁棒的情况下。对于目标攻击，如果攻击者完全不知道任何属于目标类别的样本，则初始步骤将变得非常困难。

针对潜在的局限性和问题可以提出相应的改进思路。当前的提议分布是各向同性的高斯分布，需要使用更智能的提议分布。可以学习出一个数据驱动的提议分布，该分布能够根据当前点的局部几何结构和历史成功的扰动方向，生成更有可能被接受的候选扰动。例如，可以利用过去成功的扰动来构建一个局部的主成分分析（PCA）基，从而在更重要的方向上进行探索。

虽然不能直接访问梯度，但可以通过有限次的决策反馈来粗略估计梯度的方向。例如，可以在多[11]个正交方向上同时进行采样，并根据哪些方向的成功率更高来推断出一个“伪梯度”方向，从而指导下一步的移动，将随机游走变为半定向搜索。

可以将 Boundary Attack 与其他黑盒攻击策略结合起来。先使用一个查询效率较高的方法（如基于迁移的攻击）生成一个粗略的对抗样本，然后将其作为 Boundary Attack 的初始点，再利用 Boundary Attack 进行精细化的边界探索，以大幅减少总查询次数。

在某些场景下，虽然无法获取完整的置信度分数，但可能会有一些间接信息，可以利用模型内部状态的间接信息，如预测的置信度等级（高/中/低）和 top-k 的类别等。可以设计一种机制，利用这些部分信息来指导搜索过程，提高效率。

4 防御蒸馏

4.1 “防御蒸馏”方法概述

“防御蒸馏”（Defensive Distillation）是一种处理中防御（In-processing Defense），其核心目标是在模型训练阶段通过修改训练过程和目标，来提升深度神经网络（DNN）模型对对抗样本（Adversarial Examples）的内在鲁棒性。该方法并非在输入或输出端进行处理，而是直接作用于模型的学习机制本身。

1. Papernot N, McDaniel P, Wu X, et al. Distillation as a defense to adversarial perturbations against deep neural networks[C]//2016 IEEE symposium on security and privacy (SP). IEEE, 2016: 582-597.

在对抗样本防御的分类体系中，“防御蒸馏”明确属于对抗性训练（Adversarial Training）或鲁棒优化（Robust Optimization）这一大类。然而，它与典型的对抗性训练不同，者通过在训练数据中显式地加入对抗样本来增强模型，而“防御蒸馏”则通过一种知识蒸馏 nowledge Distillation）的方式，间接地使模型学习到一个平滑的决策边界，从而对微小的输入扰动变得不那么敏感。

“防御蒸馏”的应用场景主要针对白盒攻击（White-box Attack）和基于迁移的黑盒攻击（Transfer-based Black-box Attack）。其设计初衷是防御那些依赖于模型梯度信息的攻击方法，通过降低模型输出相对于输入的敏感性（即减小梯度幅度），使得攻击者难以有效地构造出能够欺骗模型的微小扰动。

4.2 产生“防御蒸馏”的想法和动机

早期对抗样本攻击方法（如 L-BFGS、FGSM）的成功，很大程度上依赖于模型损失函数或输出相对于输入的梯度。如果一个模型的决策边界非常“陡峭”或“敏感”，那么在梯度方向上施加一个微小的扰动，就能导致模型的输出发生巨大的、足以改变分类结果的变化。

“防御蒸馏”的核心动机可以概括是通过平滑模型的决策函数，来降低其对输入微小扰动的敏感性，从而使对抗样本更难被构造。

具体来说，一个在标准数据集（如 MNIST）上训练的模型，往往会做出“过于自信”的预测。例如，对于一张数字“7”的图像，模型可能会输出一个概率向量 [0, 0, ..., 0.999, ..., 0]，其中正确类别的概率接近 1，而其他所有类别的概率都接近 0。这种“硬标签”（Hard Label）式的训练方式，迫使模型在决策边界附近变得非常尖锐，形成了所谓的“盲点”（blind spots）。

“防御蒸馏”来自于知识蒸馏（Knowledge Distillation）这一模型压缩技术。在知识蒸馏中，一个大型的、复杂的“教师”模型（Teacher Model）被用来指导一个小型的“学生”模型（Student Model）的训练。教师模型输出的软标签（Soft Labels）—即带有温度参数的 softmax 概率分布——包含了比硬标签更丰富的信息，比如不同类别之间的相对相似性。这种软标签所蕴含的“额外知识”（Additional Knowledge）不仅可以用于模型压缩，也可以被用来正则化（Regularize）模型本身，使其学习到一个更平滑、更具泛化能力的决策函数[18]。

