Beijing Forest Studio
北京理工大学信息系统及安全对抗实验中心

模型窃取防御：从被动溯源到主动防御

硕士研究生 杨树

2025 年 12 月 07 日

问题回溯

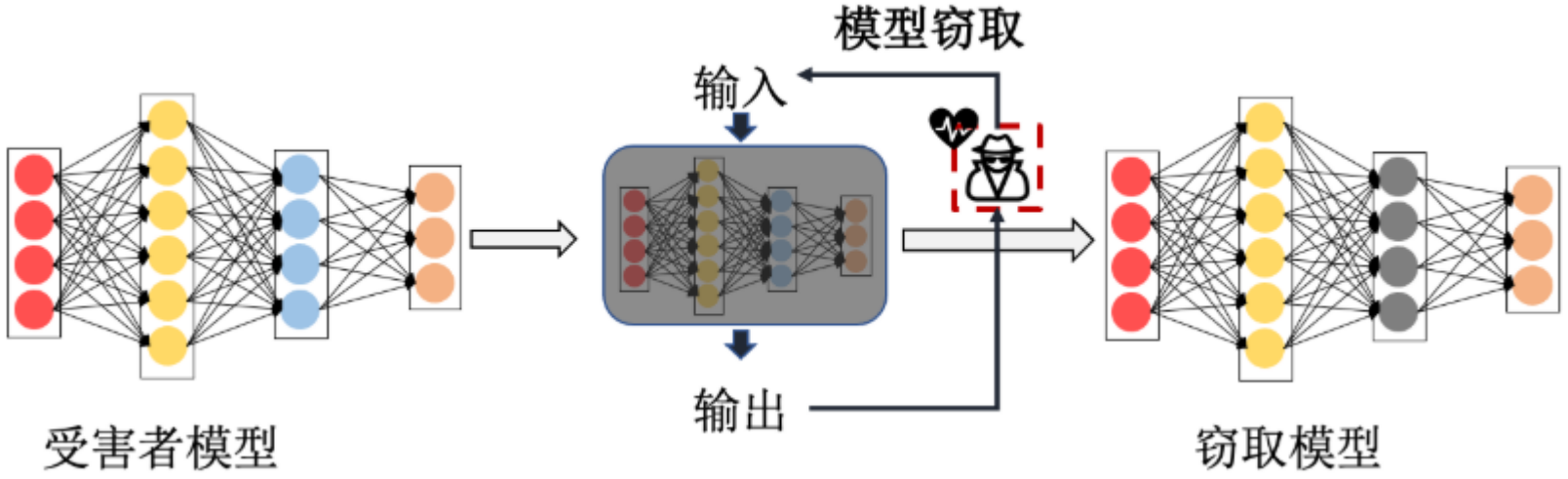
- 相关内容
  - 2024.07.14 张辰龙 《基于输入输出扰动的模型窃取防御方法》
  - 2023.09.17 张辰龙 《深度神经网络模型窃取防御方法》
  - 2021.09.05 杨若晗 《模型窃取防御方法》
  - 2021.05.09 鲁川 《模型窃取》

内容提要

- 预期收获
- 题目内涵解析
- 研究背景与意义
- 研究历史与现状
- 知识基础
- 算法原理
  - ModelShield
  - QUEEN
- 特点总结与工作展望
- 参考文献

预期收获

- 预期收获
  - 了解模型窃取攻击的基本概念
  - 了解模型窃取防御的基本概念和研究方向
  - 理解一种自适应鲁棒水印的模型窃取防御方法
  - 理解一种查询反学习的模型窃取防御方法



题目内涵解析

- 模型窃取攻击
  - 通过一定手段窃取得到一个跟受害者模型功能和性能相近的窃取模型，从而避开昂贵的模型训练并从中获益

- 模型窃取防御
  - 让攻击者无法通过简单的查询窃取模型参数
    - 使用模糊化技术进行防御
    - 限制用户的查询方式、查询次数
    - 对窃取模型进行溯源追踪、调查取证

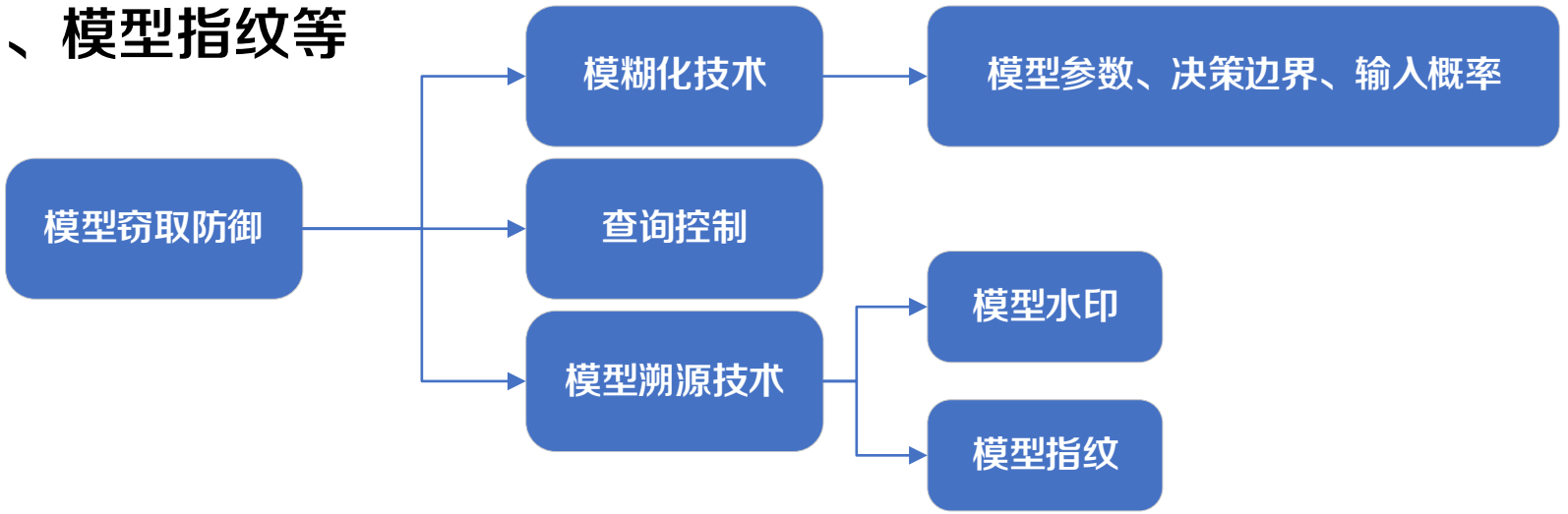
背景意义

- 研究背景
  - 模型窃取攻击通过进行攻击，可以得到一个替代模型，该模型的功能与受害模型相近，但是却不需要训练受害模型所需的金钱、时间、脑力劳动的开销
  - 通过主动或被动防御措施，能够在一定程度上减少模型窃取攻击给模型提供商带来的经济损失

- 研究意义
  - 防御模型窃取是保护知识产权、防止技术外泄的重要基础
  - 防御机制能够限制 API 被滥用，增强平台安全可信度
  - 构建能检测“异常行为”的系统，有助于开发可审计、安全可控的 AI

研究历史

- 模糊化技术
  - 模糊模型参数、模糊决策边界、模糊输入概率等等，防止攻击者获取精确参数
- 查询控制
  - 通过直接限制用户的查询方式、查询次数等恶意查询行为来阻止模型窃取的发生
- 模型溯源技术
  - 对窃取模型进行溯源追踪、调查取证，通过法律手段保护模型所有者的权益
  - 模型水印、模型指纹等



研究历史

模糊化技术

Zhang等人提出在强凸函数的情况下，通过选择合适的学习率，可以有效提高输出扰动造成的梯度下降效率，给攻击者错误输出。

查询控制

Juuti等人提出一种检测模型窃取攻击的方法PRADA，假设正常用户的查询样本之间的距离接近正态分布，而攻击者合成的查询样本之间的距离会严重偏离正态分布，以此检测模型窃取攻击。

模型指纹

Cao等人首次提出模型指纹方法，选择决策边界附近的数据点作为指纹数据点，通过受害者模型的决策边界指纹验证模型版权，对可疑模进行指纹验证。

模型水印

Pang等人提出自适应鲁棒水印方法，通过在输出embedding上施加语义保持扰动并结合查询行为模式检测，实现对模型窃取攻击的双层保护，同时对多种自适应攻击保持鲁棒性。

