T3C：具有一致性保证的测试时张量压缩

Ismail L amaakal [1] Chaymae Yahyati [1] Yassine Maleh [2] Khalid El Makkaoui [1] Ibrahim Ouahb i [1]

arXiv:2601.01299v1 [cs.CL] 3 Jan 2026

摘要

我们提出了**T3C**，一种只需训练一次、在测试时受预算条件约束的压缩框架，可将秩和精度作为可控的部署参数。T3C 将弹性张量分解（保持至最大秩）与秩绑定混合精度量化相结合，并采用轻量级控制器，将延迟/能耗/大小预算令牌映射为逐层秩/位宽分配；该策略会映射到硬件对齐配置，并随预算单调变化。快速的逐层一致性证书根据谱代理量和激活统计量计算，对 logit 漂移给出上界，并对训练进行正则化，从而以极低开销提供实用的可靠性信号。在 ImageNet-1k 上，T3C 推进了视觉帕累托前沿：对于残差网络-*50*，在准确率相同（下降 $\leq 0.5\%$）时，第50百分位延迟为**1.18 毫秒**，模型大小为**38MB**，优于 PTQ-8b（1.44 毫秒，88 MB）；对于*ViT-B/16*，T3C 的第50百分位延迟达到**2.30 毫秒**，模型大小为**59MB**，优于强劲的 PTQ/QAT 基线。因此，单个 T3C 检查点即可按需在不同设备上提供可预测且有证书支撑的准确率–延迟–模型大小权衡。

1. 引言

现代机器学习系统越来越多地运行于各种部署目标之上——包括在可变复用环境下共享的云端加速器（Yu & Chowdhury，2020年）、受热节流影响的边缘部署（Zhou 等人，2022年），以及电池供电平台，在这些平台上，动态电压/频率调节可能改变延迟/吞吐量行为（Han 等，2025年）。在这些场景中，有效的计算与内存预算并非静态不变；它们会随着共租（Yu & Chowdhury，2020年）、温度（Zhou 等人，2022年）和后台工作负载（Gujarati 等人，2020年）而波动。然而，实践中的大多数压缩流水线都是针对单一工作点编译的：在单一比特宽度下仅进行量化的导出模型（Jacob 等人，2017年；Esser 等人，2020年）、具有固定稀疏性的仅剪枝检查点（Han 等，2015a；Gale 等人，2019年），或秩固定不变的因子分解模型（Jaderberg 等人，2014年；Denton 等人，2014年；Lebedev 等人，2015年）。当运行时约束范围哪怕仅发生轻微变化时，这些静态导出模型也可能无法满足延迟服务级别目标、违反内存上限，或损失超出预期的准确率——迫使运维人员维护多个模型变体和临时路由逻辑，从而使 A/B 发布和值班操作手册变得更加复杂（Crankshaw 等人，2017年；Gujarati 等人，2020年）。

一种自然的做法是整理检查点菜单（例如，4/6/8 比特；30/50/80% 稀疏性；若干低秩截断），并在推理时在这些选项之间切换。然而，组合复杂度会迅速增长，不同设备之间的兼容性参差不齐，而且编译器/内核自动调优可能会对后端选择产生显著的敏感性（Chen等人，2018年；Zheng等人，2020年）。此外，每个检查点通常都需要单独训练或进行训练后校准（Nagel 等人，2020年；Li 等人，2021年），从而增加成本并减慢迭代速度。相比之下，实践者真正需要的是一个单一模型，能够在测试时被引导至目标延迟/大小/能耗点，并表现出可预测的准确率变化，这正是一次训练/多次专门化范式所追求的目标（Yu 等人，2019年；Yu & Huang，2019年；Cai 等人，2020年）。

以往的研究方向各自解决了这项需求的部分问题，但仍留下了关键缺口。训练后量化（PTQ）生成速度快（Nagel 等人，2020年；Li 等人，2021年），但校准条件发生变化时可能不够稳健（Yuan 等人，2023年）；对于大型模型，重尾激活异常值是已知的失效模式，因此需要专门的 PTQ 处理方法（Xiao 等人，2023年；Dettmers 等人，2022年）。量化感知训练（QAT）能够提升稳定性（Jacob 等人，2017年；Esser 等人，2020年），但仍会将导出结果锁定为固定精度。幅值剪枝和移动剪枝可以减少计算量（Han 等，2015a；Sanh 等人，2020年），但通常依赖稀疏内核，而实际加速比会因引擎或硬件而异，因此需要明确针对加速约束的方法（Frantar 和 Alistarh，2022年）。低秩方法能够以清晰的方式用准确率换取浮点运算次数（Jaderberg 等人，2014年；Denton 等人，2014年；Lebedev 等人，2015年），然而，秩的选择通常是离线决定的，且由此产生的近似误差可能与混合精度产生复杂的相互作用；因子分解的选择也出现在 Transformer 参数共享的设计中——

[1] 纳多尔多学科学院，穆罕默德·普雷米尔大学，乌季达 60000，摩洛哥
[2] 拉斯蒂实验室，恩萨克，苏丹穆莱·斯利曼大学，胡里卜盖 54000，摩洛哥。
通信作者：Ismail Lamaakal <ismail.lamaakal@ieee.org>。

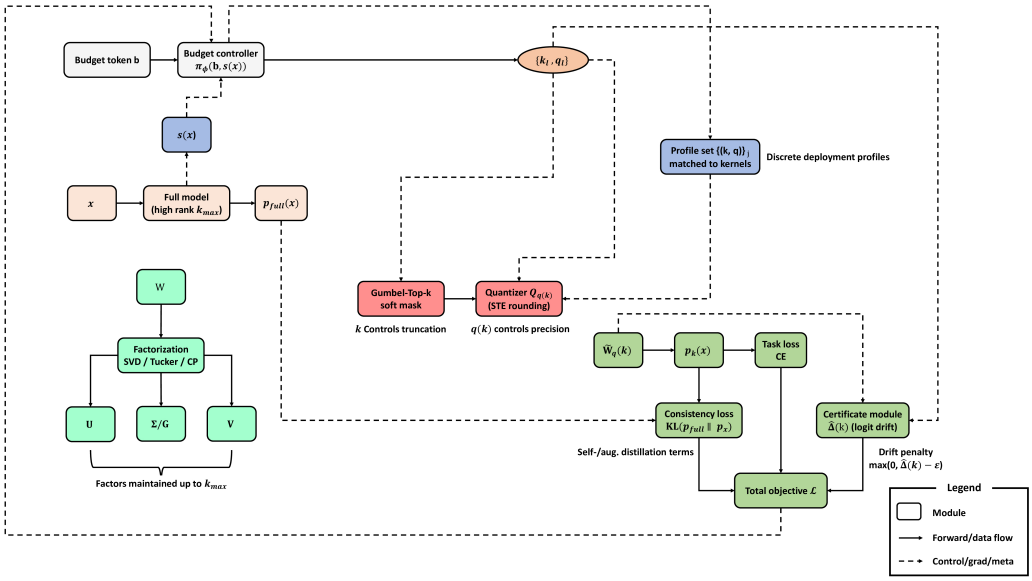

图 1. T3C流程详情（训练一次，测试时控制）。给定输入 x，在高秩 $k_{\max}$ 下运行的完整（教师）模型会生成教师逻辑值/分布 p_{full}（x）以及可选的输入摘要 s（x）。预算令牌 b（例如延迟/能耗/大小目标）与 s（x）一同输入预算控制器 π_ϕ（b，s（x））， 后者输出逐层秩和精度分配 $\{k_\ell, q_\ell\}$。与此同时，每一层的权重张量 W均以弹性分解（奇异值分解/Tucker分解/CP分解）的形式存储，分解最大维持至 $k_{\max}$。可微的 Gumbel-Top-k 软掩码会激活前 k_ℓ 个谱/张量分量（秩控制）；秩绑定的混合精度量化器 Q_{q_ℓ} 则采用 STE舍入进行 q_ℓ 位量化（位宽控制），以生成压缩权重 $\tilde{W}_q$（k）。重组后的算子生成压缩预测 p_k（x），训练时使用 (i) 任务损失（CE）和 (ii) 通过 KL（$p_{\text{full}} \| p_k$）进行的一致性/自蒸馏（也可在轻度数据增强下进行）。证书模块估计逻辑值漂移 $\hat{}\Delta$（k），并在总目标函数 L 中加入漂移惩罚项（例如 max(0, $\hat{}\Delta$（k） $-\epsilon$）。部署时，连续的（k_ℓ， q_ℓ）选择会被映射到与硬件高效内核匹配的离散配置集 $\{(k, q)\}_j$，从而实现可预测的运行时行为。实线箭头表示前向/数据流；虚线箭头表示控制/元信息或梯度路径。

中（Lan等人，2020年）。最后，“动态”网络（提前退出、自适应路由/宽度/深度）需要修改架构，而且可能更难集成到标准服务栈中（Teerapittayanon等人，2016年；Wang 等人，2018b；Graves，2016年；Hou等人，2020年）。

从运维视角来看，有三项要求。首先，部署需要**单调权衡**：收紧预算不应增加 实际延迟/大小，放宽预算也不应降低 准确率。其次，团队需要**尾部风险控制**：即使内核选择、缓存驻留和分配器抖动发生变化，p90/p99延迟和违约率也必须保持稳定（Gujarati 等人，2020年）。第三，任何解决方案都必须**设备感知**：成本模型必须将计算与内存流量结合起来，使同一制品无需针对每个目标重新训练，即可在计算受限服务器和内存受限边缘硬件之间通用（Williams等人，2009年；Chen等人，2018年）。这些要求意味着，接口在训练期间应当连续（以学习稳健行为），而在导出时应当离散（以适配硬件高效内核）；此外，还需要有原则地提供证书，概括测试时预算变化可能导致的准确率下降幅度。这一思路借鉴了能够给出明确且可验证的界的认证方法（Weng 等，2018；Cohen 等，2019）。

本文提出了 T3C，这是一种预算条件化、一次训练/测试时压缩框架，可将压缩从离线决策转变为在线控制。T3C 将弹性张量化（最高秩的 SVD/Tucker分解/CP分解）与逐因子的混合精度量化相结合，并使用轻量级策略将结构化预算令牌（延迟、字节数、可选能耗）映射为逐层秩/位分配。配置文件吸附器会将这些分配投影到由少量硬件对齐内核构成的格点上，从而确保执行稳定且快速；经过校准的一致性证书则汇总逐层残差范数和激活统计量，以给出 logit漂移的上界。我们的贡献包括：(1) 一种预算条件化参数化，在训练期间提供连续的秩–精度调节旋钮，同时可编译为离散的设备就绪配置；(2) 一种设备感知控制器，能够接收元组预算令牌，并优化混合计算量/字节数代理指标，以实现云端和边缘端之间的可移植性；(3) 一种快速证书，用于正则化训练并导出逐配置风险摘要；以及 (4) 一项广泛的实证研究，表明从单个检查点即可获得单调且硬件对齐的准确率–延迟–大小权衡，且相较于强量化、剪枝和

低秩基线，尾部风险有所降低。

本文其余部分的组织如下：第2节回顾相关工作并设定符号表示。第3节介绍弹性分解、可微截断/量化、预算控制器以及整体训练目标。第4节陈述证书，并讨论其计算与部署复杂度。第5节介绍实验设置与基线。第6节报告视觉和语言领域的结果，同时讨论局限性与更广泛的影响。第7节总结全文并展望未来方向。

2. 相关工作与预备知识

量化。训练后量化（PTQ）和量化感知训练（QAT）通过降低精度来缩小大小并改善延迟，其中 QAT 通常能提高低位宽下的稳定性（Jacob 等人，2017年；Esser 等人，2020年）。混合精度方法为每一层分配位宽，以在硬件约束下改善准确率–效率权衡（Dong 等人，2020年）。近期的训练后量化进展通过校准/重构以及更优的舍入决策提高了鲁棒性（Nagel 等人，2020年；Li 等人，2021年；Wei 等人，2022年），而无数据/蒸馏式方法则减少了对原始校准数据的依赖（Xu 等人，2020年；Li 等人，2020年）。尽管取得了进展，大多数流程仍会输出一个单一 固定精度检查点，在测试时提供的控制有限，也无法给出关于退化程度的可操作界（见附录A）。

剪枝与稀疏性。非结构化剪枝可以达到较高的稀疏性（Han 等，2015a；Sanh 等人，2020年），但实际加速比在很大程度上取决于稀疏内核的可用性和后端细节（Gale 等人，2019年）。相比之下，结构化稀疏模式（例如，N: M）旨在支持的硬件上实现更可靠的加速（Zhou 等人，2021年），近期研究则着眼于提供明确加速比保证的剪枝方法（Frantar 和 Alistarh，2022年）。通道剪枝／结构化剪枝能够带来可预测的延迟降低，但通常需要谨慎地重新训练或采用自动化调度方案（Liu 等人，2017年；He 等人，2018年）。动态推理和提前退出会根据输入调整计算量（Teerapittayanon等人，2016年；Wang 等人，2018b），但会使部署保证更为复杂。上述方法通常在导出时将工作点硬编码，而对输出漂移作出保证的情况仍不多见。

低秩方法与 LoRA 风格方法。矩阵/张量分解（例如低秩近似、Tucker/CP分解）通过以计算量/准确率换取秩来压缩层（Denton 等人，2014年；Jaderberg 等人，2014年；Lebedev 等人，2014年；Kim 等人，2015年）。LoRA 等低秩适配模块可减少微调所需的可训练参数（Hu 等人，2022年），后续方法则会调整或在训练过程中重新分配秩/预算（Zhang 等人，2023年；Valipour 等，2023年）。然而，在许多压缩流程中，秩仍通常被视为离线/导出时的选择，切换秩可能需要重新导出，并使内核假设失效；跨多个 k 的秩–位联合优化和端到端共享参数训练仍相对缺乏研究。

可认证压缩与鲁棒性界。敏感性分析以及基于 Lipschitz/松弛的方法所构建的证书，可以界定扰动下的输出变化（Weng 等，2018），其中也包括针对权重 扰动的证书（Weng 等，2020）。区间界传播为可认证训练和验证流程提供了可扩展的方案（Gowal 等，2018）。针对量化和高效部署，也有研究探讨了仅使用整数运算的认证鲁棒性（Lin 等，2021）。现有界可能过于保守或计算开销较大，因而限制了实际应用；此外，它们也很少将证书与可部署且能确保预算单调性的控制器联系起来。相比之下，我们提供了一种快速的基于校准的证书，并使其与可执行的预算策略相一致。

2.1. 预备知识

我们考虑一个包含若干层的前馈网络$\{\ell\ =1、\ldots\ldots、\ L\}$，其权重为 W_ℓ。给定输入 x，完整模型输出逻辑值 $z = f(x)$；在预算配置下，压缩模型输出 $\tilde{z} = f_k(x)$。

弹性分解。全连接层采用最高秩为 $k_{\max}$ 的奇异值分解，包含可训练因子（$U_\ell, \Sigma_\ell, V_\ell$）：

$$W_\ell \approx \underbrace{U_{\ell,k}\,\Sigma_{\ell,k}\,V_{\ell,k}^\top}_{\text{top-}k\text{ reconstruction}}, \qquad k \in [k_{\min}, k_{\max}]. \quad (1)$$

卷积/注意力张量采用 Tucker-2 或 CP 分解的对应形式；注意力头在各投影层间共享预算。可微的 top-k 掩码用于选择活动奇异值/因子。

混合精度量化。每个活动因子（或通道组）都使用与所选秩绑定的位宽 $q(k)$ 进行量化，采用直通估计器（STE）量化器 $Q_{q(k)}(\cdot)$：

$$\tilde{W}_\ell(k) \ = \ Q_{q(k)}\big(U_{\ell,k}\,\Sigma_{\ell,k}\,V_{\ell,k}^\top\big). \quad (2)$$

预算、成本与证书。控制器 π_ϕ 将预算令牌 b（例如延迟/能耗/大小目标）和可选摘要 $s(x)$ 映射为逐层 (k_ℓ, q_ℓ)。成本代理指标包括浮点运算次数和数据搬运字节数；测得的延迟/能耗用于配置文件选择。我们通过证书 $\Delta(k)$ 概述 logit 漂移，该证书由谱代理量和激活统计量计算得出（详情见第4节）；导出时，模型会附带离散的、硬件对齐的配置 $\{(k_\ell, q_\ell)\}_j$ 及其证书报告。其他符号表示和恒等式见附录 B。