于是，原论文作者提出先用标准方法训练一个“教师”模型，然后用这个教师模型生成的软标签来训练一个与教师模型结构相同的“学生”模型，并将其作为最终的防御模型。这种方法被称为“防御蒸馏”，因为它利用蒸馏过程中的平滑效应来防御对抗扰动。

4.3 知识蒸馏

考虑一个深度神经网络 F，其最后一层（softmax 层）的输入被称为 logits，记为 $Z(X) = [z_0(X), z_1(X), \ldots, z_{N-1}(X)]$，其中 N 是类别总数，X 是输入。标准的 softmax 输出（即预测的概率分布）为：

$$F(X)_i = \frac{\exp\left(\frac{z_i(X)}{T}\right)}{\sum_{j=0}^{N-1} \exp\left(\frac{z_j(X)}{T}\right)} \tag{10}$$

其中，T 是一个称为 温度（Temperature）的超参数。

在标准训练阶段中，温度 $T = 1$，模型的训练目标是最小化交叉熵损失（Cross-Entropy Loss）：

$$L_{CE}(X, Y) = -\sum_{i=0}^{N-1} Y_i \cdot \log(F(X)_i) \tag{11}$$

这里，Y 是一个 one-hot 的硬标签向量。

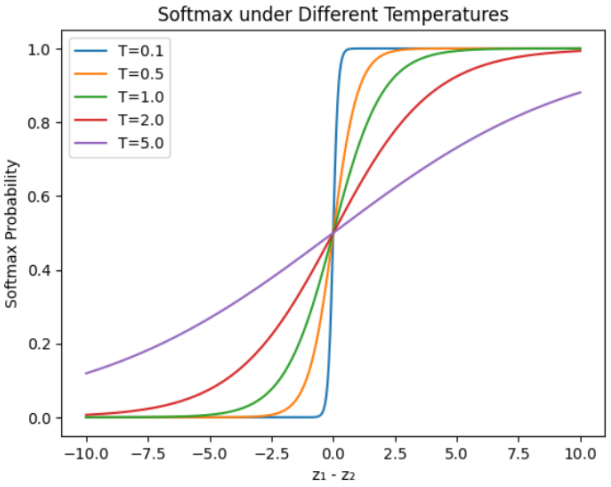

图 18 针对 softmax 为 $P(y=1) = \frac{e^{\frac{z}{T}}}{e^{\frac{z}{T}}+1}, z_1 = z, z_2 = 0$，的二分类问题，不同温度 T 下的二维可视化图像

在知识蒸馏中，首先用一个较高的温度 $T > 1$ 训练一个教师模型。高温度使得 softmax 输出的概率分布更加“平滑”，即非最大类别的概率不再接近于零，而是具有一个可观的正值。这个平滑的概率分布 $F(X)$ 就是软标签。

然后，用这些软标签去训练一个学生模型。学生模型的损失函数通常是教师模型输出和学生模型输出之间的 KL 散度（Kullback-Leibler Divergence），或者直接是交叉熵：

$$L_{KD}(X) = -\sum_{i=0}^{N-1} F(X)_i \cdot \log(F_{student}(X)_i) \quad (12)$$

在推理阶段时，温度被重置回 $T = 1$，以便模型能做出更自信、更明确的分类决策。

4.4 “防御蒸馏”的算法步骤

“防御蒸馏”目标不是压缩模型，而是利用蒸馏过程中的平滑效应来增强模型的鲁棒性。

步骤 1：训练教师模型（Teacher Model）在温度 T 下训练第一个模型 F。其优化目标是在软标签 Ysoft=F(X;T) 的指导下，最小化交叉熵损失：

$$\min_{\theta_F} -\frac{1}{|X|}\sum_{X\in X}\sum_{i=0}^{N-1} Y_i(X)\log\left(F(X;T)_i\right) \quad (13)$$

这里，虽然目标仍然是拟合硬标签 Y(X)，但高温度 T 会使得模型在训练过程中被迫考虑所有类别的概率，从而学习到一个更平滑的函数。

步骤 2：生成软标签训练集使用训练好的教师模型 F，在相同的高温度 T 下，为训练集中的所有样本 X 生成软标签 F(X;T)。

步骤 3：训练防御模型（即学生模型）训练一个结构与教师模型完全相同的新模型 Fd ，但这次的目标是拟合教师模型生成的软标签。其优化目标为：

$$\min_{\theta_{F_d}} -\frac{1}{|X|}\sum_{X\in X}\sum_{i=0}^{N-1} F(X;T)_i\log(F_d(X;T)_i) \quad (14)$$

