面向云端单节点的高效可验证隐私神经网络推理框架

田有亮[1,2]，陈埭立[1,3]，杨江迪[1,3]，李梦倩[1,3]，张馨予[1,3]，策力木格[4]
（1. 贵州省密码学与区块链技术特色重点实验室，贵州 贵阳 550025；2. 贵州大学大数据与信息工程学院，贵州 贵阳 550025；
3. 贵州大学计算机科学与技术学院，贵州 贵阳 550025；4. 电气通信大学信息理工学研究科，东京都 调布 181-8585）

摘　要：针对云端单节点推理在保障数据隐私与计算可验证性时存在的高证明开销问题，提出可验证隐私推理框架Gorak。该框架在可信执行环境下生成正确性证明，并将模型承诺与代码度量绑定于会话起点，确保推理过程的完整性。此外，通过自适应位消融优化，压缩证明密态前向轨迹，减少多项式约束规模，从而显著降低证明生成与验证的负担。实验结果表明，Gorak在公开卷积神经网络基准上将证明生成时间和验证时间降低96%和30%，在LeNet-5上相比基线降低99.01%和85.68%。

关键词：隐私保护推理；可信执行环境；可扩展透明知识论证；同态加密；可验证计算
中图分类号：TP309
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025194

Efficient verifiable privacy neural network inference framework for cloud single-node

TIAN Youliang[1,2], CHEN Shuangli[1,3], YANG Jiangdi[1,3], LI Mengqian[1,3], ZHANG Xinyu[1,3], Celimuge Wu[4]
1. Guizhou Provincial Key Laboratory of Cryptography and Blockchain Technology, Guizhou University, Guiyang 550025, China
2. College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China
3. College of Computer Science and Technology, Guizhou University, Guiyang 550025, China
4. Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofu 182-8585, Japan

Abstract: To address the high proof overhead in ensuring data privacy and computational verifiability for cloud-based single-node inference, a verifiable privacy-preserving inference framework named Gorak was proposed. Proofs were generated within a trusted execution environment, where model commitments and code measurements were bound to the session initiation to ensure the integrity of the inference process. By applying adaptive bit-pruning optimization, the encrypted forward computation trace was compressed and the polynomial constraint size was reduced, significantly alleviating the burden of proof generation and verification. Experimental results show that Gorak reduces proof latency by 96% and verification latency by 30% on public convolutional neural network benchmarks, and achieves reductions of 99.01% and 85.68% on LeNet-5 compared with the baseline.

Keywords: privacy-preserving inference, trusted execution environment, scalable transparent arguments of knowledge, homomorphic encryption, verifiable computation

0　引言

随着大数据支撑的智能化进程持续推进，云计算已将数据的生产与计算快速分离，终端设备通过网络把计算密集型任务交给算力资源富集的云或边

收稿日期：2025–08–25；**修回日期：**2025–10–17
通信作者：陈埭立，605704472@qq.com
基金项目：国家自然科学基金资助项目（No.62272123）；贵州省高层次创新型人才基金资助项目（No.[2020]6008）；贵州省科技计划基金资助项目（No.[2020]5017, No.[2022]065）
Foundation Items: The National Natural Science Foundation of China (No.62272123), The Project of High-level Innovative Talents of Guizhou Province (No.[2020]6008), The Science and Technology Program of Guizhou Province (No.[2020]5017, No.[2022]065)

缘集群完成，以达到数据资源的有效流通与利用。这种委托计算[1]的方式正逐渐成为通用的算力供给模式，为缓解端侧算力不足以及数据分散的限制，公有云如 Azure[2]、Google Cloud AI Platform[3]等通过按量付费接口提供机器学习的推理服务。云端单节点托管架构如图 1 所示，客户端在本地完成数据预处理与加密后，将密文请求提交至云端单节点进行推理，并接收回传的结果。

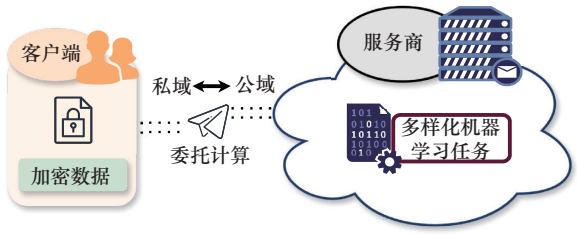

图 1　云端单节点托管架构

尽管按量托管便于落地，但数据离域与执行不可见存在 3 类关键风险，限制其在高敏行业应用。隐私保护方面，云侧可能接触原始输入或可逆中间态，漏洞、越权与多租户隔离失效可能致敏感样本外泄；可验证性方面，缺乏可独立校验的执行证据，服务方更换模型、跳步计算或偏置输出难被发现；计费方面，计量主要依赖云端自报，GPU 与内存占用及时长缺乏外部可审计依据。根本症结在于缺少一种对外可验证的会话级证据，用以关联模型版本、执行路径和资源计量。因此，在单一云节点下同时实现数据隐私保护、过程可验证与计费可审计，成为委托式推理工程化落地的首要技术瓶颈。

针对云端推理需满足隐私保护以及可核验的目标，现有研究大致沿 2 条路线前进。其一是可验证机器学习（vML, verifiable machine learning），该方向通过引入形式化证明机制，要求云端在返回推理结果的同时给出计算正确性的数学证明，从而消解委托计算中的结果可信隐忧。文献[4]提出的安全可验证学习框架与文献[5]的可验证机器学习框架，都将通用可验证计算技术迁移到机器学习场景，使客户端能够核验云端推理的完整性；后续工作进一步在证明中融入零知识性质，以隐藏服务端模型细节。其二是聚焦于隐私保护机器学习（PPML，privacy-preserving machine learning），在此框架下，研究者借助同态加密（HE, homomorphic encryption）[6]或安全多方计算（MPC, secure multi-party computation）[7]将数据和模型封装于密态域，从根本上阻断云端对明文信息的访问。然而，在隐私推理场景中，MPC 依赖多方不串通，部署与运维成本高，难以契合单节点云端形态，HE 在保护双边隐私方面成效显著，但与可验证机制耦合时会显著提高证明的生成与验证成本，甚至出现证明开销超过原始计算开销的情况。

在此背景下，单节点隐私推理需在保证数据隐私的同时提供可独立验证的计算完整性证明，但现有技术路径在这 2 个方面均存在瓶颈。首先，形式化验证虽可确保计算过程正确，但难以证明推理确由声明模型执行。为此引入的模型参数承诺及一致性检查需在证明电路中嵌入哈希或查表逻辑，涉及大量非线性与截断运算，导致证明规模与时间显著增加。其次，同态加密方案与证明后端在算术域上缺乏高效对齐，尤其在全同态加密（FHE，full homomorphic encryption）设置下，密文运算依赖余数系统（RNS，residue number system）及跨环映射，需额外执行域转换与截断位检查，从而进一步放大生成与验证复杂度。由此，如何在确保模型可信的同时保持证明生成与验证的高效性，成为单节点可验证隐私推理的核心问题。

针对上述问题，本文将同态加密、可信执行环境（TEE, trusted execution environment）与可扩展透明知识论证（STARK, scalable transparent arguments of knowledge）进行组合，以实现高效的可验证密态数据推理。然而，三者在信任边界与执行机制上存在协同挑战，同态计算与可扩展透明知识论证在电路层面需共享统一的算术描述，而可信执行环境虽可提供硬件级隔离与度量，但其度量结果通常难以形式化为可被外部验证的数学证据。若直接组合，TEE 的度量哈希与电路约束之间的映射将引入信任漂移与审计脱节问题，阻碍端到端验证闭环的建立。

为此，本文提出可验证隐私推理框架 Gorak。该框架在 TEE 内完成模型与代码的会话级度量绑定，生成可核验的参数承诺，从而避免将参数一致性哈希显式嵌入证明电路；前向阶段采用与证明后端算术域同源对齐的同态运算，将密态执行映射为低度算术轨迹，以规避非线性与截断带来的电路膨胀；最终通过可扩展透明知识论证实现对结果的外部验证，在保持执行隐私的同时实现验证域与硬件信任域的自然衔接，从而在单节点云端场景下实现

隐私保护与可验证性的统一。本文的主要工作如下。

1) 提出面向云端单节点的可验证隐私推理框架Gorak。框架在TEE内完成模型与代码的会话级度量与绑定，将经低度扩展（LDE，low-degree extension）生成的多项式码字通过默克尔树（Merkle）聚合为全局根，并将该根值写入TEE的远程证明报告（Quote）作为模型与电路的统一承诺锚点。该设计使硬件度量结果与外部验证环节在语义上保持一致，避免了将参数一致性哈希嵌入电路所带来的非线性膨胀。推理阶段在同态加密域内执行密态运算，并仅对算术轨迹生成透明知识论证，从而保证计算结果可独立验证。

2) 为高效实现工作1目标，设计基于低度代数中间值表示（AIR, algebraic intermediate representation）的自适应位消融优化算法，降低证明链路执行迹，针对椭圆曲线点加与标量乘操作构建高效的约束系统，并通过参数化快速里德所罗门交互式预言机近似性证明优化证明生成过程，显著降低生成和验证的时间开销。

3) 基于上述设计实现原型系统，在卷积神经网络（CNN, convolutional neural network）的公开基准上将证明生成时间降低96%以上，验证时间降低30%~70%；在LeNet-5卷积神经网络上较基线分别减少99.01%与85.68%，同时保持隐私与安全保障。

1 相关工作

在单节点云端推理场景中，服务器需要同时满足输入机密性、执行完整性与对外可验证性，已有研究围绕这3类性质进行探索，但各自存在局限，为厘清问题空间并界定本文研究范围，以下对前人的工作进行回顾。

同态加密为单节点密态推理提供了最自然的基础；文献[8]提出的加密数据上的神经网络应用系统在全同态加密域内完成卷积与全连接推理，首次验证了在不泄露输入的前提下进行神经网络推理的可行性。文献[9]提出的同态神经网络推理优化编译器工具将推理时间降至秒级。然而，HE方案仅解决了机密性问题，却无法保证外部可验证性，客户端在面对云端返回的密文结果时，无法区分其是否源自预期的模型推理，从而在执行完整性上存在信任缺口。

可信执行环境则从硬件隔离出发，保证加载的代码与模型不会被篡改，并且能够通过远程证明建立执行环境的完整性，文献[10]提出的基于可信执行环境的私有推理加速系统（SLALOM）与文献[11]的隐私保护推理系统是该路线的代表性工作。两者均利用TEE在安全区域内部处理非线性运算，并将线性层外包至GPU，以在保证数据隔离的同时实现毫秒级的推理性能。但TEE的可信边界止于芯片内部，无法向外部发布可公开验证的数学证据；外部用户无法复验结果是否由承诺的电路与模型计算而来，因而缺少对外部可验证性。