3. 方法

3.1. 弹性重参数化

我们使用弹性的因式分解参数化来表示每个稠密权重矩阵 $W \in \mathbb{R}^{m\times n}$，该参数化支持可变的测试时秩。我们保留前 $k_{\max}$ 个奇异分量，并在谱上学习一个可微的前 k 掩码。对于选定的秩 $k \in \{k_{\min}, \dots, k_{\max}\}$，前向传播中使用的有效权重为

$$\tilde{W}(k) \;=\; U_{:,1:k}\left(\Sigma_{1:k,1:k} \odot M_k\right) V_{:,1:k}^{\top}, \tag{3}$$

其中，U，Σ，V是截至 $k_{\max}$的奇异值分解因子，而 $M_k \in [0,1]^{k\times k}$ 是一个（训练期间放宽的）对角掩码，用于选择活跃奇异值。为将混合精度与秩关联起来，我们使用依赖于秩的位数分配 $q(k)$对因子进行量化。量化后的前向算子为

$$\tilde{W}_q(k) \;=\; Q_{q(k)}\big(U_{:,1:k}\big)\,\Big(Q_{q(k)}\big(\Sigma_{1:k,1:k}\odot M_k\big)\Big)\,Q_{q(k)}\big(V_{:,1:k}\big)^{\top}, \tag{4}$$

其中，Q_q 采用逐张量均匀仿射量化，使用 q 位，并配有可学习的缩放因子/零点；梯度使用直通估计器（详情见附录B）。

对于卷积核，我们采用 Tucker-2（仅通道）或 CP分解。卷积核 $W \in \mathbb{R}^{C_{\text{out}}\times C_{\text{in}}\times h\times w}$分解为 $U_{\text{out}} \in \mathbb{R}^{C_{\text{out}}\times r_o}$、核心张量 $G \in \mathbb{R}^{r_o\times r_i\times h\times w}$和 $U_{\text{in}} \in \mathbb{R}^{C_{\text{in}}\times r_i}$。预算约束秩（$r_o$, r_i）由第3.3节中的控制器通过单调调度 $k \mapsto (r_o(k),\ r_i(k))$ 生成。在多头注意力中，我们在一个块内对$\{W_Q,\ W_K,\ W_V,\ W_O\}$ 共享一个预算，以避免头部不平衡；为保证编译简便性，各矩阵的秩按照固定比例从 k 中拆分。混合精度遵循相同规则：每个因子分配 $q(k)$位，也可选择使用逐因子偏移（例如，$q_U = q(k)+1$、$q_G = q(k)$、$q_V = q(k)+1$）。

3.2. 可微截断与量化

我们使用基于奇异值或张量因子的 Gumbel-Top-k掩码，放宽离散的 top-k 选择。每个谱元素 s_i 都会获得一个温度控制的逻辑斯蒂得分，从而生成软掩码 $\hat{}m_i \in [0,1]$；随着温度退火，该掩码会逐渐趋近于硬性保留/丢弃决策。在前向传播过程中，我们将 Σ（或核心张量切片）乘以 diag($\hat{}m_{1:k_{\max}}$)。量化采用带有可学习缩放因子的均匀分箱和直通舍入，使梯度能够通过 Q_q传播。我们在每次迭代中随机采样一个秩 $k \sim \pi_\phi(b)$（第 3.3 节），以随机方式训练模型覆盖整个范围$k \in [k_{\min}, k_{\max}]$，并避免训练部署不匹配。

3.3. 预算条件控制器

我们通过一个轻量级策略来控制逐层秩和位宽，该策略以预算令牌为条件，并可选择以紧凑的输入摘要为条件。令 $b \in \mathbb{R}^{d_b}$ 编码部署目标函数（例如，将目标延迟/能耗/大小作为指标或嵌入），并令 $s(x) \in \mathbb{R}^{d_s}$ 为一个低维统计量（例如池化激活）。共享的多层感知机会输出逐层秩/比特提议，并将其投影到一个小型硬件友好离散集合上。我们记作

$$\{k_\ell,\ q_\ell\}_{\ell=1}^{L} \;=\; \pi_\phi\big(b,\ s(x)\big). \tag{5}$$

训练采用可微松弛（软掩码与直通离散化），或采用策略梯度目标函数，将负训练损失加上预算满足奖励作为回报。为确保可部署性，$\{k_\ell,\ q_\ell\}$ 会快照到与可用内核匹配的校准配置集，并关于 b 单调递增，从而确保更大的预算不会产生更小的秩或更低的位数。我们采用课程学习：(i) 仅使用全局预算（无 $s(x)$），以确保稳定性；(ii) 使用更紧的预算；(iii) 可选地对部分层进行输入感知细化。

3.4. 训练目标

总损失结合了任务性能、完整视图与压缩视图之间的自蒸馏、增强一致性、用于限制预测 logit 漂移的证书惩罚项，以及可微预算成本：

$$\begin{aligned}\mathcal{L} = \underbrace{\mathrm{CE}\big(f_{\text{full}}(x), y\big)}_{\text{task}} + \lambda_{\text{SD}}\ \underbrace{\mathrm{KL}(p_{\text{full}}(x)\,\|\,p_k(x))}_{\text{self distill}} \\ + \lambda_{\text{AUG}}\ \underbrace{\mathbb{E}_{\tilde{x}}\,\mathrm{KL}(p_{\text{full}}(\tilde{x})\,\|\,p_k(\tilde{x}))}_{\text{aug. consistency}} \\ + \lambda_{\text{CERT}}\ \underbrace{\max\big(0,\ \hat{\Delta}(k)-\epsilon\big)}_{\text{drift cap}} + \lambda_{\text{BUD}}\ \underbrace{\mathrm{Cost}(k,b)}_{\text{budget}}.\end{aligned} \tag{6}$$

交叉熵项训练全容量视图（在 $k = k_{\max}$ 或高秩代理上评估）以解决该任务。自蒸馏项将压缩预测 p_k 与完整模型在相同输入上的分布 p_{full} 进行对齐，从而稳定不同秩下的准确率。数据增强一致性项在扰动输入$\tilde{x}$（例如弱增强）上重复这一对齐过程，以防止特定秩的过拟合。认证惩罚项使用快速的逐层谱范数代理来计算预测logit偏移 $\Delta(k)$；如果该偏移超过容差 ϵ，控制器就会向更安全的配置靠拢（第4节）。最后，预算项使用经过校准的代理量，根据采样配置对预期的计算、内存和延迟进行惩罚；该代理量结合了浮点运算次数、数据搬运字节数和设备特定延迟表。调度方案和代理量定义见附录B。

4. 一致性证书与部署复杂度

4.1. 界的陈述

我们提供了一个证书，用于对将每层权重 W_ℓ 替换为由我们的方法生成的压缩对应项 $\tilde{W}_\ell(k)$时逻辑值的变化给出上界（见图2）。令 $f(x)$表示完整模型，令 $f_k(x)$ 表示预算旋钮为 k时的压缩模型。令 $\delta z(x;\, k) = f_k(x) - f(x)$ 表示逻辑值差异。对于每一层 ℓ，定义谱范数代理 L_ℓ，用于界定从该层输出 h_ℓ 到逻辑值的子网络的局部利普希茨因子（通过带归一化的幂迭代进行估计）（更多详情见附录C）。令 $\Delta W_\ell(k) = W_\ell - W_\ell(k)$ 和 $a_{\ell-1}$ 表示层 ℓ的输入激活。则：

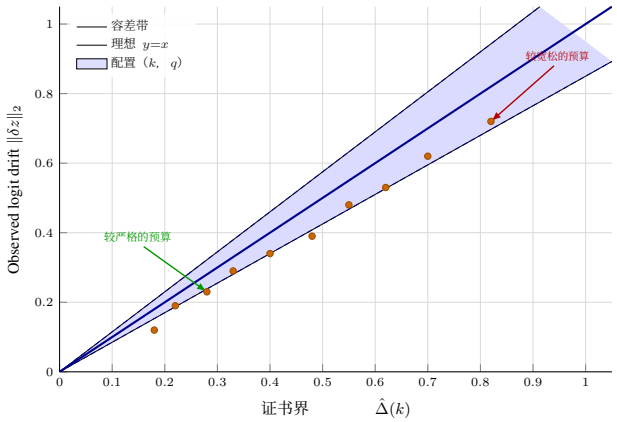

图2. 证书界 $\hat{\Delta}(k)$ 与离散配置（k，q）下观测到的 logit漂移 $\|\delta z\|_2$。阴影区域表示围绕 $y=x$ 的容差带；虚线表示其边界。箭头和标签以坐标轴为基准放置，并指向具有代表性的配置。

命题 4.1（逐层截断证书）。 对于具有标准残差块和归一化的 *ReLU/GELU* 网络，以及校准集 $\mathcal{C}$ 中的任意输入 x，*logit* 偏差满足

$$\left\|\delta z(x;k)\right\|_2 \;\le\; \sum_{\ell=1}^{L} \hat{L}_\ell \left\|\Delta W_\ell(k)\right\|_2 \left\|a_{\ell-1}(x)\right\|_2. \quad (7)$$

此外，利用校准统计量$\alpha_\ell = \sqrt{\mathbb{E}_{x\in\mathcal{C}}\|a_{\ell-1}(x)\|_2^2}$，期望 *logit* 偏差有如下界

$$\left(\mathbb{E}_{x\in\mathcal{C}}\left\|\delta z(x;k)\right\|_2^2\right)^{1/2} \;\le\; \sum_{\ell=1}^{L} \hat{L}_\ell \left\|\Delta W_\ell(k)\right\|_2 \alpha_\ell. \quad (8)$$

$\Delta W_\ell(k)$ 引起的前向扰动在第 ℓ 层至多为 $\|\Delta W_\ell(k)\|_2\|a_{\ell-1}\|_2$；将这一变化传递到$\hat{}$逻辑值时，最多会放大 L_ℓ。对各层的此类贡献求和，即可得到式 (7) 中的望远镜界。完整的技术条件和证明见附录 C。

4.2. 实用计算

$\hat{}$我们通过对雅可比代理执行少量（例如 3–5 次）幂迭代步骤，逐块计算 L_ℓ，并使用运行中的指数移动平均来稳定训练期间的估计值。截断范数 $\|\Delta W_\ell(k)\|_2$ 的获取方式有两种：对于稠密层，取掩码丢弃的最大奇异值；对于卷积层/注意力，则取 Tucker/CP 残差连接。我们通过一次幂迭代保留 $\Delta W_\ell(k)$ 的最大奇异值，以高效地界定谱范数。部署时，每种离散预算配置（k、q）都会附带一份证书报告，其中包含逐层元组（$\hat{}L_\ell$、$\|\Delta W_\ell(k)\|_2$、α_ℓ）以及聚合界 $\hat{}\Delta(k) \triangleq \sum_\ell \hat{}L_\ell\|\Delta W_\ell(k)\|_2\alpha_\ell$。我们会在模型卡中提供 ϵ 风格的摘要（例如，校准集上的第 95 百分位数）。

4.3. 紧致性与行为

当激活值呈重尾分布，或归一化层在局部增大 Lipschitz 常数时，界会变松；在实践中，权重归一化和逐块重缩放会降低 L_ℓ。依赖于数据的因子 α_ℓ 会收紧分布内输入的证书；遇到分布偏移时，我们则退回到保守的运行最大值。经验上，舍弃的奇异值能量较大的层会主导 $\Delta(k)$，这促使我们为前段卷积层和最终投影层分配更高的秩／精度。

4.4. 复杂度与部署

我们将成本建模为每层浮点运算次数、数据搬运字节数和内核启动开销的组合。控制器输出 $(k_\ell,\ q_\ell)$，并将其映射到一组小型配置集合；这些配置集合已在目标硬件上预先进行基准测试。每个配置都包含预测的延迟／能耗及其证书 $\Delta(k)$。导出产物包括所选配置的压缩权重、校准摘要 $\{\alpha_\ell\}$、谱代理量 $\{\hat{}L_\ell\}$，以及 JSON 证书台账。附录 D 提供了底层内核和张量内存布局选项，以及 TensorRT/ONNX 编译说明。

5. 实验

5.1. 设置

我们在多种模型家族和硬件目标值上，评估 T3C 在视觉和语言任务中的表现。

数据集。 *ImageNet-1k* (Deng 等人，2009年) (1.28M/50k,224×224, 单裁剪)，*CIFAR-100* (Krizhevsky，2009年)（附录 E 中可选），以及*GLUE* (Wang 等人，2018a)（MNLI-m/mm (Williams等人，2018年)、QQP、SST-2 (Socher，2013年)；宏平均分数）。小型语言模型评估报告*WikiText-103* (Merity 等人，2017年)上的困惑度。

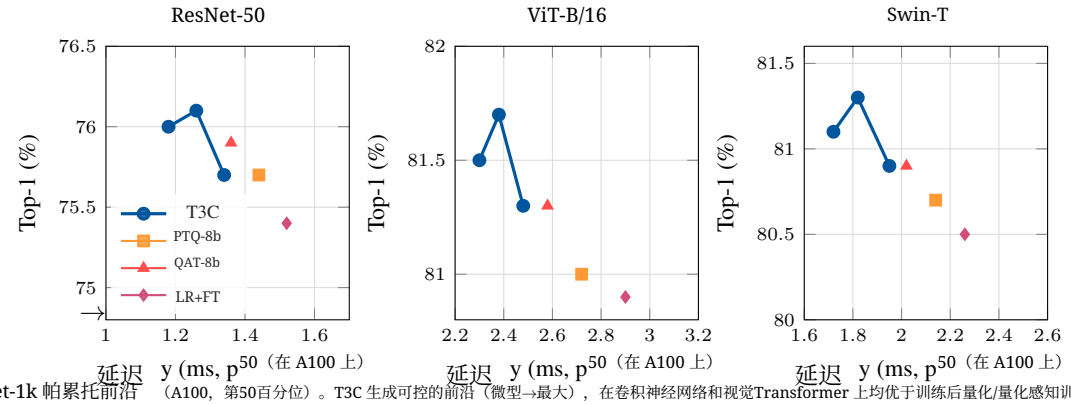

图3。 ImageNet-1k 帕累托前沿（A100，第50百分位）。T3C 生成可控的前沿（微型→最大），在卷积神经网络和视觉Transformer 上均优于训练后量化/量化感知训练和 LR+ SS 卷积神经网络和视觉Transformer。FT，覆盖

表1。 **视觉完整结果（ImageNet-1k）**。 准确率匹配（在完整模型的准确率上下浮动 0.5% 以内）和完整预算扫描。延迟为第50百分位50 [第90] 百分位，单位为毫秒。证书 ϵ 按照第4节中的方法计算。