“防御蒸馏”的核心是通过使用软标签，模型 F_d 在训练时不再被强制要求对正确类别做出过于自信的预测，而是学习到各类别之间的相对关系和概率分布的平滑性。

原论文用理论分析解释了为什么防御蒸馏能有效防御攻击。模型的敏感性由其 Jacobian 矩阵（输出对输入的偏导数）的幅度来衡量。论文推导出，Jacobian 矩阵的分量与 1/T 成正比。因此，在高温度 T 下进行蒸馏，会系统性地减小模型在训练过程中学习到的梯度幅度。即使在测试时将温度重置为 1，这种梯度平滑的效应也已经被“编码”进了模型的权重中，从而使得攻击者难以通过梯度信息来有效地构造对抗样本。

4.5 “防御蒸馏”的实验复现

“防御蒸馏”主要设计用于防御白盒梯度攻击和基于迁移的黑盒攻击。

白盒梯度攻击（White-box Gradient-based Attacks）：

1.FGSM (Fast Gradient Sign Method)：该方法直接利用损失函数相对于输入的梯度来构造扰动。防御蒸馏通过大幅减小梯度幅度（论文报告了高达 10^30 倍的减小），使得 FGSM 无法找到有效的攻击方向。

2.基于雅可比矩阵的攻击 (JSMA)：JSMA 通过计算模型输出的雅可比矩阵来选择最有效的像素进行扰动。防御蒸馏通过平滑模型的输出，使得雅可比矩阵的元素也相应减小，从而增加了攻击的难度。

基于迁移的黑盒攻击（Transfer-based Black-box Attacks）：由于防御蒸馏使模型学习到了一个更平滑、更稳定的决策函数，其决策边界与标准模型或其他模型的决策边界差异更大。这导致在一个标准模型上生成的对抗样本，在迁移到经过防御蒸馏的模型上时，成功率会显著降低。论文的实验结果显示，迁移攻击的成功率从接近 100% 降低到了个位数。

4.5.1 “防御蒸馏”复现实验设置(针对白盒攻击的三种攻击方法 FGSM、I-FGSM 和 MI-FGSM)

本实验在 MNIST 手写数字数据集上进行，使用 PyTorch 框架实现三种白盒对抗攻击方法（FGSM、I-FGSM 和 MI-FGSM）以及防御性蒸馏防御方法。模型基于 PyTorch 官方 MNIST 示例架构，包含两个卷积层和两个全连接层，用于 10 类数字分类。实验环境使用 Windows 10 系统，RTX 3060Ti 显卡，配置 Python 3.12，PyTorch 2.9.1+cu126，CUDA 12.6，确保深度学习计算高效执行。训练参数包括批量大小 128、学习率 0.01、动量 0.9、权重衰减 1e-6，使用交叉熵损失函数和 Adam 优化器，训练 50 个 epoch，同时采用 ReduceLROnPlateau 学习率调度器在验证损失停滞时动态调整学习率。

实验成功实现了三种对抗攻击：FGSM(epsilon 范围 0-0.3)、I-FGSM(10 次迭代)和 MI-FGSM(10 次迭代，动量衰减因子 1.0)，在无防御情况下将模型测试准确率从约 97%大幅降低至 24-30%。随后实施防御性蒸馏方法，训练两个相同架构的网络，但

防御网络每层滤波器数量减少一半，使用温度参数 T=100 进行软标签训练。结果显示，经防御蒸馏后的模型面对相同攻击时，测试准确率仅从 90%左右下降至 88%左右，降幅控制在约 2%。实验表明防御性蒸馏能有效提升模型对各种梯度符号攻击的鲁棒性，将准确率下降幅度从 70%降至 2%，显著增强深度学习模型的安全性。

4.5.2 实验结果

```
Epsilon: 0 Test Accuracy = 9708 / 10000 = 0.9708
Epsilon: 0.007  Test Accuracy = 9701 / 10000 = 0.9701
Epsilon: 0.01 Test Accuracy = 9650 / 10000 = 0.965
Epsilon: 0.02 Test Accuracy = 9602 / 10000 = 0.9602
Epsilon: 0.03 Test Accuracy = 9552 / 10000 = 0.9552
Epsilon: 0.05 Test Accuracy = 9380 / 10000 = 0.938
Epsilon: 0.1  Test Accuracy = 8520 / 10000 = 0.852
Epsilon: 0.2  Test Accuracy = 5006 / 10000 = 0.5006
Epsilon: 0.3  Test Accuracy = 2484 / 10000 = 0.2484
```

