基于机器遗忘的模型能力细粒度访问控制机制

岳梓岩[1,2]，许盛伟[1,2,3]，王志强[3]，杜皓华[4]

（1.北京邮电大学网络空间安全学院，北京 100876；2.北京电子科技学院密码科学与技术系，北京 100070；
3.中国科学技术大学网络空间安全学院，安徽 合肥 230026；4.北京航空航天大学网络空间安全学院，北京 100091）

摘　要：针对现有人工智能模型在部署中缺乏能力访问控制，导致模型能力可能被未授权用户滥用的问题，提出一种基于机器遗忘的模型能力细粒度访问控制机制Model-Guard，实现不需要重新训练便可对模型任务能力进行细粒度访问控制。首先，基于选择性突触衰减（SSD）算法识别敏感任务能力对应参数，并通过衰减实现模型敏感任务能力默认关闭。其次，设计授权因子计算方法，授权用户通过授权因子恢复模型能力。为保证授权因子的安全分发，Model-Guard 采用对称加密与属性基加密（CP-ABE）混合加密方式，并引入布隆过滤器降低验证开销。实验表明，Model-Guard可在图像识别任务中实现精准能力隔离与恢复，并显著降低部署与维护成本。

关键词：模型能力访问控制；选择性突触衰减算法；授权因子；属性基加密；布隆过滤器

中图分类号：TP309.0

文献标志码：A

DOI：10.11959/j.issn.1000−436x.2026066

Fine-grained model capability access control mechanism based on machine unlearning

Yue Ziyan[1,2], Xu Shengwei[1,2,3], Wang Zhiqiang[3], Du Haohua[4]

1. School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China
2. Department of Cryptography and Science Technology, Beijing Electronic Science and Technology Institute, Beijing 100070, China
3. School of Cyber Science and Technology, University of Science and Technology of China, Hefei 230026, China
4. School of Cyber Science and Technology, Beihang University, Beijing 100191, China

Abstract: A fine-grained model capability access control mechanism, named Model-Guard, was proposed to address the lack of capability access control in deployed artificial intelligence models, which may lead to unauthorized misuse of model capabilities. Without retraining, sensitive task-related parameters were identified by the selective synaptic dampening (SSD) algorithm and attenuated to disable sensitive capabilities by default. An authorization factor calculation method was designed to restore model capabilities for authorized users. To ensure secure distribution of authorization factors, a hybrid scheme combining symmetric encryption and ciphertext-policy attribute-based encryption (CP-ABE) was adopted, and a Bloom filter was introduced to reduce verification overhead. Experimental results demonstrated that Model-Guard achieved precise capability isolation and restoration in image recognition tasks. The proposed mechanism significantly reduces deployment and maintenance costs while enabling fine-grained and secure capability control.

Keywords: model capability access control, SSD, authorization factor, attribute-based encryption, Bloom filter

收稿日期：2025−12−02；**修回日期：**2026−03−01

通信作者：许盛伟，xusw@besti.edu.cn

基金项目：国家重点研发计划基金资助项目（No.2022YFB3104402）；中央高校基本科研业务费专项资金资助项目（No.3282025046）

Foundation Items: The National Key Research and Development Program of China (No.2022YFB3104402), The Fundamental Research Funds for the Central Universities (No.3282025046)

0　引言

近年来，人工智能模型在图像识别、自然语言处理、语音理解、决策推理等多个核心领域实现了前所未有的突破与广泛应用[1]。然而，随着模型在现实场景中部署的不断深化，模型能力滥用带来的安全风险也日益凸显[2-4]。

在图像识别领域，不同识别目标具有敏感度差异，例如“火箭”“导弹”“军用无人机”等目标往往涉及军事侦察、国防设施监测等高风险情境，模型对这些敏感目标的识别能力应仅限于授权用户使用。然而，现阶段模型在训练中已学习并获得对这类目标的高精度识别能力，且在部署阶段采用全能力开放模式，任何用户都可获得模型的全部能力，缺乏对敏感任务能力的访问控制，一旦被非授权用户用于军事目标识别、规避无人机监测或情报搜集辅助，就会带来潜在的国家安全风险[5]。因此，一个关键问题随之产生：是否可以控制用户对模型中不同任务能力的访问权限？针对这一问题，当前解决方案是模型生产方为用户进行专属微调[6-7]，针对不同权限用户训练生成专属模型。然而，这种方案存在根本性局限，首先，为每个用户或角色维护独立模型版本将引发“版本爆炸”问题，显著增加运维成本。其次，这种方案本质依赖于模型生产方，一旦用户权限变化，就无法对模型能力进行更新。表 1 为不同方法在能力访问控制维度上的对比。模型剪枝方法通过直接移除部分参数实现能力削弱，但其处理是不可逆的结构修改，一旦参数被删除或置零，原始能力就无法在终端侧恢复，只能重新训练模型。输出过滤方法是在模型推理完成后对结果进行规则筛选，本质上属于应用层控制，而非模型能力控制，模型内部仍完整保留敏感能力，攻击者可以绕过过滤逻辑，无法解决能力滥用问题。专属微调方法针对不同权限用户分别训练并维护独立模型版本，使不同用户获得不同能力范围的模型。该方法能够实现能力恢复与一定程度上的能力区分，但其本质仍然依赖重新训练与多模型分发，随着用户类型或权限组合增多，容易引发模型版本膨胀、维护成本升高等问题。

机器遗忘研究[8-9]为解决这一问题提供了全新视角。如图 1 所示，机器遗忘通过模型参数的重要性分析和定向干预，让模型遗忘特定类别或任务。受此研究启发，如果在部署阶段普通用户使用“已遗忘敏感任务能力”的模型，授权用户可以恢复被遗忘的模型能力，就能实现能力级别的差异化访问控制。然而，这一理念仍面临多项挑战：如何在不重新训练模型的情况下精准识别和控制与特定任务相关的参数？如何让低权限用户无法访问敏感任务能力，高权限用户可以恢复模型敏感任务能力？如何在终端设备上高效执行而不引入计算时延？

表 1　不同方法在能力访问控制维度上的对比

方法	能力可恢复	支持属性控制	能力可验证	重新训练
模型剪枝	否	否	否	是
输出过滤	否	否	否	是
专属微调	是	否	是	是
Model-Guard	是	是	是	否

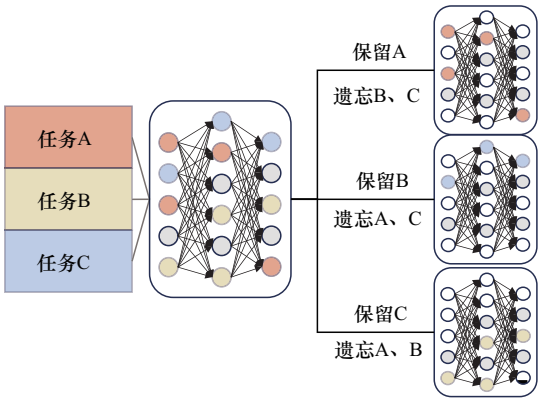

图 1　机器遗忘研究

为解决上述挑战，本文提出一种基于机器遗忘的模型能力细粒度访问控制机制 Model-Guard，其核心思想是利用机器遗忘中的模型参数重要性分析方法，在不需要重新训练模型的前提下识别与敏感任务高度相关的参数，通过模型参数衰减操作默认关闭敏感任务能力。随后，为支持授权用户恢复敏感任务能力，本文设计了可重构模型参数的授权因子，使模型能力能够在终端侧按需恢复。为保证授权因子的安全共享，Model-Guard 采用对称加密与属性基加密（CP-ABE）的混合加密策略，对称加密实现授权因子的机密性保护，CP-ABE 基于用户属性实现对称密钥的细粒度访问控制，使授权因子的分发不再局限于单个用户，而是可以扩展到整个授权群体，实现按需激活、按权访问的细粒度群体访问控制。同时，为降低端侧验证开销，Model-

Guard引入布隆过滤器，实现授权因子密钥的快速合法性判断，使模型敏感任务能力在边缘节点或终端设备上可以高效准确地恢复。本文的主要工作如下。

1) 针对模型敏感能力滥用问题，本文提出Model-Guard，通过对敏感任务所映射的授权因子进行细粒度访问控制，实现了一个模型，多种能力权限的创新部署模式，为解决模型敏感能力滥用安全隐患提供了全新思路。

2) 本文提出模型授权因子计算方法并构建混合加密方案，利用对称加密保护授权因子本身，引入CP-ABE对对称密钥进行基于属性的访问控制。同时，通过布隆过滤器对解密结果快速验证，实现细粒度且高效的访问控制。

3) 本文设计面向云−边−端场景的Model-Guard部署方案，将Model-Guard机制中的核心组件映射至云−边−端架构中的具体实体，并详细阐述了其工作流程，分析部署方案的可行性与安全性。针对模型能力、访问控制效果、成本及开销设计实验，多维度验证Model-Guard的有效性。实验表明，Model-Guard可在图像识别任务中实现精准模型敏感任务能力授权访问，成本及开销相较于专属微调方案大幅缩减，同时加解密及验证计算开销均在可接受的范围内。

4) 本文进一步分析Model-Guard的适用边界，指出其在多敏感任务组合场景和自然语言处理（natural language processing, NLP）任务中的局限性，并据此提出后续研究方向。

1 相关工作

1.1 机器遗忘研究进展

在机器学习中，模型不仅会保留训练数据的统计特征，甚至可能精确复现特定样本的输入和输出映射关系，这种“数据记忆”现象具有持久性，极易导致敏感信息泄露，引发隐私与安全风险[10-13]。为此，机器遗忘作为一项隐私保护技术应运而生，核心目标在于消除某些数据对模型参数与预测行为的残留影响，使模型在功能表现上与从未接受过该数据训练的版本无异[14]，面临的挑战在于模型不会简单孤立地分析数据点，删除单个点可能会破坏学习到的模式和依赖关系[15]，可能导致性能显著下降[16-18]。