模型系列	模型	方法/ 预算	Top-1 (%)	A100	Jetson	Android	大小 (MB)	ϵ
CNN	ResNet-50	T3C-Tiny	75.6	**1.18** [1。45]	**13.0** [16。5]	**22.4** [29。3]	**38**	**0.14**
		T3C-Med	**76.0**	**1.26** [1。58]	13.8 [17.3]	23.6 [31.0]	**42**	**0.12**
		T3C-Max	76.1	1.34 [1.62]	14.9 [18.2]	24.7 [32.1]	47	0.15
		PTQ-8b	75.7	1.44 [1.84]	18.5 [23.8]	29.2 [38.9]	88	0.22
		QAT-8b	**75.9**	1.36 [1.76]	17.4 [22.1]	27.6 [36.0]	90	0.19
	ResNet-101	T3C-Tiny	77.8	**1.64** [2。05]	**17.8** [22。6]	**30.8** [39。0]	**58**	**0.16**
		T3C-Med	**78.2**	**1.72** [2。12]	18.9 [24.0]	32.1 [40.6]	**63**	**0.14**
		T3C-Max	78.4	1.80 [2.22]	20.1 [25.4]	33.4 [42.0]	70	0.18
		LR+FT	77.6	1.98 [2.46]	23.8 [29.5]	38.9 [49.5]	98	0.28
		MagPrune	77.5	1.92 [2.40]	22.9 [28.3]	37.5 [48.1]	84	0.31
视觉 Tr ansformer	ViT-B/16	T3C-Tiny	81.2	**2.30** [2.92]	**26.0** [33.0]	**41.8** [53。9]	**59**	**0.18**
		T3C-Med	**81.5**	**2.38** [3.04]	26.5 [33.4]	42.6 [54.8]	**64**	**0.16**
		T3C-Max	81.7	2.45 [3.10]	27.8 [35.0]	44.0 [56.2]	71	0.19
		MP-QAT	**81.3**	2.58 [3.32]	33.0 [41.7]	52.9 [67.1]	134	0.25
		PTQ-8b	81.0	2.72 [3.48]	34.9 [44.1]	55.6 [70.5]	131	0.28
	ViT-L/16	T3C-Tiny	82.5	**4.10** [5。25]	**46.2** [58。0]	**74.9** [95。5]	**122**	**0.22**
		T3C-Med	**82.8**	**4.22** [5。39]	48.0 [60.2]	77.1 [97.7]	**131**	**0.20**
		T3C-Max	83.0	4.36 [5.55]	49.8 [62.0]	79.3 [100.1]	145	0.23
		LR+FT	82.3	4.78 [6.10]	57.6 [71.0]	91.9 [116.3]	212	0.34
		MovePrune	82.1	4.66 [5.98]	55.5 [68.3]	89.2 [112.0]	195	0.36
	Swin-T	T3C-Tiny	81.1	**1.72** [2。14]	**19.8** [25.2]	**31.6** [40.9]	**62**	**0.17**
		T3C-Med	**81.3**	**1.82** [2。26]	21.0 [26.6]	33.1 [42.7]	**66**	**0.16**
		T3C-Max	81.5	1.95 [2.38]	22.6 [28.4]	34.9 [45.0]	73	0.19
		MP-QAT	**81.2**	2.02 [2.46]	27.1 [33.2]	41.6 [53.2]	128	0.27
		PTQ-4b	80.7	2.14 [2.62]	29.4 [36.5]	45.1 [57.8]	96	0.39

模型。 卷积神经网络：残差网络-50/101 (He 等人，2016年)、ConvNeXt-T (Liu 等人，2022年)。视觉Transformer：ViT-B/16 和 ViT-L/16 (Dosovitskiy 等人，2021年)、Swin-T (Liu 等人，2021年)。自然语言处理Transformer：BERT-Base (Devlin 等人，2019年)、RoBERTa-Base (Liu 等人，2019年)、DistilBERT (Sanh 等人，2019年)。小型语言模型：TinyLlama-1.1B (Zhang 等人，2024年)（WikiText-103 困惑度 (Merity 等人，2017年)）。

预算。 我们为每个模型使用三种离散配置：**微型**（激进型）、**中等**（≤0.5%的下降目标）和**最大**（接近无损）。每种配置都与硬件对齐（第4节）。

基线。 PTQ-8b / PTQ-4b（Jacob 等人，2017年；Nagel 等人，2020年；Li 等人，2021年）；QAT-8b（Jacob 等人，2017年；Esser 等人，2020年）；混合精度 QAT（MP-QAT）（Wang 等人，2019年；Dong 等人，2020年）；幅值剪枝（Mag-Prune）（Han 等人，2015b）；移动剪枝（MovePrune）（Sanh 等人，2020年）；低秩 + 微调（LR+FT）（Jaderberg 等人，2014年；Denton 等人，2014年）；LoRA 压缩（LoRA-Comp）（Hu 等人，2022年）；SparseGPT（针对 TinyLlama）（Frantar 和 Alistarh，2023年）；KD-PTQ（蒸馏辅助量化）（Hinton 等人，2015年；Shin 等人，2019年；Choi 等人，2020年）。

指标。 准确率/F1 分数；语言模型的困惑度；在*A100*、*JetsonOrin*和*Androidbig.LITTLECPU*上的延迟第50百分位/第90百分位（批次=1）；通过平台计数器测得的能耗（边缘端）；大小（MB）；第4节中的证书 ϵ。完整的超参数、随机种子、数据处理和校准协议见附录 E。

表格 2 [90] 103 1 1 50 。语言完整结果。GLUE宏平均（ ϵ BERT、RoBERTa、DistilBERT）和 WikiText- PPL（TinyLlama- . B）。延迟以毫秒为单位。PPL 和 的值越低越好。

模型系列	模型	方法 /预算 GLUE/PPL		A100	Jetson	Android	大小（MB）	ϵ
编码器	BERT-Base	T3C-Tiny	81.8	**3.40** [4。35]	**41.0** [52。6]	**69.2** [88。1]	**92**	**0.20**
		T3C-Med	**82.2**	**3.18** [4。06]	43.1 [55.0]	72.4 [92.3]	**101**	**0.18**
		T3C-Max	82.4	3.26 [4.16]	44.3 [56.6]	74.0 [94.2]	112	0.21
		QAT-8b	**82.0**	3.70 [4.70]	55.8 [69.4]	92.0 [118.0]	338	0.29
	RoBERTa-Base	T3C-微型	83.9	**3.68** [4。64]	**43.9** [55。7]	**73.8** [94。0]	**108**	**0.19**
		T3C-Med	**84.2**	**3.82** [4.78]	45.6 [57.8]	76.5 [97.4]	**118**	**0.17**
		T3C-Max	84.3	3.95 [4.92]	47.3 [60.0]	78.4 [99.5]	130	0.20
		MP-QAT	**84.1**	4.26 [5.18]	57.0 [70.6]	96.2 [121.7]	372	0.27
	DistilBERT	T3C-Tiny	79.5	**2.36** [2.98]	**29.8** [38。0]	**49.9** [63.5]	**58**	**0.22**
		T3C-Med	**79.9**	**2.44** [3.06]	31.0 [39.6]	51.8 [65.8]	**63**	**0.20**
		T3C-Max	80.0	2.52 [3.16]	32.1 [41.0]	53.2 [67.7]	69	0.23
		PTQ-4b	**79.8**	2.66 [3.28]	36.4 [45.2]	59.8 [76.0]	94	0.33
解码器	TinyLlama-1.1B	T3C-Tiny	**困惑度 6.88**	**7.6** [9。8]	**92** [118]	**151** [198]	**210**	**0.26**
		T3C-Med	**困惑度 7.02**	7.9 [10.1]	96 [122]	156 [204]	225	**0.24**
		T3C-Max	困惑度 7.10	8.2 [10.5]	99 [127]	160 [210]	244	0.27
		KD-PTQ-8b	困惑度 7.04	8.8 [11.0]	121 [154]	189 [246]	418	0.33
		SparseGPT	困惑度 7.36	8.5 [10.8]	112 [144]	176 [228]	**238**	0.41
		LR+FT	困惑度 7.22	9.1 [11.6]	128 [162]	201 [258]	312	0.38
		PTQ-4b	困惑度 7.58	8.7 [11.2]	125 [159]	196 [252]	280	0.49
		MP-QAT	困惑度 7.12	9.3 [11.9]	133 [168]	209 [268]	436	0.36

6. 结果与讨论

帕累托曲线。 图3展示了 ResNet-50、ViT-B/16 和 S win-T 的准确率与 A100 第50百分位延迟之间的关系。T3C 描绘出平滑的权衡曲线（微型→最大），优于训练后量化/量化感知训练和 LR+FT。卷积神经网络与视觉 Transformer之间的斜率差异，反映了秩分配对早期卷积块与注意力投影的影响；T3C 的控制器会相应调整秩/比特预算。**准确率匹配**。 表1报告了准确率匹配时（与完整模型相差不超过 0.5%）的延迟/大小。T3C 始终能降低延迟、缩小模型占用，并报告更小（或相当）的证书 ϵ。

表3. **跨设备延迟（毫秒）**针对相同的检查点（p50 [p90]）。

模型	预算	A100	Jetson	Android CPU	移动端 NPU
ResNet-50	Tiny	**1.18** [1。45]	**13.0** [16。5]	**22.4** [29。3]	**9.2** [11。8]
	Med	1.26 [1.58]	13.8 [17.3]	23.6 [31.0]	10.0 [12.5]
	Max	1.34 [1.62]	14.9 [18.2]	24.7 [32.1]	10.7 [13.4]
ViT-B/16	Tiny	**2.30** [2。92]		**41.8** [53。9]	**18.6** [24。4]
	Med	2.38 [3.04]	26.5 [33.4]	42.6 [54.8]	19.3 [25.1]
	Max	2.45 [3.10]	27.8 [35.0]	44.0 [56.2]	20.1 [26.2]
BERT-Base	Tiny	**3.40** [4。35]	**41.0** [52。6]	**69.2** [88。1]	**28.8** [36.2]
	Med	3.18 [4.06]	43.1 [55.0]	72.4 [92.3]	30.2 [38.0]
	Max	3.26 [4.16]	44.3 [56.6]	74.0 [94.2]	31.1 [39.3]

6.1. 自然语言处理（GLUE）与小型语言模型

我们对仅编码器Transformer和一个小型仅解码器语言模型进行压缩。表2列出了不同预算和基线下的GLUE宏平均分数，以及延迟和大小（超过20行）。T3C在降低延迟和模型大小的同时，保持了接近基线的质量。在TinyLlama上，我们报告困惑度（PPL），发现T3C-Med以更低的 ϵ达到与KD-PTQ相当的表现；SparseGPT在大小方面具有竞争力，但困惑度较差。

6.2. 消融实验与行为

我们考察控制器的效用、仅秩优化与仅位数优化及联合优化，以及认证惩罚项。图4汇总了五个模型在固定延迟目标下的准确率下降。秩–位联合优化（T3C）始终表现最佳；移除认证惩罚项会增加预算违规（附录F报告了违约率以及逐层 k/q 直方图）。

6.3. 跨设备泛化

表3比较了 A100、Jetson、Android CPU 和神经网络处理器上相同检查点的 p50/p90 延迟。Mono-tone 预算行为得以保留，离散配置集可防止内核不匹配。

定性观察。(1) CNN 早期块和最终的 ˆ 投影层对证书质量 $\Delta(k)$ 的贡献最大；控制器会为它们分配更高的秩 / 位数。(2) 在 ViT 上，共享注意力预算可稳定注意力头并减少漂移。(3) 对边缘设备而言，在内存受限层（深度可分离层、MLP 投影层）采用混合精度带来的收益更大。

6.4. 讨论

我们的结果表明，所提出的预算条件化秩–比特弹性压缩，是一种实用方案，可交付一个单一检查点，使其在部署时能够适应不同的延迟、能耗或大小约束。在生产工作流中，当模型必须运行于多种硬件目标（A100、Jetson、移动端 CPU/NPU）上，或可用运行时预算会随情境（电池状态、热余量、并发任务）而波动时，该方法尤其有用。控制器提供可解释的标量或小型向量预算 b，可映射到部署策略（例如延迟目标）；同时，配置档位对齐确保与供应商内核兼容，并避免出现病态配置

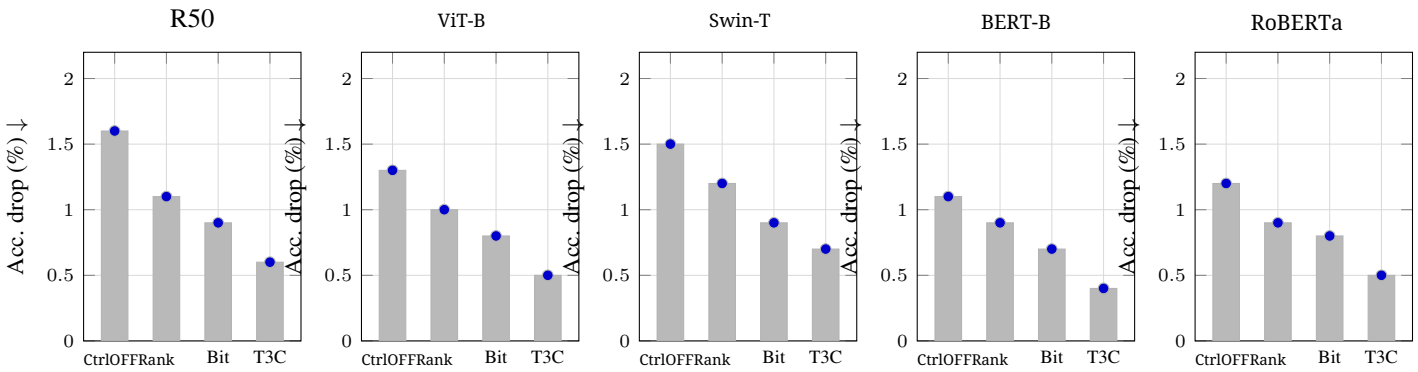

图*4*。 在固定延迟目标下的消融实验（越低越好）。与控制器联合进行秩–位学习（T3C）可在各类模型中持续降低准确率下降幅度。

。由于训练期间因子会一直保留到 $k_{\max}$，从业者可以在事后调整预算策略，而无需重新训练或重新导出模型。一致性项和证书项有助于在预算增加时维持单调性能，从而简化与调度器的集成：选择更大的预算绝不会导致秩或位宽变小，因此准确率不会更差。

将 T3C 集成到现有技术栈中十分简单。训练时的改动仅涉及参数化和损失函数；流水线的其余部分（数据、数据增强、日志记录）均保持不变。导出后会生成一个精简的制品，其中包含 (i) 逐层因子检查点，(ii) 离散配置集$\{(k_\ell, q_\ell)\}_j$ 及其对应的延迟/能耗表，以及 (iii) 证书账本。运行时集成只需一个小型策略模块，即可将应用层预算转换为配置索引。在使用 TensorRT/ONNX 或移动端推理引擎时，配置会与已实现的内核形状和量化方案相匹配；校准期间会剪除不受支持的点。我们发现，对于计算集中在少数几层的模型（例如前段卷积层、投影头），控制器尤为有效；在这些层之间重新分配秩和精度，能以较小的准确率代价换取显著的延迟降低。

6.5. 局限性

该证书依赖谱范数代理和激活统计数据；尽管这些指标与观测到的漂移高度相关，但它们较为保守，在高度非利普希茨块、无界激活分布或激进的归一化调度方案下可能会变得宽松。在这些情况下，该界可能会对紧张预算施加过重惩罚，导致分配欠佳，除非扩大校准集或采用逐块重新加权。控制器在微型数据集上或仅使用极端预算进行训练时可能不稳定；我们通过温度调度、预算课程学习以及可选的熵正则化来缓解这一问题，但它仍然会受到初始配置集合及其间距的影响。最后，部署取决于内核可用性：一些供应商仅支持特定的秩分解或位宽。尽管配置档位对齐基本上可以避开不兼容的点，但在某些神经网络处理器或数字信号处理器上，设计空间可能会受到人为限制，而且无法保证跨供应商的一致性。

7. 结论与未来工作

在本工作中，我们提出了**T3C**，一种一次训练、测试时预算条件化的压缩框架，将秩和精度作为明确的部署控制参数。该框架结合弹性张量分解、秩绑定混合精度量化，以及一个轻量级控制器；后者将预算令牌映射为逐层秩/位分配，并将其调整为一小组硬件对齐、随预算单调变化的配置文件。我们还提出了一种基于谱代理量和激活统计量的快速一致性证书，用于给出 logit 漂移的上界，从而支持训练期间的正则化以及部署时的风险感知报告。大量实验表明，单个 T3C 检查点能够在不同设备上实现可预测的准确率–延迟–模型大小权衡，并且在不同架构的 ImageNet-1k 测试中始终优于强劲的训练后量化/量化感知训练基线。未来工作将着重于在分布偏移和高度非利普希茨组件的情况下收紧并压力测试该证书，提升控制器在极端预算和低数据情形下的鲁棒性，并在供应商限制下使配置文件选择更加关注内核；此外，还将扩展控制平面，以纳入结构化稀疏性和多 SLO 优化（延迟/能耗/内存），并通过遥测驱动的自适应，同时保持预算单调性保证。

影响声明

本研究旨在推动深度模型的实用且负责任的部署，使单个训练完成的网络能够在测试时调整其计算占用（延迟、能耗和内存）

通过由预算条件控制且硬件对齐的压缩配置，降低资源消耗及相关环境成本，同时提升模型在边缘端和成本敏感型平台上的可及性。该方法还提供轻量级的一致性证书，用于估计不同配置下的输出漂移，从而在选择激进的压缩设置时支持更透明的风险评估，帮助从业者在安全或公平性相关应用中平衡效率与可靠性。潜在负面影响包括被滥用于大规模监控，或通过降低推理成本来扩大有害内容的传播；此外，如果评估不充分，激进的配置可能会导致代表性不足群体的性能受到不成比例的影响。因此，我们建议按配置报告指标，在分布偏移下进行压力测试，并在部署前针对不同人口群体和情境切片开展审计。总体而言，我们预计其主要影响将是积极的：降低实现高效推理的门槛，并提供有助于推动更安全、更负责任的压缩选择的工具。