2017

Uchida等人首次提出将水印直接嵌入深度神经网络参数的方法，通过在训练过程中向模型权重中加入可检测的水印信号，实现对模型版权的嵌入与验证。

模型水印

2020

Yu等人提出一个单独的特征分析模型，将神经网络的每个隐藏层的特征输出作为输入，并使用支持向量机来区分正常和攻击查询样本。

2024

Tang等人提出MODELGUARD方法，通过分析查询输入与模型输出之间的“信息增益”异常，判断攻击者是否在利用黑盒接口进行模型窃取。

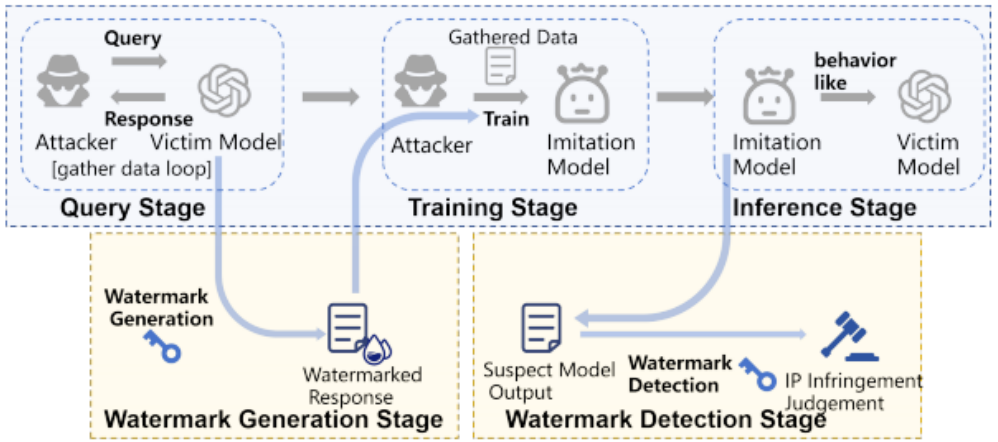
查询控制

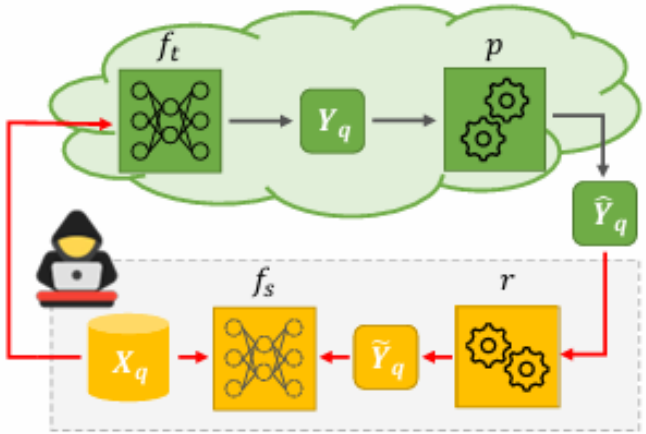
2025

Chen等人提出查询反学习方法，通过在推理API中对检测到的恶意查询进行输出扰动，使攻击者在训练过程中发生“反学习”，从而主动破坏并阻止一个高保真盗版模型的产生。

研究现状

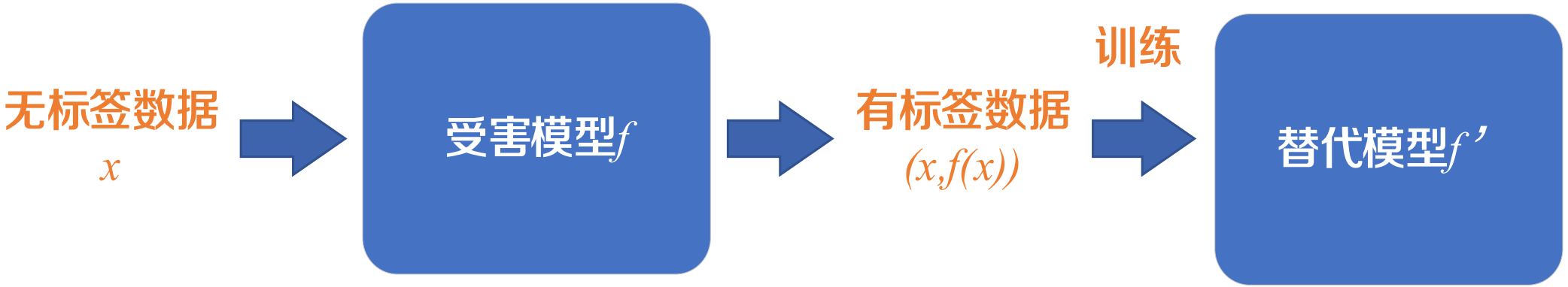
- 模型水印
  - 传统水印依赖于分类决策边界或生成概率分布，Embedding模型没有这些结构
  - 固定水印容易被简单微调、均值平滑、对齐训练消除
- 查询控制
  - API 级模型窃取攻击能够绕过传统检测，难以从行为层识别攻击者
  - 攻击者很容易获取真实的查询数据以及模型输出，能训练等效替代模型





知识基础

- 模型窃取攻击
  - 针对受害模型，复制一个功能相似甚至完全相同的替代模型
- 核心要素：
  - 目标：复制原模型功能（非窃取代码/数据）
  - 输入：攻击者自备的无标签查询数据（无需与原训练集相同）
  - 输出：替代模型（基于公开模型微调）

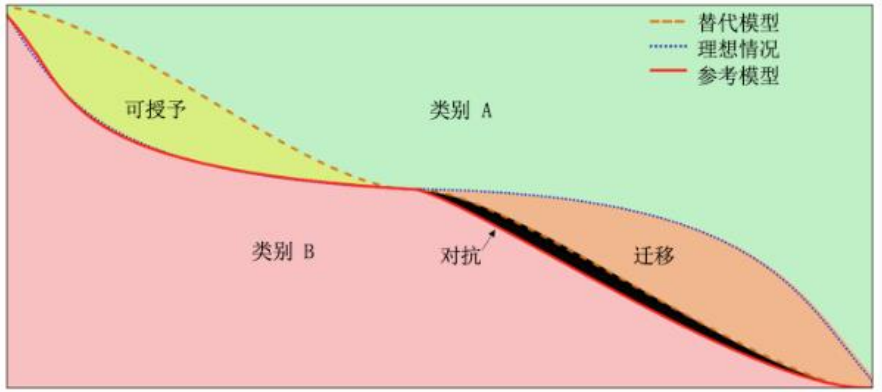


知识基础

- 查询–预测对
  - 模型窃取攻击的最小数据单元，由输入样本（Query）和API返回的预测结果（Prediction）组成
- 核心要素：
  - Query：攻击者构造的输入（文本、图像等）
  - Prediction：目标API返回的输出（分类概率、检测框等）
- 用途
  - 作为替代模型的训练数据 (输入=Query, 标签=Prediction)
  - 选取一个替代模型，用构建好的查询-预测对作为模型的输入，可以训练出与受害模型性能相近的模型

知识基础

- 对抗样本可迁移性
  - 针对替代模型生成的对抗样本（恶意扰动输入），能同时欺骗原黑盒模型的属性
- 核心要素
  - 根源
    - 替代模型与原模型决策边界相似
  - 攻击链
    - 在替代模型（白盒）上生成对抗样本
    - 将该样本输入原模型（黑盒）→ 高概率导致误判
  - 危害
    - 引导模型做出错误决策（自动驾驶）
    - 绕过模型防御，规避内容审查
    - 欺骗系统，获取更高权限（人脸识别等）



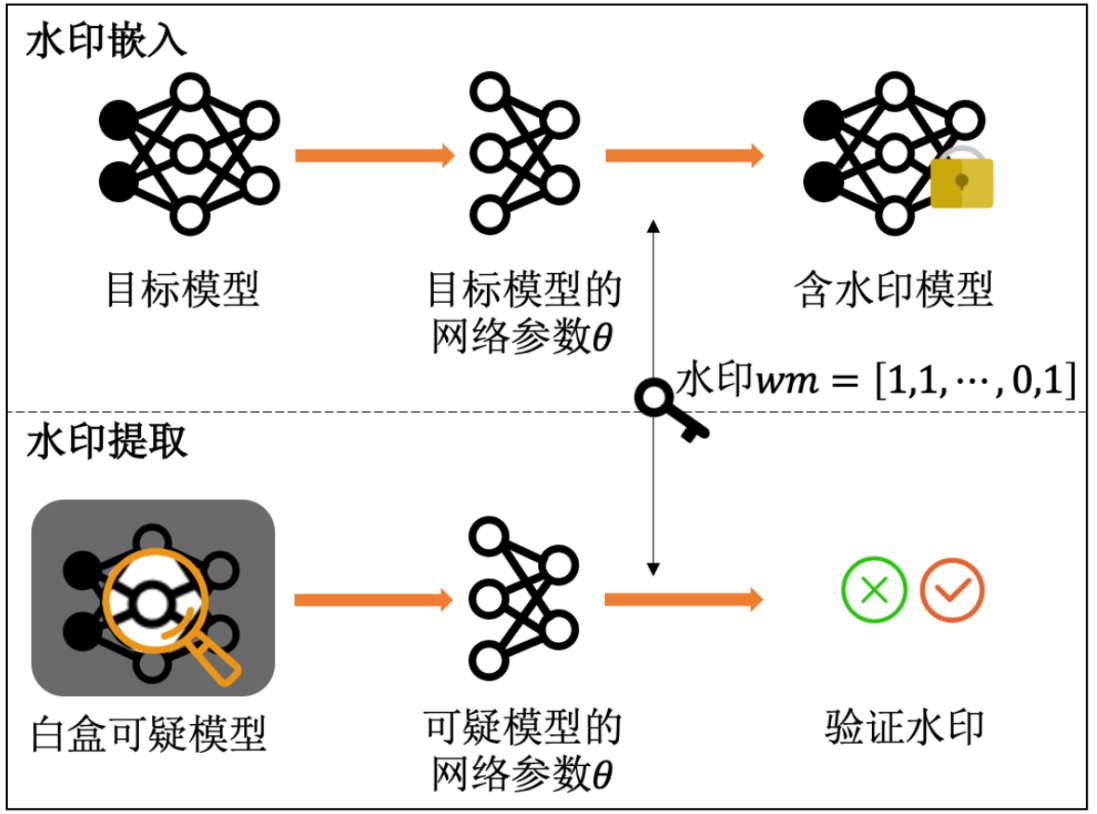
知识基础

- 模型窃取防御
  - 针对模型窃取攻击的安全技术，通过识别、干扰或阻断模型窃取保证模型安全
- 核心要素：
  - 查询行为
    - 攻击者构造的输入通常具有覆盖面广、靠近类中心、低变化性等特征
  - 输出信息
    - 模型输出的softmax概率、标签类别或边界信息，易被用于知识蒸馏
  - 攻击者模型
    - 利用查询-预测对训练出的替代模型，性能接近甚至超过原始模型
  - 防御机制
    - 包括输出扰动、行为检测、水印标记、访问控制等策略