机器遗忘可分为精确遗忘[19]与近似遗忘[20]两类[21]。在精确遗忘研究中，标准做法是将待遗忘数据从训练集中移除，并对模型进行完整重训练。然而，该方法在大规模模型与海量数据场景下面临巨大挑战，计算开销与初始训练相当，难以满足实际部署需求。因此，研究者逐渐将重心转向近似遗忘，在不进行重训练的前提下，通过分析与遗忘数据相关的特征路径或参数子集，构造一个与重训练后模型行为高度相近的“近似模型”。文献[22]提出在模型训练时存储参数空间中每个数据点引起的更新，在遗忘过程中，从模型的最终参数中减去相应更新参数以实现数据遗忘。一些学者研究基于k-means聚类算法的机器遗忘方法[23-24]，通过对训练数据进行分区，将数据对模型参数的影响限制在分区内，精准定位遗忘数据的影响，实现更高效的数据遗忘。然而，这些方法普遍存在对训练过程的依赖性，需在训练阶段引入参数缓存、梯度记录等机制，限制了在已有模型上的通用性与可部署性。

为解决上述问题，基于模型参数重要性分析的机器遗忘方法通过可解释性分析手段识别与特定任务相关的模型参数，并对其进行选择性剪枝，从而实现“行为级遗忘”。基于模型参数在某项任务上的重要性分析相关研究总结如表2所示，在卷积神经网络（convolutional neural network，CNN）、双向编码器表示的Transformer模型（BERT）、残差网络（residual network，ResNet）及混合专家模型（MoE）等模型架构中，模型参数均呈现出任务相关性特征。基于上述研究，文献[34]系统性地验证了在大语言模型中，特定任务能力以“特异性神经元”的形式高度集中地编码于神经元中。通过识别并剪除这些神经元，可在不重训练、不访问原始数据的前提下，实现对目标能力的高效删除，同时几乎不影响模型在其他任务上的表现。该工作不仅为机器遗忘提供了高效实用的新方法，也揭示了模型内部存在“功能模块化”与“知识局部化”的规则。进一步地，针对多模态大语言模型中跨模态知识高度耦合的特性，文献[35]提出模态感知神经元遗忘方法，在各模态上分别构建遗忘数据的神经元响应模型，利用梯度分析、注意力权重与激活敏感度等指标，量化神经元对特定模态中遗忘内容的相对重要性，进而识别出“关键记忆神经元”集合。随后，通过剪枝移除这些神经元，实现对敏感信息的

表 2　　**模型参数重要性分析相关研究总结**

研究	模型	任务类型	核心结论
文献[25]	CNN	图像分类	针对具体任务剔除无关滤波器，在保持准确率的同时显著压缩模型体积
文献[26]	BERT	NLP	局部剪枝优于全局剪枝，任务特定剪枝策略更有效
文献[27]	BERT	NLP	通过分析参数重要性配合剪枝可加快训练时间，保证任务的准确率不显著下降
文献[28]	ResNet	图像分类	通过任务解耦与参数重要性筛选出匹配实际需求的任务子集
文献[29]	Transformer	NLP	使用任务联合投影 + 遗传算法，实现极小精度损失下的任务感知剪枝
文献[30]	MoE	NLP	多数专家对推理贡献小，剪枝后单专家模型保留 99.3% 原始性能
文献[31]	MoE	NLP	任务与专家参数具有相关性，显著降低参数量与时延
文献[32]	MoE	图像分类	优先剪除对路由器 L2 变化最小的专家可保证性能稳定
文献[33]	MoE	NLP	专家呈现“任务特定子模块”结构，具备功能性解耦特征

精准剥离。实验表明，该方法在有效消除跨模态隐私风险的同时，显著保留了模型在其他任务上的通用能力，展现出良好的模态解耦与知识隔离能力。文献[34-35]研究表明，在模型内部，特定任务能力并非弥散地分布于整个网络，可以通过剪枝、掩码或重构等手段，定向移除这些神经元实现“遗忘”。基于这一分析，本文提出 Model-Guard，通过密码学手段设置访问权限，动态地屏蔽和激活模型参数，实现模型能力的访问控制。

1.2　选择性突触衰减算法

选择性突触衰减（selective synaptic dampening，SSD）[36]算法的核心思想在于不同模型参数对不同任务的重要性存在差异，某些特定的“遗忘任务”D_f高度依赖于某些参数，利用这种不均匀性，识别并抑制那些对遗忘任务高度敏感的参数，从而最大限度保护主要任务性能的同时，有选择地削弱与D_f相关的模型能力。定义模型参数$\varphi_{\boldsymbol{\theta}}$和数据集$D$，根据式(1)计算费舍尔（Fisher）信息矩阵的一阶导数并分析每个参数在保留数据集和遗忘数据集上的重要度，记为$\boldsymbol{I}_{D,i}$和$\boldsymbol{I}_{D_f,i}$。设定选择阈值α判断参数在遗忘任务中的相对重要性，然后对参数进行衰减计算，遗忘后模型参数为$\varphi_{\boldsymbol{\theta}'}$，SSD 流程如算法 1 所示，其中$\partial\boldsymbol{\theta}$表示对模型参数向量$\boldsymbol{\theta}$求偏导，$\dfrac{\partial \ln p(D|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$表示对数似然函数关于参数的梯度。

$$\boldsymbol{I}_D = E\left[\left(\left(\frac{\partial \ln p(D|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)\left(\frac{\partial \ln p(D|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^{\mathrm{T}}\right)|\boldsymbol{\theta}_D^*\right] \quad (1)$$

算法 1　SSD 流程

输入　$\varphi_{\boldsymbol{\theta}}$、$D_f$、$D$

中间参数　α、λ

输出　$\varphi_{\boldsymbol{\theta}'}$

1) 根据式(1)计算并存储$\boldsymbol{I}_D$

2) 根据式(1)计算并存储$\boldsymbol{I}_{D_f,i}$

3)　for $i = 1{:}1{:}|\varphi_{\boldsymbol{\theta}}|$

4)　　if $\boldsymbol{I}_{D_f,i} > \alpha \boldsymbol{I}_{D,i}$ then

5)　　$\boldsymbol{\theta}_i' = \min\left(\dfrac{\lambda_{D,i}}{\boldsymbol{I}_{D_f,i}}\boldsymbol{\theta}_i, \boldsymbol{\theta}_i\right)$

6)　　end if

7)　end for

8)　return $\varphi_{\boldsymbol{\theta}'}$

1.3　模型安全防护研究

现有研究已提出多种模型安全防护机制，例如，模型水印[37]在推理阶段验证模型归属；可信执行环境研究通过硬件隔离技术在模型运行时提供安全封闭空间[38]；同态加密推理则试图从数据隐私的角度提供保护[39]；差分隐私作为保护训练数据隐私的重要方法，致力于在不暴露个体数据的前提下提取群体特征信息，广泛用于医疗等敏感数据场景[40]；联邦学习[41]通过多方协同训练实现“数据不出本地”的隐私保护目标，在工业应用中已获得广泛关注。尽管上述方法在模型归属验证、执行隔离、数据隐私与协同训练方面各具优势，但均未涉及模型能力的访问可控性，难以满足实际部署中对“部分能力可用、部分能力受限”的访问控制需求。

针对模型能力访问控制研究，文献[42]提出了 SECNeuron 框架，将神经元加密与访问策略绑定，该工作是针对特定任务能力动态控制的初步探索。

与之相比，Model-Guard 在理论支撑、控制对象选择以及验证机制等方面呈现出不同的设计路径。在理论基础方面，SECNeuron 主要基于神经元重要性启发式分析；Model-Guard 建立在机器遗忘与参数重要性理论之上，采用 Fisher 信息矩阵计算参数对任务的贡献程度，能力抑制具有明确的统计学与信息论解释，使能力削弱具有可分析性与可解释性。SECNeuron 对神经元进行整体加密，敏感能力的控制依赖对所有神经元实施加密与解密操作，计算与存储开销随模型规模线性增长；Model-Guard 则基于参数重要性筛选，仅对与敏感任务高度相关的参数施加抑制与授权因子控制，不需要对全部神经元执行加密处理，降低了加解密负担。SECNeuron 在解密阶段无法直接判断神经元参数是否正确恢复，需要依赖密文随机性检测等统计手段判断，计算流程复杂；Model-Guard 引入布隆过滤器校验，能够在密钥层面直接验证授权有效性，更具轻量化优势。华为公司在盘古大模型（Pangu Pro）MoE 的部署中，已率先探索面向企业级场景的模型访问控制机制[43]，体系涵盖了多层级身份认证与基于角色的授权，体现出将传统网络安全理念融入模型运行时权限管理的系统性思路，但其权限机制仍以身份为中心。本文提出的 Model-Guard 融合了 CP-ABE，实现了从“以身份为中心”到“以属性为中心”的权限管理范式转变，通过属性来定义用户，实现群体级别的细粒度能力授权。如图 2 所示，模型参数重要性分析与机器遗忘研究构成本文工作的理论支撑，Model-Guard 对应的安全防护方向属于“防止模型能力滥用”范畴，是对现有防护框架的有益补充。

2　Model-Guard

2.1　Model-Guard 概述

为了在模型部署阶段实现对多任务模型能力的精细化、可验证、可控的访问管理，本文提出 Model-Guard，其核心思想是在模型发布前先对敏感任务对应模型参数进行抑制，使模型默认处于“受限”状态；随后通过对授权因子进行对称加密与 CP-ABE 属性加密，实现“密钥即能力”的访问控制。只有满足访问策略的用户，才能解密获得对应任务的授权因子，并以此恢复相应敏感任务的模型能力。Model-Guard 实现从参数抑制、授权因子加密、密钥访问控制到敏感任务能力恢复的完整闭环，不需要重新训练即可对模型多任务能力进行按需授权。

Model-Guard 总体设计如图 3 所示，在 Model-Guard 中，将系统参与者划分为模型生产方与模型部署方，核心流程如下。

1) 模型生产方首先训练多任务模型，并基于 Fisher 信息矩阵计算不同参数在不同任务上的重要度。

2) 根据敏感任务与通用任务的训练样本重要度，分析对敏感任务有显著贡献的模型参数，并为每个参数计算衰减因子和授权因子，衰减因子用于在参数层面实施抑制，得到默认弱化的模型。