参考文献

Benoit-Cattin, T., Velasco-Montero, D., 和 Fernández- Berni, J. 热节流对基于CPU的边缘设备中长期视觉推理的影响。*Electronics*，9(12)：2106，2020年。

Boroumand, A., Ghose, S., Akin, B., Narayanaswami, R., Oliveira, G. F., Ma, X., Shiu, E., 和 Mutlu, O. 面向边缘设备的Google神经网络模型：分析并缓解机器学习推理瓶颈。载于*2021*年第*30*届并行架构与编译技术国际会议（*PACT*），第159–172页。IEEE，2021年。

Cai, H.、Gan, C.、Wang, T.、Zhang, Z. 和 Han, S. Once-for-all：训练一个网络并针对高效部署进行专门化。在国际学习表征会议 *(ICLR)*，2020年。网址：https://arxiv.org/abs/1908.09791。Chen, T.、Moreau, T.、Jiang, Z.、Shen, H.、Yan, E.、Wang, L.、Hu, Y.、Ceze, L.、Guestrin, C. 和 Krishnamurthy, A. TVM：用于深度学习的自动化端到端优化编译器。在第 *13* 届 *USENIX* 操作系统设计与实现研讨会 *(OSDI 18)*，2018年。网址：https://arxiv.org/abs/1802.04799。Choi, J.、Wang, Z.、Venkataramani, S.、Chuang, P. I.-J.、Srini-vasan, V. 和 Gopalakrishnan, K. Pact：用于量化神经网络的参数化裁剪激活。*arXiv* 预印本 *arXiv:1805.06085*，2018年。Choi, Y.、Yu, J.、Wook Shin, Y.、Lee, J. 和 Kim, M. 通过对抗性知识蒸馏进行无数据网络量化。在 *IEEE/CVF*计算机视觉与模式识别会议研讨会 *(CVPRW)*论文集，2020年。网址：https://openaccess.thecvf.com/content_CVPRW_2020/papers/w40/Choi_Data-Free_Network_Quantization_With_Adversarial_Knowledge_Distillation_CVPRW_2020_paper.pdf。

Cohen, J., Rosenfeld, E., 和 Kolter, Z. 通过随机平滑实现可认证对抗鲁棒性。载于第*36*届国际机器学习会议（*ICML*）论文集，2019年。网址：https://proceedings.mlr. press/v97/cohen19c.html。

Crankshaw, D., Wang, X., Zhou, G., Franklin, M. J., Gonza- lez, J. E., 和 Stoica, I. Clipper：一种低延迟在线预测服务系统。载于第*14*届*USENIX*网络系统设计与实现研讨会（*NSDI 17*），2017年。网址：https://www.usenix.org/conference/nsdi17/technical-sessions/presentation/crankshaw。

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., 和 Fei-Fei, L. ImageNet：一个大规模分层图像数据库。载于*IEEE* 计算机视觉与模式识别会议论文集（*CVPR*），2009年。网址：https://www.image-net.org/static_files/papers/imagenet_cvpr09.pdf。

Denton, R., Zaremba, W., Bruna, J., LeCun, Y., 和 Fer- gus, R. 利用卷积网络中的线性结构实现高效评估。载于神经信息处理系统进展，2014年。网址：https://arxiv.org/abs/1404.0736。

Dettmers, T., Lewis, M., Belkada, Y., 和 Zettlemoyer, L. LLM.int8()：面向大规模 Transformer 的 8位矩阵乘法。载于神经信息处理系统进展，2022年。网址 https://arxiv.org/abs/2208.07339。Dettmers, T., Pagnoni, A., Holtzman, A., 和 Zettlemoyer, L. Qlora：量化大语言模型的高效微调。神经信息处理系统进展，36:10088–10115，2023年。Devlin, J., Chang, M.-W., Lee, K., 和 Toutanova, K. BERT：用于语言理解的深度双向变换器预训练。载于*2019*年北美计算语言学协会分会会议论文集：人类语言技术（*NAACL-HLT*），2019年。网址 https: // aclanthology.org/N19-1423.pdf。Dong, Z., Yao, Z., Gholami, A., Mahoney, M. W., 和 Keutzer, K. Hawq：神经网络的 Hessian 感知量化

网络的混合精度。载于 *IEEE/CVF*国际计算机视觉会议论文集，第 293–302 页，2019年。

Dong, Z., Yao, Z., Arfeen, D., Gholami, A., Mahoney, M. W., and Keutzer, K. Hawq-v2：神经网络的海森感知迹加权量化。发表于神经信息处理系统进展（*NeurIPS*），2020年。网址：https://proceedings.neurips.cc/paper/2020/hash/d77c703536718b95308130ff2e5cf9ee-Abstract.html。

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. 一张图像胜过16×16个词：面向大规模图像识别的Transformer。发表于国际学习表征会议（*ICLR*），2021年。网址：https://arxiv.org/abs/2010.11929。Esser, S. K., McKinstry, J. L., Bablani, D., Appuswamy, R., and Modha, D. S. 学习步长量化。发表于国际学习表征会议（*ICLR*），2020年。网址：https://openreview.net/forum?id=rkgO66VKDS。Evci, U., Gale, T., Menick, J., Castro, P. S., and Elsen, E. 改造彩票：让所有彩票票据都成为赢家。发表于国际机器学习会议，第2943–2952页。PMLR，2020年。Fedus, W., Zoph, B., and Shazeer, N. Switch Transformer：以简单高效的稀疏性扩展至万亿参数模型。机器学习研究期刊，23(120)：1–39，2022年。Frankle, J. and Carbin, M. 彩票票据假说：寻找稀疏可训练神经网络。*arXiv*预印本 *arXiv:1803.03635*，2018年。Frantar, E. and Alistarh, D. SPDY：以速度提升保证实现准确剪枝。发表于第*39*届国际机器学习会议（*ICML*）论文集，2022年。网址：https://arxiv.org/abs/2201.13096。Frantar, E. and Alistarh, D. SparseGPT：大型语言模型可通过单次操作实现准确剪枝。*arXiv*预印本 *arXiv:2301.00774*，2023年。网址：https://arxiv.org/abs/2301.00774。Frantar, E., Ashkboos, S., Hoefler, T., and Alistarh, D. GPTQ：面向生成式预训练Transformer的准确训练后量化。*arXiv*预印本 *arXiv:2210.17323*，2022年。

Gale, T., Elsen, E., and Hooker, S. 深度神经网络中的稀疏性现状。*arXiv* 预印本 *arXiv:1902.09574*，2019年。网址：https://arxiv.org/abs/1902.09574。Gowal, S., Dvijotham, K., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Arandjelovic, R., Mann, T., and Kohli, P. 区间界传播在训练可验证鲁棒模型中的有效性。*arXiv* 预印本 *arXiv:1810.12715*，2018年。网址：https://arxiv.org/abs/1810.12715。Graves, A. 循环act神经网络的自适应计算时间。*arXiv* 预印本 *arXiv:1603.08983*，2016年。网址：https://arxiv.org/abs/1603.08983。Gu, J., Chowdhury, M., Shin, K. G., Zhu, Y., Jeon, M., Qian, J., Liu, H., and Guo, C. Tiresias：用于分布式深度学习的 GPU 集群管理器。载于第 *16* 届 *USENIX* 网络系统设计与实现研讨会（*NSDI 19*）。USENIX 协会，2019年。Gujarati, A., Karimi, R., Alzayat, S., Hao, W., Kaufmann, A., Vigfusson, Y., and Mace, J. 像发条机制一样服务深度神经网络：自底向上的性能可预测性。载于第 *14* 届 *USENIX* 操作系统设计与实现研讨会（*OSDI 20*），2020年。网址：https://www.usenix.org/conference/osdi20/presentation/gujarati。Han, S., Mao, H., and Dally, W. J. 深度压缩：通过剪枝、训练后量化和霍夫曼编码压缩深度神经网络。*arXiv* 预印本 *arXiv:1510.00149*，2015a。网址：https://arxiv.org/abs/1510.00149。Han, S., Pool, J., Tran, J., and Dally, W. 高效神经网络的权重与连接联合学习。载于神经信息处理系统进展（*NeurIPS*），2015b。网址：https://arxiv.org/abs/1506.02626。Han, Y., Nan, Z., Zhou, S., and Niu, Z. GPU 上的 DVFS 感知深度神经网络推理：延迟建模与性能分析。*arXiv* 预印本 *arXiv:2502.06295*，2025年。网址：https://arxiv.org/abs/2502.06295。He, K., Zhang, X., Ren, S., and Sun, J. 用于图像识别的深度残差学习。载于*IEEE* 计算机视觉与模式识别会议（*CVPR*）论文集，2016年。网址：https://openaccess.thecvf.com/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf。He, Y., Lin, J., Liu, Z., Wang, H., Li, L.-J., and Han, S. AMC：用于模型压缩的自动化机器学习

以及移动设备上的加速。载于欧洲计算机视觉会议（*ECCV*），2018年。网址：https://openaccess.thecvf.com/ content_ECCV_2018/papers/Yihui_He_AMC_AutoML_for_ECCV_2018_paper.pdf。

Hinton, G., Vinyals, O., 和 Dean, J. 神经网络中的知识蒸馏。*arXiv*预印本 *arXiv:1503.02531*，2015年。网址：https://arxiv. org/abs/1503.02531。

Hou, L., Huang, Z., Shang, L., Jiang, X., Chen, X., 和 Liu, Q. DynaBERT：具有自适应宽度和深度的动态BERT。收录于神经信息处理系统进展，2020年。网址：https://arxiv.org/abs/ 2004.04037。

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., 和 Chen, W. LoRA：大语言模型的低秩适配。收录于国际学习表征会议（*ICLR*），2022年。网址：https: //openreview.net/forum?id=nZeVKeeFYf9。

Huang, G., Chen, D., Li, T., Wu, F., Van Der Maaten, L. 和 Weinberger, K. Q. 用于资源高效图像分类的多尺度密集网络。*arXiv* 预印本 *arXiv:1703.09844*，2017年。

Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H. 和 Kalenichenko, D. 面向高效纯整数运算推理的神经网络量化与训练。*arXiv* 预印本 *arXiv:1712.05877*，2017年。网址：https://arxiv.org/abs/1712.05877。

Jaderberg, M., Vedaldi, A. 和 Zisserman, A. 通过低秩展开加速卷积神经网络。*arXiv* 预印本 *arXiv:1405.3866*，2014年。网址：https://arxiv.org/abs/1405.3866。

Kim, Y.-D., Park, E., Yoo, S., Choi, T., Yang, L., 和 Shin, D. 面向快速、低功耗移动应用的深度卷积神经网络压缩。*arXiv* 预印本 *arXiv:1511.06530*，2015年。网址：https: //arxiv.org/abs/1511.06530。

Krizhevsky, A. 从微型图像中学习多层特征。技术报告，多伦多大学，2009年。网址：https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf。

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Shar ma, P., andSoricut, R. ALBERT：用于自监督学习语言表示的轻量级 BERT。发表于国际学习表征会议 *(ICLR)*，2020年。网址：https://arxiv.org/abs/1909.11942。

Lebedev, V., Ganin, Y., Rakhuba, M., Oseledets, I., and Lempitsky, V. 使用微调 CP 分解加速卷积神经网络。*arXiv* 预印本 *arXiv:1412.6553*，2014年。网址：https://arxiv.org/ abs/1412.6553。Lebedev, V., Ganin, Y., Rakhuba, M., Oseledets, I., and Lempitsky, V. 使用微调 CP 分解加速卷积神经网络。发表于国际学习表征会议 *(ICLR)*，2015年。网址：https://arxiv.org/abs/1412.6553。

Lee, N., Ajanthan, T., 和 Torr, P. H. SNIP：基于连接敏感度的单次网络剪枝。*arXiv* 预印本 *arXiv:1810.02340*，2018年。

Li, B., Huang, K., Chen, S., Xiong, D., Jiang, H., 和 Claesen, L. DFQF：无数据量化感知微调。发表于第 *12* 届亚洲机器学习会议（*ACML*）论文集，2020年。网址：https:// proceedings.mlr.press/v129/li20a.html。

Li, Y., Gong, R., Tan, X., Yang, Y., Hu, P., Zhang, Q., Yu, F., Wang, W., 和 Gu, S. BRECQ：通过块重建突破训练后量化的极限。载于 国际学习表征会议（*ICLR*），2021年。网址：https://arxiv.org/abs/2102.05426。

Lin, C., Wang, K., Li, Z., 和 Pu, Y. 一种适用于移动设备、能够应对并发任务且具备鲁棒性的工作负载感知动态电压频率调节方法。载于第*29*届移动计算与网络国际年会论文集，第1–16页，2023年。

Lin, H., Lou, J., Xiong, L., 和 Shahabi, C. 面向量化神经网络的纯整数运算认证鲁棒性。载于 *IEEE/CVF*计算机视觉国际会议（*ICCV*）论文集，2021年。网址：https://openaccess.thecvf.com/content/ICCV2021/papers/Lin_Integer-Arithmetic-Only_Certified_Robustness_for_Quantized_Neural_Networks_ICCV_2021_paper.pdf。

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. RoBERTa：一种稳健优化的 BERT 预训练方法。*arXiv* 预印本 *arXiv:1907.11692*，2019年。URL https://arxiv.org/abs/1907.11692。

Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S., and Zhang, C. 通过网络瘦身学习高效卷积网络。载于 *IEEE* 国际计算机视觉会议论文集（*ICCV*），2017年。网址 https: //openaccess.thecvf.com/content_ICCV_2017/papers/Liu_Learning_Efficient_Convolutional_ICCV_2017_paper.pdf。

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin Transformer：采用移位窗口的分层视觉Transformer。载于*IEEE/CVF* 国际计算机视觉会议论文集（*ICCV*），2021年。网址 https://openaccess.thecvf.com/content/ICCV2021/papers/ Liu Swin Transformer Hierarchical_ _ _ _Vision_ Transformer_Using_Shifted_Windows_ICCV_2021_ paper.pdf。

Liu, Z.、Mao, H.、Wu, C.-Y.、Feichtenhofer, C.、Darrell, T. 和 Xie, S. 面向2020年代的卷积网络。在*IEEE/CVF* 计算机视觉与模式识别会议（*CVPR*）论文集中，2022年。URL https: // openaccess.thecvf.com/content/ CVPR2022/papers/Liu_A_ConvNet_for_the_2020s_CVPR_2022_paper.pdf。Lou, Q.、Guo, F.、Liu, L.、Kim, M. 和 Jiang, L. AutoQ：自动化逐核神经网络量化。*arXiv* 预印本 *arXiv:1902.05690*，2019年。Louizos, C.、Welling, M. 和 Kingma, D. P. 通过 *l*_0正则化学习稀疏神经网络。*arXiv* 预印本 *arXiv:1712.01312*，2017年。Luo, J.、Wu, J. 和 Lin, W. ThiNet：一种用于深度神经网络压缩的滤波器级剪枝方法。在*IEEE* 国际计算机视觉会议（*ICCV*）论文集中，2017年。Merity, S.、Xiong, C.、Bradbury, J. 和 Socher, R. 指针哨兵混合模型。在国际学习表征会议（*ICLR*）中，2017年。URL https: // arxiv.org/abs/1609.07843。Micikevicius, P.、Stosic, D.、Burgess, N.、Cornea, M.、Dubey, P.、Grisenthwaite, R.、Ha, S.、Heinecke, A.、Judd, P.、Kamalu, J. 等。用于深度学习的 FP8 格式。*arXiv* 预印本 *arXiv:2209.05433*，2022年。Nagel, M.、Baalen, M. v.、Blankevoort, T. 和 Welling, M. 通过权重均衡与偏差校正实现无数据量化。在*IEEE/CVF* 国际计算机视觉会议论文集中，第 1325–1334 页，2019年。Nagel, M.、van Baalen, M.、Blankevoort, T. 和 Welling, M. 向上还是向下？用于训练后量化的自适应舍入。在第 *37* 届机器学习国际会议（*ICML*）论文集中，2020年。URL https://proceedings.mlr.press/ v119/nagel20a.html。Novikov, A.、Podoprikhin, D.、Osokin, A. 和 Vetrov, D. P. 张量化神经网络。神经信息处理系统进展，第 28 卷，2015年。