零知识证明为可验证计算体系提供了形式化的外部验证能力。文献[12]提出基于椭圆曲线配对的零知识证明协议，该协议能够在常数时间内完成验证，但需要逐电路的可信设置；文献[13]提出了基于多项式约束的零知识证明协议，该协议将可信设置成本降低为一次通用初始化，却仍存在残余信任假设；透明体系如文献[14]的多项式交互式零知识证明体系框架（SPARTAN）与文献[15]的可扩展透明知识论证协议（STARK）则不需要可信设置。其中SPARTAN是基于多项式的交互式证明，通过将秩一约束系统（R1CS，rank-1 constraint system）转化为多项式求值问题并结合文献[16]提出的求和校验协议完成验证，该证明系统采用双线性配对的多项式承诺保证一致性，其证明体积小、验证高效，但仍继承了R1CS的“门爆炸”问题。相较之下，STARK将执行过程表示为有限域上的执行迹，并通过多项式整除约束保证状态转移的正确性，利用快速里德所罗门交互式预言近似性证明（FRI，fast Reed-Solomon interactive oracle proof of proximity）协议进行低度性测试，从而具备并行友好性与后量子安全优势，多项式约束在矩阵乘与卷积累加等算术密集算子上具有高效性，但在处理哈希与权重一致性检查时仍需引入布尔控制与查表门，电路规模会随之退化，开销也会显著增加。

在加密域下结合零知识证明更具挑战。从算子级别的运算层面看，针对全同态加密的证明研究虽已在理论层面建立了可行性，但在原型实现中依然存在巨大的性能瓶颈。FHE通常依赖模数链来控制噪声增长，而在执行乘法后往往需要进行重缩放与模数切换等维护操作，这些操作本身具有非线性特征。将其嵌入零知识电路时，不仅需要将复杂的除

法和取整过程细化为算术或布尔约束，还必须通过查表或范围证明来确保正确性，从而导致证明电路急剧膨胀。文献[17]提出的全同态加密计算的可验证框架（vFHE）、文献[18]的基于FHE的零知识验证框架与文献[19]的基于格密码的可验证加密计算原型系统等工作均表明，即使仅验证一层密态推理，也会引入数量庞大的多项式和查表约束，使证明生成时间远超实用阈值。因此，尽管理论上FHE与零知识证明（ZK, zero-knowledge proof）的结合能够同时提供隐私与可验证性，但在实践中证明开销由非线性操作所主导，尚难以达到可部署水平。在此背景下，许多实用化方案退而采用半同态加密如ElGamal与Paillier。与全同态相比，半同态的运算规则简洁，不依赖复杂的模数链维护，也避免了乘法运算所引入的非线性开销，使得在零知识证明中能够以较低成本表达和验证线性运算。

在密态可验证计算实际应用到机器学习推理的研究方向上，现有方案在单节点部署、全流程密态处理与计算完整性验证的需求面前仍存在显著局限。典型代表之一是文献[20]提出的基于随机置换与安全多方计算的高效机密神经网络推理方案（CENTAUR），该方案通过随机置换与安全多方计算协议实现大模型推理下的隐私与效率平衡，但其强隐私保障本质上依赖于多方参与和高交互成本，因而不适用于单服务器环境。类似地，文献[21]提出的基于加密令牌剪枝的隐私化Transformer推理方案，其显著优化了Transformer类模型的私有推理性能，但其安全性同样建立在两方安全计算和可信初始化之上，缺乏非交互式的公开可验证能力。这类工作在单节点密态数据推理和执行可验证的条件下，并不具备可直接迁移性。相较于多方交互式方案，近年来也有研究开始探索在单节点环境下实现可验证推理的路径，文献[22]提出了可验证隐私神经网络推理框架（vPIN），其在单节点设定下结合ElGamal加密与Spartan证明，实现了隐私计算与正确性验证，然而，由于其证明后端是基于椭圆曲线配对的建模方式，证明开销仍然偏高。文献[23]提出了基于混合策略与零知识证明的深度学习可验证推理方案，使复杂模型也能实现端到端验证；文献[24]构建了面向分布式推理的轻量级零知识验证系统（ezDPS）；文献[25]与文献[26]分别提出了针对卷积网络与通用深度学习的零知识验证框架；文献[27]则将零知识验证拓展至大语言模型场景。上述研究在明文输入下验证了可验证推理的可行性，但尚未在密态输入场景中高效支持。文献[28]提出的隐私保护可验证卷积神经网络（pvCNN）通过模型分拆实现部分隐私保护，但难以覆盖全流程密态输入。

近期多项综述与框架性研究[29-31]均指出，在密态输入条件下实现对外可验证性仍是一个开放挑战。大多数现有系统不得不采用混合架构，以规避因密态和明文的域切换、哈希一致性检查以及非线性操作证明所带来的高昂开销。为了更直观地对比方案之间的差异，本文在表1中总结了典型方案的特征与不足，内容涵盖推理过程所采用的隐私保护方式、计算正确性的验证机制、在单节点环境下独立完成隐私推理与验证的适配能力，并在备注中对各方案的关键特征与局限进行了简要说明，以突出本文工作的改进价值。

表1　　代表性方案功能对比

方案	数据隐私保护	执行完整性验证	单节点适配	备注
文献[10]	内存保护	TEE	√	1)利用TEE实现高效私有推理 2)完整性由硬件保障 3)缺乏密码学外部验证
文献[24]	明文	ZK	√	1)验证明文数据的正确推理 2)非隐私保护
文献[21]	安全多方计算	协议内一致性	×	1)混合框架平衡性能 2)完整性由安全多方计算协议保障 3)无法向外部第三方验证
文献[25]	明文	ZK	√	1)为明文CNN预测提供端到端验证 2)仅保护模型隐私，不保护客户端数据
文献[28]	部分HE	ZK	√	1)非全流程客户端数据隐私 2)依赖可信设置的ZK证明
文献[22]	ElGamal	ZK	√	1)在单节点实现密态可验证推理 2)证明生成开销较大
文献[17]	FHE (BGV)	原型验证	√	1)为FHE计算的正确性生成证明 2)原型模拟实现
本文	ElGamal	ZK、TEE	√	1)在单节点实现密态可验证推理 2)证明开销低于同类方案 3) 明确模型参数绑定

2 预备知识

本节主要介绍后续方案设计与安全证明中涉及的关键密码学技术与可信硬件基础，包括加法同态加密（AHE, additive homomorphic encryption）ElGamal、STARK透明论证系统，以及可信执行环境相关定义与安全模型，为全文描述建立统一的技术语境。

2.1 加法同态加密

同态加密允许对密文直接执行特定运算而不需要解密。本文采用的算法为AHE，基于椭圆曲线群G的ElGamal方案，支持密文加法和标量乘法运算，其定义与安全性如下。

设安全参数为λ，取一素数阶q的椭圆曲线群G，群生成元记为$[\![1]\!]\in G$，对于消息$m\in\mathbb{Z}_q$记$[\![m]\!]=[\![1]\!]^m\in G$，为将明文$m$嵌入群$G$的编码，方案包括3类核心算法。

1) 密钥生成$\mathrm{KeyGen}(1^\lambda)$：输入安全参数$\lambda$，随机选取私钥$x\leftarrow\mathbb{Z}_q$，计算公钥$h=[\![1]\!]^x$，最终输出公私钥对$(\mathrm{pk},\mathrm{sk})=(([\![1]\!],h),x)$。

2) 加密算法$\mathrm{Enc}(\mathrm{pk},m)$：给定公钥$\mathrm{pk}=([\![1]\!],h)$和消息$m\in\mathbb{Z}_q$，随机选择$r\leftarrow\mathbb{Z}_q$，计算并输出密文$C=(c_1,c_2)=([\![r]\!],h^r+[\![m]\!])\in G^2$。

3) 解密算法$\mathrm{Dec}(\mathrm{sk},C)$：输入私钥$\mathrm{sk}=x$和密文$C$，解密明文消息$m=\log_{[\![1]\!]}([\![m]\!])\in\mathbb{Z}_q$。

其中，离散对数计算仅针对经量化编码后的有限域整数执行，可在受控区间内通过查表或硬件加速完成，确保解密效率而不影响推理精度。ElGamal AHE方案具备2个关键的同态性质，即密文的同态加法与密文和标量的同态乘法，具体定义如下。

1) 密文同态加法：给定2个消息$m_1,m_2\in\mathbb{Z}_q$，以及它们对应的密文$C_1,C_2\in\mathbb{Z}_q$，其密文相加操作（记为$\oplus$）定义$C_1\oplus C_2=([\![r_1+r_2]\!],h^{r_1+r_2}+[\![m_1+m_2]\!])$，并满足$\mathrm{Dec}(\mathrm{sk},C_1\oplus C_2)=m_1+m_2$。

2) 密文标量同态乘法：给定消息$m\in\mathbb{Z}_q$与标量$\delta\in\mathbb{Z}_q$，对密文定义标量乘法操作（记为$\odot$）为$\delta\odot\mathrm{Enc}(\mathrm{pk},m)=([\![\delta\cdot r]\!],h^{\delta\cdot r}+[\![\delta\cdot m]\!])$，解密后得到的明文为标量乘法的结果$\delta\cdot m$。

上述同态特性为推理计算提供基础支持，使服务器能够直接对客户端提交的加密数据进行同态运算，保障数据隐私，本文使用的ElGamal AHE方案满足在选择明文攻击下的不可区分性（IND-CPA, indistinguishability under chosen-plaintext attack）。具体而言，对任意概率多项式时间（PPT, probabilistic polynomial-time）算法攻击者，其攻击成功的概率与随机猜测的概率仅相差可忽略量$\mathrm{negl}(\cdot)$，形式化定义如下。

定义一个安全参数λ，任意2个长度相等的明文消息m_0,m_1，攻击者的优势满足

$$\left|\Pr[\mathcal{A}(\mathrm{pk},\mathrm{Enc}(\mathrm{pk},m_0))=1]-\Pr[\mathcal{A}(\mathrm{pk},\mathrm{Enc}(\mathrm{pk},m_1))=1]\right|\leqslant\mathrm{negl}(\lambda)\tag{1}$$

即攻击者无法有效区分2个明文对应密文的差异，从而保障加密数据在传输和同态计算期间的数据机密性。

本文方案消息m被编码为曲线点$M=mG$。解密后直接得到M，而恢复m等价于在曲线上求离散对数。本文系统中m已通过量化映射限定在小范围内，因此该离散对数可通过查表或硬件辅助在可行时间内完成。