3) 针对每类敏感任务，系统为其授权因子集合生成独立对称密钥并执行对称加密，同时将对称密钥的校验值写入布隆过滤器，用于部署方的轻量

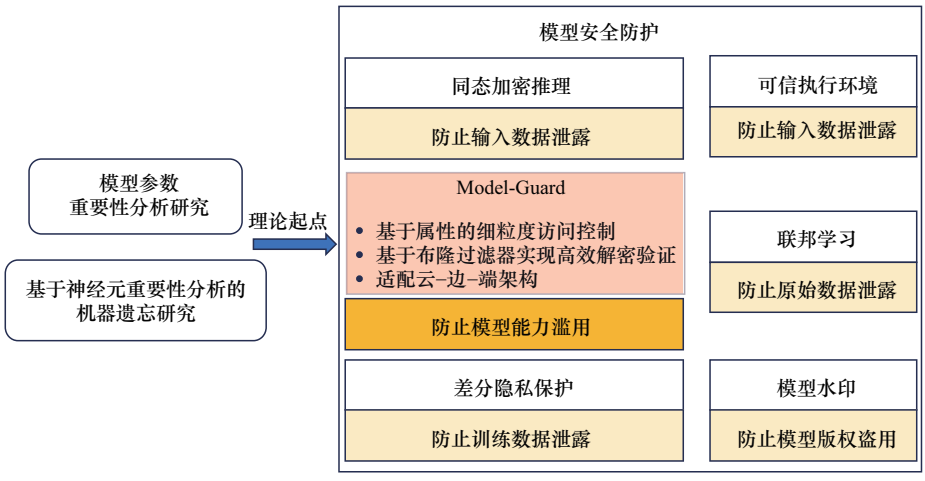

图2　本文的理论基础与研究边界

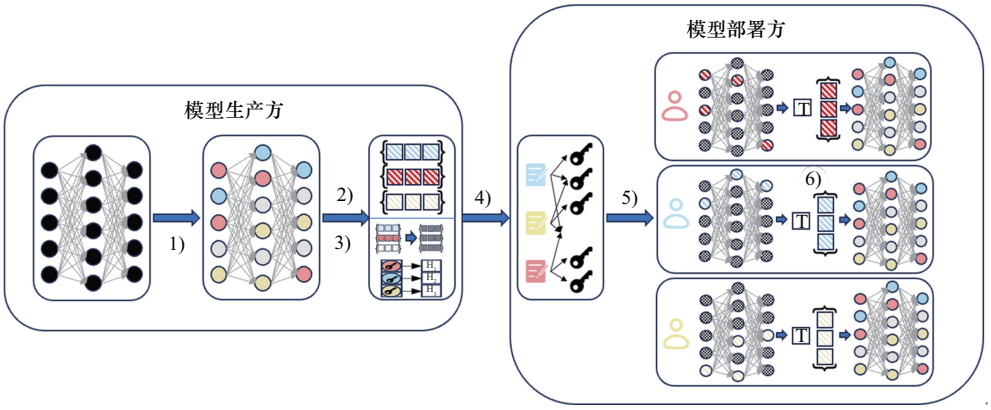

图3　Model-Guard 总体设计

级密钥正确性验证。

4) 为实现按属性授权的访问控制，系统为对称密钥设置访问策略，只有当部署方用户属性满足对应访问策略时，才能解密对称密钥。

5) 用户利用自身属性私钥解密获得对称密钥，随后通过布隆过滤器验证密钥有效性，最后解密授权因子。

6) 用户侧根据获得的授权因子执行参数更新，从而恢复敏感任务能力，实现按需、可控的敏感任务能力解锁。

2.2　授权因子计算

为了实现对模型能力的精细化授权控制，本文基于 SSD 提出授权因子计算方法，模型生产方在不重新训练模型的前提下，识别出与某个敏感任务相关的模型参数，并为模型参数生成对应的授权因子，使模型在默认状态下处于“抑制能力”的状态，拥有权限的用户可通过授权因子恢复获得对应任务的完整能力。定义训练完成的模型参数集合 φ_θ 抑制后的模型参数集合 $\varphi_{\theta'}=\{\theta_i'\}$，敏感任务训练集 $D_{\text{authorized}}$、通用任务数据集 D_{general}，依据 Fisher 信息矩阵的一阶近似形式分别计算每个参数 θ_i 在不同任务上的重要度，计算并存储授权因子 σ，授权因子 σ 计算流程如算法 2 所示。

算法 2　授权因子 σ 计算流程

输入　φ_θ、$D_{\text{authorized}}$、D_{general}

中间参数　α、λ

输出　$\varphi_{\theta'}$

1) 根据式(1)计算并存储 $\boldsymbol{I}_{D_{\text{authorized}}}$

2) 根据式(1)计算并存储 $\boldsymbol{I}_{D_{\text{general}}}$

3)　for i = 1:1:$|\varphi_\theta|$

4)　　if $\boldsymbol{I}_{D_{\text{authorized},i}} > \alpha \boldsymbol{I}_{D_{\text{general},i}}$ then

5)　　$\beta_{i,D_{\text{authorized}}} = \min\left(\dfrac{\lambda \boldsymbol{I}_{D_{\text{general},i}}}{\boldsymbol{I}_{D_{\text{authorized},i}}}, 1\right)$

6)　　$\boldsymbol{\theta}_i' = \beta_{i,D_{\text{authorized}}} \boldsymbol{\theta}_i$

7)　　$\sigma_{i,D_{\text{authorized}}} = \dfrac{1}{\beta_{i,D_{\text{authorized}}}}$

8)　　end if

9)　end for

10)　return $\varphi_{\theta'}$、$\sigma_{i,D_{\text{authorized}} i \in |\varphi_\theta|}$

2.3　授权因子加密

在 Model-Guard 中，为实现授权因子分发过程中的机密性，对授权因子进行对称加密。定义 D_{general} 中包含 k 类任务，对于每类任务所对应的授权因子 $\sigma_{i,j\in k}$，系统生成一个对称密钥 k_j，对授权因子进行加密，输出加密后的授权因子 $C_{i,j}$，加密过程形式化定义如算法 3 所示。

算法 3　授权因子加密算法

输入　$\sigma_{i,D_j i\in|\varphi_\theta| j\in k}$、$k_j$

输出　C_j

1) for j = 1:1:k

2)　$C_{i,j} = \text{Sym.Enc}\left(k_j, \sigma_{i,D_j}\right)$

3) end for

4)　return $C_{i,j\ i\in|\varphi_\theta| j\in k}$

2.4　基于布隆过滤器的授权因子校验机制

为了在部署侧实现对授权因子对称密钥解密有效性的验证，本文提出基于布隆过滤器的授权因子校验机制。该机制保证隐私前提下实现对授权解密结果的有效性确认，对于每类任务 j 对应的授权因

子加密密钥对称密钥k_j，计算校验值h_j，过程定义如式(2)所示。

$$h_j = \mathrm{Hash}\left(k_j \| \mathrm{task} - \mathrm{id}_j\right) \tag{2}$$

在计算校验值过程中加入任务id，避免因哈希碰撞相互干扰，随后h_j被依次插入布隆过滤器作为后续授权验证的依据。

2.5 授权因子密钥访问控制

Model-Guard引入CP-ABE，确保模型在部署后，只有满足特定属性集的用户，才能获得对称密钥并解密使用授权因子，从而实现“按需授权”的模型能力分发。$C_{i,j\ i\in|\varphi_\theta|,j\in k}$授权因子对应的对称密钥为$k_j$，根据任务授权策略$\mathcal{P}_j$，使用CP-ABE加密算法对对称密钥进行加密，生成密钥密文，加密过程形式化定义如算法4所示。

算法4 基于CP-ABE的密钥访问控制算法

输入 k_j、$\mathcal{P}_j$

输出 $\tilde{k}_j$

1) for j = 1:1:k

2) $\tilde{k}_j$= CPABE.Enc$\left(\mathcal{P}_j, k_j\right)$

3) end for

4) return $\tilde{k}_{j\in k}$

在加密过程中，访问策略$\mathcal{P}_j$被嵌入密钥$\tilde{k}_j$之中，只有拥有满足该策略属性集合的用户，才能成功解密获得对应密钥。

2.6 授权因子解密及校验

部署方用户获取模型后，当希望获得敏感任务能力j时，用户根据自身属性集合$\mathcal{A}$与对应的属性私钥集$\mathrm{SK}_{\mathcal{A}}$，解密被CP-ABE加密的对称密钥$\tilde{k}_j$，$\tilde{k}_j$对应的访问策略为$\mathcal{P}_j$，仅当用户属性$\mathcal{A}$满足策略$\mathcal{P}_j$时，才能正确恢复对称密钥$k_j$，否则，解密失败。解密过程形式化定义如式(3)所示。

$$\widehat{k}_j = \mathrm{CPABE.Dec}\left(\mathrm{SK}_{\mathcal{A}}, \tilde{k}_j\right) \tag{3}$$

解密获得$\widehat{k}_j$后，计算校验值$\widehat{h}_j = \mathrm{Hash}\left(\widehat{k}_j \| \mathrm{task} - \mathrm{id}_j\right)$，向布隆过滤器查询，如果返回false，说明无权限或解密失败；如果返回true，说明解密对称密钥成功。在获取对称密钥k_j后，部署用户对密文$C_{i,j\ i\in|\varphi_\theta|,j\in k}$进行解密，得到授权因子$\sigma_{i,j\ i\in|\varphi_\theta|,j\in k}$，解密过程形式化定义如式(4)所示。

$$\sigma_{i,D_j} = \mathrm{Sym.Dec}\left(k_j, C_{i,j}\right) \tag{4}$$

解密获得授权因子后，执行参数更新算法恢复敏感任务能力j，流程如算法5所示。

算法5 模型参数更新算法

输入 $\varphi_{\theta'}$、σ_{i,D_j}

输出 φ_θ

1) for i = 1:1:$|\varphi_{\theta'}|$

2) $\boldsymbol{\theta}_i = \sigma_{i,D_j}\boldsymbol{\theta}'_i = \sigma_{i,D_j}\beta_{i,D_j}\boldsymbol{\theta}_i$