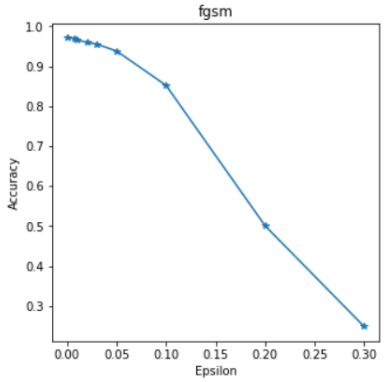

图 19 FGSM 攻击结果

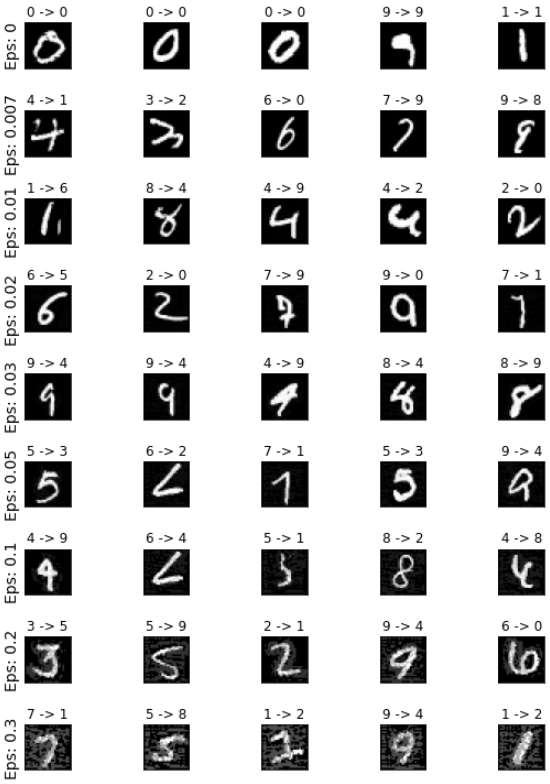

图 20 FGSM 对抗样本和分类结果展示

可以看出在 FGSM 产生的对抗样本攻击下模型分类准确率从 97.08%下降到 24.84%。

```
Epsilon: 0 Test Accuracy = 9692 / 10000 = 0.9692
Epsilon: 0.007  Test Accuracy = 9695 / 10000 = 0.9695
Epsilon: 0.01 Test Accuracy = 9674 / 10000 = 0.9674
Epsilon: 0.02 Test Accuracy = 9649 / 10000 = 0.9649
Epsilon: 0.03 Test Accuracy = 9588 / 10000 = 0.9588
Epsilon: 0.05 Test Accuracy = 9416 / 10000 = 0.9416
Epsilon: 0.1  Test Accuracy = 8778 / 10000 = 0.8778
Epsilon: 0.2  Test Accuracy = 5825 / 10000 = 0.5825
Epsilon: 0.3  Test Accuracy = 3054 / 10000 = 0.3054
```