NVIDIA。TensorRT：NVIDIA 推理优化器和运行时。https://developer.nvidia.com/tensorrt，2017年。ONNX 社区。ONNX：开放神经网络交换格式。https://onnx.ai/，2017年。Parashar, A., Rhu, M., Mukkara, A., Puglielli, A., Venkatesan, R., Khailany, B., Emer, J., Keckler, S. W., and Dally, W. J. SCNN：一种用于压缩稀疏卷积神经网络的加速器。*ACM SIGARCH* 计算机体系结构新闻，45(2):27–40，2017年。Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. 探索使用统一的文本到文本转换器进行迁移学习的极限。机器学习研究期刊，21 (140):1–67，2020年。Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., and Hsieh, C.-J. DynamicViT：通过动态令牌稀疏化实现高效视觉变换器。神经信息处理系统进展，34:13937–13949，2021年。Reddi, V. J., Cheng, C., Kanter, D., Mattson, P., Schmuelling, G., Wu, C.-J., Anderson, B., Breughe, M., Charlebois, M., Chou, W., 等。MLPerf 推理基准测试。载于*2020*年 *ACM/IEEE* 第*47*届计算机体系结构国际年会（*ISCA*），第446–459页。IEEE，2020年。Sanh, V., Debut, L., Chaumond, J., and Wolf, T. DistilBERT：BERT 的蒸馏版本，更小、更快、更便宜、更轻量。*arXiv* 预印本 *arXiv:1910.01108*，2019年。网址：https://arxiv.org/abs/1910.01108。Sanh, V., Wolf, T., and Rush, A. M. 移动剪枝：通过微调实现自适应稀疏性。*arXiv* 预印本 *arXiv:2005.07683*，2020年。网址：https://arxiv.org/abs/2005.07683。Shin, S., Boo, Y., and Sung, W. 用于优化量化深度神经网络的知识蒸馏。*arXiv* 预印本 *arXiv:1909.01688*，2019年。网址：https://arxiv.org/abs/1909.01688。Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. 在情感树库上实现语义组合性的递归深度模型。载于*2013*年自然语言处理实证方法会议（*EMNLP*）论文集，2013年。网址：https://aclanthology.org/D13-1170/。Teerapittayanon, S., McDanel, B., and Kung, H. T. BranchyNet：通过深度神经网络提前退出实现快速推理。载于*2016*年第*23*届国际模式识别会议（*ICPR*），2016年。网址：https://arxiv.org/abs/1709.01686。

Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., 等。MLP-Mixer：一种用于视觉的全 MLP 架构。神经信息处理系统进展，34:24261–24272，2021年。

Valipour, M., Rezagholizadeh, M., Kobyzev, I., 和 Gh- odsi, A. DyLoRA：使用动态免搜索低秩适配对预训练模型进行高效参数调优。载于欧洲计算语言学协会第 *17* 届会议论文集（*EACL*），2023年。网址：https://aclanthology. org/2023.eacl-main.239/。

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., 和 Bow- man, S. R. GLUE：用于自然语言理解的多任务基准和分析平台。载于*2018*年*EMNLP BlackboxNLP*研讨会论文集，2018a。网址：https://aclanthology.org/W18-5446. pdf。

Wang, K., Liu, Z., Lin, Y., Lin, J., 和 Han, S. HAQ：采用混合精度的硬件感知自动量化。载于*IEEE/CVF* 计算机视觉与模式识别会议论文集（*CVPR*），2019年。网址：https: //openaccess.thecvf.com/content_CVPR_2019/papers/Wang_HAQ_Hardware-Aware_Automated_Quantization_With_Mixed_Precision_CVPR_2019_paper.pdf。

Wang, S., Li, B. Z., Khabsa, M., Fang, H., 和 Ma, H. Linformer：具有线性复杂度的自注意力。*arXiv* 预印本 *arXiv:2006.04768*，2020年。

Wang, X., Yu, F., Dou, Z.-Y., Darrell, T., 和 Gonzalez, J. E. Skipnet：在卷积网络中学习动态路由。载于欧洲计算机视觉会议（*ECCV*）论文集，2018b。URL https: //openaccess.thecvf.com/content_ECCV_2018/html/Xin_Wang_SkipNet_Learning_Dynamic_ECCV_2018_paper.html。

Wei, X., Gong, R., Liu, Y., Yu, F., Wang, W., Huang, J., Zhang, Q., 和 Gu, S. Qdrop：用于极低比特后训练量化的随机量化丢弃。*arXiv* 预印本 *arXiv:2203.05740*，2022年。URL https://arxiv.org/abs/2203.05740。

Weng, T.-W., Zhang, H., Chen, H., Song, Z., Hsieh, C.- J., Boning, D., Dhillon, I. S., and Daniel, L. 迈向 ReLU 网络认证鲁棒性的快速计算。发表于第 *35* 届国际机器学习大会（*ICML*）论文集，2018年。网址：https://arxiv.org/abs/1804.09699。

Weng, T.-W., Zhao, P., Liu, S., Chen, P.-Y., Lin, X., and Daniel, L. 迈向针对权重扰动的认证模型鲁棒性。发表于 *AAAI* 人工智能大会（*AAAI*）论文集，2020年。网址：https://ojs.aaai.org/index.php/AAAI/article/view/6105/5961。

Williams, A., Nangia, N. 和 Bowman, S. R. 用于通过推理进行句子理解的广覆盖挑战语料库。载于*2018*年北美计算语言学协会分会会议论文集：人类语言技术（*NAACL-HLT*），2018年。网址：https://aclanthology.org/N18-1101/。

Williams, S., Waterman, A. 和 Patterson, D. Roofline：一种针对多核架构的深刻直观的性能模型。《*ACM* 通讯》，52(4):65–76，2009年。doi：10.1145/1498765.1498785。

Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J. 和 Han, S. SmoothQuant：面向大语言模型的准确且高效的训练后量化。载于第*40*届国际机器学习会议（*ICML*）论文集，2023年。网址 https://proceedings.mlr.press/v202/xiao23c.html。

Xiao, W., Bhardwaj, R., Ramjee, R., Sivathanu, M., Kwatra, N., Han, Z., Patel, P., Peng, X., Zhao, H., Zhang, Q., Yang, F., 和 Zhou, L. Gandiva：面向深度学习的内省式集群调度。载于第 *13* 届 *USENIX* 操作系统设计与实现研讨会（*OSDI 18*），第 595–610 页。USENIX 协会，2018年。

Xu, S., Li, H., Zhuang, B., Liu, J., Cao, J., Liang, C., 和 Tan, M. 生成式低比特宽度数据无关量化。载于欧洲计算机视觉会议（*ECCV*），2020年。网址 https://www.ecva.net/papers/eccv_2020/papers_ECCV/papers/123570001.pdf。

Yu, J. 和 Huang, T. 通用可伸缩网络与改进的训练技术。载于*IEEE/CVF* 国际计算机视觉会议论文集（*ICCV*），2019年。网址 https: //openaccess.thecvf.com/content_ICCV_2019/papers/Yu_Universally_Slimmable_Networks_and_Improved_Training_Techniques_ICCV_2019_paper.pdf。

Yu, J., Yang, L., Xu, N., Yang, J., 和 Huang, T. 可伸缩神经网络。载于国际学习表征会议（*ICLR*），2019年。网址：https:// openreview.net/forum?id=H1gMCsAqY7。

Yu, P. 和 Chowdhury, M. 深度学习应用的细粒度 GPU 共享原语。载于机器学习与系统会议论文集，第 2 卷，

2020年。网址：https://proceedings.mlsys. org/paper_files/paper/2020/hash/d9cd83bc91b8c36a0c7c0fcca59228f2-Abstract. html。

Yuan, Z.、Liu, J.、Wu, J.、Yang, D.、Wu, Q.、Sun, G.、Liu, W.、Wang, X. 和 Wu, B. 训练后量化可靠性的基准测试。*arXiv* 预印本 *arXiv:2303.13003*，2023年。URL https://arxiv. org/abs/2303.13003。

Zhang, H.、Weng, T.-W.、Chen, P.-Y.、Hsieh, C.-J. 和 Daniel, L. 使用通用激活函数实现高效的神经网络鲁棒性认证。神经信息处理系统进展，31，2018年。

Zhang, P.、Zeng, G.、Wang, T. 和 Lu, W. TinyLlama：开源小型语言模型。*arXiv* 预印本 *arXiv:2401.02385*，2024年。URL https://arxiv. org/abs/2401.02385。

Zhang, Q., Chen, M., Bukharin, A., Karampatziakis, N., He, P., Cheng, Y., Chen, W., 和 Zhao, T. AdaLoRA：面向参数高效微调的自适应预算分配。*arXiv* 预印本 *arXiv:2303.10512*，2023年。URL：https://arxiv.org/abs/2303.10512。

Zheng, L., Jia, C., Sun, M., Wu, Z., Yu, C. H., Haj-Ali, A., Wang, Y., Yang, J., Zhuo, D., Sen, K., Gonzalez, J. E., 和 Stoica, I. Ansor：为深度学习生成高性能张量程序。载于 第*14*届 *USENIX* 操作系统设计与实现研讨会（*OSDI 20*），2020年。URL https://arxiv.org/abs/2006. 06762。Zhou, A., Ma, Y., Zhu, J., Liu, J., Zhang, Z., Zhang, L., Zhu, W., 和 Huang, G. 从头开始学习 n:m 细粒度结构化稀疏神经网络。载于 国际学习表征会议（*ICLR*），2021年。URL https://openreview.net/forum? id= K9bw7vqp_s。Zhou, S. 等人。Dorefa-net：使用低比特宽度梯度训练低比特宽度卷积神经网络，2016年。

Zhou, Y., Liang, F., Chin, T.-W., 和 Marculescu, D. 保持冷静：动态移位可防止热降频。*arXiv* 预印本 *arXiv:2206.10849*，2022年。URL https://arxiv.org/abs/2206.10849。

A. 扩展背景与相关工作

A.1 扩展的动机与生态系统需求

现代机器学习部署所处的硬件、延迟目标和可靠性约束环境错综复杂，且每小时、每位用户、每个应用都可能发生变化；同一模型可能这一分钟运行在多路复用云服务中，下一分钟运行在热降频边缘加速器上，最后又运行在受电池限制、以内存带宽而非浮点运算次数为主要瓶颈的移动设备 CPU 上。（Xiao 等人，2018年；Gu 等人，2019年；Benoit-Cattin 等人，2020年；Lin 等人，2023年；Boroumand 等人，2021年）在这类场景中，主要痛点源于导出产物的僵化、运行时预算的不确定性以及压缩时缺乏可靠的防护机制。

首先，经典流程会针对每个工作点生成一个压缩检查点（例如，为服务器准备一个 8 位量化感知训练模型，为移动设备准备一个 4 位训练后量化变体，为神经网络处理器准备一个剪枝变体）。这会增加存储需求，使 A/B 测试和回滚更加复杂，导致监控支离破碎，并迫使系统采用脆弱的路由逻辑，而这些逻辑容易受到设备特性差异和内核可用性的影响。其次，真实世界中的预算具有随机性：图形处理器上的共租和调度器噪声、边缘设备上的温度和动态电压频率调节、交互式工作负载中用户可见的帧截止时间，以及会随其他应用而变化的设备端内存压力。因此，运维人员需要在推理时具备弹性：一种无需重新训练或重新导出、也不违反硬件约束（内核形状、量化格式、对齐）的调节手段，使模型能够沿着准确率—延迟—体积前沿移动。

第三，生产环境可靠性日益要求失效模式可预测，尤其是在压缩以准确率换取效率时。服务级别协议和安全审查不仅会问“速度有多快”和“准确率有多高”，还会问“减少 20% 的计算量时，输出可能偏移多大？”目前的答案依赖经验，且往往很脆弱；形式化鲁棒性研究很少能转化为可实际操作的部署控制参数，而许多高性能压缩方法提供的保证也有限。生态系统的发展趋势进一步加剧了这些压力：模型家族规模更大、异质性更强（卷积神经网络、视觉Transformer、MLP-Mixer、编码器/解码器语言模型），(Dosovitskiy 等人, 2021年; Tolstikhin 等人, 2021年; Raffel 等人, 2020年) 加速器支持多种数据类型（ INT8 、 INT4 、 FP8 、混合逐通道方案） (Micikevicius 等人 , 2022 年) 以及高度依赖特定供应商的稀疏和低秩内核，(Parashar 等人, 2017年) 隐私和延迟要求也推动更多推理任务转移到边缘端。

运营人员需要一个单一、可移植的检查点，能够在测试时进行调控，以便在不熟悉的设备上达到预算目标，同时具备单调行为（更大的预算绝不会损害准确率）以及可量化风险的遥测接口。从工具链的角度来看，该制品应能与常见的导出/编译步骤（ONNX/TensorRT、移动端运行时）集成（ONNX 社区，2017年；NVIDIA，2017年），并能贴合离散的硬件配置以避免内核间隙，还应提供轻量级证书，为允许的压缩所引入的偏移设定界限。最后，评估本身也必须与时俱进：异构设备帕累托曲线、p50/p90 延迟、单次推理能耗、包含激活缓冲区在内的内存占用，以及预算达标率（请求的配置达到其延迟/能耗目标的频率），与单一数值的准确率同样重要。（Reddi 等人，2020年）简而言之，生态系统需要自适应、可认证、硬件感知且易于运维的压缩技术，将压缩从离线工程分支转变为与生产服务级目标保持一致的在线控制平面。

A.2 详细综述（量化、剪枝、低秩、动态网络、验证）

量化。 训练后量化（PTQ）会基于留出集校准缩放因子／零点，因速度快、简单而颇具吸引力；但在超低精度或分布偏移情况下，其准确率可能下降。缓解措施包括逐通道缩放因子、激活裁剪、偏差校正，以及将高方差通道单独设为更高精度的离群值感知方案。量化感知训练（QAT）以基于直通估计器的替代函数取代硬舍入，从而学习缩放／裁剪参数，通常能提升鲁棒性，但代价是需要额外训练，且导出时位宽固定。混合精度方法将位分配建模为搜索或可微松弛（利用延迟代理指标提供奖励／梯度），以便在各层之间分配位数；当所选混合方案无法与供应商内核顺畅匹配时，部署就会遇到阻碍，因此需要进行性能分析剪枝。近期的训练后量化／量化感知训练混合方法加入了知识蒸馏和跨层均衡；它们推动了帕累托前沿，但在导出时仍会固定 精度，也无法对输出漂移给出明确的界。（Choi 等，2018；Nagel 等，2019；Zhou 等，2016；Dong 等，2019；Lou 等，2019；Frantar 等，2022）

剪枝与稀疏性。非结构化剪枝能够实现较高的参数稀疏度和较小的磁盘占用，但要真正获得加速比，需要使用细粒度稀疏内核，并承担不可忽视的开销；结构化剪枝或通道剪枝通过缩减整个滤波器／投影层，可稳定降低延迟，但往往需要迭代重新训练和精心设计调度方案，以避免准确率骤降。基于移动和幅值的准则、L0／L1 松弛以及彩票假说式回溯共同构成了这一设计空间；动态稀疏性增强了输入自适应性，但也使编译和可预测性更加复杂。（Han 等，2015a；Luo 等，2017；

Lee 等人，2018年；Frankle 和 Carbin，2018年；Louizos 等人，2017年；Evci 等人，2020年）

低秩与张量分解。 SVD/Tucker/CP 分解可减少密集层和卷积层中的参数及内存流量；在 Transformer 中，应用于投影矩阵和多层感知机模块的分解会以准确率为代价换取较低的秩。LoRA 风格适配器会在微调期间注入低秩更新，而低秩替换 则会压缩基础权重；大多数方法会离线固定秩，并针对每种秩重新导出模型，这与运行时自适应性和内核可用性相冲突。联合秩与精度分配的研究相对不足：量化会与秩相互作用，因为截断会改变动态范围和奇异值谱，进而影响尺度选择和误差传播。（Novikov 等人，2015年；Wang 等人，2020年；Dettmers 等人，2023年）