3) end for

4) return φ_θ

在授权因子解密与模型参数恢复过程中，模型参数需要通过授权因子解密及线性变换恢复。从算法机制角度分析，模型参数恢复过程本质为一组可逆线性变换。定义恢复后的模型参数$\hat{\boldsymbol{\theta}}_i$，根据算法2与算法5，$\boldsymbol{\theta}'_i = \beta_{i,D_j}\boldsymbol{\theta}_i$，$\hat{\boldsymbol{\theta}}_i = \sigma_{i,D_j}\boldsymbol{\theta}'_i = \left(\dfrac{1}{\beta_{i,D_j}}\right)\beta_{i,D_j}\boldsymbol{\theta}_i$，在实数域条件下，模型参数恢复过程是可逆的，恢复后的参数$\hat{\boldsymbol{\theta}}_i$与原始参数$\boldsymbol{\theta}_i$相同。在加解密过程中，授权因子采用对称分组加密算法，其处理对象为二进制比特流，不涉及浮点数运算，因此不会引入数值误差。CP-ABE仅用于对称密钥的访问控制，不参与模型参数的数值计算，对参数精度没有影响。

在未引入额外量化或数据格式转换的实现条件下，模型权重误差主要来自参数恢复阶段的浮点乘法舍入误差，根据IEEE-754单精度浮点表示标准，其机器精度为$\varepsilon = 2^{-23} \approx 1.19 \times 10^{-7}$，浮点乘法满足$\mathrm{fl}(a \cdot b) = (a \cdot b)(1+\delta)$，其中$|\delta| \leqslant \varepsilon$，设模型共有$N$个参数，恢复误差的均方误差（mean square error, MSE）可表示为$\mathrm{MSE} = \dfrac{1}{N}\sum_{i=1}^{N}\left(\hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i\right)^2$，其中$\hat{\boldsymbol{\theta}}_i = \boldsymbol{\theta}_i + \boldsymbol{\epsilon}_i$且$\left|\boldsymbol{\epsilon}_i\right| \leqslant \varepsilon\left|\boldsymbol{\theta}_i\right|$，因此$\mathrm{MSE} = \dfrac{1}{N}\sum_{i=1}^{N}\boldsymbol{\epsilon}_i^2 \leqslant \varepsilon^2 \cdot \dfrac{1}{N}\sum_{i=1}^{N}\boldsymbol{\theta}_i^2$，模型权重的平方均值通常满足$\dfrac{1}{N}\sum_{i=1}^{N}\boldsymbol{\theta}_i^2 = O\left(10^{-2} \sim 1\right)$，可得MSE上界为$O(10^{-14} \sim 10^{-12})$，该误差远低于模型训练阶段的梯度噪声、数值截断误差以及常见模型量化误差，对模型推理输出不产生明显影响。

3　面向云–边–端场景的 Model-Guard 部署方案

3.1　方案概述

为使 Model-Guard 能够在云–边–端环境中适配，本文提出一套 3 层部署架构。系统由模型生产方、模型部署方/证书授权中心（CA）、云层、边缘层和端层构成。面向云–边–端场景的 Model-Guard 部署架构如图 4 所示。

3.2　系统部署实体

3.2.1　模型生产方

模型生产方是系统中的完全可信实体，负责多任务模型的训练、参数重要性分析以及授权因子生成，在实际流程中负责以下任务。

1) 模型训练与分析：模型生产方访问云层训练数据，训练多任务模型，并调用部署在云层的参数重要性分析模块计算各任务的 Fisher 信息矩阵。

2) 授权因子生成与加密：模型生产方依据各任务的 Fisher 信息矩阵计算每类任务的授权因子，并构建布隆过滤器用于后续的密钥校验。随后，对授权因子执行对称加密，得到加密授权因子。

3) 上传模型与授权因子：训练完成的模型、加密后的授权因子与布隆过滤器最终存储于云层，供模型部署方执行访问控制。

3.2.2　模型部署方/CA

CA 是系统中的访问控制核心实体，承担属性密钥管理与策略制定功能，其主要职责如下。

1) 数据采集与策略制定：CA 负责从多源传感器采集训练数据，并为具有敏感任务能力需求的终端制定细粒度的 CP-ABE 访问控制策略。

2) 属性密钥分发：CA 向注册终端分发属性私钥，确保只有满足属性集合的终端才能解密对应任务能力。

3)对称密钥访问控制：对授权因子对应的对称密钥执行 CP-ABE 加密，只有满足策略的终端才可以解密对称密钥。

3.2.3　云层

云层作为系统的核心存储与调度中心，承担以下任务。

1) 托管模型与加密授权因子：包括模型参数、加密授权因子、布隆过滤器与密钥密文。

2) 授权因子交付：根据端侧请求向终端分发加密授权因子与密钥密文。

3) 边缘协同工作：将布隆过滤器同步到边缘节点以降低授权验证延时。

3.2.4　边缘层

边缘层位于靠近终端的位置，能够提供低时延响应，但安全性弱于云层。本文方案假设边缘层不适合长期存储核心密钥，其职能如下。

1) 布隆过滤器缓存与授权验证：边缘节点从云层同步布隆过滤器，终端解密获得对称密钥后，可在边缘层完成快速哈希验证，减少云层交互延迟。

2) 外包解密：当终端计算能力受限时，可将 CP-ABE 解密任务外包给边缘层。外包过程中，终端不会暴露完整私钥，而是以可验证方式委托边缘节点执行计算。

3.2.5　端层

端层是模型敏感任务能力恢复的执行方。端侧

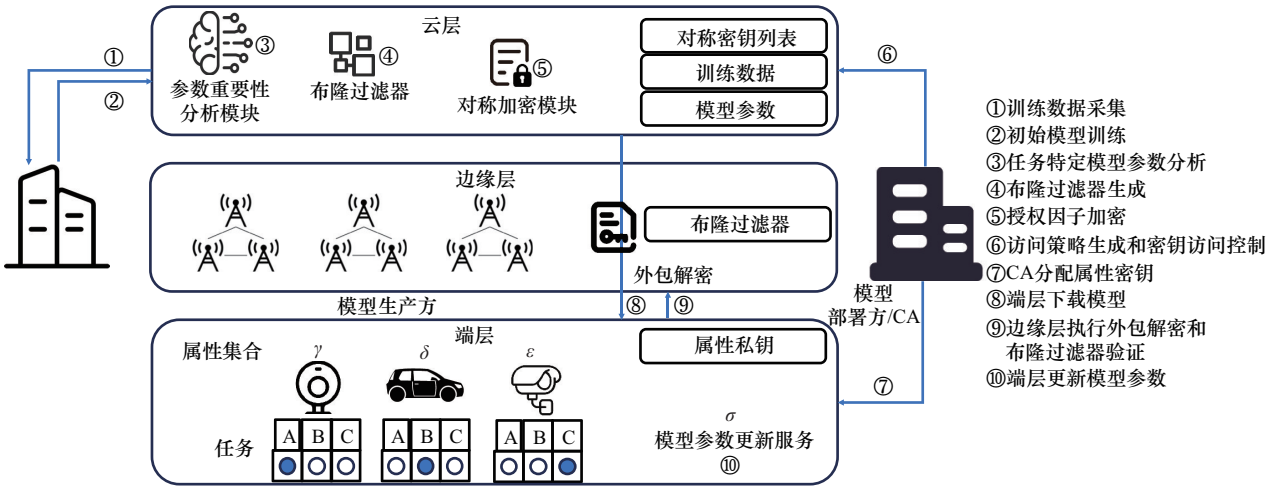

图 4　面向云–边–端场景的 Model-Guard 部署架构

设备通常暴露在公共环境，可能受到物理攻击，因此本文方案假设终端具备可信执行环境可以安全完成解密流程。端层主要完成以下任务。

1) 加载默认抑制模型：端层从云层下载默认“能力抑制模型”，该模型不具备敏感任务能力。

2) 获得授权因子：端层使用属性私钥解密CP-ABE密文，获得对称密钥，若终端计算不足，则可将解密过程安全外包至边缘层。

3) 密钥校验：端层计算对称密钥的哈希并提交至边缘缓存的布隆过滤器进行验证，验证失败则视为未授权或解密失败。

4) 恢复敏感任务能力：对加密授权因子进行对称解密，使用授权因子更新模型参数，恢复敏感任务能力。

3.3 面向云−边−端场景的Model-Guard部署流程

在3.2节实体划分基础上，Model-Guard在云−边−端环境中的运行过程可划分为模型生产与能力抑制阶段、授权数据封装阶段、模型分发与授权请求阶段以及敏感任务能力恢复阶段4个阶段，此节符号定义与第2节内容一致。

3.3.1 模型生产与能力抑制阶段

首先，模型生产方在云层利用训练数据完成多任务模型的训练，获得完整能力模型参数集合φ_{θ}。同时，生产方基于敏感任务数据集与通用任务数据集，调用参数重要性分析模块计算各参数在不同任务上的Fisher信息矩阵，识别与敏感任务高度相关的参数子集。

针对该参数子集，模型生产方执行SSD衰减操作，得到默认能力被抑制的模型参数集合$\varphi_{\theta'}$。此时模型仅保留通用任务能力，敏感任务能力被抑制，模型生产方计算各任务的授权因子σ_{i,D_j}，为后续能力恢复做准备。

3.3.2 授权数据封装阶段

为实现安全分发，模型生产方对每个任务的授权因子采用对称加密算法进行加密，生成授权因子密文$C_{i,j}$。对于每类任务对应的对称密钥k_j，计算其校验哈希值并构建布隆过滤器，用于后续的密钥正确性验证。CA根据终端属性设计访问策略$\mathcal{P}_j$，并利用CP-ABE对每个任务的对称密钥k_j进行加密，生成密钥密文$\tilde{k}_j$。最终云层托管$\varphi_{\theta'}$、$C_{i,j}$、$\tilde{k}_j$和布隆过滤器等数据。

3.3.3 模型分发与授权请求阶段

当终端设备请求模型时，仅能从云层下载默认能力抑制模型$\varphi_{\theta'}$，此时模型处于安全状态，敏感任务能力被抑制。若终端需要激活某一敏感任务能力j，首先使用自身属性私钥$\mathrm{SK}_{\mathcal{A}}$对密钥密文$\tilde{k}_j$进行CP-ABE解密，得到对称密钥$\widehat{k}_j$。为降低时延，终端可将布隆过滤器校验请求提交至边缘层，边缘节点利用布隆过滤器计算$\widehat{h}_j = \mathrm{Hash}\left(\widehat{k}_j \| \mathrm{task}-\mathrm{id}_j\right)$进行查询，验证密钥有效性。当终端计算能力受限时，可将CP-ABE解密任务安全外包至边缘节点，终端仅保留验证所需的关键数据，避免私钥暴露。