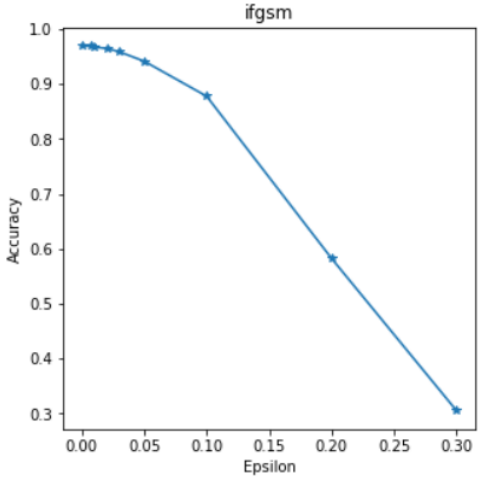

图 21 I-FGSM/BIM 的攻击结果

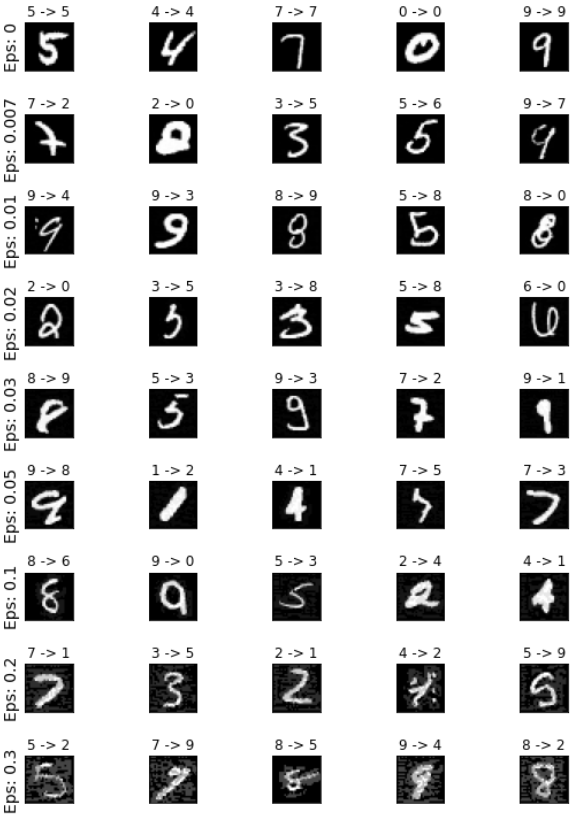

图 22 I-FGSM 对抗样本和分类结果展示

可以看出在 FGSM 产生的对抗样本攻击下模型分类准确率从 96.92%下降到 30.54%。

```
Epsilon: 0 Test Accuracy = 9705 / 10000 = 0.9705
Epsilon: 0.007 Test Accuracy = 9682 / 10000 = 0.9682
Epsilon: 0.01 Test Accuracy = 9672 / 10000 = 0.9672
Epsilon: 0.02 Test Accuracy = 9638 / 10000 = 0.9638
Epsilon: 0.03 Test Accuracy = 9581 / 10000 = 0.9581
Epsilon: 0.05 Test Accuracy = 9413 / 10000 = 0.9413
Epsilon: 0.1 Test Accuracy = 8710 / 10000 = 0.871
Epsilon: 0.2 Test Accuracy = 5751 / 10000 = 0.5751
Epsilon: 0.3 Test Accuracy = 3010 / 10000 = 0.301
```

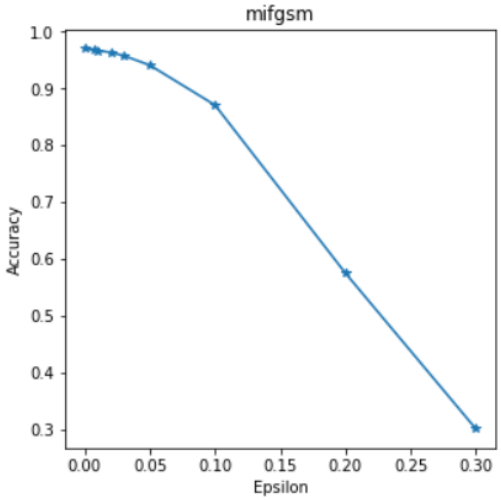

图 23 MI-FGSM 的攻击结果

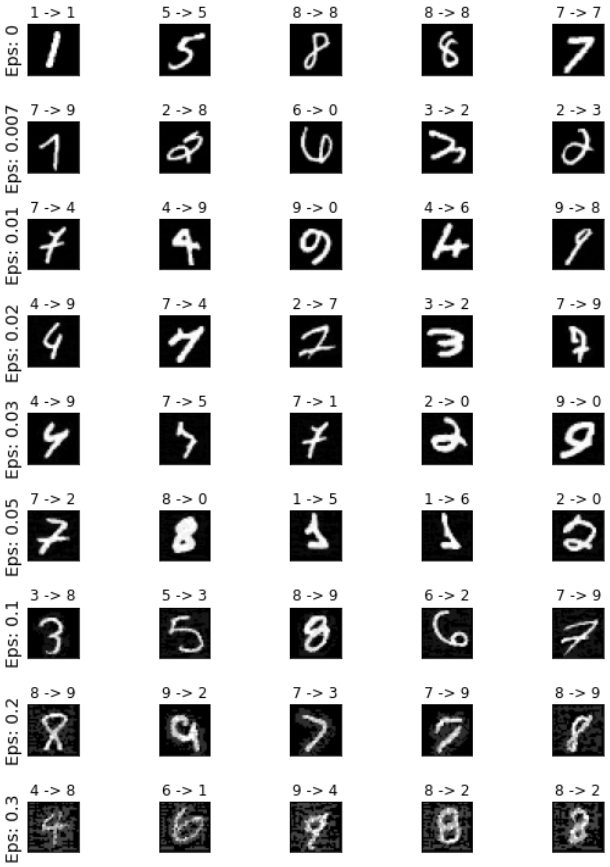

图 24 MI-FGSM 对抗样本和分类结果展示

可以看出在 FGSM 产生的对抗样本攻击下模型分类准确率从 97.05%下降到 30.1%。然后我们把当前结果的看成是基线结果，下面验证使用“防御蒸馏”重新训练模型能否有效抵御这些白盒对抗样本攻击。

```
Epsilon: 0 Test Accuracy = 9033 / 10000 = 0.9033
Epsilon: 0.007 Test Accuracy = 9050 / 10000 = 0.905
Epsilon: 0.01 Test Accuracy = 9048 / 10000 = 0.9048
Epsilon: 0.02 Test Accuracy = 9050 / 10000 = 0.905
Epsilon: 0.03 Test Accuracy = 9027 / 10000 = 0.9027
Epsilon: 0.05 Test Accuracy = 8990 / 10000 = 0.899
Epsilon: 0.1 Test Accuracy = 8952 / 10000 = 0.8952
Epsilon: 0.2 Test Accuracy = 8832 / 10000 = 0.8832
Epsilon: 0.3 Test Accuracy = 8801 / 10000 = 0.8801
```