2.2 可信执行环境

在委托计算架构中，可信执行环境提供硬件级保护，用以对抗不可信操作系统乃至物理主机运行时代码与数据的篡改。本文以Intel SGX为例，处理器将物理地址空间划分普通区（$\mathrm{Reg_N}$, normal region）以及隔离区（$\mathrm{Reg_E}$, enclave page cache），其中$\mathrm{Reg_N}$用于常规应用和操作系统访问；$\mathrm{Reg_E}$由SGX硬件保护，仅允许在安全执行状态下访问。当程序计数器$\mathrm{pc}\in\mathrm{Reg_E}$指向隔离区页面时，处理器进入安全飞地（enclave）模式时，对通用寄存器与微架构状态的刷新屏蔽，使得任意外部主体对enclave内存的窥探或注入的概率不超过$\mathrm{negl}(\lambda)$。

为使远端调用方确信enclave已加载期望程序P且其初始状态未被篡改，SGX提供远程证明协议$\Pi_{\mathrm{TEE}}=(\mathrm{Setup},\mathrm{Init},\mathrm{Attest},\mathrm{Verify})$。其中，Setup在制造阶段生成平台签名密钥对$(\mathrm{pk},\mathrm{sk})$；Init在enclave加载二进制$P$时计算度量$M=\mathrm{Meas}(P)\in\{0,1\}$；随后$\mathrm{Attest}(\mathrm{nonce},\mathrm{sk_{TEE}})$产生Quote，其中nonce为一次性挑战值对应的随机数；定义签名为$\mathrm{Sig_{sk}}(M\|\mathrm{nonce})$。客户端调用$\mathrm{Verify}(\mathrm{pk},\mathrm{Quote},M^\star,\mathrm{nonce})$并在式(2)成立时接受。

$$\text{Verify}=1 \Leftrightarrow \text{Verify}_{\text{pk}}(\text{Quote},M\|\text{nonce})=1 \wedge M=M^{\star} \tag{2}$$

远程证明链上的平台签名体制满足消息的不可伪造性（EUF-CMA, existential unforgeability under chosen-message attack）；TEE 提供内存隔离与测量绑定。任意多项式时间对手若在未持有平台私钥的情形下生成通过验证且绑定任意度量值的 Quote，将与 EUF-CMA 假设矛盾；若在加载后篡改 enclave 内存而不改变测量，将与 TEE 隔离性矛盾。由此通过验证的 Quote 等价于本次会话由指定二进制与初始化状态启动。

2.3　可扩展透明知识论证

为确保密态计算过程具备外部可验证性，本文引入基于可扩展透明知识论证框架的非交互式证明系统，用以对运算过程中的多项式计算轨迹建立完整的结构化验证路径。

以下给出整体 STARK 协议流程定义。

1) $\text{pp} \leftarrow \text{Setup}(1^{\lambda},F)$：给定安全参数 λ 和待验证计算任务 F，生成公共参数 pp。其中，pp 包括有限域 $\mathbb{F}$、哈希函数 H、FRI 协议参数（低阶多项式阶数限制）以及代数约束集合 $\mathcal{C}$。

2) $\text{cm} \leftarrow \text{Commit}(w,r,\text{pp})$：给定见证 $w \in \mathbb{F}^{l}$、随机数 $r \in \mathbb{F}^{l}$ 和公共参数 pp，证明方构造多项式承诺 cm 作为见证多项式插值的默克尔树根（Merkle root），其哈希为 $\text{cm}=\text{H}(w,r)$。

3) $\pi \leftarrow \text{Prove}(x,w,r,\text{pp})$：给定公开输入 $x \in \mathbb{F}^{m}$、见证 $w \in \mathbb{F}^{l}$、随机数 r 和公共参数 pp，首先计算执行迹 Trace 并进行 LDE 扩展。随后，根据代数约束集合，构造约束多项式并计算验证多项式，再利用 FRI 协议生成低度证明，输出完整的非交互证明 π，形式上，证明 π 包含 $\pi=(\pi_{\text{trace}},\pi_{\text{constraints}},\pi_{\text{FRI}})$。

4) $\{0,1\} \leftarrow \text{Verify}(x,\text{cm},\pi,\text{pp})$：验证方给定公开输入 x、承诺 cm、证明 π 与公共参数 pp，执行以下 3 个子过程：验证执行迹多项式的 Merkle 承诺及路径；验证约束多项式在随机挑战点的评估值；执行 FRI 协议验证多项式的低阶性质。若以上过程全部验证通过，则输出 1 接受，否则输出 0 拒绝。

该证明协议具备如下核心性质。

1) 完备性：若服务器按规范执行函数 F 并产生正确结果，则验证方以极高概率接受证明，形式化为 $\Pr[V(\text{stmt},\pi)=1 \wedge \pi \leftarrow P(c_{\text{in}},F)]=1$，其中 stmt 表示验证语句。

2) 知识健全性：若服务器未按规范执行，则在未破译底层密码假设的前提下，攻击者几乎无法生成被验证方接受的错误证明，攻击成功概率至多为 $\Pr[V(\text{stmt},\pi)=1 \wedge \neg\text{Correct}(\text{stmt})] \leqslant 2^{-\lambda}$。

3) 零知识性：存在模拟器仅给定公开输入 x 与 pp 即可生成 π，使其与真实执行下的分布计算不可区分，且不泄露关于 w 的任何额外信息。实现上通过对轨迹约束加入随机遮蔽并保持度数边界，从而在不破坏可验证性的前提下隐藏见证细节。

3　系统模型

基于第 2 节介绍的基础，本节提出 Gorak 系统模型，明确其架构、核心对象与威胁模型。系统在单节点环境中将 ElGamal 同态推理、STARK 证明和 TEE 远程认证结合在一起。服务器在本地可信内核中加载已承诺的模型，对客户端提交的密文输入执行推理，并在推理的同时生成 STARK 证明。客户端只需依据服务器返回的扩展证明报文 Quote 即可验证执行环境的可信性和结果的正确性，不需要依赖第三方。ElGamal 保证整个推理过程始终处于密文域，STARK 提供一次性公开可验证的证明，远程证明将模型参数哈希和证明后端的代码哈希写入度量值，从而确保执行期间的代码与参数不会被篡改。

3.1　核心对象

系统模型如图 2 所示，系统由客户端（可兼任验证方）与托管可信执行环境的云端服务器构成。整个推理与验证流程围绕 3 类核心机制展开，ElGamal 密文交互、TEE 远程证明 Quote 与 STARK 对应的透明可验证推理计算。

在会话启动阶段，平台完成 TEE 初始化与 enclave 创建，会话启动时，服务器在 enclave 内加载模型并计算参数承诺根 cm_{model}（LDE 码字的 Merkle 根），同时添加证明器代码的测度值。上述两者与会话随机量被写入远程证明报文的可报告域，由平台组件签发 Quote，从而在会话开始时，将执行环境、代码版本、参数承诺根 cm_{model} 三者绑定并对外背书。随后，客户端通过云端推理接口提交 ElGamal 加密的输入密文；enclave 在密态域内完成前向推理，并按图中记录执行迹、LDE 扩展、FRI 折叠的流程生成 STARK 证明。在组合多项式约束下对执行迹进行承诺与验证，确保计算是由根为

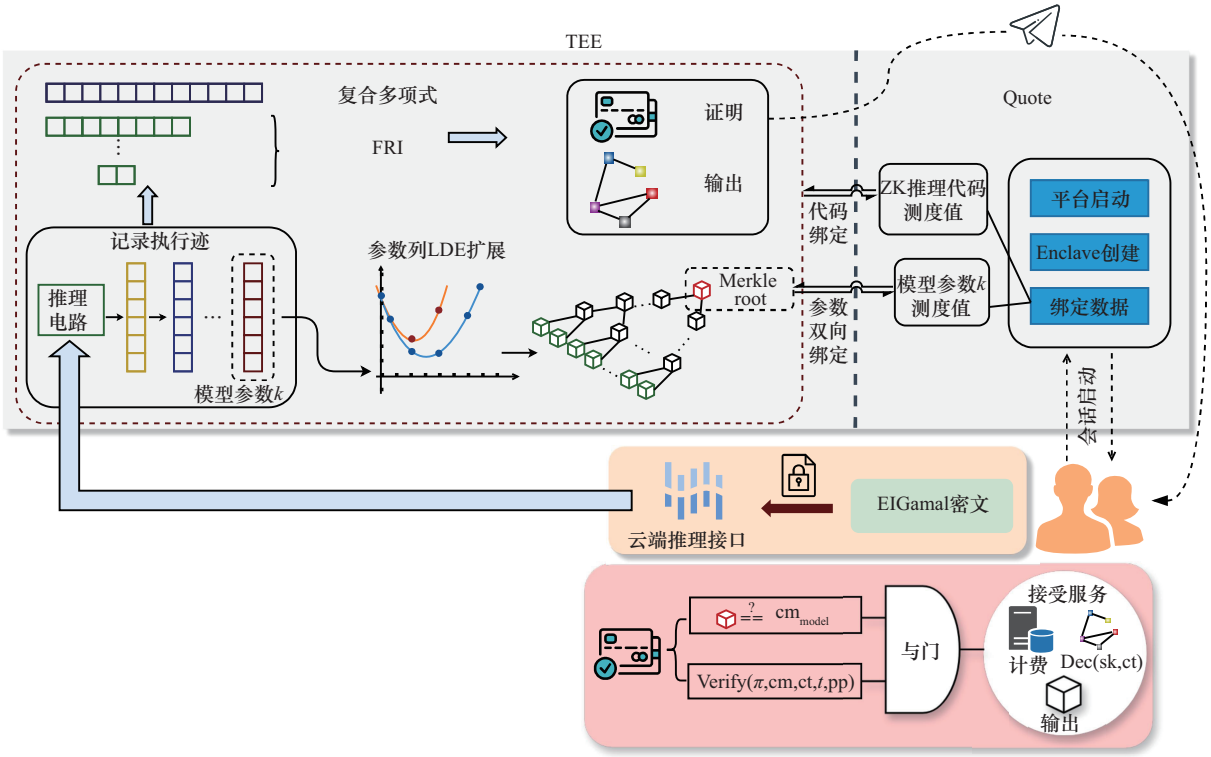

图2　系统模型

$\mathrm{cm}_{\mathrm{model}}$ 的参数实例完成。客户端侧先验证 Quote 的证书链与报文一致性，再比对 Quote 中的 $\mathrm{cm}_{\mathrm{model}}$ 与证明是否一致，继而验证 STARK 证明；仅当两者均通过时，方对返回的输出密文解密并进入后续计费流程。这样，参数根 $\mathrm{cm}_{\mathrm{model}}$ 在 Quote 与 STARK 两条链路中被同时绑定，远程可信性与外部可验证性在同一会话内达成一致。