3.3.4 敏感任务能力恢复阶段

在验证通过后，终端利用对称密钥k_j解密授权因子密文$C_{i,j}$，恢复授权因子σ_{i,D_j}。随后执行模型参数更新$\boldsymbol{\theta}_i = \sigma_{i,D_j}\boldsymbol{\theta}_i'$，敏感任务对应的参数被恢复，从而实现按需敏感能力激活。

3.4 方案可行性分析

1) 成熟的云−边−端协同架构为Model-Guard提供系统基础

当前云−边−端协同计算体系已在车联网及工业互联网等领域得到大规模应用，为Model-Guard的系统化部署提供了高度成熟的运行环境。云层具备强大的计算与存储能力，可承担模型训练、参数分析及授权因子计算等任务；边缘节点由于物理上靠近终端，能够支持低时延的布隆过滤器校验与外包解密操作；终端设备则可稳定接入网络，执行轻量级的模型能力恢复流程。得益于这一成熟的协同架构，Model-Guard中模型生产方、CA、云层、边缘层与端层之间的交互流程可以在现有基础设施上直接运行，不需要额外构建新的通信或安全体系，从而确保了该机制在系统架构维度上的可行性。

2) 所依赖的密码学技术体系完备，保障Model-Guard的自主可控落地

Model-Guard使用的密码学原语均已具备工程级成熟度，并拥有广泛的行业实践基础。首先，对称加密部分可采用SM4算法，其软件吞吐量已达数Gbit/s，结合硬件加速更能突破十余Gbit/s，能够满足大规模模型参数授权因子的加密需求；基于SM3算法构建布隆过滤器为授权因子的密钥校验提供轻量化快速验证能力，非常适合部署在边缘设备上。此外，CP-ABE已发展多年，研究涵盖外包解

密机制、密钥撤销、多授权中心架构以及访问策略隐藏等安全功能扩展。在大规模物联网环境下，访问策略更新与属性变更频繁发生，导致密钥管理与生命周期维护的系统复杂度增加。尤其在终端数量快速增长的场景下，属性私钥分发、撤销与更新机制的设计直接影响系统的管理开销，这类问题属于属性加密技术在大规模动态系统中的通用挑战。Model-Guard 本身并不依赖特定的 CP-ABE 实现形式，未来可结合支持高效属性撤销与分层管理的扩展方案，增强在大规模动态环境下的部署能力。因此，Model-Guard 不需要构建新的密码学原语，全部安全组件均可直接从标准化库中调用，证明了方案在技术层面的可行性和可扩展性。

3) 终端侧计算负载低，适配大规模分布式部署需求

Model-Guard 在设计中考虑了边缘与终端设备计算能力有限的问题，通过“默认抑制+按需恢复”的模型能力管理方式，不需要重新训练模型，也不需要下载和存储多版本模型，仅需解密授权因子并执行参数更新即可完成能力恢复。恢复过程仅涉及对称密钥解密与参数逐元素乘法运算，计算开销低；若 CP-ABE 解密对终端来说负担过重，还可通过边缘节点执行外包解密，进一步降低终端压力。Model-Guard 有效避免了重复训练与大规模存储等高成本操作，能够轻量化地部署在资源受限终端中，从而保证了在实际应用场景中的端侧可承载性。

3.5　方案安全性分析

1) 模型能力与授权因子的机密性保障

Model-Guard 通过默认能力抑制和授权因子加密实现了对模型能力层面的机密性保护。模型生产方在发布模型前，将敏感任务相关模型参数进行抑制，使默认模型仅拥有通用任务能力，而无法直接使用敏感任务能力。此外，与任务能力对应的授权因子均通过对称加密算法进行加密，确保在分发过程中授权因子不会被未授权实体窃取或逆向推断。由于授权因子是恢复任务能力的唯一途径，因此其机密性等价于模型能力的机密性，即使攻击者获得了模型参数文件，也无法在没有授权因子的情况下恢复敏感任务能力。只能通过加密授权因子与抑制后的模型分离发布，Model-Guard 保证了模型能力的安全隔离，实现了敏感任务能力在未经授权前的隐藏。

2) 基于 CP-ABE 的细粒度访问控制

Model-Guard 引入 CP-ABE 对授权因子的对称密钥进行保护，从而实现基于属性的细粒度访问控制。在该机制中，每个任务 j 对应的对称密钥 k_j 均通过 CP-ABE 加密，密文中嵌入访问策略 $\mathcal{P}_j$。只有当终端设备持有满足策略要求的属性私钥 $\mathrm{SK}_{\mathcal{A}}$ 时，才能成功解密获得 k_j。因此，即使攻击者窃取了授权因子密文 $C_{i,j}$，在没有满足访问策略的属性密钥时仍无法进行解密。此外，现有的 CP-ABE 方案已实现抗共谋攻击能力，多个未授权用户无法通过组合属性私钥恢复密钥，从而有效防止多终端共谋获取敏感任务能力。

3) 默认抑制带来的能力滥用防护

在 Model-Guard 中，模型以“能力抑制”的状态进行下发，这一设计从根源上消除了敏感任务能力被滥用的风险。未经授权的用户仅能获得抑制后的模型 $\varphi_{\theta'}$，模型已被削弱对敏感任务的响应能力。由于授权因子 σ 是恢复能力的必要条件，未获得授权因子的用户无法恢复模型能力。此外，抑制机制作用于参数，不同任务具有独立的抑制因子，使模型能力呈现独立性。因此，Model-Guard 不仅基于密钥控制能力，还在模型参数层面实现能力隔离，增强了能力滥用防护能力。

4) 云–边–端的密钥与模型隔离保障

Model-Guard 通过云–边–端分层部署实现了密钥与模型参数的物理隔离，降低了系统的整体攻击面。云层负责存储模型参数和加密授权因子，但不持有任何属性密钥；边缘节点仅承担布隆过滤器校验与可选的外包解密任务，而无法访问模型参数；终端设备虽然能够访问模型参数，但必须通过属性密钥解密云层下发的密钥密文，才能获得授权因子 σ。由此，模型参数 $\varphi_{\theta'}$ 和对应的密钥 k_j 不同时出现在同一实体上，即便边缘节点或终端设备被入侵，攻击者也无法直接恢复完整模型能力。

5) 基于布隆过滤器的密钥正确性验证

为了确保终端所恢复的授权因子对称密钥有效且未被篡改，Model-Guard 引入基于布隆过滤器的轻量级密钥正确性验证机制。模型生产方在生成每个任务的对称密钥 k_j 后，计算其哈希值 $h_j = \mathrm{Hash}\left(k_j \| \mathrm{task}-\mathrm{id}_j\right)$ 并插入布隆过滤器。终端设备在

解密获得$\widehat{k}_j$后，根据相同方式计算$\widehat{h}_j$并向边缘节点查询布隆过滤器以验证密钥是否有效。此外，该校验机制的计算和存储开销极低，适合边缘节点的大规模快速查询场景。由于布隆过滤器存在假阳性概率，理论上可能出现错误密钥的哈希值被判定为“存在”的情况，在此情况下，终端设备将基于错误密钥解密得到随机授权因子并执行参数恢复操作，随机因子仅会对模型参数引入无结构噪声，表现为模型精度下降，而不会重建敏感任务对应的语义表示结构，也不会破坏Model-Guard所建立的能力隔离安全边界。模型精度下降时作为可观测的运行时异常反馈信号，增强了Model-Guard在部署阶段的可诊断性。在概率方面，布隆过滤器误判率在任务数量有限的场景下可忽略。

6) 基于成熟密码学假设的安全性

Model-Guard的核心创新点在于提出一种模型能力授权与恢复机制，而非设计新的底层密码算法。因此，Model-Guard在构建各安全功能模块时严格采用已被广泛研究、分析并标准化的密码学原语，包括对称加密算法、哈希函数、布隆过滤器以及CP-ABE。这些密码学手段在理论与实践中均已得到充分的安全性论证。CP-ABE依据双线性群上的困难问题已被形式化证明满足密文不可区分性与抗共谋攻击，本文直接调用这些经过严格审查的标准化构造，因此不再重复对其进行安全性证明。此外，Model-Guard在机制层面遵循安全组合原则，在不修改底层密码构造的前提下，将模型参数抑制、授权因子对称加密、布隆过滤器构建与CP-ABE访问控制按模块化方式组合。因此Model-Guard的整体安全性可以视为构建在这些成熟密码原语之上的系统级安全性，由底层算法提供形式化安全保证，由上层机制提供访问控制与能力约束，从而确保整个机制既具备理论安全性，又具备工程可落地性。

7) 对抗少样本微调恢复攻击

Model-Guard在参数处理机制方面并非通过简单噪声扰动或参数遮盖实现能力控制，而是基于Fisher信息矩阵对参数重要性进行筛选，通过$\beta_{i,D_j}\boldsymbol{\theta}_i$改变模型中与敏感任务高度相关的参数方向，使原本承载敏感任务表征的参数维度在权重空间中被压缩至低能级区域。在微调机制方面，敏感任务能力的恢复本质上等价于重新学习该任务在模型参数空间中的有效表示结构。攻击者若仅掌握少量敏感任务样本，其梯度信息不足以稳定重建被压缩的高重要性参数子空间。若要恢复至接近原始模型的敏感任务性能，攻击者需要持续提供大规模、高质量数据并进行长时间训练，其成本将接近重新训练该任务的代价，而非简单的快速微调。此外，在Model-Guard部署模式下，未经授权用户无法获得授权因子，也无法准确定位被抑制参数，进一步提高了攻击难度。因此，通过少样本微调的方式难以低成本绕过Model-Guard中的能力抑制机制，增强了实际部署中的抗敏感能力恢复攻击能力。