动态网络。 提前退出、令牌/通道丢弃和输入条件路由可根据难度调整计算量，从而降低平均延迟。然而，为避免出现异常退出，这些方法需要架构钩子、重新训练和谨慎校准；其保证通常是统计性的（预期成本），而非针对每个预算的 确定性保证。此外，动态策略可能与批处理、缓存或编译器融合发生对抗性相互作用，导致尾延迟不稳定。（Huang 等人，2017年；Rao 等人，2021年；Fedus 等人，2022年）

验证与证书。 鲁棒性证书为输入或参数扰动下的输出变化设定上界（例如 Lipschitz 界、区间/zonotope 界、随机平滑）。参数扰动保证与压缩相关，可为量化噪声、剪枝或低秩截断的影响设定上界，但许多方法过于保守，或难以在大规模场景下计算，而且很少有方法被纳入部署时控制环。数学上便于处理的假设与生产环境中的实际细节（如残差连接、归一化、融合内核和校准漂移）之间存在差距，阻碍了这些方法的实际应用。（Zhang 等人，2018年；Cohen 等，2019年；Weng 等，2018年）

定位。 在此背景下，实用的方法应当：1）只训练一次，但提供连续的 控制，并在推理时将其映射到离散的硬件配置；2）联合考量秩和精度，因为二者的误差会叠加，成本也与内存带宽和内核形状相互关联；3）提供快速的逐层证书，为所选配置下的 logit 漂移设定上界，并可将其呈现给监控系统；4）保持预算单调性，使运维人员能够提高目标值，而不会面临准确率回退的风险。现有研究提供了许多强大的技术：固定比特下精度很高的训练后量化/量化感知训练、有效的剪枝计划、性能强劲的低秩适配器，以及严谨但计算开销大的证书；然而，它们很少将这些技术整合成弹性、可认证且具备硬件意识的流水线，也很少采用生产服务级目标的表述方式。T3C 旨在弥合这一差距：通过在不同秩之间共享参数，以可部署的方式将位宽与秩关联，并附加一种计算成本足够低的证书，使其适用于训练时正则化和导出时报告，从而将压缩转变为一种运维控制手段，而非离线分支。

B. 方法详情

B.1 符号表示与张量分解（奇异值分解/Tucker分解/CP分解变体）

基本符号表示。 我们考虑一个前馈网络，其层的索引为 $\ell \in \{1、\ldots\ldots、 L\}$。对于稠密（全连接）层 ℓ，令 $W_\ell \in \mathbb{R}^{m_\ell \times n_\ell}$ 将 $a_{\ell-1} \in \mathbb{R}^{n_\ell}$ 映射到 $h_\ell = W_\ell a_{\ell-1}$ （为简洁起见省略偏差）；激活值可能包含归一化和非线性函数 $h_\ell \mapsto a_\ell$。对于卷积核， $W_\ell \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times h \times w}$ 使用与基础模型相同的步幅/填充。给定输入 x，完整模型输出逻辑值 $z = f(x)$；在预算配置 k 下，若逐层秩/比特为 $(k_\ell,\ q_\ell)$，~则将压缩模型记为 $f_k(x)$，其逻辑值为 $\tilde{z}$。

（矩阵）奇异值分解。 对于 $W \in \mathbb{R}^{m \times n}$，秩为 r 的截断奇异值分解为 $W_r = U_{:,\ 1:r} \Sigma_{1:r} V_{:,\ 1:r}^\top$ 其奇异值为$\sigma_1 \geq \cdots \geq \sigma_r$。我们保留截至 $k_{\max}$ 的因子，并提供可微的前 $k \leq k_{\max}$ 项选择（见 B.2 节）。我们采用两种数值稳定的参数化：(i) 谱形式 $U,\ \Sigma,\ V$通过基于 Householder更新的隐式重新正交化来施加正交约束；(ii) 乘积形式 $AB^\top$, $A \in \mathbb{R}^{m \times k_{\max}}$, $B \in \mathbb{R}^{n \times k_{\max}}$，并通过谱惩罚项促使 A 和 B 逼近奇异方向。谱形式可以直接控制残差范数$\|W - W_k\|_2 = \sigma_{k+1}$ 和 $\|W - W_k\|_F = (\sum_{i>k} \sigma_i^2)^{1/2}$，有助于生成证书。

用于卷积的 Tucker-2。 对于 $W \in \mathbb{R}^{C_o \times C_i \times h \times w}$，我们采用仅沿通道的Tucker-2分解：

$$W \approx (U_o, U_i) \cdot G, \quad U_o \in \mathbb{R}^{C_o \times r_o},\ U_i \in \mathbb{R}^{C_i \times r_i},\ G \in \mathbb{R}^{r_o \times r_i \times h \times w}. \tag{9}$$

实际的前向计算为 Conv(U_o Conv(G, $U_i^\top \cdot$))，它会编译为三个内核（逐点 1×1、空间 $h\times w$、逐点 1×1）。预算 k 通过单调调度映射为 ($r_o(k)$ $r_i(k)$)。

深度可分离层/注意力块的 CP分解。 对于可分离性很强的类深度可分离层权重或 MLP投影权重，秩为 r 的 CP分解 $W \approx \sum_{j=1}^{r} a_j^{(1)} \otimes a_j^{(2)}$ 可减少内存流量；在注意力中，我们可以选择在 $\{W_Q, W_K, W_V, W_O\}$ 之间共享单个 k，以避免头部不平衡。

可辨识性与条件性。 Tucker分解和 CP分解在缩放/置换意义下并不唯一。我们通过对每个因子进行 ℓ_2 归一化，并采用置换固定规则（按因子范数降序排列）来消除退化。为防止秩较小时出现病态条件，我们通过 $\sum_i \max(0, \sigma_i - \sigma_{i+1} - \delta)$ 对谱进行正则化，以防间隙塌缩（较小的 δ）。

B.2 损失项、退火调度， λ扫描

总目标函数（展开式）。 对于一个小批量$\mathcal{B}$ 和采样得到的预算配置 k，损失函数为

$$\mathcal{L} = \underbrace{\frac{1}{|\mathcal{B}|}\sum_{(x,y)\in\mathcal{B}} \mathrm{CE}(f_{\text{full}}(x), y)}_{\text{task (full)}} + \lambda_{\text{SD}} \underbrace{\frac{1}{|\mathcal{B}|}\sum_{x\in\mathcal{B}} \mathrm{KL}\big(p_{\text{full}}(x) \| p_k(x)\big)}_{\text{self-distill}}$$
$$+ \lambda_{\text{AUG}} \underbrace{\mathbb{E}_{\tilde{x}\sim\mathcal{T}(x)} \mathrm{KL}\big(p_{\text{full}}(\tilde{x}) \| p_k(\tilde{x})\big)}_{\text{augmentation consistency}} + \lambda_{\text{CERT}} \underbrace{\max\big(0,\, \hat{\Delta}(k) - \epsilon\big)}_{\text{certificate penalty}} + \lambda_{\text{BUD}} \underbrace{\mathrm{Cost}(k; b)}_{\text{budget proxy}}, \tag{10}$$

采用 $p(\cdot) = \mathrm{softmax}(z/T)$（温度为 $T=1$，除非另有说明）。任务（完整）项用于训练高秩视图（$k = k_{\max}$或高秩代理），以锚定教师质量。两个 KL 项将压缩预测与完整模型在原始输入和轻度增强输入上的输出对齐。证书惩罚项使用第4节（正文）中的界以及运行估计。预算代理指标是相对于目标值的预期归一化延迟/能耗/大小。

退火调度方案。 *Top-k* 温度。对于 Gumbel-Top-k 掩码 logit g_i，我们使用温度 $\tau_t = \max(\tau_{\min}, \tau_0 \cdot \gamma^{t/T_{\text{anneal}}})$，其中 $\gamma \in$ 介于 0,1 之间，经过 T_{anneal} 步退火后保持不变。秩采样。课程学习初期：从 $[k_{\min}, k_{\max}]$ 上的宽 Beta分布(α=0.75, β=0.75) 中采样 k，然后通过混合离散配置上的分类分布 $\{k^{(j)}\}_j$，使采样偏向部署相关配置。系数递增。我们在训练最初的 10–20% 阶段，对 $\lambda_{\text{SD}}, \lambda_{\text{AUG}}, \lambda_{\text{CERT}}$ 使用线性预热，以避免过早过度正则化。

λ 扫描与稳定性。我们建议采用由粗到细扫描：先固定 λ_{BUD}，使其在验证设备上达到目标利用率（延迟/能耗）；然后搜索 λ_{CERT}，以限制违约率（即满足 $\Delta(k) > \epsilon$ 的样本比例）；最后调整 λ_{SD}和 λ_{AUG}，以便在预算紧张时恢复准确率。实践中，一个不错的初始范围是

$$\lambda_{\text{SD}} \in [0.3, 1.0], \quad \lambda_{\text{AUG}} \in [0.1, 0.5], \quad \lambda_{\text{CERT}} \in [0.05, 0.5], \quad \lambda_{\text{BUD}} \in [0.1, 1.0]. \tag{11}$$

我们监控：(i) 单调性（准确率应随预算增加而不下降），(ii) % 预算违规率，以及 (iii) $\Delta(k)$ 与观测漂移之间的校准误差。

B.3 量化器定义、校准、舍入技巧

均匀仿射量化。对于实数张量 T， q 位对称逐张量量化使用缩放因子 $s > 0$ 和整数网格 $\mathcal{G}_q = \{-2^{q-1}+1, \ldots, 2^{q-1}-1\}$:

$$\mathrm{Quantize}_q(T) \;=\; s\,\mathrm{clip}\left(\left\lfloor \frac{T}{s} \right\rceil,\; \min \mathcal{G}_q,\; \max \mathcal{G}_q\right), \qquad s = \frac{\max(|T|)}{2^{q-1}-1}. \tag{12}$$

对于沿输出通道（行）的矩阵，以及沿 C_{out}的卷积，我们还使用逐通道缩放因子 s_c。

直通估计器（STE）。在反向传播中，我们将 $\partial\lfloor u \rceil / \partial u \approx 1$ 视为在裁剪范围内取值、范围外为 0。对于缩放 s，我们学习 s（对数参数化），其梯度为 $\partial \text{量化}_q(T)/\partial s \approx$ ~~$(\lfloor T/s \rceil - T/s)$~~（在范围内），以促使缩放与信号范围相匹配。

舍入变体。随机舍入 $\lfloor u \rceil_{\text{stoch}} = \lfloor u \rfloor$ 以概率 $1 - (u - \lfloor u \rfloor)$ 取值，否则取 $\lceil u \rceil$，在训练时使用以减少低 q时的偏差；为保证确定性，导出时禁用。偏差校正在校准期间向反量化张量添加折叠偏移量 $\delta = \mathbb{E}[T - \text{量化}_q(T)]$（逐通道），随后将其融合到偏置参数中。离群值感知拆分会将一小部分 ρ 幅值最大的通道分配到 $q+\Delta q$ 位数；当硬件支持在层内使用混合精度时可启用此功能。我们将其作为配置选项提供，以便控制器为离群值组分配更高的位数。

校准。 我们通过校准集 $\mathcal{C}$ 采用以下任一方法计算 s（对于非对称量化，还可选计算零点）：**(i)最大范围**（如上所述），或 **(ii) 百分位裁剪** $s = \text{百分位}_p(|T|)/(2^{q-1} - 1)$，其中 $p \in [99.0, 99.99]$ 经过调优，以最小化该层浮点输出与量化输出之间的验证集 KL 散度。训练期间，我们维护 s 的指数移动平均统计量，以稳定部署尺度。

秩绑定位分配。 我们对每个因子组采用单调映射 $q(k) = \min\{q_{\max}, \lfloor a \log\, k + b \rfloor\}$，并使用较小的逐因子偏移（例如 $q_U = q(k)+1$, $q_G = q(k)$, $q_V = q(k)+1$），确保较高秩至少分配到同样多的位数。这与谱衰减相得益彰：较小的秩会截断更多能量，因此从额外精度中获益较少。

引理（随机舍入下的期望反量化误差）。设 X 为满足 $|X| \le M$ 且缩放因子固定为s的标量。在网格 $s\mathcal{G}_q$ 上进行随机舍入且不进行裁剪时，反量化误差 $\varepsilon = \text{Quantize}_q(X) - X$ 满足 $\mathbb{E}[\varepsilon \mid X] = 0$，且 $\text{Var}(\varepsilon \mid X) \le s^2/4$。证明。在给定 $u = X/s$ 的条件下，舍入以概率 $1 - (u - \lfloor u \rfloor)$ 取 $\lfloor u \rfloor$，否则取 $\lceil u \rceil$；其期望等于 u。乘以 s 即可得到无偏性；步长为 s的伯努利变量的条件方差上界为 $s^2/4$。□

B.4 控制器架构与训练（松弛、基线）

预算词元化。预算 b 编码目标延迟/能耗/大小，并可选地编码设备 ID。我们将 b 嵌入为 $e_b = \text{Embed}_{\text{dev}}(\text{id}) \oplus \phi(\text{延迟、能耗、大小})$，其中 ϕ 是作用于归一化标量的可学习多层感知机；$\oplus$ 表示拼接。

策略头。 我们使用一个共享的两层多层感知机 π_ϕ，输出逐层离散配置的逻辑值，并可选择输出一个投影到这些配置上的连续提议：

$$\hat{k}_\ell, \hat{q}_\ell = \text{Proj}(W_2\, \sigma(W_1[e_b \oplus s(x)])), \qquad \text{Proj} : \mathbb{R}^2 \to \mathcal{P}_\ell \subset \{(k, q)\}. \tag{13}$$

这里，$s(x)$ 是可选的输入摘要（例如，在批次上取平均的池化倒数第二层激活）。$\mathcal{P}_\ell$ 是由硬件对齐的 (k, q) 对组成的层特定菜单。

离散松弛。 我们对 $\mathcal{P}_\ell$ 使用温度为 τ 的 Gumbel-Softmax 和直通技巧，使前向传播得到独热（配置索引），反向传播得到软分布。这样既保留了端到端可微性，也能在训练控制器的同时联合训练因子和量化器。

控制器的训练目标。 控制器通过总损失（公式 10）隐式训练；梯度会通过可微松弛以及预算/证书惩罚反向传播。当仅使用离散选择（不进行松弛）时，我们会添加一个 REINFORCE 项：

$$\nabla_\phi \mathbb{E}_{\pi_\phi}\big[-\mathcal{L}\big] \approx \mathbb{E}\big[(-\mathcal{L} - b)\, \nabla_\phi \log \pi_\phi\big], \tag{14}$$

并使用学习基线 b（价值头）来降低方差。实际应用中，Gumbel-Softmax +直通技巧已足够；退火后切换为纯 argmax。

单调性约束。 我们通过保序约束确保更高的预算不会产生更小的秩/位数：对于两个预算令牌 $b_1 \prec b_2$（逐分量），我们添加铰链损失

$$\lambda_{\text{ISO}} \sum_\ell \max\big(0,\, k_\ell(b_1) - k_\ell(b_2)\big) + \max\big(0,\, q_\ell(b_1) - q_\ell(b_2)\big). \tag{15}$$

At e导出时，我们剔除任何违反约束的配置，并重新校准剩余配置集以保持偏序 .