3.2　威胁模型

Gorak 框架针对单云节点推理场景而设计，目标是在整个推理生命周期内同时保障输入与输出的隐私、执行结果的可验证性以及系统计费过程的可审计性。一旦任意环节遭到篡改或泄露，客户端应能通过验证流程即时拒绝。

系统由客户端与托管 TEE 的服务器组成。客户端被视为诚实方，按照协议提交加密数据并验证返回的证明。服务器在 enclave 外的部分被视为潜在的恶意对手，能够完全控制操作系统、网络与调度逻辑，可观察所有公开信息并主动发起攻击。然而，TEE 的内核和远程认证链在给定安全参数下被认为可信，其签发的 Quote 由平台密钥保护，不可伪造或篡改。服务器一旦完成远程认证并加载指定模型，enclave 内的执行被视为安全且不可更改。

基于上述假设，Gorak 框架所抵御的敌手攻击主要分为 3 类。

1)隐私性攻击：敌手试图窃取用户原始输入或最终预测结果。

2) 完整性攻击与欺诈行为：敌手在维持密态计算外部特征的前提下，返回错误的推理结果，或夸大资源使用以实施计费欺诈。

3) 真实性攻击：伪造远程认证报告或冒充可信环境。

3.3　安全模型

在上述威胁模型下，定义客户端 $\mathcal{C}$、服务器 $\mathcal{S}$ 与可信执行环境 $\mathcal{T}$。敌手 $\mathcal{A}$ 可控制 $\mathcal{S}$ 的外部环境并观测所有公开信息，但无法突破 $\mathcal{T}$ 的硬件隔离，也不能伪造由平台密钥签发的 Quote。系统的安全性在于即使 $\mathcal{A}$ 拥有完全的执行与通信控制权，也无法泄露输入、篡改输出或伪造证明。

在此定义下，Gorak 框架的安全目标由 3 个性质刻画。

1) 隐私性：系统保证 $\mathcal{A}$ 无法从密文或公开证明中恢复输入 x 或输出 y。形式上，对任意两组等长输入 (x_0,x_1)，若 $c_b = \mathrm{Enc}_{\mathrm{pk}}(x_b), b \in \{0,1\}$，则敌手

的区分优势满足$\left|\Pr\left[\mathcal{A}\left(c_0\right)=1\right]-\Pr\left[\mathcal{A}\left(c_1\right)=1\right]\right| \leqslant \operatorname{negl}(\lambda)$。

2) 完整性：若客户端验证通过，则输出必然与模型$f(x)$一致。记π为证明，则$\operatorname{Verify}\left(\pi, f(x)\right)=1 \Rightarrow y=f(x)$。否则验证失败，保证敌手无法生成伪造的证明使错误结果被接受。

3) 真实性：系统保证模型承诺、代码度量与执行环境的绑定真实可信。当且仅当$\operatorname{Verify}(\mathrm{pk}, H(f_W), k)=1$时，客户端接受返回结果；若敌手伪造$\mathrm{Quote}^*$并通过验证，其成功概率$\Pr[\operatorname{Verify}(\mathrm{Quote}^*)=1] \leqslant \operatorname{negl}(\lambda)$。这意味着所有计算均由经认证的TEE实例完成。

4　方案设计

为实现第3节提出的安全目标与可验证推理需求，本节详细设计Gorak的7个核心算法。系统所证明的陈述为给定公开实例，即模型承诺以及输入输出的密文承诺，存在见证，使各层的线性算子满足相应的AIR约束；层间连线与状态滚动保持一致；末态寄存器与输出承诺相符。任何一个被接受的证明均意味着整条密态前向是按照规范执行的，并且与声明输出一致。基于此，再分别构建算子成面的点加与标量乘的AIR，并给出轨迹压缩与FRI参数化的相关优化。

系统形式化定义由7个算法构成，即$\mathrm{Gorak}=(\mathrm{Setup},\mathrm{Attest},\mathrm{Comm},\mathrm{Enc},\mathrm{Infer},\mathrm{Verify},\mathrm{Dec})$，其中包含初始化、建立可信执行环境、可验证推理以及结果验证4个流程。

4.1　系统初始化

在系统启动阶段，首先依据安全参数λ、模型参数规模上界l以及TEE的配置参数$\mathrm{tee}_{\mathrm{params}}$执行初始化过程$\mathrm{Setup}(1^{\lambda}, l, \mathrm{tee}_{\mathrm{params}})$。该过程生成ElGamal密钥对$(\mathrm{pk},\mathrm{sk})$、STARK系统的公共参数pp，并输出包含SGX enclave测量值与版本信息的上下文描述$\mathrm{tee}_{\mathrm{ctx}}$。其中，密钥对用于加密与解密推理输入和输出，公共参数支撑透明证明的生成与验证，而上下文描述则为后续远程证明提供度量基准和版本校验依据。

4.2　建立可信执行环境

在系统初始化完成后，客户端与验证方需确保服务器端运行的推理环境与声明一致，从而在会话开始时建立可信根。为此，验证方首先向服务器发送一次随机挑战（challenge），该挑战将作为远程证明的输入之一，以防止重放攻击。服务器端的SGX Enclave在接收到挑战后，利用其测量值和版本信息$\mathrm{tee}_{\mathrm{ctx}}$生成远程证明报告，并通过$\mathrm{Attest}(\mathrm{challenge},\mathrm{tee}_{\mathrm{ctx}})$输出签名凭证$\sigma_{\mathrm{tee}}$及平台证明信息$\mathrm{attest}_{\mathrm{id}}$。

随后，服务器在TEE内加载推理所需的私有模型权重W，并基于随机数r及STARK系统公共参数pp计算其承诺值$\mathrm{cm}=\mathrm{Comm}\left(W, r, \mathrm{pp}, \mathrm{tee}_{\mathrm{ctx}}\right)$，该承诺作为公开数据返回。不泄露$W$的具体内容，但可作为后续验证中识别模型一致性的全局标识。为了保证cm对应的权重确实已被TEE内部正确加载，远程证明报告Quote中包含了执行环境的测量值、版本信息以及cm，并由平台密钥签名背书。验证方在收到cm与远程证明后，执行以下检查。

1) 验证Quote中的测量值与版本信息是否与预期匹配，确保执行环境可信且运行的代码与声明一致。

2) 验证Quote中包含的cm，客户端从官方发布渠道获取$\mathrm{cm}_{\mathrm{pub}}$，并在会话开始即校验$\mathrm{cm}=\mathrm{cm}_{\mathrm{pub}}$，若不一致直接拒绝进入推理。

通过上述绑定，系统在建立可信执行环境阶段即将模型承诺与可信执行环境的度量信息强关联，使得后续透明证明中的模型一致性验证具备可信基础，防止执行环境被替换或模型参数被篡改。

4.3　可验证推理

在推理阶段，采用ElGamal的加密后，卷积、全连接与平均池化等线性运算在密文域都可统一表示为权重缩放和向量求和的群运算组合。设明文前向为$y=Wx+b$。将第i个输入分量加密为密文$\mathrm{ct}_i=\mathrm{Enc}(\mathrm{pk}, x_i)$，偏置加密为$\mathrm{ct}_b=\mathrm{Enc}(\mathrm{pk}, b)$，则该层在密文域的同态前向写为

$$\mathrm{ct}_{\mathrm{out}}=\left(\sum\nolimits_i w_i \odot \mathrm{ct}_i\right) \oplus \mathrm{ct}_{\mathrm{b}} \tag{3}$$

其中，$\oplus$表示密文同态加法，$\odot$表示标量同态乘法（明文权重作用于密文点）。卷积层可看作式(3)在每个感受野上的一次实例，对任意输出位置r，将其$k\times k$感受野内的密文按固定顺序拉平成$\{\mathrm{ct}_t\}_{t=1}^{k^2}$，对应卷积核权重拉平成$\{f_t\}$，则该位置的输出密文为

$$\mathrm{ct}_{\mathrm{out}}^{(r)}=\left(\sum_{t=1}^{k^2} f_t \odot \mathrm{ct}_t\right) \oplus \mathrm{ct}_b \tag{4}$$

由式(3)、式(4)可见，要保证一层线性前向在密文域按规范完成，核心在于反复执行两类原子运算 $\oplus$ 与 $\odot$。由于ElGamal密文对应椭圆曲线上的点，$\oplus$ 等价于曲线点加，$\odot$ 等价于曲线标量乘；因此本文在STARK中为二者运算分别构造约束。验证方接受的证明即表明式(4)的等式在密文域成立，进而神经网络的卷积、全连接、平均池化均被端到端可验证。

1)点加法约束：为将椭圆曲线点加法嵌入AIR框架，本文为每个时间步构造7列轨迹 $(P_x,P_y,Q_x,Q_y,F,R_x,R_y)$，其中 (P_x,P_y) 和 (Q_x,Q_y) 为输入点，(R_x,R_y) 为输出点，布尔位 $F\in\{0,1\}$ 用于区别真实加法与简单复制两种转移模式。令曲线方程为 $y^2=x^3+ax+b$，给出式(5)的如下9条转移约束来确保运算正确性。

$$
\begin{aligned}
&\mathrm{C}_0{:}P_y^2-(P_x^3+a\cdot P_x+b)=0\\
&\mathrm{C}_1{:}R_y^2-(R_x^3+a\cdot R_x+b)=0\\
&\mathrm{C}_2{:}(1-F)\cdot[Q_y^2-(Q_x^3+a\cdot Q_x+b)]=0\\
&\mathrm{C}_3{:}F\cdot(R_x-P_x)=0\\
&\mathrm{C}_4{:}F\cdot(R_y-P_y)=0\\
&\mathrm{C}_5{:}(1-F)\cdot[(Q_x-P_x)^2(R_x+P_x+Q_x)-\\
&(Q_y-P_y)^2]=0\\
&\mathrm{C}_6{:}(1-F)\cdot[(Q_x-P_x)(R_y+P_y)-(Q_y-P_y)\\
&(P_x-R_x)]=0\\
&\mathrm{C}_7{:}P_{\tilde{x}}-R_x=0\\
&\mathrm{C}_8{:}P_{\tilde{y}}-R_y=0
\end{aligned}
\tag{5}
$$