3.6 实验分析

为评估面向云–边–端场景的Model-Guard部署方案的有效性与实用性，本文从敏感任务能力隔离效果和成本及开销两个方面设计实验展开验证分析。

3.6.1 模型能力授权验证

CIFAR-100[44]是经典图像分类基准数据集，广泛用于评估模型在多类别视觉识别任务中的性能，所有类别被组织为20个超类，每个超类包含5个语义相关的子类。为系统评估Model-Guard在多任务模型能力控制场景下的有效性，本文基于ResNet-18与Vision Transformer两种模型，在CIFAR-100数据集上构建粗粒度、细粒度与全局细粒度3类图像识别任务授权场景。根据任务粒度定义3类敏感任务能力，并设计了具有不同访问权限的用户类型，如表3所示。图5为不同用户类型对应的任务访问范围，模型能力访问权限的粒度定义如下。

表3 不同访问权限的用户类型

用户类型	常规任务范围	敏感任务
类型A-1	除交通工具2超类外的所有类别	交通工具2超类
类型A-2	除水果与蔬菜超类外的所有类别	水果与蔬菜超类
类型B-1	交通工具2超类中除火箭类外的所有类别	火箭类
类型B-2	大型自然户外场景超类中除海洋类外的所有类别	海洋类
类型C-1	除火箭子类外的所有类别	火箭子类
类型C-2	除蘑菇子类外的所有类别	蘑菇子类

类型A（粗粒度）：训练数据集为全量数据集，设定某个超类图片的识别能力为敏感任务能力，需要授权才可以获取。

类型 B（超类内的细粒度）：训练数据集为超类数据集，设定某个超类内的某个子类图片的识别能力为敏感任务能力，需要授权才可以获取。

类型 C（细粒度全局）：训练数据集为全量数据集，设定某一子类图片的识别能力为敏感任务能力，需要授权才可以获取。

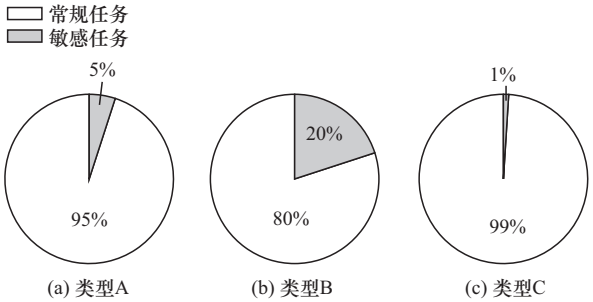

图 5　不同用户类型对应的任务访问范围

本文实验基于算法 2 对模型参数进行敏感任务能力抑制，形成下发给所有用户的常规模型。此时，模型在敏感任务上的能力被显著削弱甚至完全抑制，用户只能访问其常规任务范围对应的识别能力。当用户满足访问控制策略并成功获取授权因子后，可利用授权因子恢复模型对应的敏感任务能力，从而在不需要重新训练模型的前提下，实现按需解锁敏感任务能力。表 4 展示了各类用户在常规模型能力（敏感任务被抑制）与授权模型能力（使用授权因子恢复）下的通用任务准确率与敏感任务准确率。

1) 敏感任务能力抑制的有效性

从表 4 可以看出，在类型 A 和类型 C 用户条件下，类型 A 与类型 C 用户的敏感任务准确率均被抑制至 0，这表明 Model-Guard 可有效识别模型中与敏感任务高度相关的参数，并在不损伤通用任务性能的前提下对其进行抑制，实现近乎完全的“敏感能力屏蔽”。在类型 B2 实验设置中，敏感任务“海洋”类别与常规任务同属于“大型自然户外场景”超类，语义与视觉特征高度重叠。该超类内部各类别（如森林、山脉、天空）共享低频纹理、颜色分布及边缘结构等底层视觉表示，因此在模型参数空间中，其重要性参数子空间并非严格独立，呈现出交叠现象。Model-Guard 的能力抑制机制建立在“任务重要参数存在相对可分子空间”的假设基础之上。当敏感任务与常规任务在语义层和特征层高度耦合时，其对应的 Fisher 信息重要性分布趋于一致，导致同一参数同时承载多任务表征功能。此时对敏感任务相关参数进行衰减，在参数级别难以构造清晰的能力边界，从而出现敏感任务和通用任务在抑制状态下仍保留较高准确率的现象。本文采用的 SSD 算法仅作为参数重要性分析的一种实现方式，并非机制的唯一选择。当敏感任务抑制效果不足或任务隔离边界模糊时，可通过替换参数重要性方法进行优化。

2) 授权后敏感任务能力恢复效果

授权后，用户获得对应任务的授权因子，更新模

表 4　常规模型与授权模型准确率对比

用户类型	模型	Model-Guard 常规模型能力（敏感能力被抑制）		Model-Guard 授权模型能力（授权因子恢复）	
		通用任务准确率	敏感任务准确率	通用任务准确率	敏感任务准确率
类型 A-1	ResNet-18	82.97%	0	82.69%	80.41%
	Vision Transformer	93.12%	0	95.73%	95.22%
类型 A-2	ResNet-18	82.38%	0	82.31%	86.90%
	Vision Transformer	95.71%	0	95.59%	97.57%
类型 B-1	ResNet-18	82.43%	2.17%	82.54%	79.34%
	Vision Transformer	95.13%	5.12%	95.73%	94.53%
类型 B-2	ResNet-18	81.72%	75.35%	82.37%	96.27%
	Vision Transformer	95.57%	97.05%	95.67%	99.22%
类型 C-1	ResNet-18	74.54%	0	76.27%	80.90%
	Vision Transformer	88.90%	0	88.88%	94.70%
类型 C-2	ResNet-18	75.59%	0	76.28%	80.12%
	Vision Transformer	88.82%	0	88.87%	94.88%

型参数以获得敏感任务能力。实验结果显示，授权后敏感任务准确率达到基线模型水准，验证了通过授权因子更新模型能够在不需要重新训练模型的情况下恢复模型的敏感任务能力，实现细粒度的能力授权。

3.6.2 成本与开销分析

为全面评估Model-Guard在云−边−端协同架构中的运行成本与部署可行性，云层由模型生产方与CA共同构成，由一台服务器（Intel Core i5-14600KF处理器、RTX 4070显卡、16 GB内存）模拟，用于执行模型训练、参数重要性分析、授权因子计算以及对称密钥与CP-ABE加密操作。边缘层采用一台移动边缘计算（mobile edge computing，MEC）设备（Intel i5-8250U，16 GB内存），部署Ubuntu 18.04系统，用于执行布隆过滤器校验与属性解密的外包计算。端层为DJI M30T无人机，其负载软件开发工具包（payload software development kit，SDK）单元集成Orin Nano（四核ARMv8处理器），用于执行对称密钥解密与本地模型参数恢复。

1) 训练成本对比分析

在多任务模型能力访问控制场景中，专属微调方案需要为不同权限等级的用户分别训练模型。以CIFAR-10为例，当某一类识别对象被标记为敏感任务时，如果采用专属微调方案，则需要为普通用户训练一个不含敏感类的模型，为高权限用户训练一个包含全部10类的模型。如果需要更细粒度的能力拆分，例如每个类在某些场景都属于敏感任务，则需要从训练集删除该类数据，然后训练10份模型。为量化Model-Guard机制在训练阶段的优势，本文以ResNet-18 + CIFAR-10为基准，分别测试了“传统10次微调方案”与“Model-Guard一次训练+参数分析”的训练成本。传统方案需要对10组任务分别进行微调，总训练时间达到4 781.2 s。相比之下，Model-Guard仅需495.92 s就可以完成一次完整模型训练，随后对10类任务执行算法2，总耗时仅16.7 s。整体来看，Model-Guard将训练成本从4 781.2 s降低至495.92 s，训练开销降低约89.63%，并且不随任务规模线性增长。此外，Model-Guard使模型能力的调整不再依赖重新训练。当普通用户获得新的能力授权时，不需要重新训练模型，只需解密对应的授权因子即可即时恢复任务能力；传统方案需要重新微调模型，既增加了计算负担又降低了系统灵活性。

2) CP-ABE加解密开销

为了评估基于属性对密钥进行访问控制的计算开销，本文对CP-ABE在不同策略下对对称密钥加解密的运算时间进行了基准测试。实验基于MEC设备，使用charm-crypto v0.50库实现，统计1 000次加解密运算时间的平均值，CP-ABE加解密计算开销随属性数量的变化如图6所示。由图6可知，加解密计算开销随属性数量的增加而增加，总体仍然在毫米级。

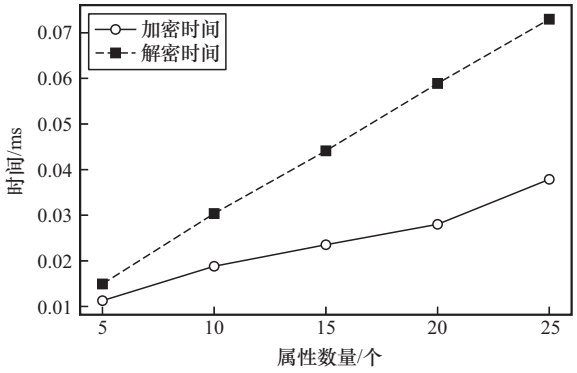

图6 CP-ABE加解密计算开销随属性数量的变化

3) 布隆过滤器构建与查询开销分析

为评估Model-Guard在边缘侧引入的布隆过滤器轻量级授权校验机制的计算开销，测试了不同规模布隆过滤器的构建时间与查询时间。实验中分别向布隆过滤器插入10、20和100个授权因子密钥校验值，对应任务规模，并记录构建时间和查询时间，结果如表5所示。在100种任务规模下，构建时间小于70 ms，完全可在模型训练阶段一次性生成，此外，对于端层或边缘层在线授权校验而言成本极低，不会对模型解密和能力恢复产生可感知时延。

表5 布隆过滤器计算开销

任务规模/种	构建时间 /ms	查询时间/μs
10	8.219 5	768.108 3
20	14.963 4	665.252 1
100	63.586 1	621.345 0

4) 模型参数更新成本分析

在Model-Guard的授权执行流程中，当终端用户成功解密获得授权因子后，将根据算法5对模型参数进行恢复更新。模型更新计算开销如表6所示，ResNet-18的模型参数恢复耗时仅为6.18 ms；