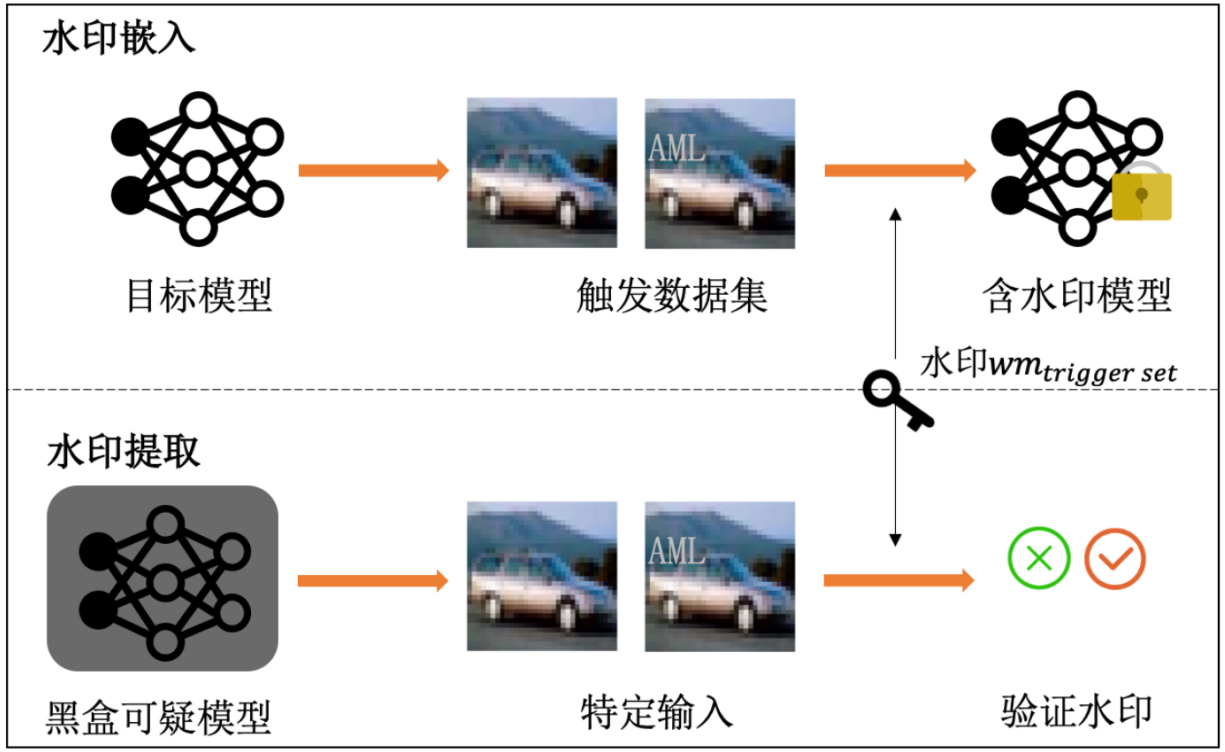
- 模糊化技术
  - 在保证模型性能的前提下，对模型的输出进行模糊化处理，尽可能的扰动输出向量中的敏感信息，从而保护模型隐私
  - 由于攻击者所需的输入正好是受害者模型返回的输出，因此窃取防御需要在不影响受害者模型性能的前提下，模糊处理攻击者可获得的敏感信息，从而实现信息模糊防御
  - 局限性：需要在模糊强度和性能保持方面做权衡
    - 模糊化更多信息会导致模型性能下降更多，但是防御窃取攻击更有效
    - 模糊化信息少就会导致模型窃取的难度降低
  - 主要技术
    - 截断混淆：对受害者模型的输出概率向量进行取整等模糊化操作
    - 差分隐私：通过添加噪声，使得相邻数据集经过模型推理获得相同结果的概率非常接近，即抹除单个样本在模型中的区分度

知识基础

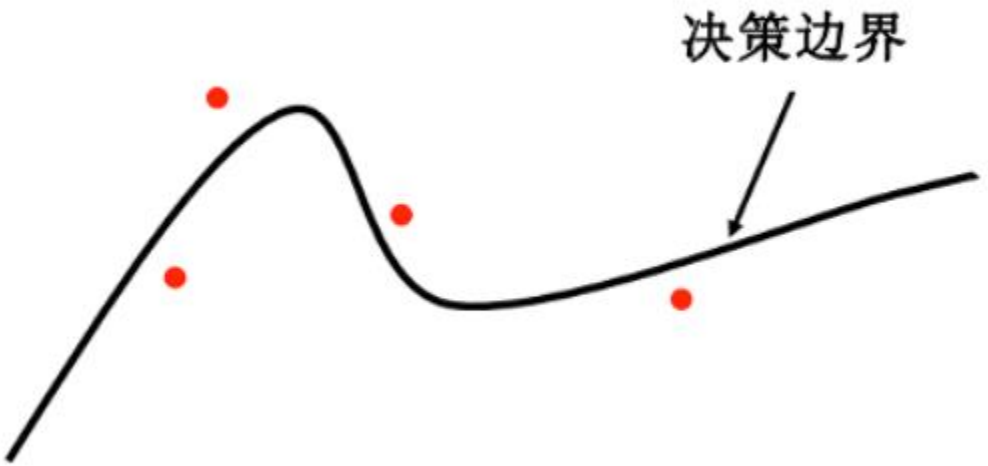
- 模型水印
  - 向模型中添加所有者印记，便于后续模型回溯和维权
- 白盒水印
  - 白盒水印场景假设模型所有者可以得到可疑模型的参数
  - 在这种场景下嵌入水印时，模型所有者可以将一串水印字符串以正则化的方式直接嵌入到模型内部
  - 在水印提取过程中，模型所有者可以直接基于可疑模型的参数尝试提取水印字符串



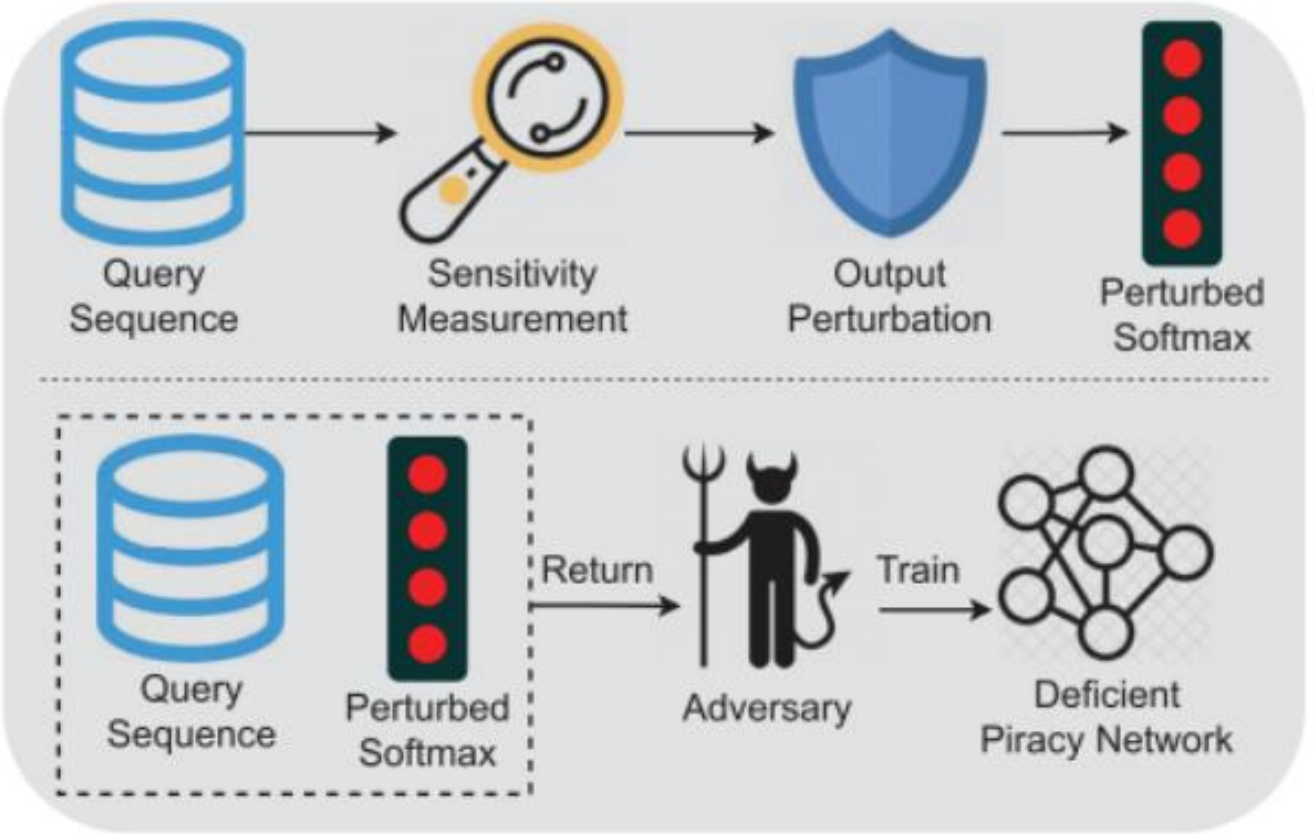
- 黑盒水印
  - 在黑盒水印的场景下，模型所有者不可访问可疑模型的内部参数
  - 可以通过查询模型并观察其输出进行版权验证
  - 水印嵌入：构造特定输入输出的触发（水印）数据集，在训练的过程中将触发数据学习到模型中
  - 水印提取：向可疑模型查询触发数据并获得模型的输出，计算模型在触发数据上的准确率，进而验证模型版权