约束将椭圆曲线上的加法以无除法的多项式形式嵌入AIR，在每一时间步，轨迹包含被加数 (P_x,P_y)、加数 (Q_x,Q_y) 与本步输出 (R_x,R_y) 等列，并在需要时通过布尔位选择真实加法或复制，使其与密文域中的 $\oplus$ 运算对应。为了消除除法并避免伪解，式(5)的9条约束按斜率关系与坐标更新一致性的顺序约束整个轨迹，先通过 C_0 和 C_1 同时要求 (P,R) 落在曲线上，随后仅在加法分支激活时对 Q 的成员性检查，即 $\mathrm{C}_2{:}(1-F)\cdot[Q_y^2-(Q_x^3+a\cdot Q_x+b)]=0$，避免在复制模式下施加不必要的约束，通过两条复制等式 C_3、C_4 强制保持输出一致；相应地，在 $F=0$ 的真实加法分支，以无除法的仰弦关系与坐标更新多项式直接刻画 $R=P+Q$，分别对应 C_5 与 C_6，两式等价于传统点加公式在清分母后的多项式形式，不引入除法与分支判断。跨步写回通过 C_7 与 C_8 将本步输出作为下一步的累加器，实现轨迹的顺序推进。

2) 点乘法约束：在AIR框架中完整刻画标量乘法 $R=kP$，定义一条11维轨迹 $(A_x,A_y,C_x,C_y,Q_x,Q_y,R_x,R_y,b,s_d,s_a)$，其中 $\left(C_x,C_y\right)$ 保存当前累加器 C 的倍点坐标；$\left(Q_x,Q_y\right)$ 为本轮参与加法的点 Q；布尔位 b 为控制位点；$\left(R_x,R_y\right)$ 记录本行结果；(A_x,A_y) 为本次累加器；s_d、s_a 为倍点、加法的斜率中间变量，全局已强制所有点在椭圆曲线上。

$$
\begin{aligned}
&\mathrm{C}_0{:}2A_ys_d-(3A_x^2+a)=0\\
&\mathrm{C}_1{:}C_x-(s_d^2-2A_x)=0\\
&\mathrm{C}_2{:}C_y-(s_d(A_x-C_x)-A_y)=0\\
&\mathrm{C}_3{:}b(b-1)=0\\
&\mathrm{C}_4{:}b((C_y-Q_y)-s_a(C_x-Q_x))=0\\
&\mathrm{C}_5{:}(1-b)(R_x-C_x)+b(R_x-(s_a^2-C_x-Q_x))=0\\
&\mathrm{C}_6{:}(1-b)(R_y-C_y)+b(R_y-(s_a(C_x-R_x)-C_y))=0\\
&\mathrm{C}_7{:}A_x^{\mathrm{next}}-R_x=0\\
&\mathrm{C}_8{:}A_y^{\mathrm{next}}-R_y=0
\end{aligned}
\tag{6}
$$

为确保运算正确性与语义完整性，整个轨迹需同时满足9条转移约束，令曲线方程为 $y^2=x^3+ax+b$，给出9条转移约束如式(6)来确保运算正确性。首先由 C_0、C_1、C_2 在不做除法的情况下以切线公式完成倍点 $C=2A$，其中 $\mathrm{C}_0{:}2A_ys_d-(3A_x^2+a)=0$ 确定倍点斜率，$\mathrm{C}_1{:}C_x-(s_d^2-2A_x)=0$、$\mathrm{C}_2{:}C_y-(s_d(A_x-C_x)-A_y)=0$ 给出倍点坐标；随后由 C_3 把控制位限制在0和1，并用 C_4、C_5、C_6 在同一组多项式里区分两种分支，当 $b=1$ 时，C_4 确定仰弦斜率，C_5、C_6 由 $R_x-(s_a^2-C_x-Q_x)$、$R_y-(s_a(C_x-R_x)-C_y)$ 计算输出 $R=C+Q$；当 $b=0$ 时，复制 $R=C$。最后由 C_7、C_8 将 R 写回为下一行累加器 $A^{\mathrm{next}}=(R_x,R_y)$，从而把本步的整体更新固定为 $A^{\mathrm{next}}=2A+bQ$,再与下一位的同类约束级联，得到整条位级轨迹。上述构造中，倍点仅依赖 s_d，加法仅依赖 s_a，列语义与运算阶段一一对应并且与群运算法则完全等价。

为降低STARK证明生成与验证开销，本文设计自适应位消融优化算法（如算法1所示）与FRI参数化策略。前者通过压缩标量乘法的执行迹，减少约束实例化次数；后者通过优化域扩展与查询次数，在保证安全性的前提下提升证明效率。

算法1 自适应位消融优化算法

输入 椭圆曲线基点 G、标量 w 的二进制位序

列 $\boldsymbol{b}=(b_{n-1},\cdots,b_0)$

输出　执行迹矩阵 $\boldsymbol{T}$，约束实例化集合 $\mathcal{C}$

1) 初始化累加器 $\boldsymbol{a}\leftarrow\boldsymbol{O}$，位指针 $i\leftarrow n-1$，$\boldsymbol{T}\leftarrow\varnothing$，$\mathcal{C}\leftarrow\varnothing$
2) while $i\geqslant 0$ do：
3) 读取当前位 b_i，置标志 $B\leftarrow b_i$
4) 倍点计算 $\boldsymbol{a}'\leftarrow 2\boldsymbol{a}$；生成倍点行 $\boldsymbol{r}_i$ 并追加到 $\boldsymbol{T}$
5) 实例化倍点在曲线上的约束
6) if $b_i=1$ then：　// 条件加：当前位为 1
7) 点加计算 $\boldsymbol{a}''\leftarrow\boldsymbol{a}'+G$；生成加法行 $R=C+Q$ 并追加到 $\boldsymbol{T}$
8) 实例化点加约束
9) 更新累加器 $\boldsymbol{a}\leftarrow\boldsymbol{a}''$
10) else
11) 更新累加器 $\boldsymbol{a}\leftarrow\boldsymbol{a}'$
12) end if
13) $i\leftarrow i-1$
14) end while
15) 边界断言 $\boldsymbol{a}_{\text{start}}=O,\boldsymbol{a}_{\text{end}}=[w]G$ }；

本文设计的自适应位消融优化算法，其结合点乘运算的双倍加流水与AIR约束生成，对标量权重逐位扫描，若当前比特 $b_i=0$ 为某情况，仅执行倍点步骤并回写累加器，不生成额外加法行；$b_i=1$ 则按常规路径执行倍点与条件加。AIR中的复制掩码约束保证轨迹连续性和语义完备性，当控制位为0时，强制本行输出与上一行累加器一致，不破坏曲线合法性与运算链路。算法1在生成执行迹时实例化多项式约束，这种压缩在不改变证明语义的前提下提升了生成与验证端吞吐，且与高并发算子调度兼容。

随后对FRI进行参数化配置，如表2所示，让低度检验有足够度数余量，且在常数级折叠与查询次数下达128位量级知识健全性。本文选128位素数域 $\mathbb{F}$，采用LDE膨胀因子4。点加AIR最高多项式度为3、点乘AIR最高度为5，4倍扩域后，验证多项式有效度上界约为评估域大小 $\frac{1}{4}$，给FRI留安全余量，不需要扩域。FRI折叠因子 $\eta=2$ 选取，既保证每轮折叠后度域比二分递减，又避免高折叠因子致常数放大；给定评估域规模，折叠轮数固定。查询次数设为55，结合128位素域随机挑战，可压低低度检验误受理概率 2^{-126}。为抵御挑战可预测性与碰撞带来的边际松弛，引入20位研磨因子，要求证明者提交承诺前完成 2^{20} 前导零试探。评估域选取偏移为3的陪集，避免低度多项式在零因子处对齐问题，提升LDE数值稳定性。综合这些参数，在文献[32]的标准分析框架下，STARK不完备概率由度域比与FRI查询项控制，确保最终的整体误差上界不超过 2^{-128} 的安全目标值。

表2　　FRI协议参数

参数名称	参数值	说明
SECURITY_LEVEL	128位	安全级别
BLOWUP_FACTOR	4	膨胀因子
FRI_NUMBER_OF_QUERIES	55次	FRI查询次数
COSET_OFFSET	3	陪集偏移量
GRINDING_FACTOR	20位	工作量证明难度

4.4　结果验证

验证方接收STARK证明、模型参数承诺、输出密文、会话随机性与公共参数后，首先运行验证算法 Verify $(\pi,\text{cm},\text{ct},t,\text{pp})$ 以核验执行迹是否满足全部AIR约束，并通过FRI检查低度性质；随后检验扩展Quote的完整性与真实性，确认测量值、版本信息与模型承诺与预期一致且与证明器哈希匹配。只有当STARK证明与扩展Quote同时通过验证时，客户端才会对输出密文进行解密，获得推理结果，并进入计费阶段。

计费设计不依赖服务器报告的GPU和内存使用量或运行时长，而是直接将费用计算绑定于推理电路执行过程中可验证的工作量度量。在执行迹中，针对每类算术原语，系统都会设置相应的计数寄存器，并与其代数约束同步更新。每当算子被调用时，执行迹既需输出运算结果，同时每类算子的计数寄存器均与递增约束绑定，在执行迹中随算子调用强制递增，以实现对工作量的可验证计量，从而形成一个与电路语义紧密对应、无法伪造的算子计数向量。

所得的计数向量由STARK证明机制保证其与逐步运算过程保持一致，阻止服务方通过跳过步骤或伪造结果来制造虚假的计量。同时，TEE的扩展Quote将模型承诺、代码度量与计费规则版本绑定在一起，防止通过替换模型或篡改计费逻辑规避费

用。客户端或独立审计方在验证STARK与TEE的正确性之后，即可根据公开费率表重现计费结果，从而完成账单的独立验证。

5 安全性证明

为验证第4节方案的安全性，本节基于ElGamal、STARK与TEE的安全假设和3.3节的安全模型，形式化证明Gorak的隐私性、完整性、真实性。记安全参数为λ，$\mathrm{negl}(\lambda)$表示可忽略函数，PPT表示多项式时间算法。系统协议$\Pi=(\mathrm{Setup},\mathrm{Attest},\mathrm{Comm},\mathrm{Enc},\mathrm{Infer},\mathrm{Verify},\mathrm{Dec})$，一次会话涉及的公开对象均来自第4节中的7步流程，其中公共输入含系统参数pp、会话标识s_{id}、代码度量哈希H_{code}、模型参数承诺$\mathrm{cm}_{\mathrm{model}}$，TEE远程认证生成Quote，STARK证明记为π以及随机挑战记为nonce。验证关系$\mathcal{R}$仅由第4节给出的7个算法确定，其中定义A1为EC点加AIR；A2为EC标量乘AIR；A3为密态线性层前向；A4为轨迹掩码和随机化；A5为STARK证明生成；A6为STARK验证；A7为TEE Quote绑定的执行环境测量值、版本信息、模型参数承诺与随机挑战。后续每个性质的论证均在这些算法的输出之上展开。