ResNet-34 为 42.92 ms；即便是参数量较大的 Vision Transformer，其更新过程仍可在 65.13 ms 内完成，因此模型参数更新算法完全可以在终端设备中实时执行，不会引入明显的授权时延。

表 6　模型更新计算开销

模型	参数量/MB	参数更新耗时/ms
ResNet-18	11.69	6.18
ResNet-34	21.80	42.92
Vision Transformer	86.39	65.13

4　Model-Guard 机制扩展分析

4.1　多类敏感任务能力情况下的隔离效果与适用边界探讨

3.6.1 节实验证明了 Model-Guard 在单类敏感任务能力隔离场景中表现出显著优势，但当需要对多个敏感任务同时进行授权时，其能力隔离效果表现出衰减。本文基于 ResNet-18 与 CIFAR-10 进行了两组对比实验，通过改变“敏感任务集合”的组成方式，观察不同类别组合下能力抑制效果的变化。

在第一组实验中，将飞机、汽车、轮船、卡车 4 类定义为常规任务，其余 6 类定义为敏感任务。如表 7 所示，常规任务类别的性能保持稳定，敏感类别的性能出现大幅下降。

表 7　第一组常规模型与授权模型准确率对比

类别	Model-Guard 常规模型	Model-Guard 授权模型	Δ
飞机	95.40%	99.00%	+3.60%
汽车	96.70%	97.10%	+0.40%
轮船	96.60%	85.60%	−11.00%
卡车	94.90%	86.60%	−8.30%
鸟	92.50%	47.70%	−44.80%
猫	86.80%	31.20%	−55.60%
鹿	94.80%	45.60%	−49.20%
狗	88.40%	24.60%	−63.80%
青蛙	95.00%	17.60%	−77.40%
马	94.50%	42.10%	−52.40%

在第二组实验中，将猫、鹿、狗、马 4 类作为常规任务，其余 6 类定义为敏感任务。如表 8 所示，虽然敏感任务有一定降幅，但常规任务同样受到抑制，相较于单敏感类的隔离效果仍有差距。

表 8　第二组常规模型与授权模型准确率对比

类别	Model-Guard 常规模型	Model-Guard 授权模型	Δ
猫	86.80%	81.40%	−5.40%
鹿	94.80%	97.80%	+3.00%
狗	88.40%	77.90%	−10.50%
马	94.50%	81.60%	−12.90%
飞机	95.40%	71.20%	−24.20%
汽车	96.70%	89.70%	−7.00%
鸟	92.50%	81.00%	−11.50%
青蛙	95.00%	80.50%	−14.50%
轮船	96.60%	81.00%	−15.60%
卡车	94.90%	71.50%	−23.40%

Model-Guard 的核心机制依赖于对敏感任务相关参数进行识别，其隐含前提是不同任务在参数空间中存在相对可分的高重要性子空间。当多个敏感任务在语义表示层面共享表征时，其对应的重要参数子空间会发生重叠，导致单一参数同时服务于多个任务功能。当敏感任务集合变大或类别之间存在较强特征耦合时，Model-Guard 在参数层面实现能力隔离的线性可分性会显著降低，模型的重要参数隔离边界变得模糊，最终表现为多敏感任务隔离效果不如单敏感任务场景，因此紧密耦合的任务组合需要更精细的参数选择策略或结构级的隔离设计，未来可从参数级扩展至结构级能力标识，从而实现更灵活的敏感任务组合。

4.2　Model-Guard 在 NLP 任务中的适用性探讨

Model-Guard 的理论基础在于模型参数对不同任务呈现出差异化的重要性分布，并据此实现参数级能力抑制与可逆恢复。为进一步探讨该机制在不同模型结构与任务类型下的适用性，本文选取 MoE 模型作为研究对象，从结构行为层面分析其是否具备天然的“能力模块化”特征。MoE 模型通过路由机制在多个专家之间动态分配计算资源，理论上不同任务可能在专家维度上呈现差异化激活模式。若不同任务对应相对稳定且可分离的专家子集，则有可能通过结构级模块控制实现能力隔离。因此，本文通过统计专家使用频率，对模型在 NLP 任务场景中的结构行为进行分析，评估其潜在的能力可分离性。

本文采用基于混合专家结构的 Switch-Trans-

former模型，模型包含8个专家模块。GLUE数据集[45]是广泛用于评估预训练语言模型语义理解能力的标准基准，任务涵盖多种句级与句对语义推理场景。本文从GLUE的9项核心任务中选取斯坦福情感树库二分类（SST-2）、多体裁自然语言推理（MNLI）、Quora问题对匹配（QQP）和文本蕴含识别（RTE）4项任务作为实验对象，并在这4项基准任务上进行了联合微调。上述4项任务分别对应单句语义判别、跨领域逻辑推理、语义相似度判断与文本蕴含推理等典型自然语言理解范式，覆盖了预训练语言模型在下游应用中最常见的核心理解能力类型。因此，该任务组合能够较为全面地反映模型在语义理解层面的能力分布特征，适合作为分析模型参数级能力可分离性的实验载体。本文在GLUE基准任务的验证集上，对编码器与解码器共12层的专家路由输出进行统计并绘制专家使用频率热力图，如图7~图9所示。

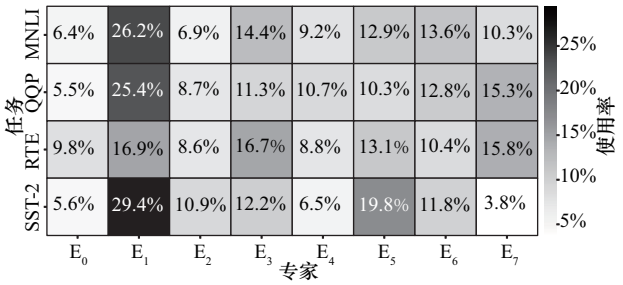

图7　Encoder第1层专家使用频率热力图

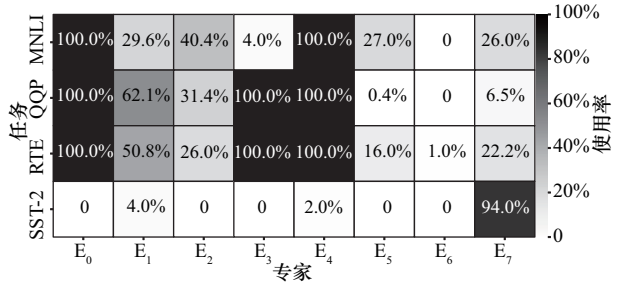

图8　Encoder第9层专家使用频率热力图

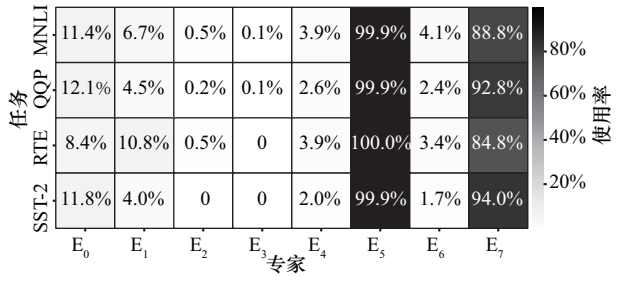

图9　Decoder第9层专家使用频率热力图

通过分析使用频率，模型在NLP任务中呈现以下特征。

1) 专家使用的高重叠性

不同任务在专家维度上的激活分布高度相似，多数专家在多任务场景中均被频繁调用，未呈现明显的任务特定激活模式。这一现象表明，在当前训练设置下，语言模型在多个任务间共享大量语义表示空间。

2) 未观察到清晰的任务与专家映射关系

与图像分类任务中类别特征可相对局部化的现象不同，在多任务语言模型中，各任务在语义、逻辑与句法层面相互交织，未发现可直接用于能力划分的稳定专家子集。

3) 上下文依赖性导致的全局退化效应

NLP任务依赖自注意力机制，每个词元的语义表征与上下文全局相关。当对部分专家参数执行衰减操作时，信息传递链条被破坏，可能会扩散至整句的语义表示，造成整体性能下降而非局部能力抑制。

综上分析，在基于Switch-Transformer的MoE架构下，不同NLP任务在专家维度呈现出高度重叠的激活分布，难以形成稳定的任务与专家对应关系。因此，仅依赖专家使用频率作为结构级划分依据，难以直接实现精细粒度的能力隔离。该观察结论并不意味着Model-Guard在NLP任务中不可适用，而是表明专家模块本身并不天然构成清晰的能力边界，仍需进一步结合参数级重要性度量或更细粒度结构单元分析，才能实现更稳定的模型能力隔离。

5　结束语

云–边–端协同智能系统迅速发展的背景下，模型能力按需授予与控制已成为关键需求。针对不同任务分别训练不同模型、维护多套模型版本等方式存在训练成本高、存储开销大、无法灵活调整权限等突出问题，本文提出Model-Guard，在不需要重新训练模型的前提下，实现模型能力的按需授权、可验证恢复与精细化控制。Model-Guard基于机器遗忘SSD算法，使模型默认处于敏感任务能力被抑制的安全状态，再结合对称加密、CP-ABE与布隆过滤器轻量验证，实现安全可控的授权因子分发与验证。在此机制基础上，本文进一步面向真实部署需求，设计了云–边–端3层协同架构，使授

权因子计算、密钥分发、布隆过滤器验证和外包解密能够在各层之间高效协同，从而在确保安全性的同时兼顾终端侧算力限制，同时对本文方案进行了可行性、安全性以及实验分析。实验结果表明，Model-Guard能够在图像分类任务中实现精细粒度的能力隔离，在常规模型中敏感任务能力可被完全抑制，而授权后可准确恢复至基线水平。此外，实验还从训练成本、计算开销与模型参数更新计算开销等方面验证了该机制在实际部署中的高效性。Model-Guard采用国密算法，为国密算法在人工智能领域的应用拓展提供了新的实践场景，同时将CP-ABE从传统用于云存储密文访问控制场景拓展至模型安全防护场景，为模型安全防护提供了新的思路。

本文使用SSD算法作为参数重要性分析与抑制手段，通过实验验证了该算法在参数级能力隔离方面的可行性。然而，在类型B2等高语义耦合任务场景中，敏感任务与常规任务共享底层特征表征，SSD算法的参数级可分性受到限制，导致能力隔离强度下降。需要指出的是，SSD算法并非Model-Guard的必要组成部分，而是当前实现的一种选择。Model-Guard的创新在于“能力重要参数识别+参数可逆抑制+授权恢复”这一控制逻辑。因此未来工作将继续研究更细粒度的结构级能力隔离，从参数级扩展到注意力头、通道、专家模块等结构级能力标识与加密，增强多任务间的可分性以及针对NLP模型语义耦合强的特点，研究能力区域划分、语义分片等新方法，提高隔离效果，从而优化Model-Guard的效果。

参考文献：

[1] Gursoy D, Cai R Y. Artificial intelligence: an overview of research trends and future directions[J]. International Journal of Contemporary Hospitality Management, 2025, 37(1): 1-17.