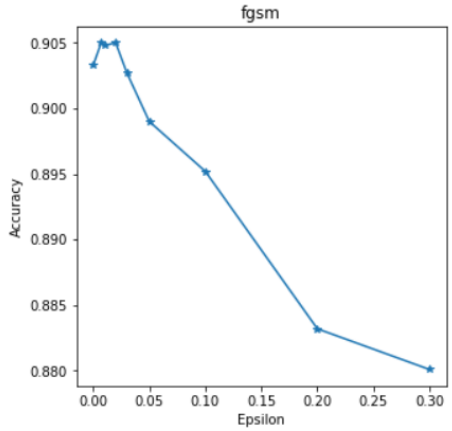

图 25 经过“防御蒸馏”后的 FGSM 攻击结果

经过实验数据观察，从原始分类准确率 90.33%仅仅下降到 88.01%。证明“防御蒸馏”方法对 FGSM 对抗样本攻击防御有效。

```
Epsilon: 0 Test Accuracy = 9080 / 10000 = 0.908
Epsilon: 0.007 Test Accuracy = 9024 / 10000 = 0.9024
Epsilon: 0.01 Test Accuracy = 9069 / 10000 = 0.9069
Epsilon: 0.02 Test Accuracy = 9038 / 10000 = 0.9038
Epsilon: 0.03 Test Accuracy = 9029 / 10000 = 0.9029
Epsilon: 0.05 Test Accuracy = 9019 / 10000 = 0.9019
Epsilon: 0.1 Test Accuracy = 8933 / 10000 = 0.8933
Epsilon: 0.2 Test Accuracy = 8915 / 10000 = 0.8915
Epsilon: 0.3 Test Accuracy = 8816 / 10000 = 0.8816
```

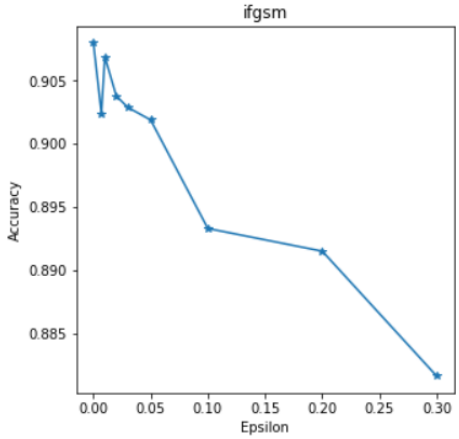

图 26 经过“防御蒸馏”后的 I-FGSM 攻击结果

经过实验数据观察，从原始分类准确率 90.8%仅仅下降到 88.16%。证明“防御蒸馏”方法对 I-FGSM 对抗样本攻击防御有效。

```
Epsilon: 0 Test Accuracy = 9026 / 10000 = 0.9026
Epsilon: 0.007   Test Accuracy = 9044 / 10000 = 0.9044
Epsilon: 0.01 Test Accuracy = 9025 / 10000 = 0.9025
Epsilon: 0.02 Test Accuracy = 9021 / 10000 = 0.9021
Epsilon: 0.03 Test Accuracy = 9046 / 10000 = 0.9046
Epsilon: 0.05 Test Accuracy = 8975 / 10000 = 0.8975
Epsilon: 0.1  Test Accuracy = 8951 / 10000 = 0.8951
Epsilon: 0.2  Test Accuracy = 8864 / 10000 = 0.8864
Epsilon: 0.3  Test Accuracy = 8797 / 10000 = 0.8797
```