安全假设如下。

1) STARK满足完备性、知识健全性与零知识性[32]。

2)ElGamal同态加密满足IND-CPA[33]安全性。

3)哈希承诺抗碰撞且具绑定性[34]。

4)TEE远程证明签名体制满足EUF-CMA[35-36]，不可伪造；TEE隔离性保证测量与初始化后的内存不可被外界修改。

在一次会话执行中，敌手$\mathcal{A}$的真实视图定义为

$$\mathrm{View}^{\mathrm{real}}_{\Pi,\mathcal{A}}=\left(\mathrm{pp},H_{\mathrm{code}},\mathrm{cm}_{\mathrm{model}},\mathrm{nonce},\mathrm{ct}_{\mathrm{in}},\mathrm{ct}_{\mathrm{out}},\pi,\mathrm{Quote}\right) \tag{7}$$

其中，给定明文x加密得到密文$\mathrm{ct}_{\mathrm{in}}=\mathrm{Enc}(x)$，$\mathrm{ct}_{\mathrm{out}}$为密态前向的密文输出。相应的模拟视图$\mathrm{View}^{\mathrm{sim}}_{\Pi,\mathcal{A}}$由安全性证明中的模拟器产生。

5.1 隐私性

隐私性的随机化来源仅有2处，分别是A4的轨迹掩码与A5的零知识模拟接口；A3的密态前向始终在密文半群上运算，不触及明文。证明中第一步用模拟证明替换π对应A5的模拟器；第二步把输入输出密文替换为等维零密文的加密利用同一公钥的IND-CPA，不影响A7的Quote，因为Quote只包含测量、版本、承诺、挑战等公开量。

在隐私实验$\mathrm{Exp}^{\mathrm{priv}}_{\Pi,\mathcal{A}}(\lambda)$中，敌手$\mathcal{A}$指定等长输入$\left(x_0,x_1\right)$。挑战者采样$b\leftarrow\{0,1\}$，返回$\mathrm{ct}_{\mathrm{in}}=\mathrm{Enc}(x_b)$，并运行$\Pi$，返回视图$\mathrm{View}=(H_{\mathrm{code}},\mathrm{cm}_{\mathrm{model}},\mathrm{ct}_{\mathrm{in}},\mathrm{ct}_{\mathrm{out}},s_{\mathrm{id}},\pi,\mathrm{Quote})$交给$\mathcal{A}$。$\mathcal{A}$输出猜测$b'$。实验输出1当且仅当$b'=b$，定义优势为$\mathrm{Adv}^{\mathrm{priv}}_{\Pi,\mathcal{A}}(\lambda)=\left|\Pr[b'=b]-\frac{1}{2}\right|$。

定理1 输入与输出隐私。

证明 设$\mathrm{View}^{\mathrm{real}}$为真实执行视图，做如下2次标准混合步骤。

1) 保持$\left(H_{\mathrm{code}},\mathrm{cm}_{\mathrm{model}},\mathrm{ct}_{\mathrm{in}},\mathrm{ct}_{\mathrm{out}},s_{\mathrm{id}}\right)$、Quote不变，仅用同一公开实例的STARK模拟证明替换π。由STARK零知识性，该替换后的视图与$\mathrm{View}^{\mathrm{real}}$计算不可区分。

2) 在上一步基础上，仅把$\mathrm{ct}_{\mathrm{in}}=\mathrm{Enc}(x_b)$替换为对等维零向量的加密$\mathrm{Enc}(0)$；同理把$\mathrm{ct}_{\mathrm{out}}=\mathrm{Enc}(y_b)$替换为$\mathrm{Enc}(0)$。由ElGamal的IND-CPA可知，这一步替换与前一视图亦计算不可区分。

在完成上述2步后，得到的视图与比特b无关，输入输出均为零明文的随机加密，证明π为对公开实例的模拟，Quote只含公开量，因此敌手成功概率为$\Pr[b'=b]=\frac{1}{2}\pm\mathrm{negl}(\lambda)$。即得$\mathrm{Adv}^{\mathrm{priv}}_{\Pi,\mathcal{A}}(\lambda)\leqslant\mathrm{negl}(\lambda)$。证毕。

5.2 完整性

完整性依赖两层语义，其中算术语义由A1、A2给出的点加与标量乘AIR约束，将正确前向的门级条件写成多项式等式；另外业务语义由A3用式(3)、式(4)把密态线性层的正确性映射到输出密文；同时由A5和A6提供知识提取与验证。从而当A6接受时，可从π提取到满足A1、A2、A3的执行迹；若输出密文与A3指定的正确前向不一致，则与可提取见证矛盾。另一分支由A7处理，若Quote非法或绑定不符，则落到签名绑定性的安全性上，完整性实验直接拒绝。

在完整性实验$\mathrm{Exp}^{\mathrm{int}}_{\Pi,\mathcal{A}}(\lambda)$中，$\mathcal{A}$与挑战者按照$\Pi$交互后输出报文$(\tilde{\pi},\widetilde{\mathrm{cm}}_{\mathrm{model}},\widetilde{\mathrm{ct}}_{\mathrm{in}},\widetilde{\mathrm{ct}}_{\mathrm{out}},\mathrm{Quote})$。若$\mathrm{Verify}(\tilde{\pi},\widetilde{\mathrm{cm}}_{\mathrm{model}},\widetilde{\mathrm{ct}}_{\mathrm{in}},\mathrm{Quote})=1$且$\mathrm{Dec}(\widetilde{\mathrm{ct}}_{\mathrm{out}})\neq F(x)$，则实验输出1，否则输出0。定义优势为$\mathrm{Adv}^{\mathrm{int}}_{\Pi,\mathcal{A}}(\lambda)=\Pr[\mathrm{Exp}^{\mathrm{int}}_{\Pi,\mathcal{A}}(\lambda)=1]$。

定理 2　计算完整性。

证明　假设存在 PPT 敌手以非忽略概率 ϵ 产出 $(\tilde{\pi}, \widetilde{\mathrm{cm}}_{\mathrm{model}}, \widetilde{\mathrm{ct}}_{\mathrm{in}}, \widetilde{\mathrm{ct}}_{\mathrm{out}}, \mathrm{Quote})$ 使 Verify = 1 且解密输出与正确前向 $F(x)$ 不一致，分以下 2 种情形。

1) Quote 合法且绑定的 $(H_{\mathrm{code}}, \mathrm{cm}_{\mathrm{model}})$ 情况与客户端预期相符。依据 TEE 的隔离性可知，证明生成是在指定代码内完成的；若 $\tilde{\pi}$ 通过验证但输出错误，触发 STARK 知识提取器得到与 AIR 满足的见证，从而与输出错误矛盾，破坏了知识健全性。

2) 若 Quote 被伪造或绑定了非预期 $(H_{\mathrm{code}}, \mathrm{cm}_{\mathrm{model}})$ 内容，那么敌手能够构造出伪造的平台签名，或者在 s_{id} 包含的消息上进行二次签名，这违反了 EUF-CMA；若试图以不同参数重放旧报告，就需要为哈希承诺找到碰撞或第二原像，这与抗碰撞绑定性相矛盾。

这两种情形均与基本假设冲突，因此 ϵ 可以忽略。所以同时满足知识健全性与 EUF-CMA，则 $\mathrm{Adv}^{\mathrm{int}}_{\Pi,\mathcal{A}}(\lambda) \leqslant \mathrm{negl}(\lambda)$，证毕。

5.3　执行真实性

真实性涉及 A7 的 Quote 绑定与 A6 验证端复现挑战并检查绑定，没有合法硬件密钥与测量绑定，无法产出可验 Quote；即便重放旧报告，也因随机挑战不同而在 A6 处被拒。

在真实性实验 $\mathrm{Exp}^{\mathrm{auth}}_{\Pi,\mathcal{A}}$ 中，敌手企图在不经合法 TEE 的情况下生成可验的 Quote 绑定到某度量 H_{code}，使得 $\mathrm{Verify}(\widetilde{\mathrm{cm}}_{\mathrm{model}}, H_{\mathrm{code}}, \mathrm{Quote}, \mathrm{nonce}) = 1$，成功则实验输出 1，否则输出 0。优势定义为 $\mathrm{Adv}^{\mathrm{auth}}_{\Pi,\mathcal{A}}(\lambda) = \Pr[\mathrm{Exp}^{\mathrm{auth}}_{\Pi,\mathcal{A}}(\lambda) = 1]$。

定理 3　执行真实性。

证明　认证通过意味着存在合法硬件对消息 $(H_{\mathrm{code}}, \mathrm{cm}_{\mathrm{model}}, s_{\mathrm{id}})$ 的有效签名，TEE 隔离性排除运行时篡改，故谁执行且执行何物被绑定。若尝试以旧证明跨会话重放，s_{id} 进入 Fiat-Shamir 启发式生成的随机挑战，导致低度检验实例改变；成功重放将要求找到哈希碰撞或复用式签名，与哈希抗碰撞和 EUF-CMA 矛盾，在未持有私钥且不破坏隔离的前提下无法伪造或者跨会话重放，则 $\mathrm{Adv}^{\mathrm{auth}}_{\Pi,\mathcal{A}}(\lambda) \leqslant \mathrm{negl}(\lambda)$。证毕。

归纳 5.1 节~5.3 节的安全性分析，若 A6 接受公开对象 π，则存在由 A1、A2、A3、A4 构成的执行迹见证使 $R = 1$ 即语义完备；同时，敌手对输入输出的区分优势由 A4、A5 控制、对错误结果被接受的优势由 A1、A2、A3、A5、A6 控制，以及对 Quote 伪造和重放的优势由 A7 控制均为可忽略量。至此，整个协议流程中的 7 个算法 $\Pi = (\mathrm{Setup}, \mathrm{Attest}, \mathrm{Comm}, \mathrm{Enc}, \mathrm{Infer}, \mathrm{Verify}, \mathrm{Dec})$ 与隐私性、完整性及执行真实性在同一验证关系上闭环。

本节对 Gorak 框架的安全性进行了形式化分析与证明，系统性地阐述了其在隐私性、完整性及执行真实性 3 个核心维度的安全保障能力。证明了 Gorak 在云端单节点推理环境中能够同时实现输入与输出的隐私保护、计算过程的公开可验证性以及执行环境的真实性保障。

6　实验分析

在 vPIN[22]框架基础上，本文重构并接入了 Gorak 证明链路，新增 2 100 行代码，其中 Rust 实现基于 lambdaworks 密码计算库的 STARK 证明生成与验证流程，Python 封装同态推理、TEE 接口及系统集成。主要改动包括将 FRI 与椭圆曲线 AIR 约束整合入 lambdaworks；将证明生成逻辑移入内核，并绑定 Quote 生成与证书链校验以保障执行完整性；针对点加与标量乘约束引入自适应位消融以缩短执行迹。ElGamal 同态运算及 CNN 前向逻辑保持 vPIN 原实现不变，新的证明器原位替代旧版 Spartan 生成器。