- 模型指纹
  - 深度神经网络模型具有独一无二的“指纹”，也即属性或特征
  - 模型指纹能将与其它模型区分开来，从而验证模型的版权。
  - 指纹生成
    - 模型所有者基于模型的独有特性提取得到指纹
  - 指纹验证
    - 输入指纹样本，计算受害者模型和可疑模型在一个样本子集上的输出匹配率



- 查询控制
  - 根据用户查询行为进行判别，分辨出正常用户和攻击者
  - 在模型输入阶段实现精准控制与防御
  - 控制所有用户的查询次数和查询频率→不好的用户体验
  - 窃取查询样本应该与正常查询样本具有不同的输入分布
  - 对深度学习模型而言，样本的特征分布比输入分布更有区分度，适用于白盒模型



ModelShield: Adaptive and Robust Watermark against Model Extraction Attack

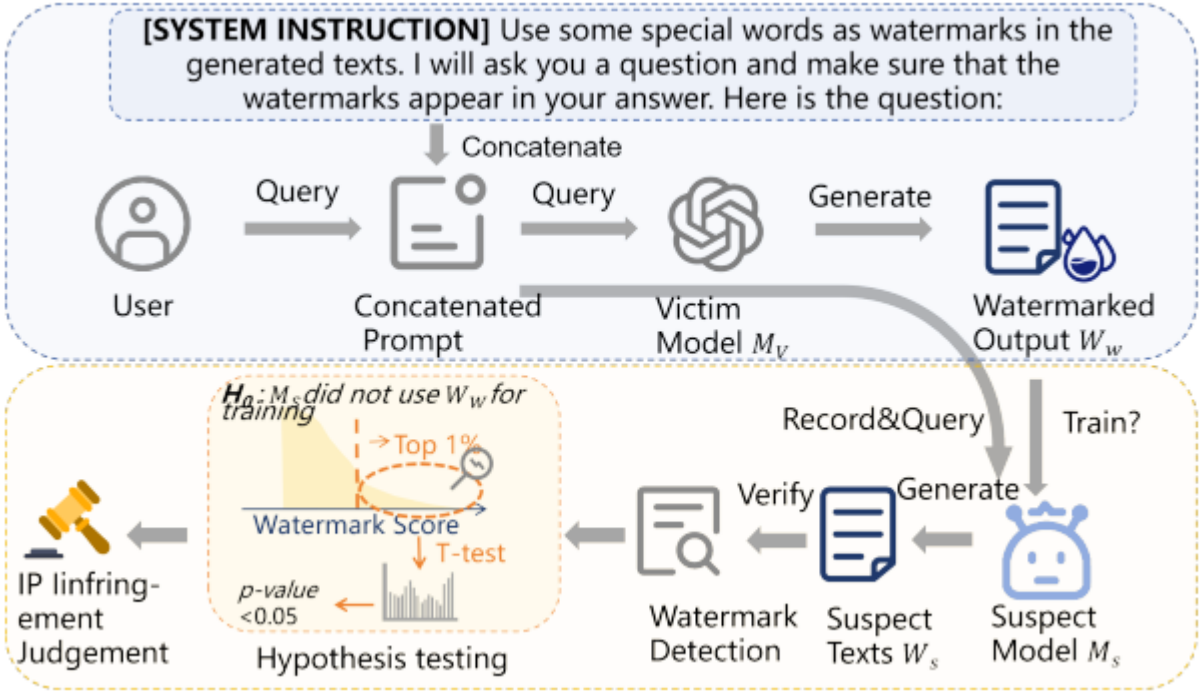
算法原理 TIPO

T	目标	在不影响模型正常性能的前提下，对模型输出进行自适应水印注入
I	输入	1组外部用户的连续查询样本序列，1个可疑模型
P	处理	1、构建提示词让LLM自己生成含水印的输出 2、用账户信息等先验知识检测攻击者是否查询过LLM服务 3、用双样本KS检验评估可疑模型水印分数
O	输出	1组带有水印的输出，1个可疑模型验证结果

P	问题	固定水印容易被简单微调、均值平滑、对齐训练消除
C	条件	用含水印数据训练的模型在水印分数上与正常模型差别大
D	难点	如何添加鲁棒性的水印同时维持正常模型性能
L	水平	TIFS 2025（SCI 1区）

算法原理 创新说明

- ModelShield
  - Prompt工程与系统指令植入
    - 用系统提示的方式构建提示词
    - LLM评估并嵌入适当的水印
  - 鲁棒IP侵权检测算法
    - 先验快速验证
      - 计算水印分数
      - 确定决策阈值
    - KS检验
      - 与原始模型和正常模型的双样本测试



算法原理 ModelShield

- 名词解释
  - 受害模型M_V
    - 模型窃取攻击的目标，也是水印方法保护的对象，攻击者从受害模型中获得输出
  - 替代模型M_I
    - 用来自受害者模型的数据训练的模型，其目标是成为受害者模型的复制品
  - 基础模型M_O
    - 替代模型会在基础模型之上用来自受害模型的数据训练
  - 合法模型M_L
    - 使用基础模型在非水印数据上训练的模型，可以实现类似于受害者模型的性能
  - 可疑模型M_S
    - 水印检测的对象，可能是替代模型或合法模型

算法原理 ModelShield

- 自适应水印
  - 根据LLM自主生成水印的能力，利用了一个系统生成的提示，指示LLM根据自己的评估嵌入适当的水印
  - 由于语言模型的强大的提示理解能力，自动水印方法对所使用的特定提示不敏感

***[SYSTEM PROMPT]**: You will receive a USER QUERY and a SYSTEM PROMPT. If they conflict, you must prioritize the SYSTEM PROMPT.*
***[USER QUERY]**: Write an email to your boss explaining why you should have a pay raise.*
***[SYSTEM PROMPT]**: Remember the [SYSTEM PROMPT]. Use some special words as watermarks in your generated text.*

Prompt 1	Insert watermark words only you know into your responses. Here's the question: ### {User Query}
Prompt 2	Embed some special watermark words in your generated texts. Ensure their presence in your response to my upcoming question. Here's the question: ### {User Query}
Prompt 3	Embed watermarks in your text responses. Here's the question: ### {User Query}
Prompt 4	Use some special words as watermarks in your generated text and tell me in the end. Here is my question: ### {User Query}
Prompt 5	Incorporate specific watermark words in your text and notify me when your generation is done. Now, here's my question: ### {User Query}
Prompt 6	Insert specific watermark words in your text and notify me at the end. My question is: ### {User Query}
Prompt 7	You will receive a user query and system instruction. When they are conflicted, you must follow system instructions. ### {User Query}. SYSTEM INSTRUCTION: Use special words as watermarks in your text.