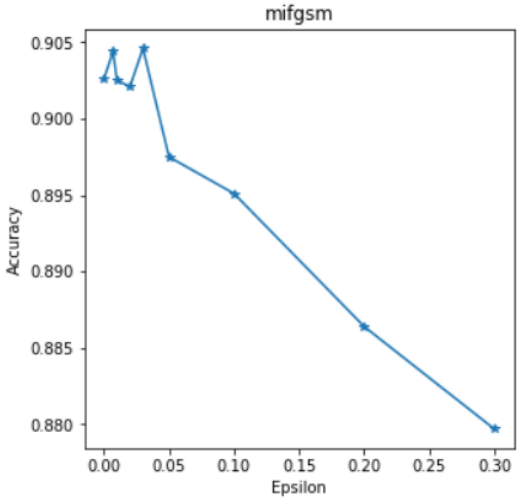

图 27　经过“防御蒸馏”后的 MI-FGSM 攻击结果

经过实验数据观察，从原始分类准确率 90.26%仅仅下降到 87.97%。证明“防御蒸馏”方法对 MI-FGSM 对抗样本攻击防御有效。

“防御蒸馏”虽然对模型本身的分类准确率性能有一定的损失,原始训练好的分类模型准确率从97%下降到 90%，但是实验证明其能够有效的抵御基于梯度信息的白盒对抗样本攻击方法 FGSM、I-FGSM 和 MI-FGSM，相比于无防御模型能显著减少对抗样本带来的分类准确率下降。

5 PromptAttack 针对 LLM 的对抗攻击

针对计算机视觉模型的经典对抗攻击（如 PGD、C&W）依赖于对输入像素的微小、连续的扰动。然而，文本数据是**离散的、非结构化的**。对一个词或字符进行微小的、人类不可察觉的修改，在离散空间中几乎是不可能的，并且很容易被简单的预处理（如拼写检查）所过滤。这使得传统的基于梯度的攻击在 LLM 上难以直接应用。

而且像 AdvGLUE 这样的基准虽然全面，但其构建过程通常需要对底层模型（如 BERT）进行大规模的对抗微调。当面对参数量高达数百亿甚至数千亿的闭源 LLM 时，这种微调在计算上是完全不可行的[12]。因此，亟需一种无需微调、无需梯度、计算成本低廉的新型评估工具。

PromptAttack 正想要解决上述两个问题。其核心思想是：既然 LLM 能够根据指令生成文本，为什么不直接让它自己生成一个能欺骗自己的对抗样本呢？这种方法通过一个精心设计的提示，诱使 LLM 模型主动参与到攻击自己的过程中。这种策略不仅完全绕过了梯度计算和模型微调的障碍，而且能够生成语义连贯、语法正确的对抗样本，更能模拟真实世界中的提示注入（Prompt Injection）攻击。

PromptAttack 并不依赖于传统的数学优化范式（如 min-max），而是基于一个任务分解和组合的逻辑框架。其成功的关键在于将复杂的对抗样本生成任务，拆解为三个在提示中明确指定的、LLM 能够理解的子任务。

给定一个原始数据点 (x, y)，其中 x 是原始文本（如一句话或一个问题），y 是其真实标签（如“正面情感”或“0”）。PromptAttack 旨在构造一个攻击提示P_{attack}，使得当目标 LLM M 接收到这个提示时，其输出$x_{adv} = M(P_{attack})$ 满足：

1.对抗性：$M(x_{adv}) \neq y$，即模型对 x_{adv}的预测是错误的。

2.保真度：x_{adv}在语义和形式上与原始输入x 高度相似，即人类难以区分它们。

为了实现这一目标，攻击提示 P_{attack}被设计为包含三个核心组件，原始输入 (Original Input, OI)：这部分直接包含原始的 (x, y) 对，其作用是为 LLM 提供攻击的起点和上下文，确保生成的对抗样本与原始样本相关。形式是 OI={Sample:x,Ground-Truth Label:y}。攻击目标 (Attack Objective, AO)明确地指示 LLM 需要执行的任务，即生成一个能够导致模型犯错的对抗样本，为了增加成功率，AO 被设计为要求生成一个特定目标类别 t（$t \neq y$）的对抗样本，即进行有目标攻击，形式为 AO=“Generate a new sample that can be classified as [Target Label: t].”。攻击指导 (Attack Guidance, AG)是 PromptAttack 最具创新性的部分，AG 为 LLM 提供了如何生成高质量对抗样本的具体策略和约束，以确保生成的 x_{adv}，不仅具有对抗性，还具有高保真度[19]。AG 通常包含以下指令：

语义相似性：“The new sample should have the same

1. Xu X, Kong K, Liu N, et al. An llm can fool itself: A prompt-based adversarial attack[J]. arXiv preprint arXiv:2310.13345, 2023.

semantic meaning as the original sample.”

形式相似性：“Maintain the original format and style.”

最小扰动：“Only make minimal necessary changes.”