6.1　实验参数配置

本文的性能评测在 SGX 仿真模式下进行；所有安全结论均依赖 SGX 硬件及数据中心认证原生证明方案（DCAP）的标准安全性质，与仿真实现无关。面向更大模型与更高内存占用的场景，可平滑迁移至虚拟化的安全嵌套分页（SEV-SNP）或 Intel 可信域扩展（TDX, trust domain extensions）平台以获得整机和整虚拟机级的受保护内存；在不改变协议与可信边界的前提下，可保留本文的模型承诺绑定与透明证明链路，并利用更大受保护内存承载同态前向与证明生成，从而复用相同的推理与验证流程。

密码学主体由表 3 给出，STARK 素数域同时承担有限域与椭圆曲线基域双重角色。曲线参数 $a = 1$ 使加法公式可简化为一次乘法与 2 次平方，进一步降低 AIR 度数；幂根空间为 192 次幂，使 LDE 膨胀因子定在 4 时仍能留出安全裕量。在该曲线阶 q 上实现 ElGamal 加解密，避免额外模切换。

表3 椭圆曲线参数

参数名称	参数值	说明
CURVE_PRIME	0x800000000000011000000000000000000000000000000000000000000000000000000000000001	STARK 素数域模数
CURVE_A	1	椭圆曲线参数 a
CURVE_B	0x6f21413efbe40de150e596d72f7a8c5609ad26c15c915c1f4cdfcb99cee9e89	椭圆曲线参数 b
GENERATOR_X	0x1EF15C18599971B7BECED415A40F0C7DEACFD9B0D1819E03D723D8BC943CFCA	生成点 x 坐标
GENERATOR_Y	0x5668060AA49730B7BE4801DF46EC62DE53ECD11ABE43A32873000C36E8DC1F	生成点 y 坐标
CURVE_ORDER	0x800000000000010FFFFFFFFFFFFFFFFB781126DCAE7B2321E66A241ADC64D2F	椭圆曲线阶数
TWO_ADICITY	192	2 的幂次数
TWO_ADIC_ROOT	0x5282db87529cfa3f0464519c8b0fa5ad187148e11a61616070024f42f8ef94	2 次本原单位根

6.2 CNN 实验结果与分析

本节在 5 条不同规模的 CNN（记为 A~E）上复现实验，对比 vPIN 与本文方案的端到端表现，实验结果如图 3 所示。

图 3 汇总了 CNN A~E 在 vPIN 与 Gorak 框架下的 3 项性能指标对比。在生成时间方面，如图 3(a) 所示，vPIN 随网络规模由 23 889 增至 83 417 ms，Gorak 则稳定在 882~3 438 ms，生成时间整体缩短 96%~97%。在验证时间方面，如图 3(b) 所示，vPIN 从 219 ms 增至 371 ms，Gorak 则为 65 ~238 ms，平均加速幅度为 30%~70%，其中在最复杂模型上验证仍保持在 0.25 s 以内。在证明文件体积方面，如图 3(c)所示，vPIN 为 253~367 KB，Gorak 稳定在 770~865 KB，体积约为 vPIN 的 2.4 倍。需要指出的是，在 CNN C~E 上，Gorak 的生成时间、验证时间与证明文件体积几乎重合，这主要源于 STARK 内部执行迹受快速傅里叶变换幂级约束的特性，当网络规模未跨越下一个幂级阈值时，电路规模填充于固定长度的迹表中，不会触发额外的折叠轮次或域扩展，从而使开销曲线趋于平坦。整体而言，Gorak 通过以空间换算力的策略，在保持公开可验证的同时，将生成时间降低 96%~97%、验证时间减少 30%~70%，仅增加半兆级证明体积，显著提升了单节点推理的吞吐与响应能力，而不会因证明体积成为新的性能瓶颈。

为验证基于低度代数中间值表示的自适应位消融优化效果，本文在 5 个不同规模的网络模型上进行了对比实验。实验保持证明系统参数不变，仅切换位消融以观察其影响。表 4 给出了点乘数、原始执行迹行数、填充执行迹行数、填充占比、生成时间与验证时间等总体结果；表 5 进一步分解了随机访问多项式（RAP)、复合多项式、域外点评估（OOD）、FRI、FRI 验证、验证复合多项式深度与恢复挑战等阶段的时间开销。结果表明，

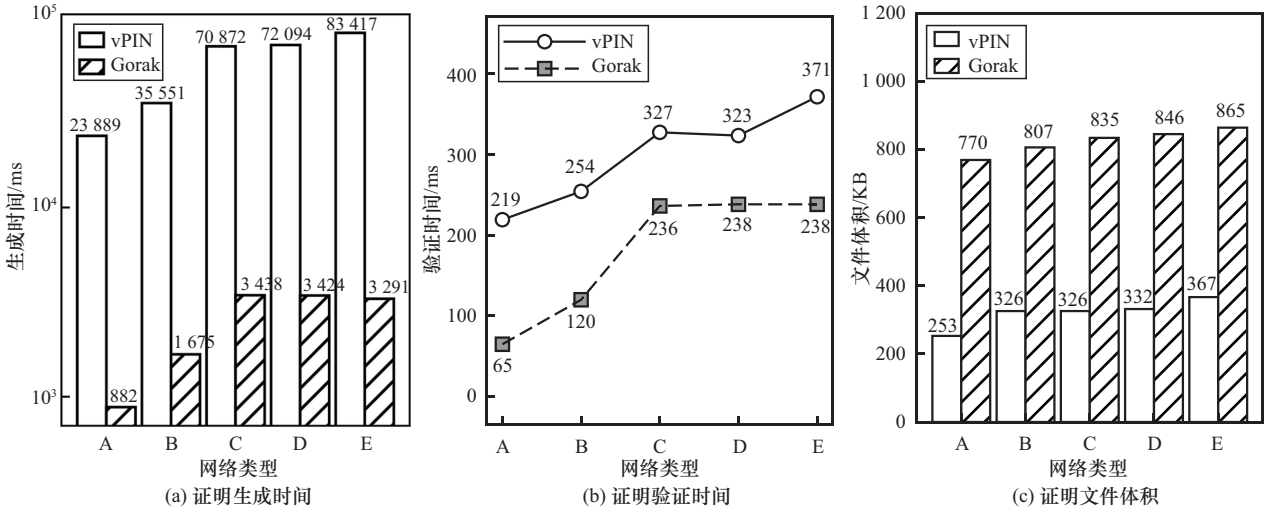

图3 证明生成验证及大小性能对比

表 4　位消融优化前后执行迹规模与性能对比

网络	点乘数	位消融	原始执行迹行数	填充执行迹行数	填充占比	生成时间/ms	验证时间/ms
A	178	ON	30 653	32 768	6.45%	932	49
A	178	OFF	40 632	65 536	38.00%	1 935	98
B	210	ON	36 897	65 536	43.70%	1 984	97
B	210	OFF	49 060	65 536	25.14%	2 014	100
C	562	ON	103 906	131 072	20.73%	4 234	221
C	562	OFF	137 800	262 144	47.43%	8 623	423
D	594	ON	110 784	131 072	15.48%	4 137	212
D	594	OFF	147 128	262 144	43.88%	8 488	453
E	658	ON	123 589	131 072	5.71%	4 088	208
E	658	OFF	164 488	262 144	37.25%	8 691	426

表 5　位消融优化前后分阶段性能剖析

网络	点乘数	位消融	RAP/ms	复合多项式/ms	OOD/ms	FRI/ms	FRI 验证/ms	验证复合多项式深度/ms	恢复挑战/ms
A	178	ON	299.69	472.76	43.45	116.88	3.41	1.10	47.80
A	178	OFF	601.61	924.03	70.25	236.14	3.54	1.09	94.00
B	210	ON	599.21	974.64	75.75	228.82	3.70	1.18	97.13
B	210	OFF	599.54	976.82	69.50	255.65	3.59	1.13	96.01
C	562	ON	1 342.11	1 985.60	147.49	531.94	4.18	1.26	216.37
C	562	OFF	2 761.72	4 138.89	292.59	990.33	4.15	1.25	418.42
D	594	ON	1 352.22	1 952.73	145.90	466.78	3.99	1.21	207.35
D	594	OFF	2 718.19	4 069.15	269.61	993.27	4.31	1.28	447.53
E	658	ON	1 338.01	1 929.41	143.90	453.59	3.91	1.20	202.79
E	658	OFF	2 801.66	4 150.06	272.08	1 009.47	4.14	1.21	419.79

位消融平均使原始执行迹行数减少约 25%，带来填充执行迹行数与填充占比的同步下降。当执行迹长度跨越 STARK 后端的二次幂填充边界时，生成时间和验证时间近似减半；若填充阶未变，则收益有限。例如，网络 C 的原始执行迹行数由 137 800 行降至 103 906 行，生成时间由 8 623 ms 降至 4 234 ms，验证时间由 423 ms 降至 221 ms。RAP 与复合多项式是生成端主要耗时环节，位消融显著缩短二者；验证端的挑战恢复阶段也明显加速。总体而言，位消融有效减少执行迹规模，在不改变系统参数和可信边界的前提下显著提升了证明生成与验证的效率。

6.3　LeNet 实验结果与分析

Gorak 在 LeNet 上的表现如表 6 所示，其汇总了 Gorak 与对比方案在 LeNet-5 的 7 个计算层上生成证明的核心指标，包括生成和验证的时间以及文件体积。

表 6　Gorak 在 LeNet 上的表现

层级	网络类型	Gorak			vPIN		
		生成时间/s	验证时间/s	文件体积/KB	生成时间/s	验证时间/s	文件体积/KB
L1	卷积层	0.086	0.007	543.29	29.085	0.236	297.88
L2	池化层	0.352	0.048	367.39	3.866	0.073	104.05
L3	卷积层	0.248	0.016	628.46	116.072	0.484	476.64
L4	池化层	0.126	0.012	302.02	1.313	0.061	82.53
L5	卷积层	1.648	0.133	802.09	376.62	2.44	856.47
L6	全连接层	1.769	0.214	748.74	28.917	0.234	297.88
L7	全连接层	1.486	0.103	660.13	17.173	0.194	222.63
总计	—	5.715	0.533	4 052.12	573.046	3.722	2 338.08

在 LeNet-5 的 7 个前向计算层的实验中，Gorak 在证明生成和验证时间均表现出显著优势。如表 6 所示，在卷积层 L1 中，Gorak 的证明生成时间仅为 0.086 s，vPIN 则需要 29.085 s；在卷积层 L5 中，二