基线（用于消融实验）。**(i) 仅秩**：q 每层固定；**(ii) 仅位数**：固定秩；**(iii) 贪心背包**：根据实测延迟表，按收益成本比的顺序将k/q 分配给各层；**(iv) 均匀**：各层使用相同的 k, q。这些方案展示了联合学习和控制器分配策略的价值。

B.5 各层类型的算法复杂度与内存分析

令 b_w 表示权重位宽，b_a 表示推理时的激活位宽（通常为 8），并用 $\mathcal{B}(\cdot)$ 表示字节占用。我们分别统计计算浮点运算次数和内存字节数（权重 +激活的读写操作），因为在许多边缘设备上，后者占主导地位。

全连接层 $W \in \mathbb{R}^{m\times n}$。 完整 GEMV 浮点运算次数（批次为 1）和权重字节数：

$$\mathrm{FLOPs}_{\text{full}} = 2mn, \qquad \mathcal{B}_W = mn \cdot \frac{b_w}{8}. \tag{16}$$

秩-k *SVD* 路径： 按 $(U_{m\times k}\Sigma_{k\times k}V^{\top}_{n\times k})a$ 计算，通过 $(V^{\top}a) \in \mathbb{R}^k$，乘以 Σ，然后 U：

$$\mathrm{FLOPs}_{\mathrm{SVD}}(k) \approx 2nk + k + 2mk \;\approx\; 2(n+m)k. \tag{17}$$

权重字节数（每个因子；混合精度时将 b_w 替换为 $q(k)$）：

$$\mathcal{B}_U = mk \cdot \frac{b_w}{8}, \qquad \mathcal{B}_V = nk \cdot \frac{b_w}{8}, \qquad \mathcal{B}_\Sigma = k \cdot \frac{b_w}{8}. \tag{18}$$

卷积（Tucker-2）。设特征图的空间大小为 $H \times W$。完整 卷积浮点运算次数：

$$\mathrm{FLOPs}_{\text{full}} = 2\,C_o C_i hw\,HW. \tag{19}$$

Tucker-2，即将 →沿 →（空间）维度压缩 1×1，再通过 1×1扩展：

$$\mathrm{FLOPs}_{\mathrm{Tucker2}}(r_o, r_i) \;=\; 2HW\Big(C_i r_i \;+\; r_o r_i hw \;+\; C_o r_o\Big). \tag{20}$$

秩绑定 $q = q\,(k)$ 时的权重字节数：

$$\mathcal{B}_{U_o} = C_o r_o \cdot \frac{q}{8}, \quad \mathcal{B}_{U_i} = C_i r_i \cdot \frac{q}{8}, \quad \mathcal{B}_G = r_o r_i hw \cdot \frac{q}{8}. \tag{21}$$

注意力投影层和多层感知机。 当 $d_{\text{model}}{=}d$ 且多层感知机隐藏层维度为 d_{ff}时，四个投影层 $W_Q,\ W_K,\ W_V,\ W_O \in \mathbb{R}^{d\times d}$ 和 $W_1 \in \mathbb{R}^{d_{\text{ff}}\times d},\ W_2 \in \mathbb{R}^{d\times d_{\text{ff}}}$。对一个 $d \times d$ 矩阵乘法应用秩为k 的奇异值分解，得到浮点运算次数的大致缩放关系：

$$\frac{\mathrm{FLOPs}_{\text{rank-}k}}{\mathrm{FLOPs}_{\text{full}}} \;\simeq\; \frac{2dk}{2d^2} \;=\; \frac{k}{d}. \tag{22}$$

延迟/能耗代理指标（按设备校准）。

$$\widehat{\mathrm{Lat}}(k) \;=\; \alpha_0 \;+\; \sum_{\ell=1}^{L}\Big(\alpha_\ell^{\mathrm{comp}} \cdot \mathrm{FLOPs}_\ell(k) \;+\; \alpha_\ell^{\mathrm{mem}} \cdot \mathrm{Bytes}_\ell(k)\Big), \tag{23}$$

$\widehat{\ }$，其系数通过在配置网格上进行最小二乘拟合得到。类似地，Energy（k）使用其自身的系数。

证书聚合成本。 每层维护：(i) $\hat{}\,L_\ell$，通过 s 次幂迭代得到；(ii) 残差谱范数 $\|\Delta W_\ell\ (k)\ \|_2$，通过 1–2 次幂迭代得到；(iii) 激活均方根 α_ℓ，通过指数移动平均得到。聚合如下：

$$\hat{\Delta}(k) \;=\; \sum_{\ell=1}^{L} \hat{L}_\ell\ \big\|\Delta W_\ell(k)\big\|_2\ \alpha_\ell. \tag{24}$$

秩-k 何时胜出（阈值分析）？ 何时类 GEMV 推理能从奇异值分解中获益

$$2(n+m)k \;\ll\; 2mn \quad\Longrightarrow\quad k \;\ll\; \frac{mn}{m+n}. \tag{25}$$

对于方阵 $m{=}n{=}d$:

$$k \;\ll\; \frac{d}{2} \quad \text{(compute-only; bandwidth effects tighten this bound).} \tag{26}$$

对于卷积 Tucker-2，要求

$$C_i r_i \;+\; r_o r_i hw \;+\; C_o r_o \;\ll\; C_o C_i hw. \tag{27}$$

设置 $r_o = \rho C_o, r_i = \rho C_i$ 得到

$$\rho(C_i + C_o) \;+\; \rho^2\, C_o C_i hw \;\ll\; C_o C_i hw \quad\Rightarrow\quad \rho \ll 1 \;\text{ and }\; \rho \lesssim \frac{1}{\sqrt{hw}}. \tag{28}$$

内存摘要。秩-k 奇异值分解（字节数）：

$$\mathcal{B}_W^{\mathrm{SVD}}(k) \;\approx\; (mk + nk + k)\cdot \frac{q(k)}{8}. \tag{29}$$

Tucker-2（字节数）：

$$\mathcal{B}_W^{\mathrm{T2}}(r_o, r_i) \;\approx\; \Big(C_o r_o + C_i r_i + r_o r_i hw\Big)\cdot \frac{q(k)}{8}. \tag{30}$$

命题（预算 $\Rightarrow$ 单调递减的代理成本）。若每层的配置集合 P_ℓ 是全序的，使得 $(k', q') \succeq (k, q)$ 蕴含 浮点运算次数$_\ell(k', q') \geq$ 浮点运算次数$_\ell(k, q)$，且字节数$_\ell(k', q') \geq$ 字节数$_\ell(k, q)$，并且控制器对所有 ℓ 强制执行$b_1 \prec b_2 \Rightarrow (k_\ell, q_\ell)(b_1) \preceq (k_\ell, q_\ell)(b_2)$，那么

$$\widehat{\mathrm{Lat}}(b_1) \;\leq\; \widehat{\mathrm{Lat}}(b_2). \tag{31}$$

$\widehat{\ }$证明概要。Lat 对非负系数的线性性质和逐层单调性，在对各层求和时保持偏序。□

T3C 相较于固定压缩的计算开销。训练会增加：(i) 因子化前向传播，(ii) 直通估计器量化运算，(iii) 间歇性地对每个块执行少量 PI 步骤。实测表明，

$$\text{Overhead}_{\text{train}} \;\approx\; (1.1 \text{ to } 1.4) \times \text{QAT wall-clock}, \tag{32}$$

由于无需针对每个工作点重新训练，因此开销可在所有秩/预算之间摊销。

C. 证书与证明

C.1 形式化假设（Lipschitz 代理、归一化）

我们对网络进行形式化描述，并阐明证书所需的温和正则性条件。

网络与　　扰动模型。设 $f\colon \mathbb{R}^{d_0} \to \mathbb{R}^{d_L}$ 是一个由 L 个模块组成的前馈网络。每个模块　　k

$$h_\ell \;=\; \Phi_\ell\big(h_{\ell-1};\, W_\ell\big), \qquad \ell = 1, \ldots, L, \quad h_0 \equiv x, \tag{33}$$

可以是 (i) 一个仿射层 $h \mapsto W_\ell h + b_\ell$ 后接逐点非线性函数，(ii) 一个带有残差连接的卷积/注意力块，或 (iii) 一个具有固定（评估模式）统计量的˜　　˜归一化 N_ℓ。设 f_k 为通过将 W_ℓ 替换为 $W_\ell(k)$ 得到的压缩网络（例如，按 k 索引的秩/位配置）。定义逐层残差

$$\Delta W_\ell(k) \;:=\; W_\ell - \tilde{W}_\ell(k). \tag{34}$$

层后雅可比增益。对于每一层 ℓ，将 $T_{\ell\to L}$ 表示为层后映射，它将第 ℓ 层的输入（即 h_ℓ）映射到逻辑值 $z \in \mathbb{R}^{d_L}$，其中所有后续模块均保持不变。其雅可比算子范数为

$$L_\ell^\star(x) \;:=\; \Big\| J_{T_{\ell\to L}}(h_\ell(x)) \Big\|_2. \tag{35}$$

实践中，我们估计一个p　　ˆ近似 L_ℓ 通过对高效线性化进行几次幂迭代步骤；我们假设存在有界的松弛　　k $\eta_\ell \geq 1$ 使得

$$L_\ell^\star(x) \;\leq\; \hat{L}_\ell \;\leq\; \eta_\ell\, L_\ell^\star(x), \qquad \text{uniformly over } x \in \mathcal{C}. \tag{36}$$

分块利普希茨性与归一化。 对于每个块，映射 $u \mapsto \Phi_\ell(u; W_\ell)$ 在 u 上局部为 L_Φ-Lipschitz，并且在 $(h_{\ell-1}, W_\ell)$ 附近对 W_ℓ 局部为 L_W-光滑：

$$\big\|\Phi_\ell(u;W_\ell)-\Phi_\ell(v;W_\ell)\big\|_2 \le L_{\Phi,\ell}\|u-v\|_2, \qquad \big\|\Phi_\ell(h_{\ell-1};W_\ell)-\Phi_\ell(h_{\ell-1};\tilde W_\ell)\big\|_2 \le L_{W,\ell}\,\|W_\ell-\tilde W_\ell\|_2\,\|h_{\ell-1}\|_2. \tag{37}$$

对于具有 1-Lipschitz 激活函数（ReLU/GELU 在近线性区域内近似）的仿射逐点模块，由次乘性可得。归一化层在评估模式下使用（例如，冻结的批归一化、具有固定参数的层归一化）时，被视为具有算子增益的固定线性仿射变换，并将该增益吸收到中。

残差连接。对于残差块，其具有 Lipschitz 性，层后增益满足

$$L^\star_{\ell-1}(x) \;\le\; (1 + L_{\Psi,\ell})\, L^\star_\ell(x). \tag{38}$$

在我们的证书中，我们不将残差增益相乘；我们通过直接进行雅可比估计，将所有层后放大效应纳入。

C.2 logit 漂移界的完整证明（逐层→网络）

我们界定 $\delta z(x;\, k) := f_k(x) - f(x)$ 由逐层 扰动经层后雅可比传播 s.

单层替换界。 固定 x 和某个 ℓ。考虑混合网络 $f^{(\ell)}$：它仅在第 ℓ 层使用 $\tilde W_\ell(k)$，而在其他层使用W_j。令 $h_{\ell-1}$ 为 f 中的预-ℓ 激活（由于层 $< \ell$ 相同，它在 $f^{(\ell)}$中也相同）。第 ℓ 层的激活变化为

$$\Delta h_\ell \;:=\; \Phi_\ell(h_{\ell-1};\tilde W_\ell) - \Phi_\ell(h_{\ell-1};W_\ell). \tag{39}$$

根据 W-光滑性,

$$\big\|\Delta h_\ell\big\|_2 \;\le\; L_{W,\ell}\,\big\|\tilde W_\ell - W_\ell\big\|_2\,\big\|h_{\ell-1}\big\|_2. \tag{40}$$

将这一变化传播经过后续各层 $\ell{+}1, \dots, L$，得到 logit 差

$$\delta z^{(\ell)}(x;k) \;:=\; f^{(\ell)}(x) - f(x) \;=\; T_{\ell\to L}\big(h_\ell + \Delta h_\ell\big) - T_{\ell\to L}(h_\ell). \tag{41}$$

根据中值形式（或 h_ℓ附近的 $T_{\ell\to L}$ 的 Lipschitz 性质），

$$\big\|\delta z^{(\ell)}(x;k)\big\|_2 \;\le\; L^\star_\ell(x)\,\big\|\Delta h_\ell\big\|_2 \;\le\; L^\star_\ell(x)\,L_{W,\ell}\,\big\|\Delta W_\ell(k)\big\|_2\,\big\|h_{\ell-1}\big\|_2. \tag{42}$$

逐层望远镜求和 由三角不等式。 逐层替换，构建完全压缩网络，直至 me:

$$f \xrightarrow{\ell=1} f^{(1)} \xrightarrow{\ell=2} f^{(2)} \xrightarrow{\cdots} f^{(L)} = \tilde f_k. \tag{43}$$

Then

$$\tilde f_k(x) - f(x) \;=\; \sum_{\ell=1}^{L} \big(f^{(\ell)}(x) - f^{(\ell-1)}(x)\big), \quad f^{(0)} := f, \tag{44}$$

结合三角不等式和 (42),

$$\big\|\delta z(x;k)\big\|_2 \;\le\; \sum_{\ell=1}^{L} L^\star_\ell(x)\,L_{W,\ell}\,\big\|\Delta W_\ell(k)\big\|_2\,\big\|h_{\ell-1}(x)\big\|_2. \tag{45}$$

实用代理量与证书表述。 将 $L_{W,\ell}$ 吸收到 $L^\star_\ell(x)$ 中（对于仿射变换→逐点作用且非线性函数为 1-利普希茨非线性函数的情形，该值等于 1），并将 $L^\star_\ell(x)$ 替换为其代理量 $\hat{} L_\ell$（通过幂迭代估算）。由此得到可部署的界

$$\boxed{\big\|\delta z(x;k)\big\|_2 \;\le\; \sum_{\ell=1}^{L} \hat L_\ell\,\big\|\Delta W_\ell(k)\big\|_2\,\big\|a_{\ell-1}(x)\big\|_2,} \tag{46}$$

与 $a_{\ell-1} \equiv h_{\ell-1}$ 层 ℓ的输入相对应。这就是所声称的逐点logit 漂移证书。

C.3 通过激活范数进行依赖于数据的收紧

现在，我们推导一个期望（经数据集校准的）证书，它通常更紧致且更稳定 e.

校准统计量与詹森不等式/柯西–施瓦茨不等式。令$\mathcal{C}$为一个校准集。定义逐层激活均方根

$$\alpha_\ell := \left(\mathbb{E}_{x\in\mathcal{C}}\|a_{\ell-1}(x)\|_2^2\right)^{1/2}. \tag{47}$$

对（46）两边平方，应用詹森不等式，然后对每一项应用柯西–施瓦茨不等式，可得

$$\left(\mathbb{E}_{x\in\mathcal{C}}\|\delta z(x;k)\|_2^2\right)^{1/2} \le \sum_{\ell=1}^{L} \hat{L}_\ell \left\|\Delta W_\ell(k)\right\|_2 \left(\mathbb{E}_{x\in\mathcal{C}}\|a_{\ell-1}(x)\|_2^2\right)^{1/2}. \tag{48}$$

因此，

$$\boxed{\left(\mathbb{E}_{x\in\mathcal{C}}\|\delta z(x;k)\|_2^2\right)^{1/2} \le \sum_{\ell=1}^{L} \hat{L}_\ell \left\|\Delta W_\ell(k)\right\|_2 \alpha_\ell = \hat{\Delta}(k).} \tag{49}$$

由于α_ℓ反映了分布内输入的典型激活能耗，所以当激活值呈现ˆ非对抗性变化时，（49）会收紧（46）。在部署时，我们报告$\Delta(k)$在$\mathcal{C}$上的分位数。

关于残差/归一化堆叠的备注。上述证明不需要将整个网络中的逐层 Lipschitz 常数ˆ相乘。所有层后的放大效应都包含在L_ℓ中，并在工作点（架构、归一化、残差拓扑）处直接估计，因此得到更紧且在经验上稳定的界。

C.4 反例与紧致性讨论

（1）重尾激活 $\Rightarrow$ 宽松的逐点界。考虑一个单独的仿射层$z = Wa$，其上游的 ReLU 会产生重尾的$\|a\|_2$。即使$\|\Delta W\|_2$很小，乘积$\|\Delta W\|_2\|a\|_2$仍可能以不可忽略的概率变得很大，从而使逐点界（46）变得宽松。期望形式（49）通过将$\|a\|_2$替换为α来缓解这一问题，但仍会反映任何真实存在的重尾。改进紧致性：激活裁剪或范数感知正则化可减小α；权重归一化可ˆ减小L_ℓ。

(2) 输入依赖归一化。训练模式下的批归一化依赖于批次，因此其关于输入的 Lipschitz 性质并非全局且与数据无关。我们的证书假设使用评估模式统计量（固定仿射映射），否则$L_\ell^\star$可能会激增。ˆ收紧方法：认证期间冻结 BN（评估模式）；在精确的部署图下估计L_ℓ。