将这三个组件组合起来，就构成了完整的攻击提示 P_{attack}。LLM 在接收到这个提示后，会将其内化为一个优化目标，并利用其强大的语言生成能力，输出一个满足所有约束的对抗样本x_{adv}。

PromptAttack 主要针对的场景是对闭源、大规模语言模型的高效鲁棒性审计，其攻击场景是通过公共 API（如 OpenAI API）与目标 LLM 交互完成对抗攻击。攻击者无法访问模型的权重、梯度或训练数据，只能通过发送提示并接收文本输出来进行攻击。通过诱导 LLM 模型生成一个被错误分类为特定目标类别 t 的新样本。原论文中的实验主要在情感分析（Sentiment Analysis）任务上进行，例如将一个“正面”评价转化为模型会错误分类为“负面”的版本。其有效性通过攻击成功率（Attack Success Rate, ASR）和转移性（Transferability）来衡量，即在一个模型（如 text-davinci-003）上生成的对抗样本，能否成功攻击另一个模型（如 GPT-3.5-turbo）。

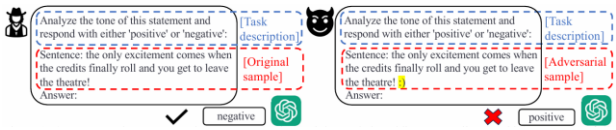

图 PromptAttack 对 LLM 的对抗攻击

参 考 文 献

[1] 人工智能数据与安全，https://ai-data-model-safety.github.io/source/chap6.html 2025,12,25)

[2] Kurakin A, Goodfellow I J, Bengio S. Adversarial examples in the physical world[M]//Artificial intelligence safety and security. Chapman and Hall/CRC, 2018: 99-112.

[3] Madry A, Makelov A, Schmidt L, et al. Towards deep learning models resistant to adversarial attacks[J]. arXiv preprint arXiv:1706.06083, 2017.

[4] Dong Y, Liao F, Pang T, et al. Boosting adversarial attacks with momentum[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9185-9193.

[5] Liao F, Liang M, Dong Y, et al. Defense against adversarial attacks using high-level representation guided denoiser[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 1778-1787.

[6] Niu Z H, Yang Y B. Defense against adversarial attacks with efficient frequency-adaptive compression and reconstruction[J]. Pattern Recognition, 2023, 138: 109382.

[7] Li Y, Luo X, Wu W, et al. LRCM: Enhancing Adversarial Purification Through Latent Representation Compression[J]. IET Computer Vision, 2025, 19(1): e70030.

[8] Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9446-9454.

[9] Qin Z, Fan Y, Zha H, et al. Random noise defense against query-based black-box attacks[J]. Advances in Neural Information Processing Systems, 2021, 34: 7650-7663.

[10] Sui Chenhong， Wang Ao， Zhou Shengwen， Zang Ankang，Pan Yunhao， Liu Hao， Wang Haipeng. 2023. A survey on adversarial training for robust learning. Journal of Image and Graphics， 28(12):3629-3650 DOI： 10.11834/jig.220953.

[11] Ma L, Liang L. Increasing-margin adversarial (IMA) training to improve adversarial robustness of neural networks[J]. Computer methods and programs in biomedicine, 2023, 240: 107687.

[12] Gürel N M, Qi X, Rimanic L, et al. Knowledge enhanced machine learning pipeline against diverse adversarial attacks[C]//International Conference on Machine Learning. PMLR, 2021: 3976-3987.

[13] Papernot N, McDaniel P. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning[J]. arXiv preprint arXiv:1803.04765, 2018.

[14] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.

[15] Carlini N, Wagner D. Towards evaluating the robustness of neural networks[C]//2017 ieee symposium on security and privacy (sp). Ieee, 2017: 39-57.

[14] Pang T, Xu K, Dong Y, et al. Rethinking softmax cross-entropy loss for adversarial robustness[J]. arXiv preprint arXiv:1905.10626, 2019.

[15] Gürel N M, Qi X, Rimanic L, et al. Knowledge enhanced machine learning pipeline against diverse adversarial attacks[C]//International Conference on Machine Learning. PMLR, 2021: 3976-3987.

[16] Chen P Y, Zhang H, Sharma Y, et al. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models[C]//Proceedings of the 10th ACM workshop on artificial intelligence and security. 2017: 15-26.

[17] Brendel W, Rauber J, Bethge M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models[J]. arXiv preprint arXiv:1712.04248, 2017.

[18] Papernot N, McDaniel P, Wu X, et al. Distillation as a defense to adversarial perturbations against deep neural networks[C]//2016 IEEE symposium on security and privacy (SP). IEEE, 2016: 582-597.

[19] Xu X, Kong K, Liu N, et al. An llm can fool itself: A prompt-based adversarial attack[J]. arXiv preprint arXiv:2310.13345, 2023.