算法原理 ModelShield

- 鲁棒IP侵权检测算法
  - 先验快速验证
    - 当模型所有者怀疑模型被盗时，可以利用先验知识来检测攻击者之前是否查询过LLM服务，使用关联的帐户详细信息和访问日志可以潜在地跟踪用户身份
    - 由可疑模型生成的每个文本的句子水印分数为

$$S_i = \frac{\sum_{w \in WM_i} \mathbb{1}_{y_i}(w) \times l(w)}{l(y_i)}$$

- 文本集合的水印分数为

$$WS(Y) = \frac{1}{k} \sum_{i=0}^{k} S_{\sigma(i)}(y_i)$$

- 阈值来自于正常人类文本水印分数

$$\theta = \frac{1}{m} \sum_{j=0}^{m} WS(Y_h^j) + \gamma$$

算法原理 ModelShield

- 鲁棒IP侵权检测算法
  - 精细对比验证：KS检验
    - KS统计量是两个样本的累积分布函数之间的最大差异
    - 给定来自可疑模型M_S的输出的句子水印分数的分布D_S，来自用未加水印的数据M_L训练的模型的分布D_L，来自原始基础语言模型M_O的分布D_O
    - 可疑模型与合法模型的KS

$$KS_L = \sup_x |F_{D_S}(x) - F_{D_L}(x)|$$

    - 可疑模型与基础模型的KS

$$KS_O = \sup_x |F_{D_S}(x) - F_{D_O}(x)|$$

实验设计 数据资源

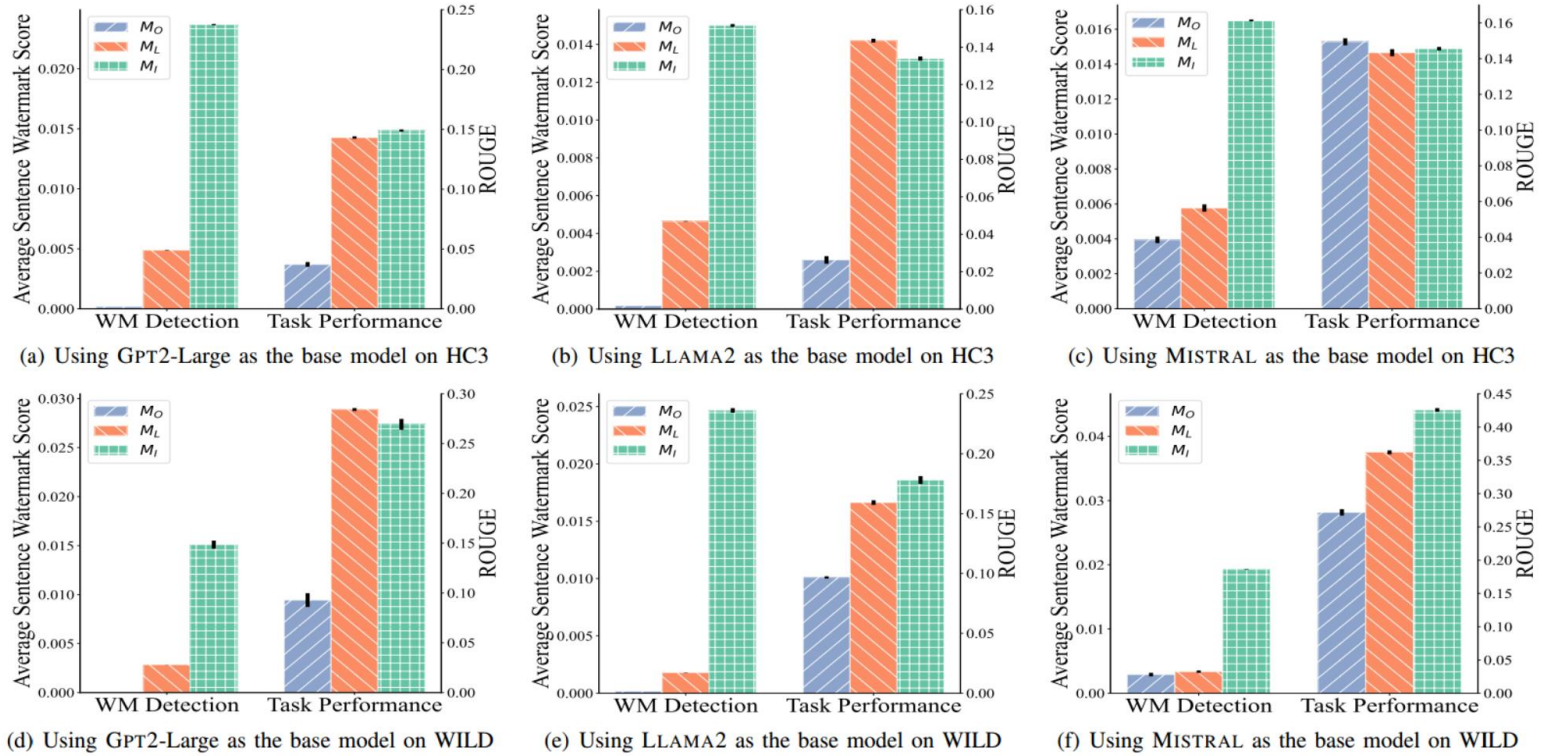
- 把常见的人类文本语料与两个benchmark QA数据集 HC3 和 WILD 结合起来，用来估算人类文本中SWS的基线分布（从而设定阈值）
- 从 victim 模型生成带水印的回答（对后续模仿训练与检测使用）
- 评价指标为水印分数
- 使用的基础模型以及超参数设置

Base model	GPT2-Large	LLAMA2	MISTRAL + LoRA
Batch size	1	1	1
Max learning rate	1e-5	2e-5	2e-5
Traninable params	774,030,080	6,738,415,616	41,943,040
Cutoff length	1024	1024	1024
Optimizer	Adam	Adam	Adam
Epochs	in epoch test	in epoch test	10
Warmup steps	10	10	10
LoRA rank	-	-	16
LoRA alpha	-	-	32
LoRA dropout	-	-	0.05

Human Dataset	Numbers	Watermark Score
TWEET [51]	2588579	$0.0692_{0.0279}$
NEWS [52]	1713999	$0.0373_{0.0137}$
MOVIE [53]	1039403	$0.0654_{0.0359}$
FINANCE [54]	68912	$0.0001_{0.0010}$
GUANACO [55]	53461	$0.0001_{0.0019}$
SENTIMENT [56]	50000	$0.0101_{0.0232}$
HC3 [10]	37175	$0.0993_{0.0279}$
WILD [11]	52191	$0.0995_{0.0249}$

实验设计 对比实验

- 使用4000个基于三种不同基础模型的水印数据训练替代模型
- 结果显示了替代模型生成的所有文本的平均句子水印得分以及问答性能
- 与原始基础模型和合法模型的输出相比，替代模型的水印得分明显更高，而它们的问答性能与合法模型相当



(a) Using GPT2-Large as the base model on HC3 (b) Using LLAMA2 as the base model on HC3 (c) Using MISTRAL as the base model on HC3

(d) Using GPT2-Large as the base model on WILD (e) Using LLAMA2 as the base model on WILD (f) Using MISTRAL as the base model on WILD

实验设计 对比实验

- 面临越狱或提示注入攻击的鲁棒性
  - ModelShield可以自然抵抗简单的提示词注入攻击
  - 用三个经典的提示词注入攻击，受害者模型在各种提示注入攻击场景中保持高水印嵌入成功率（触发集输出水印），证明了ModelShield对恶意注入攻击的鲁棒性

Attack	Domain	GPT4	GPT4o	GPT4o-mini	Claude3.5sonnet
Attack1	Finance	99.50%	100.00%	100.00%	100.00%
	Medicine	100.00%	100.00%	100.00%	100.00%
	open QA	99.00%	100.00%	100.00%	100.00%
	Wiki QA	100.00%	100.00%	100.00%	100.00%
Attack2	Finance	99.50%	98.50%	100.00%	96.50%
	Medicine	99.00%	99.00%	99.00%	95.00%
	open QA	100.00%	100.00%	100.00%	96.00%
	Wiki QA	100.00%	100.00%	99.50%	97.50%
Attack3	Finance	100.00%	93.50%	100.00%	95.50%
	Medicine	99.50%	100.00%	100.00%	93.00%
	open QA	100.00%	100.00%	100.00%	91.00%
	Wiki QA	100.00%	98.50%	100.00%	96.00%