者分别为1.648 s与376.62 s。将7层生成时间累计，Gorak仅用5.715 s即可完成全网证明生成，vPIN需573.046 s，整体生成时间缩短约99.01%。在验证阶段，Gorak的性能同样显著优于vPIN。7层累计验证时间仅为0.533 s，vPIN为3.722 s，验证时间降低约85.68%。其中，卷积层差距最为明显，这得益于Gorak采用了哈希友好的FRI度数检查，替代了vPIN在椭圆曲线上执行的高开销配对承诺运算。总体而言，性能提升主要源于在证明电路中融合了门级压缩、执行迹优化与多核并行等协同策略，从根本上削减执行迹长度与运算开销；虽然多轮FRI折叠与追加Merkle路径使证明体积增加至3.95 MB，较vPIN增加约1.7 MB，但在千兆链路下传输时间仅十余毫秒，远低于生成端节省的数百秒开销，可在多数场景中忽略不计。

此外，为评估系统端到端可部署性，本文对客户端整体处理时间与吞吐率进行了测量。实测结果显示，在完整密态推理链路下，单次请求平均时间为316.35 s，对应吞吐率约为0.003 2 QPS（每秒处理0.003 2次请求），其中主要瓶颈为椭圆曲线解密过程。TEE认证与STARK验证阶段耗时可忽略，表明验证链路对总体性能影响较小。

6.4 可信配置分析

在本文仿真环境中，远程认证流程从生成证明报文到客户端完成基于P-256椭圆曲线的签名链验证，平均耗时约为45 ms。该过程包括在安全执行环境内部调用EREPORT指令生成证明报告，向平台证书缓存服务获取硬件平台证书，执行椭圆曲线数字签名算法验证，并通过在线证书状态协议查询吊销信息。由于仅涉及一次在线查询，局域网与典型云内部链路之间的时间差异可以忽略不计。与随后的密态推理阶段相比，这一认证过程的耗时占比极小，推理阶段的执行时间往往远高于认证过程的总时间。因此，即便在每次任务冷启动时都重新执行一次远程认证，整体系统时间依旧主要由密态推理阶段决定。

7 结束语

针对云端单节点隐私推理在同时满足数据保密性与计算可验证性要求时，常面临证明生成与验证开销较高的问题，本文提出一种可验证隐私推理框架Gorak。该框架以可信执行环境为信任锚点，将模型与代码的会话级度量结果嵌入透明论证体系中，实现模型承诺与执行验证的统一，显著降低了模型一致性校验带来的电路开销。在计算与验证层面，框架在有限域中对密态运算生成低度算术轨迹，并通过自适应位消融压缩与批量FRI折叠优化，有效减少证明生成与验证时间。实验结果表明，Gorak在典型网络LeNet上的证明生成时间和验证时间分别较基线方案降低99.0%与85.7%，验证所提方案在保持隐私与可验证性前提下的高效性与可行性。未来的研究工作将进一步扩展框架对复杂非线性算子的密态支持，以提升系统在多样化神经网络结构中的适用性。同时，将面向大语言模型与Transformer等长序列结构，探索分层验证与分块证明机制，通过局部轨迹重用与增量证明优化计算与验证开销，从而在不改变框架核心设计原则的前提下，实现更复杂网络结构与更长推理序列的高效可验证支持，推动Gorak框架向通用化和高可扩展性方向发展。

参考文献：

[1] SHAN Z, REN K, BLANTON M, et al. Practical secure computation outsourcing: a survey[J]. ACM Computing Surveys, 2018, 51(2): 1-40.
[2] Microsoft. Azure ai services [R] 2025.
[3] GOOGLE LLC. Google cloud AI platform[R]. 2025.
[4] GHODSI Z, GU T Y, GARG S. SafetyNets: verifiable execution of deep neural networks on an untrusted cloud[C]//Proceedings of the 31st International Conference on Neural Information Processing System. Massachusetts: MIT Press, 2017: 4675-4684.
[5] ZHAO L C, WANG Q, WANG C, et al. VeriML: enabling integrity assurances and fair payments for machine learning as a service[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(10): 2524-2540.
[6] GENTRY C. A fully homomorphic encryption scheme[D]. Palo Alto: Stanford University, 2009.
[7] CRAMER R, DAMGÅRD IB, NIELSEN J B. Secure multiparty computation and secret sharing[M]. Cambridge: Cambridge University Press, 2015.
[8] DOWLIN N, GILAD-BACHRACH R, LAINE K, et al. CryptoNets: applying neural networks to encrypted data with high throughput and accuracy[C]//16: Proceedings of the 33rd International Conference on International Conference on Machine Learning. New York: ACM Press, 2016: 201-210.
[9] DATHATHRI R, SAARIKIVI O, CHEN H, et al. CHET: an optimizing compiler for fully-homomorphic neural-network inferencing[C]//Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation. New York: ACM Press, 2019: 142-156.
[10] TRAMER F, BONEH D. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware[J]. arXiv Preprint, arXiv: 1806.03287, 2018.
[11] GROVER K, TOPLE S, SHINDE S, et al. Privado: practical and secure DNN inference with enclaves[J]. arXiv Preprint, arXiv:

1810.00602, 2018.
[12] GROTH J. On the size of pairing-based non-interactive arguments[C]// Advancesin Cryptology – EUROCRYPT2016. Berlin: Springer, 2016: 305-326.
[13] GABIZON A, WILLIAMSON Z J, CIOBOTARU O M. PLONK: permutations over Lagrange-bases for oecumenical noninteractive arguments of knowledge[J]. IACR Cryptology ePrint Archive, 2019: 953.
[14] SETTY S. Spartan: efficient and general-purpose zkSNARKs without trusted setup[C]//Advancesin Cryptology – CRYPTO 2020. Berlin: Springer, 2020: 704-737.
[15] BEN-SASSON E, BENTOV I, HORESH Y, et al. Scalable, transparent, and post-quantum secure computational integrity[J]. IACR Cryptology ePrint Archive, 2018: 46.
[16] BAGAD S, DOMB Y, THALER J. The sum-check protocol over fields of small characteristic[J]. IACR Cryptology ePrint Archive, 2024: 1046.
[17] KNABENHANS C, VIAND A, MERINO-GALLARDO A, et al. vFHE: verifiable fully homomorphic encryption[C]//Proceedings of the 12th Workshop on Encrypted Computing & Applied Homomorphic Cryptography. New York: ACM Press, 2024: 11-22.
[18] BADAWI A A, ALEXANDRU A, BATES J, et al. OpenFHE: open-source fully homomorphic encryption library[J]. IACR Cryptology ePrint Archive, 2022: 915.
[19] BOTTAZZI E. Greco: fast zero-knowledge proofs for valid FHE RLWE ciphertexts formation[J]. IACR Cryptology ePrint Archive, 2024: 594.
[20] LUO J, CHEN G, ZHANG Y, et al. Centaur: bridging the impossible trinity of privacy, efficiency, and performance in privacy-preserving transformer inference[J]. arXiv Preprint, arXiv: 2412.10652, 2024.
[21] ZHANG Y, XUE J, ZHENG M, et al. Cipherprune: efficient and scalable private transformer inference[J]. arXiv Preprint, arXiv: 2502.16782, 2025.
[22] RIASI A, GUAJARDO J, HOANG T. Privacy-preserving verifiable neural network inference service[J]. arXiv Preprint, arXiv: 2024.00063, 2024.
[23] WENG C K, YANG K, XIE X, et al. Mystique: efficient conversions for zero-knowledge proofs with applications to machine learning[J]. IACR Cryptology ePrint Archive, 2021: 730.
[24] WANG H D, HOANG T. ezDPS: an efficient and zero-knowledge machine learning inference pipeline[J]. Proceedings on Privacy Enhancing Technologies,2023, 2023(2): 430-448.
[25] LIU T Y, XIE X, ZHANG Y P. zkCNN: zero knowledge proofs for convolutional neural network predictions and accuracy[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. NewYork: ACM Press, 2021: 2968-2985.
[26] SUN H C, BAI T H, LI J, et al. zkDL: efficient zero-knowledge proofs of deep learning training[J]. IEEE Transactions on Information Forensics and Security, 2025, 20: 914-927.
[27] SUN H C, LI J, ZHANG H Y. zkLLM: zero knowledge proofs for large language models[C]//Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2024: 4405-4419.
[28] WENG J S, WENG J, TANG G, et al. pvCNN: privacy-preserving and verifiable convolutional neural network testing[J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 2218-2233.
[29] XING Y, WANG L, LI Q, et al. A comprehensive survey on privacy-preserving deep learning[J]. ACM Computing Surveys, 2023, 56(3): 1-36.
[30] PENG H, LIU S, WU J, et al. Towards verifiable and privacy-preserving deep learning in the cloud: a framework and challenges [J]. IEEE Transactions on Dependable and Secure Computing. 2025, 22(1): 1-15.
[31] SCARAMUZZA D, PATEL J, WHITEHOUSE O, et al. Bridging the gap: verifiable computation for encrypted data in machine learning[J]. Proceedings of the IEEE, 2025, 113(1): 1-25.
[32] BEN-SASSON E, BENTOV I, HORESH Y, RIABZEV M. Fast reed-solomon interactive oracle proofs of proximity[C]//Proceedings of the 45th International Colloquium on Automata, Languages, and Programming. Springer Nature: Leibniz International Proceedings in Informatics, 2018: 1-17.
[33] KATZ J, LINDELL Y. Introduction to modern cryptography[M]. 3rd ed. Boca Raton: CRC Press , 2020.
[34] MERKLE R. A certified digital signature [C]//Conference on the Theory and Application of Cryptology. Berlin: Springer, 1990: 218-238.
[35] INTEL. Intel® SGX data center attestation primitives (DCAP) white paper[R]. 2020.
[36] National Institute of Standards and Technology. Digital signature standard:FIPS 186-4[R]. 2013.

[作者简介]

田有亮（1982– ），男，贵州盘州人，博士，贵州大学教授、博士生导师，主要研究方向为博弈论、密码学与安全协议、大数据隐私保护。

陈壕立（1996– ），男，侗族，贵州贵阳人，贵州大学硕士生，主要研究方向为隐私保护、可验证计算、零知识证明技术等。

杨江迪（1997– ），男，仡佬族，贵州遵义人，贵州大学硕士生，主要研究方向为隐私保护、图像哈希、图像处理、门限签名、生成式人工智能安全框架等。

李梦倩（1997– ），女，河北衡水人，贵州大学博士生，主要研究方向为联邦学习、隐私保护、差分隐私技术等。

张馨予（1995– ），女，山西太原人，贵州大学博士生，主要研究方向为隐私保护机器学习、可验证计算、博弈论等。

策力木格（1979– ），男，博士，日本电气通信大学教授，主要研究方向为无线网络、物联网系统、人工智能、边缘计算等。