[2] Apicella A, Isgrò F, Prevete R. Don’t push the button! Exploring data leakage risks in machine learning and transfer learning[J]. Artificial Intelligence Review, 2025, 58(11): 339.

[3] Anderljung M, Hazell J, von Knebel M. Protecting society from AI misuse: when are restrictions on capabilities warranted?[J]. AI & Society, 2025, 40(5): 3841-3857.

[4] Rakin A S, Chowdhuryy M H I, Yao F, et al. DeepSteal: advanced model extractions leveraging efficient weight stealing in memories[C]// Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2022: 1157-1174.

[5] Tang Q, Su C, Tian Y, et al. YOLO-SS: optimizing YOLO for enhanced small object detection in remote sensing imagery[J]. The Journal of Supercomputing, 2025, 81: 303.

[6] Panigrahi A, Saunshi N, Zhao H, et al. Task-specific skill localization in fine-tuned language models[C]//International Conference on Machine Learning. New York: ACM Press, 2023: 27011-27033.

[7] Si C, Shi Z, Zhang S, et al. Unleashing the power of task-specific directions in parameter efficient fine-tuning[C]//The Thirteenth International Conference on Learning Representations. Vancouver: ICLR, 2024: 1-24.

[8] 李梓童, 孟小峰, 王雷霞, 等. 机器遗忘综述[J]. 软件学报, 2025, 36(4): 1637-1664.
Li Z T, Meng X F, Wang L X, et al. Survey on machine unlearning[J]. Journal of Software, 2025, 36(4): 1637-1664.

[9] 何黎松, 杨洋. 遗忘学习综述[J]. 计算机科学与探索, 2024, 18(11): 2872-2886.
He L S, Yang Y. Review of machine unlearning[J]. Journal of Frontiers of Computer Science and Technology, 2024, 18(11): 2872-2886.

[10] Chodey M D, Gouthami E, Rao K, et al. Privacy-preserving machine learning models[C]//Proceedings of the 2025 International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI). Piscataway: IEEE Press, 2025: 1521-1527.

[11] Schelter S. amnesia-towards machine learning models that can forget user data very fast[C]//Proceedings of the1st International Workshop on Applied AI for Database Systems and Applications (AIDB19). Saarland: DBLP, 2019: 1-4.

[12] Zanella-Béguelin S, Wutschitz L, Tople S, et al. Analyzing information leakage of updates to natural language models[C]//Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2020: 363-375.

[13] Shokri R, Stronati M, Song C Z, et al. Membership inference attacks against machine learning models[C]//Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2017: 3-18.

[14] Cao Y Z, Yang J F. Towards making systems forget with machine unlearning[C]//Proceedings of the 2015 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2015: 463-480.

[15] Serra J, Suris D, Miron M, et al. Overcoming catastrophic forgetting with hard attention to the task[C]//International Conference on Machine Learning. New York: PMLR, 2018: 4548-4557.

[16] Lee J, Mai Z, Yoo J, et al. Continual unlearning for text-to-image diffusion models: a regularization perspective[PP]. V2. (2025-11-11)[2025-12-02]. arXiv: arXiv. 2511.07970.

[17] Chen M, Zhang Z K, Wang T H, et al. When machine unlearning jeopardizes privacy[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2021: 896-911.

[18] Schelter S, Grafberger S, Dunning T. HedgeCut: maintaining randomised trees for low-latency machine unlearning[C]//Proceedings of the 2021 International Conference on Management of Data. New York: ACM Press, 2021: 1545-1557.

[19] Bourtoule L, Chandrasekaran V, Choquette-Choo C A, et al. Machine unlearning[C]//Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2021: 141-159.

[20] Tarun A K, Chundawat V S, Mandal M, et al. Fast yet effective machine unlearning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(9): 13046-13055.

[21] 何可, 王建华, 于丹, 等. 基于自适应采样的机器遗忘方法[J]. 信息网络安全, 2025, 25(4): 630-639.
He K, Wang J H, Yu D, et al. Adaptive sampling-based machine unlearning method[J]. Netinfo Security, 2025, 25(4): 630-639.

[22] Graves L, Nagisetty V, Ganesh V. Amnesiac machine learning[J]. Pro-

ceedings of the AAAI Conference on Artificial Intelligence, 2021, 35 (13): 11516-11524.
[23] Ginart A, Guan M, Valiant G, et al. Making ai forget you: data deletion in machine learning[C]//Proceedings of the 33rd International Conference on Neural Information Processing Systems. Massachusetts: MIT Press, 2019: 3518-3531.
[24] Mirzasoleiman B, Karbasi A, Krause A. Deletion-robust submodular maximization: data summarization with “the right to be forgotten”[C]// International Conference on Machine Learning. New York: PMLR, 2017: 2449-2458.
[25] Wang Y T, Shi B J, Zhang H. TSP: task-specific pruning for personalized image classification on edge devices[C]//Proceedings of the ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway: IEEE Press, 2025: 1-5.
[26] Mishra A K, Chakraborty M. Does local pruning offer task-specific models to learn effectively? [C]//Proceedings of the Student Research Workshop Associated with RANLP 2021. [S.l.:s.n.], 2021: 118-125.
[27] Wang G R, Yang J, Sun Y R. Task-oriented memory-efficient pruning-adapter[PP]. V2. (2023-04-06)[2025-12-02]. arXiv: arXiv. 2303.14704.
[28] Zhou J X, Bao W D, Wang J, et al. CUT: pruning pre-trained multi-task models into compact models for edge devices[C]//International Conference on Intelligent Computing. Berlin: Springer, 2025: 164-177.
[29] Reda W, Jangda A, Chintalapudi K. How many parameters does your task really need? task specific pruning with LLM-sieve[PP]. V2. (2025-10-04)[2025-12-02]. arXiv: arXiv. 2505.18350.
[30] Chen T Y, Huang S H, Xie Y, et al. Task-specific expert pruning for sparse mixture-of-experts[PP]. V2. (2022-06-02) [2025-12-02]. arXiv: arXiv. 2206.00277.
[31] Lu X D, Liu Q, Xu Y H, et al. Not all experts are equal: efficient expert pruning and skipping for mixture-of-experts large language models[C]// Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Stroudsburg: ACL, 2024: 6159-6172.
[32] Chowdhury M N R, Wang M, El Maghraoui K, et al. A provably effective method for pruning experts in fine-tuned sparse mixture-of-experts[PP]. V3. (2024-05-30)[2025-12-02]. arXiv: arXiv. 2405.16646.
[33] Han Z Y, Liu X T, Zhou R T, et al. Faster, smaller, and smarter: task-aware expert merging for online MoE inference[PP]. V2. (2026-01-23) [2025-12-02]. arXiv: arXiv. 2509.19781.
[34] Pochinkov N, Schoots N. Dissecting language models: machine unlearning via selective pruning[PP]. V2. (2024-07-24) [2025-12-02]. arXiv: arXiv. 2403.01267.
[35] Liu Z Y, Dou G Y, Yuan X C, et al.Modality-aware neuron pruning for unlearning in multimodal large language models[C]//Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL Press, 2025: 5913-5933.
[36] Foster J, Schoepf S, Brintrup A. Fast machine unlearning without retraining through selective synaptic dampening[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2024, 38(11): 12043-12051.
[37] Zhang J, Chen D D, Liao J, et al. Deep model intellectual property protection via deep watermarking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(8): 4005-4020.
[38] Huang W, Wang Y G, Cheng A D, et al. A fast, performant, secure distributed training framework for LLM[C]//Proceedings of the ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway: IEEE Press, 2024: 4800-4804.
[39] Lloret-Talavera G, Jorda M, Servat H, et al. Enabling homomorphically encrypted inference for large DNN models[J]. IEEE Transactions on Computers, 2022, 71(5): 1145-1155.
[40] Ji Z L, Lipton Z C, Elkan C. Differential privacy and machine learning: a survey and review[PP]. V1. (2014-12-24) [2025-12-02]. arXiv: arXiv. 1412.7584.
[41] Li L, Fan Y X, Tse M, et al. A review of applications in federated learning[J]. Computers & Industrial Engineering, 2020, 149: 106854.
[42] Wang Z Q, Du H H, Wang J Y, et al. SECNeuron: reliable and flexible abuse control in local LLMs via hybrid neuron encryption[PP]. V1. (2025-06-05)[2025-12-02]. arXiv: arXiv. 2506.05242.
[43] Tang Y H, Li X S, Liu F C, et al. Pangu pro MoE: mixture of grouped experts for efficient sparsity[PP]. V2. (2025-05-28) [2025-12-02]. arXiv: arXiv. 2505.21411.
[44] Krizhevsky A, Hinton G. Convolutional deep belief networks on cifar-10[J]. Unpublished Manuscript, 2010, 40(7): 1-9.
[45] Wang A, Singh A, Michael J, et al. GLUE: A multi-task benchmark and analysis platform for natural language understanding[C]//Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. [S.l.:s.n.], 2018: 353-355.

[作者简介]

岳梓岩（1998– ），男，新疆哈密人，北京邮电大学博士生，主要研究方向为属性加密、密码算法识别。

许盛伟（1977– ），男，江西吉安人，博士，北京电子科技学院教授、博士生导师，主要研究方向为密码学与信息安全。

王志强（1998– ），男，山东青岛人，中国科学技术大学博士生，主要研究方向为数据安全、智能体访问控制。

杜皓华（1990– ），女，陕西眉县人，博士，北京航空航天大学副教授，主要研究方向为移动计算、智能感知、数据隐私保护、物联网安全。