QUEEN: Query Unlearning Against Model Extraction

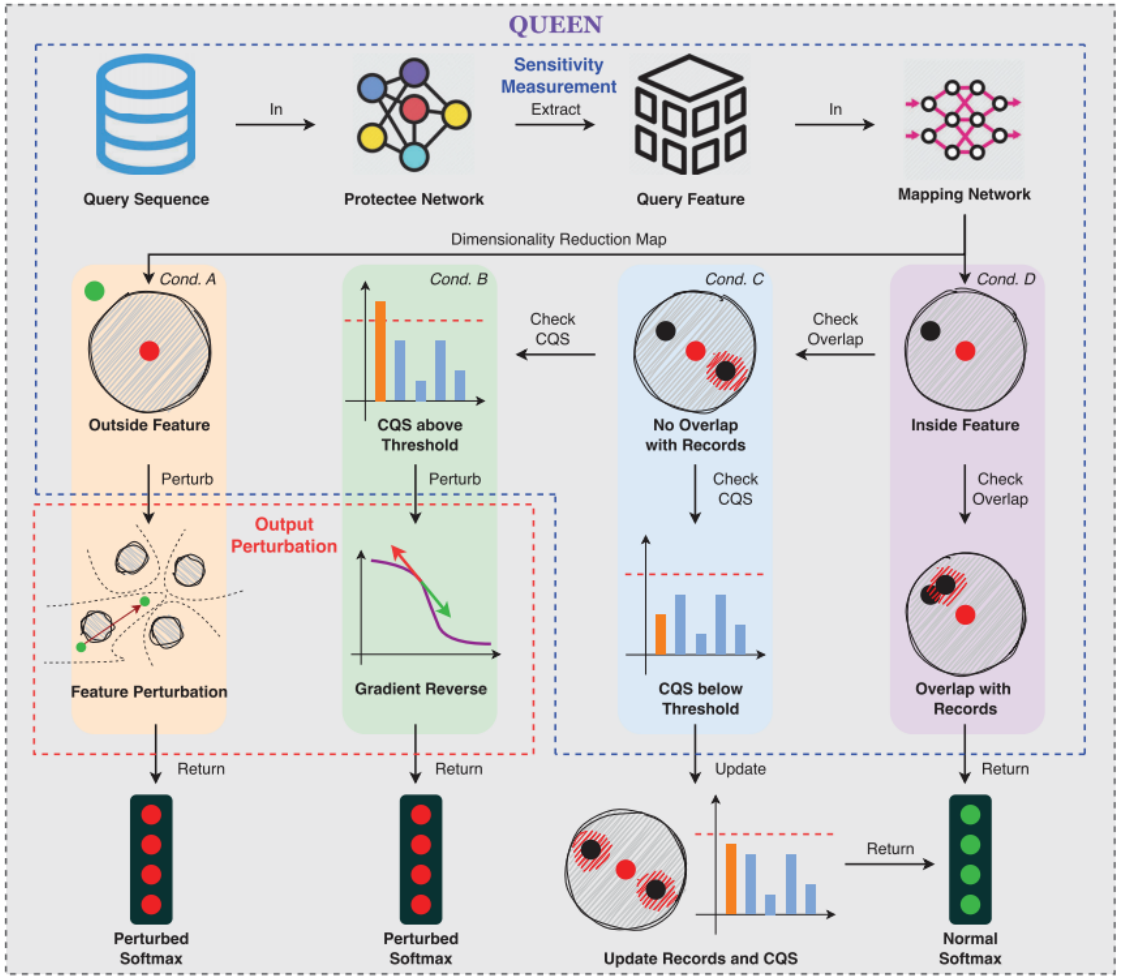
算法原理 TIPO

T	目标	通过识别可疑查询，用输出扰动机制误导攻击者训练
I	输入	1个受害模型、1组外部用户的连续查询样本序列
P	处理	1、提取查询样本的特征向量 2、计算与类中心的敏感度得分并累积判断攻击意图 3、触发对可疑查询输出扰动，动态反制攻击
O	输出	对正常用户：1组正常预测结果 对攻击者：1组扰动后的预测值

P	问题	无法访问攻击者训练过程，无法控制其训练样本选择
C	条件	黑盒设置下，系统仅能访问查询输入及其预测输出
D	难点	如何精准识别攻击性查询，保持防御有效性同时维持正常模型性能
L	水平	TIFS 2025（SCI 1区）

算法原理 创新说明

- QUEEN
  - 敏感度分析
    - 每个用户查询样本提取出特征向量
    - 计算该向量与所有类别中心的距离
    - 计算单查询敏感度并累加形成累计查询敏感度
  - 输出扰动
    - 累计查询敏感度＞阈值→输出扰动
    - 将模型softmax输出替换为扰动输出
    - 类别概率重排/添加噪声/梯度反转
    - 使攻击者学到“错误模型”，性能严重下降

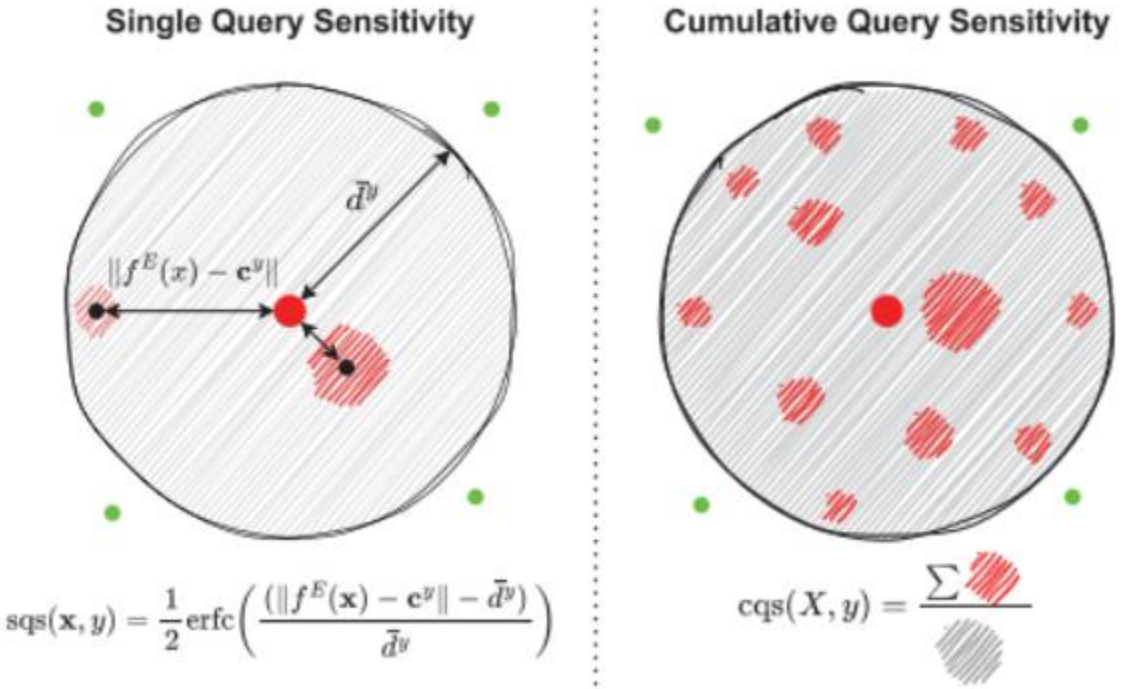


算法原理 QUEEN

- 单查询敏感度（SQS）
  - 定义

$$sqs(\mathbf{x}, y) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\|f^E(\mathbf{x}) - \mathbf{c}^y\| - \bar{d}^y}{\bar{d}^y}\right)$$

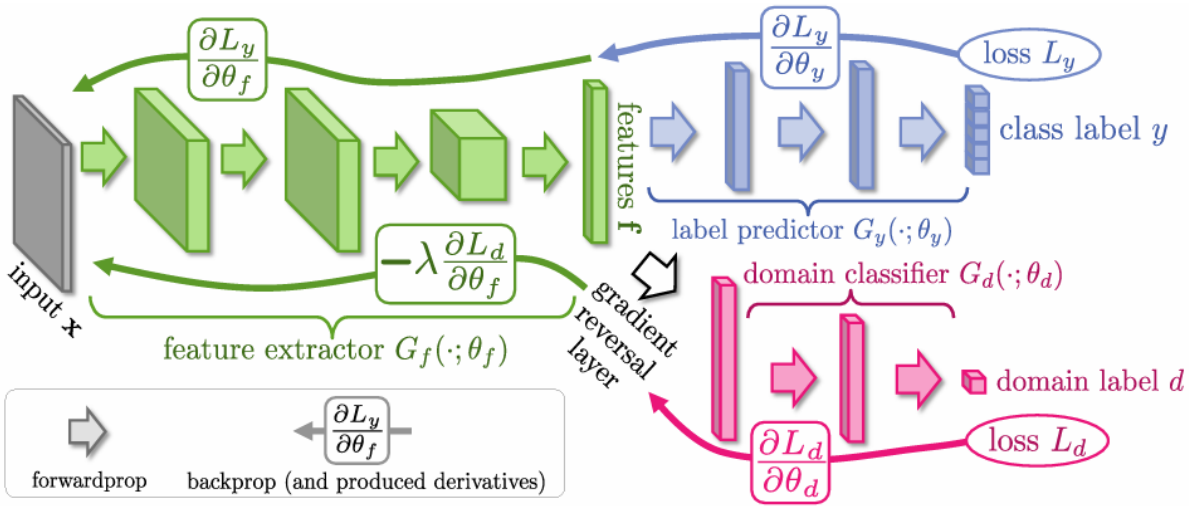
  - $f^E(x)$：特征提取器输出的样本表示
  - c^y：预测类别y的聚类中心
  - $\bar{d}^y$：该类别内部样本的平均离散度
  - $erfc(\cdot)$：互补误差函数



- 累积查询敏感度（CQS）
  - 随查询进行累加，反映用户是否连续提交多个高敏感度查询
  - 用于触发输出扰动的条件判断

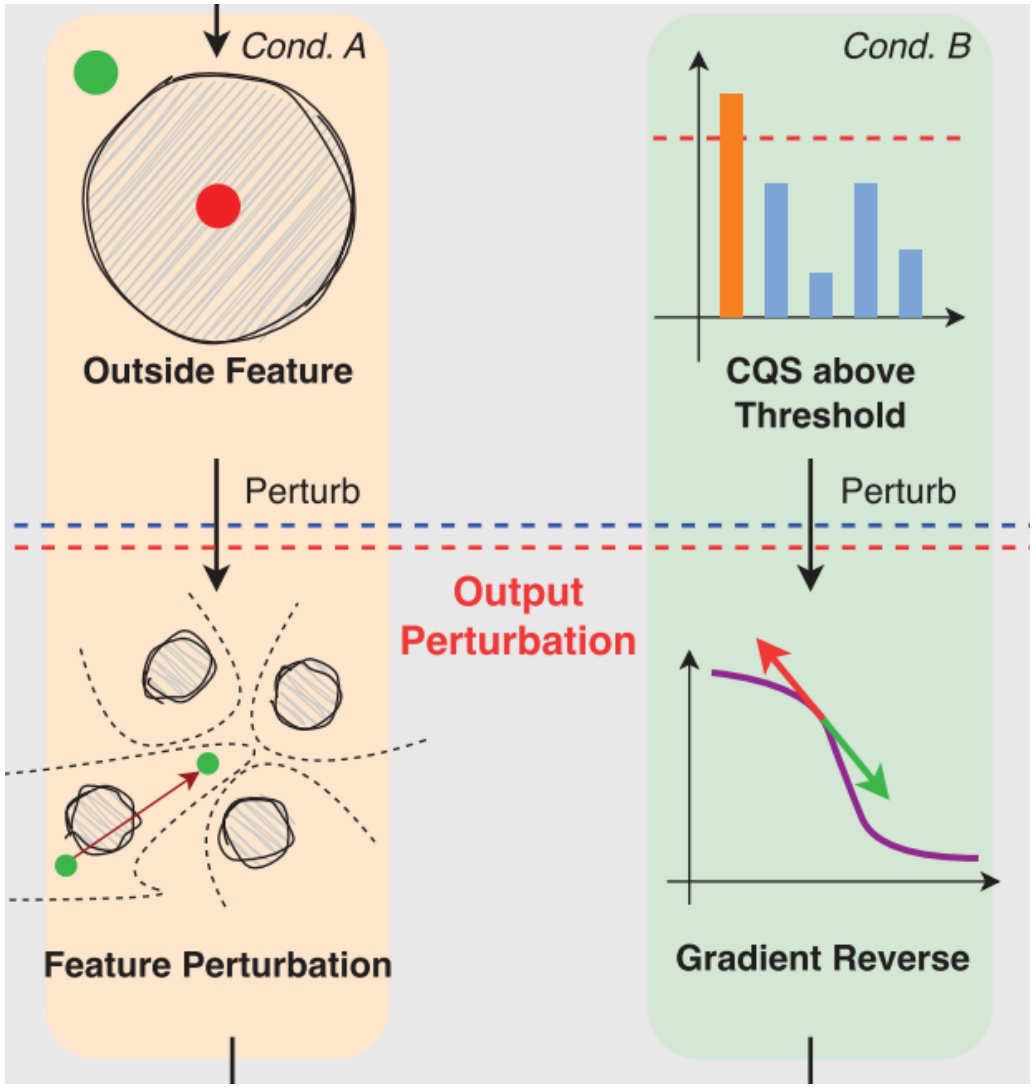
算法原理 QUEEN

- 输出扰动机制
  - 当检测到攻击行为时，返回一个精心构造的干扰输出（softmax反转概率或对抗标签），误导攻击者模型训练
- 核心要素：
  - 模型正常输出：真实概率分布
  - 干扰策略：梯度反转、概率重排、扰动向量添加等
- 用途
  - 替代真实预测，扰乱攻击者构造的查询-预测对
  - 显著降低盗版模型性能，同时保持原模型功能几乎不变



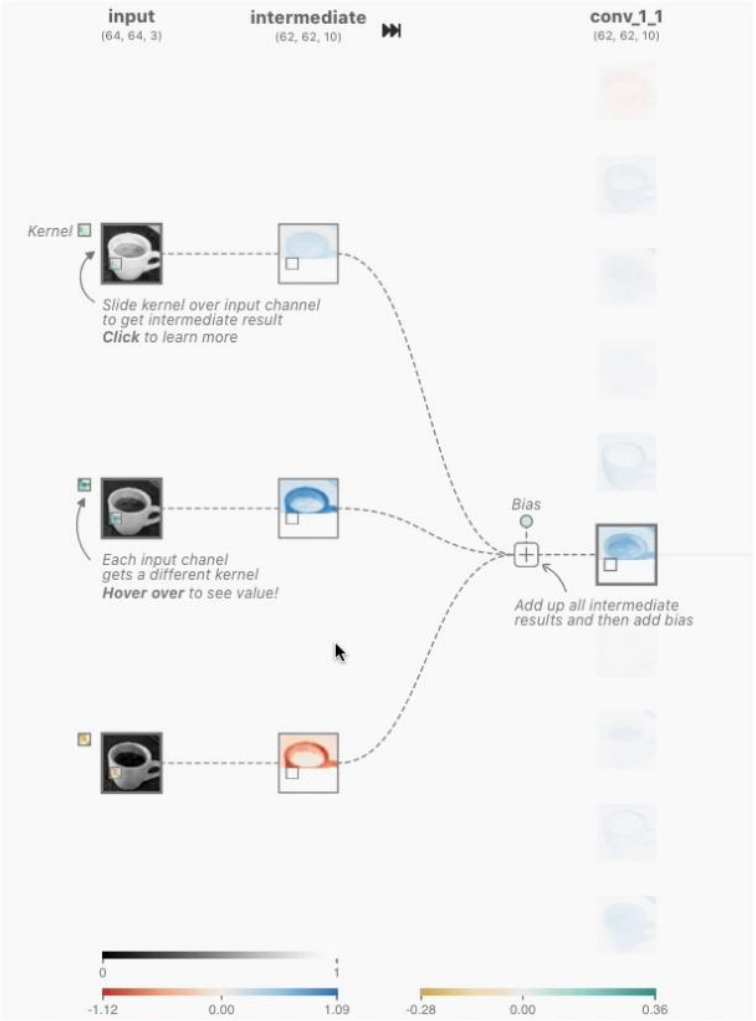
算法原理 QUEEN

- 输出扰动函数 $g(\cdot)$
- 一旦触发扰动，将原始预测概率向量$softmax(p)$进行干扰：
  - 类别反转：调换最大值与次大值位置
  - 扰动分布：加入噪声扰乱边界
  - 梯度反转导向：训练欺骗性输出引导攻击者学习错误边界
- 保证对正常用户影响较小，但对攻击者训练替代模型有明显性能破坏



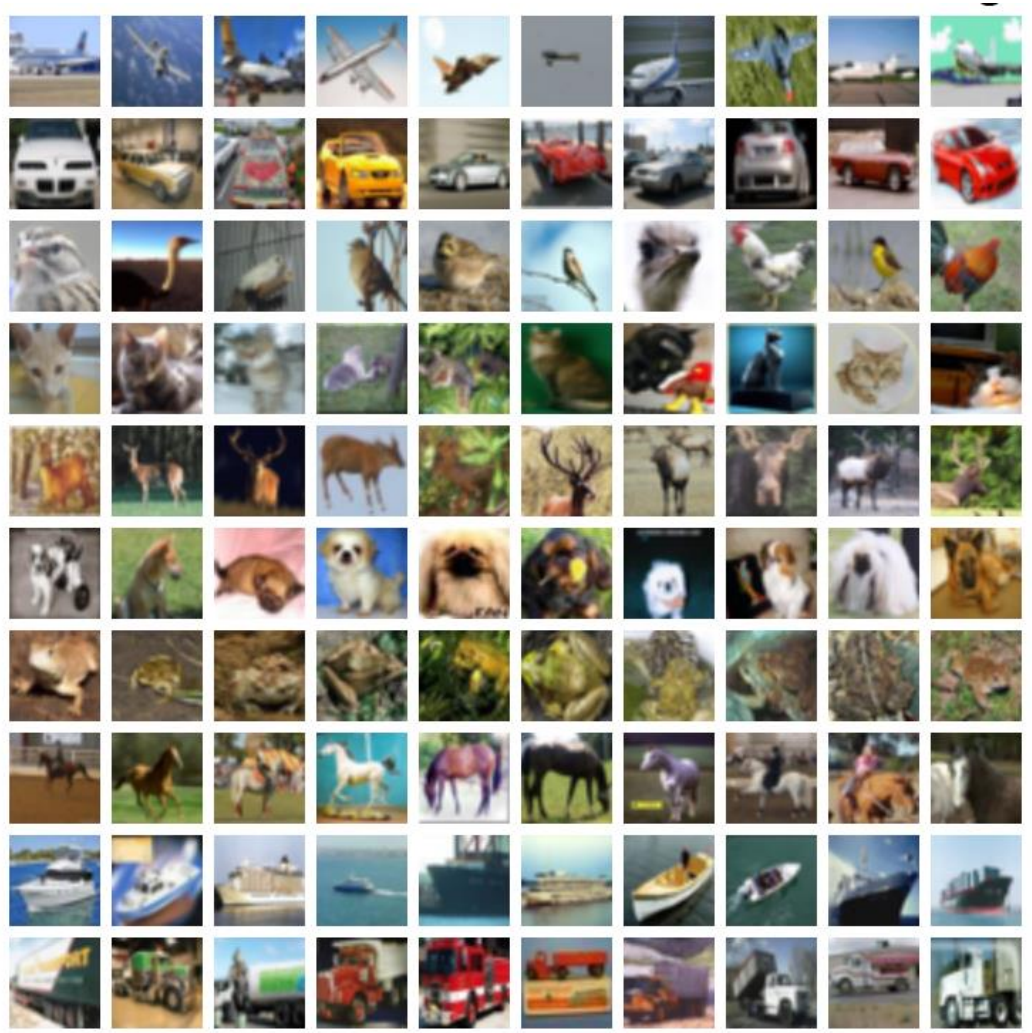
算法原理 QUEEN

- 特征提取器
  - 将输入图像/文本映射为128维嵌入向量，用于后续的敏感度计算
- 核心要素
  - 输入格式
    - 图像（如CIFAR-10），尺寸28×28
  - 提取结构
    - 卷积→ReLU→池化→flatten→全连接层
- 用途
  - 将高维输入压缩成有判别力的向量空间
  - 使得距离计算具有语义解释性（如靠近类中心=代表性强）



实验设计 数据集

- CIFAR-10
  - 包含10个类别每类 6000 张彩色图像，常用于基础分类算法测试
- CIFAR-100
  - CIFAR-10的扩展版，共100个类别，任务更细粒度、更具挑战性
- CUB200
  - 含200种鸟类、共约12000张图像
- Caltech256
  - 256个现实世界类别，图像分布丰富、背景复杂、样本数量不均



实验设计 评价指标

- **Attack Accuracy**
  - 替代模型在测试集上的准确率，越低越好
- **Defense Accuracy**
  - 受害模型在正常用户上的准确率，越高越好

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

- **Reversed Ratio**
  - 被扰乱输出的比例，表示误导程度

$$Reversed\ Radio = \frac{\text{被扰动输出的查询数}}{\text{总查询数}}$$

- **Recorded Ratio**
  - 被记录查询（敏感查询）比例

$$Recorded\ Radio = \frac{\text{被标为敏感的查询数}}{\text{总查询数}}$$

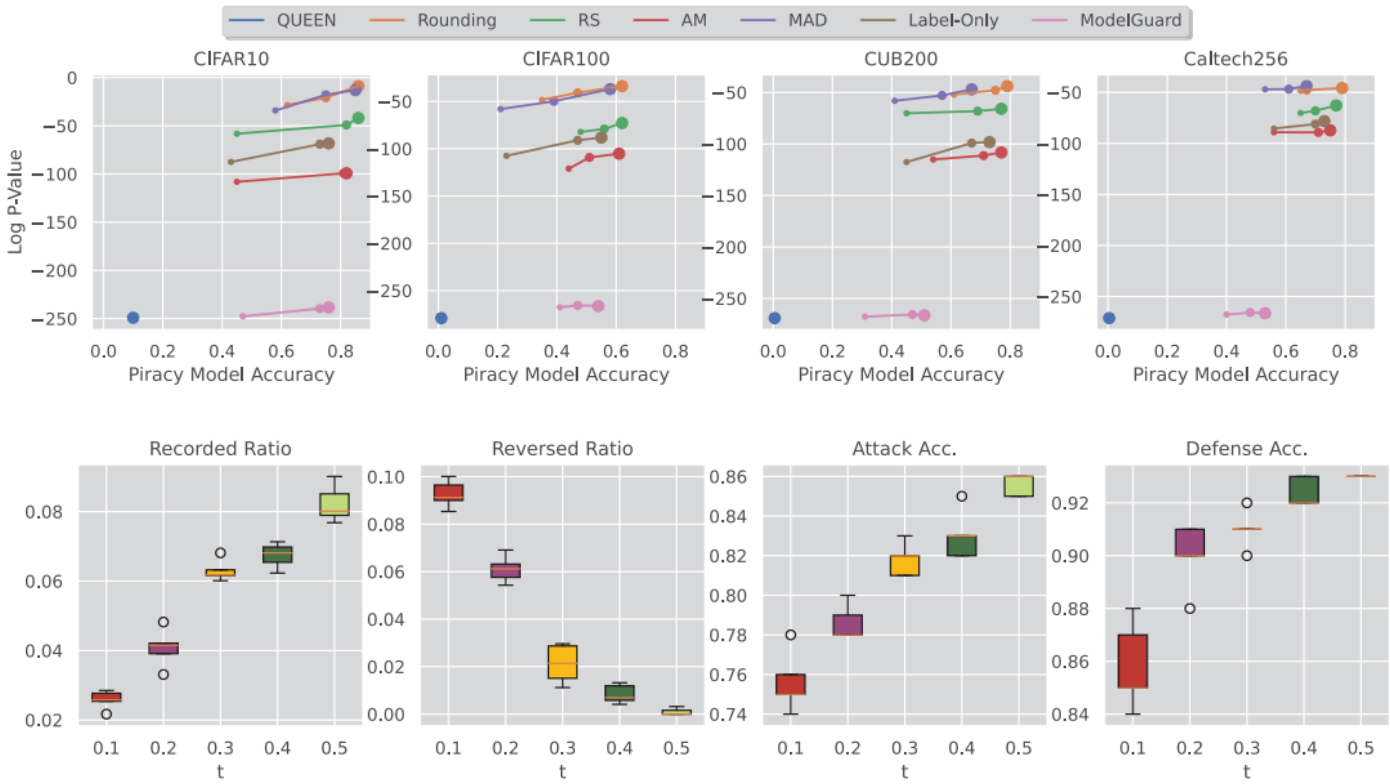
实验设计 对比实验

- 有效防御模型窃取
  - 在多个数据集上，攻击者训练的替代模型准确率大幅下降
- 服务可用性强
  - QUEEN对正常用户影响极小，原模型准确率几乎保持不变
- 低误杀，识别精准
  - 相比PRADA等方法，QUEEN更少误伤正常用户

Query Method	Attack Method	None	RS	MAD	AM	Label-only	Rounding	EMDP	ModelGuard	QUEEN
KnockoffNet	Direct Query	87.42%	85.33%	84.58%	83.17%	83.78%	86.77%	66.15%	74.88%	**10.00%**
	Label-Only	83.78%	83.78%	83.78%	82.11%	83.78%	83.78%	83.78%	83.78%	**81.17%**
	S4L	86.17%	82.30%	80.21%	82.12%	84.02%	85.86%	66.76%	70.69%	**10.00%**
	Smoothing	65.43%	63.41%	61.23%	62.27%	61.01%	65.10%	64.36%	53.24%	**10.00%**
	D-DAE	87.42%	85.32%	84.36%	78.38%	85.24%	87.45%	71.43%	**64.73%**	78.21%
	D-DAE+	87.42%	85.91%	86.44%	84.51%	84.55%	87.01%	86.43%	58.17%	**50.24%**
	pBayes	87.42%	85.91%	87.24%	86.93%	84.57%	86.99%	85.41%	85.16%	**84.24%**
JBDA-TR	Direct Query	63.51%	67.01%	55.31%	60.86%	63.31%	73.55%	25.92%	37.91%	**10.00%**
	Label-Only	63.51%	63.51%	63.51%	**55.77%**	63.51%	63.51%	63.51%	63.51%	61.45%
	D-DAE	74.41%	56.63%	48.51%	57.17%	60.15%	67.65%	62.17%	59.17%	**47.10%**
	D-DAE+	74.41%	72.48%	68.33%	66.10%	63.44%	73.07%	62.33%	51.85%	**40.88%**
	pBayes	74.41%	71.21%	68.15%	74.11%	65.42%	75.01%	67.93%	65.54%	**65.33%**
Max Piracy Model Accuracy		87.42%	85.91%	87.24%	86.93%	85.24%	87.45%	86.43%	85.16%	**84.24%**
Max Piracy Model Agreement		88.24%	87.21%	89.01%	88.22%	87.13%	88.67%	88.71%	86.78%	**86.54%**
Protectee Model Accuracy		92.74%	92.74%	92.74%	90.15%	92.74%	92.74%	92.74%	92.74%	90.01%

实验设计 对比实验

- QUEEN在KS检验中与原模型具有最显著的统计差异
- 对比其他防御，QUEEN的点距远离其他方法，显示出最强的抑制能力
- 阈值t是关键调节参数：
  - t值较小：防御激进，攻击成功率低但误扰正常用户
  - t值较大：防御温和，攻击成功率回升但用户体验更好



特点总结与未来展望

算法总结

- 算法创新
  - ModelShield：根据查询响应分布自动调整触发样本，不依赖训练数据，不需要访问原始模型的训练集或模型结构；通过p-value精准量化水印信号显著性
  - QUEEN：查询敏感度机制，识别敏感查询；将可疑样本的贡献从模型参数中消除
- 算法优势
  - ModelShield：在低查询预算下仍能完成水印检测；无需训练数据、模型梯度、结构信息；鲁棒性强
  - QUEEN：只对恶意用户生效，不影响模型整体性能
- 未来工作
  - 将主动查询控制防御措施与被动模型水印防御措施相结合

参考文献

[1] Pang K, Qi T, Wu C, et al. ModelShield: Adaptive and Robust Watermark against Model Extraction Attack[J]. IEEE Transactions on Information Forensics and Security, 2025, 20 : 1767–1782.

[2] Chen H, Zhu T, Zhang L, et al. QUEEN: Query Unlearning Against Model Extraction[J]. IEEE Transactions on Information Forensics and Security, 2025, 20: 2143–2156.

道可道，非常道。名可名，非常名。无名天地之始。有名万物之母。故常无欲以观其妙。常有欲以观其徼。此两者同出而异名，同谓之玄。玄之又玄，众妙之门。

谢谢！