(3) Softmax 尖锐的注意力。自注意力包含 softmax$(QK^\top/\sqrt{d})V$。Softmax 的雅可比矩阵算子范数有界，但当逻辑值趋于平坦时，该范数可能接近 1；若再与较大的$\|Q\|_2\|K\|_2$、$L_\ell^\star$耦合，整体范数可能很大。收紧方法：采用温度调节ˆ正则化或谱控制（例如权重缩放），以界定Q，K；在进行此类缩放后估计L_ℓ。

(4) 由构造导致的非利普希茨块。如果某个块会显式放大范数，例如在残差分支中包含$h \mapsto \gamma h$和$\gamma \gg 1$，却没有相应的补偿，那么即使是微小的$\|\Delta W\|_2$也可能造成较大的漂移。我们的估计ˆL_ℓ会如实反映ˆ这一点；只有低估了L_ℓ时，证书才会过松。收紧方法：采用更保守的幂迭代（增加迭代步数）、残差缩放或谱归一化。

(5) 量化失配。证书是针对推理图计算的。如果训练使用直通估计器，而部署使用舍入或饱和方式不同的整数内核，那么在浮点数中测得的ΔW_ℓ可能会低估实际扰动。收紧：在精确内核生成的反量化张量上计算$\|\Delta W_\ell\|_2$；在算子中纳入逐张量缩放因子/零点。

全局紧致性评论。对于后续模块在$h_\ell(x)$附近可由线性映射局部良好近似的网络，界 (45) 是一阶紧致的；任何非线性曲率都会引入估计器ˆ忽略的二阶残差，使该界变得保守（安全）。从经验上看，通过 (i) 在部署图中测量L_ℓ，(ii) 使用依赖于数据的α_ℓ，以及 (iii) 聚焦于压缩算子的谱范数残差，我们得到的证书与观测漂移具有良好相关性，同时计算开销很小。

保证概要。 在所述假设下，逐点证书和期望证书 (46)–(49) 均成立。ˆ 只有在代理量 L_ℓ 低估了真实的层后增益，或者部署图与经过认证的部署图不同时，才可能出现违例；这两种情况分别通过保守估计（增加幂迭代次数）和认证精确的运行时内核来处理。

D. 部署与工程

D.1 内核选择（通用矩阵乘法与张量核心路径；内存布局）

算子映射。 每个因子化算子都必须降低为具体内核，并满足数据类型、分块粒度和内存布局的要求。对于 SVD 式层，我们将 $\tilde{W}(k) = U_{m\times k}\,\Sigma_{k\times k}\,V_{n\times k}^{\top}$ 实现为两个通用矩阵乘法，中间进行低开销的对角缩放；而 Tucker-2 则采用一系列操作：1×1 降维、$\to$ 空间核心张量、$\to 1\times 1$ 扩展。我们遵循三种模式：

- **标量GEMM路径**（CPU/较旧的 GPU）：适用于内维度较窄的情况 $k \ll \min(m,n)$，并兼容 FP32/FP16/INT8。若张量核心分块利用率会极低，则优先采用此路径。
- **张量核心 MMA 路径**（现代 GPU/神经网络处理器）：当收缩维度和前导维度与原生分块倍数对齐时（例如，具体取决于数据类型，为 8、16 或 32），可获得最佳吞吐量。当 k 较小时，可将多个独立收缩分组，以充分利用分块。
- **深度卷积/逐点卷积融合**（Tucker-2）：围绕空间核心张量连续地安排 $U_{\text{在}}$ 归约和 $U_{\text{在}}$ 扩展，以尽量减少读写操作；在神经网络处理器上，可用时使用厂商分组卷积原语。

数据类型和缩放因子。 混合精度必须匹配内核原生格式：

- **INT8/INT4**：优先使用对称的逐通道权重缩放因子和逐张量激活缩放因子；确保内维度满足 MMA分块所要求的倍数。
- **FP8（E4M3/E5M2）**：当谱能量集中时，适用于 U 和 V；保留 Σ 或 Tucker 核心张量为 FP16/INT8，以避免较小奇异值下溢。

布局。 沿收缩维度选择连续存储，以减少跨步加载：

- 对于 $V^{\top}a$，存储 V，使其 k轴连续；对于后续的 U（$\Sigma\cdot$），存储 U，使 k连续。
- 对于 Tucker-2，选择 NCHW 还是 NHWC，应与目标内核系列相匹配；在移动端神经网络处理器上，NHWC 通常可以减少 1×1 阶段的转置操作。

微优化，无需编写代码。 将 k 填充至最接近的内核倍数，把相互独立的低秩乘积（例如注意力投影）批量合并为一次分组调用，并在导出时将因子预先打包为适合内核的分块，以减少设备端预处理。

D.2 离散预算配置与对齐规则

配置集合。 设每个层 ℓ 都允许一组由目标平台上的内核和布局支持的离散可部署配置 $\mathcal{P}_\ell = \{(k,q)\}$。全局配置是受小型模板集$\mathcal{S} = \{s_j\}_{j=1}^{J}$约束的笛卡尔积，其中每个$s_j$ 都映射到逐层选择 $\{(k_\ell^{(j)}, q_\ell^{(j)})\}$。这样可以保持编译稳定，并支持缓存复用。

对齐策略。 给定连续控制器输出 $(\hat{k}_\ell, \hat{q}_\ell)$ 和设备预算令牌 b，我们按照单调规则对齐至最近的可行邻项：

1. **可行性优先**：选择 $(k,\ q) \in \mathcal{P}_\ell$，使 $|k-\hat{k}_\ell| + \beta|q-\hat{q}_\ell|$ 最小，同时满足内核约束和对齐粒度。
2. **预算单调性**：如果 $b' \succ b$（预算更宽松），则确保 $k_\ell(b') \geq k_\ell(b)$ 和 $q_\ell(b') \geq q_\ell(b)$ 逐分量成立。
3. **证书安全性**：拒绝任何预测漂移 $\Delta_\ell(k,\ q)$ 超出逐层容差的候选项；升级到下一个更安全的邻项。

跨模型模板。 对于包含重复块的系列（例如，Transformer 阶段），重复使用每个阶段的绑定配置，以减少不同内核的数量。模板示例：

- **CNN 模板**：前段卷积层和最终分类器的 k 较高，中期块的 k 适中。
- **Transformer 模板**：在一个块内，$\{W_Q, W_K, W_V, W_O\}$ 共享预算；MLP 投影层的 k 略高。

延迟/能耗门控。 给定校准代理量 Lat(s_j) 和可选的 En(s_j)，选择

$$j^\star = \arg\min_j \widehat{\mathrm{Lat}}(s_j) \quad \text{s.t.} \quad \widehat{\mathrm{Lat}}(s_j) \le \text{budget}, \ \hat{\Delta}(s_j) \le \epsilon. \tag{50}$$

如果没有可行的 s_j 满足 b oth 约束，则选择延迟最低的配置，并在运行时日志中标记证书警告 g.

D.3 导出格式；设备端运行时；兼容性说明

导出产物。 每个目标的导出包包含：

- **每个配置 的因子权重**： SVD 因子 U， V、对角项 Σ 或 Tucker-2 $U_{\text{输出}} G$， $U_{\text{输入}}$，以内核所需的确切数据类型存储（例如，带缩放因子的 INT8）。
- **量化元数据**：逐张量或逐通道缩放因子/零点、舍入模式，以及运行时反量化路径（如有）所用的动态范围摘要。
- **配置清单**：一个精简的类 JSON 索引，列出 s_j、各自逐层的（k， q）、内核对齐填充，以及 ˆ校准成本元组（延迟，能耗 Δ）。
- **内存布局提示**：张量步幅、内存顺序，以及导出时应用的任何免转置改写，以避免设备端格式转换。

运行时选择路径。 在推理开始时，应用程序提供预算令牌 b （例如延迟目标或功耗模式）。运行时将：

1. 使用单调控制器头和取整规则将 b 映射到候选配置索引。 2. 加载或以内存映射方式访问适用于 s_j 的预打包因子，通过在请求之间复用持久分配来最大限度地减少缺页。 3. 按照调度顺序 {归约 $\to$ 核心 $\to$ 扩展} 调度 Tucker-2 的内核，或按 $\{V^\top a \to \Sigma \to U\}$ 调度 SVD 的内核，且不产生中间主机端拷贝。

如果发生热事件或服务质量事件，运行时可以热切换到更严格的 s_{j^-} ，并保证其单调安全且内核兼容。

兼容性说明。

- **对齐和填充**：序列化因子时强制使用分块倍数，从而无需在设备上进行填充；将逻辑形状与填充后的步幅一同存储。
- **激活格式**：当供应商内核不支持逐通道激活量化时，优先使用逐张量激活缩放因子（针对 INT8）；在清单中记录此选择，以防止反量化路径不匹配。
- **算子可用性**：某些神经网络处理器禁止在特定矩阵乘加运算中使用非正方形 k ，或要求遵守分组卷积限制；配置生成器应在导出时剔除这些选项，以避免运行时回退。
- **确定性**：如有需要，在导出时固定随机量化选项和 Gumbel 随机种子，以实现可复现编译；在清单中记录确定性标志。

失效模式及缓解措施。

- **加载时内核不匹配**：回退到同一 s_j 阶段内最近的更小值（k，q），或回退到s_{j^-}；输出诊断信息，其中包含被拒绝的瓦片大小。
- **运行时证书违规**：如果测得的漂移代理指标（例如轻量级激活范数）超出校准界限，则下调一个配置并重新评估；记录相关层标识符，以便离线重新校准。
- **大型加速器利用率不足**：对请求进行批处理，或将注意力投影层聚合为分组通用矩阵乘法，以使张量核心保持饱和；如果无法进行批处理，则选择其 k 与更大瓦片相匹配的配置。

E. 实验方案

本附录详细说明数据处理、模型/训练配置、测量方法和可复现性控制，以便精确复现所有结果。除非另有说明，所有数值均基于下文定义的验证集，并采用正文所述的配置文件选择和证书计算方法。

E.1 数据集、预处理、数据增强

视觉。**ImageNet-1k**（1,281,167 张训练图像、50,000 张验证图像、1,000 个类别）。图像通过 JPEG 解码，并采用双线性插值和标准化的 RGB 排序。训练 变换：（i）随机调整大小裁剪至 224×224，面积范围为 $[0.08, 1.0]$，宽高比范围为 $[3/4, 4/3]$；（ii）以 0.5 的概率进行随机水平翻转；（iii）颜色抖动，亮度/对比度/饱和 $=0.4$，色调 $=0.1$；（iv）对于 ViT 变体，可选用 RandAugment，N=2，幅度为 M=9；（v）仅对 ViT/ConvNeXt 运行使用 mixup α=0.2 和 Cut Mix α=1.0；（vi）按通道均值（0.485，0.456，0.406）和标准差（0.229，0.224，0.225）进行归一化。评估 变换：将较短边调整为 256（双三次插值），中心裁剪至 224×224，并进行相同的归一化。除非另有说明，否则使用单次 224^2 裁剪。

CIFAR-100（5 万张训练集/1 万张测试集，共 100 个类别）。训练：随机裁剪，32×32 使用 4 像素填充；随机水平翻转，概率为 0.5；根据数据集均值/标准差进行归一化。评估：中心裁剪（不填充），使用相同的归一化。

语言（编码器）。**GLUE**开发集：MNLI（匹配/不匹配）、QQP、SST-2。分词使用与预训练检查点匹配的 WordPiece/BPE；除非另有说明，SST-2/QQP 的最大序列长度为 128，MNLI 为 256。训练：按批次进行动态填充，使用基础模型中的丢弃设置，不进行数据增强，仅执行标准文本归一化。评估：单次前向传播，使用任务专用头；报告 GLUE宏平均。

语言模型。**WikiText-103**用于计算困惑度。分词和去分词与发布的 TinyLlama-1.1B 设置一致；上下文长度为 512。使用滑动窗口计算困惑度，且不进行重叠的损失掩码。

数据完整性和划分。 除官方发布版本之外，不进行类别再平衡或额外去重。对于涉及仅校准子集的任何运行（例如训练后量化基线），使用训练集中的一个不相交的 5,000 个样本切片；T3C 使用单独的 10,000 个样本校准切片来计算证书统计量（E.3 详细说明了其作用）。

E.2 模型配置、训练调度方案、HPO 网格

架构。**卷积神经网络**：ResNet-50/101（标准瓶颈结构，{3、4、6、3}/ {3、4、23、3} 模块，宽度乘数为 1。0）。**视觉变换器**：ViT-B/16 和 ViT-L/16，图像块大小为 16，隐藏层维度为 {768、1024}，多层感知机比例为 {4。0、4。0}，注意力头为 {12、16}。**Swin-T**：窗口大小为 7，深度为 {2、2、6、2}。**编码器**：BERT-Base、RoBERTa-Base、DistilBERT，使用默认注意力头维度。**语言模型**：TinyLlama-1.1B（隐藏层维度 2048、22 层、32 个注意力头）、旋转位置嵌入，词汇表采用发布版本。

优化与调度（T3C）。所有 T3C 模型均使用 AdamW 训练；权重衰减 $\in [0.01, 0.05]$。基础学习率随加速器数量 $\times$ 全局批次按线性规则缩放：

$$\mathrm{LR}_{\text{base}} = \eta_0 \cdot \frac{\text{GlobalBatch}}{256}, \quad \eta_0 \in \{1.0\times10^{-3},\, 2.0\times10^{-3}\} \text{ (vision)}, \;\; \eta_0 \in \{2.0, 3.0\}\times10^{-5} \text{ (NLP)}. \tag{51}$$

余弦衰减，并进行 5–10 个训练轮次的线性预热（视觉），或预热总步数的 5%（自然语言处理）。除非另有说明，ImageNet 使用 0.1 的标签平滑。

T3C 规格 fic schedule. 秩采样从宽泛的均匀分布逐渐退火为按预算加权的分布 :

$$p_t(k) \propto \Big[\gamma_t\, \mathbb{U}[k_{\min}, k_{\max}] + (1-\gamma_t)\, \mathbf{1}\{k \in \mathcal{K}_{\text{profiles}}\}\Big], \quad \gamma_t = \max\Big(0, 1 - \frac{t}{T_{\text{anneal}}}\Big), \tag{52}$$

在训练的前三分之一阶段。Gumbel-Top-k 温度 τ呈指数衰减： τ_t =最大值($\tau_{\min}$， $\tau_0 \cdot \alpha^{t/T}$)，默认使用 ($\tau_0$， $\tau_{\min}$， α) = (2.0， 0.3， 0.5)。证书惩罚项权重 λ_{CERT} 在同一时间窗口内从 0 线性递增至目标值，以避免训练初期过度正则化。

超参数优化网格（代 表性）。我们采用小型且考虑硬件特性的网格进行搜索，以避免对单一目标值过拟合 et.

- 视觉：学习率 $\in\{1\times10^{-3}, 2\times10^{-3}\}$，权重衰减 $\in\{0.02, 0.05\}$，mixup/CutMix 开启/关闭（仅适用于 ViT/ConvNeXt），标签平滑 $\in\{0, 0.1\}$， λCERT $\in\{0.1, 0.2, 0.4\}$， $\epsilon \in\{0.10, 0.15\}$（漂移容差），秩上限 k最大为每种层类型完整秩的 {0.6, 0.8, 1.0}。
- **自然语言处理**：学习率 $\in\{2, 3\} \times 10^{-5}$，权重衰减 $\in\{0.01, 0.05\}$，丢弃率 $\in\{0.1, 0.2\}$， $\lambda_{\text{CERT}} \in\{0.05, 0.1\}$， $k_{\text{最大}}$ 位于$\{0.5, 0.7\}$：投影层取满秩时的比例；多层感知机取$\{0.6, 0.8\}$。

每个网格点使用三个随机种子进行训练（E. 4），并选取验证目标函数最佳的设置（任务损失 + 约束松弛惩罚）。

训练轮次和批量大小。 ImageNet：ViT/ConvNeXt 训练 300 轮，残差网络训练 120 轮；全局批次大小为 4096（ViT）或 2048（残差网络）。GLUE：MNLI 训练 3 轮，QQP 训练 3 轮，SST-2 训练 6 轮；批量大小为 128 个序列（动态）。

E.3 测量工具（延迟第50百分位/第90百分位、能耗）

吞吐量与延迟状态。 除非另有明确标注，所有延迟数值均以单请求、批次= 1 状态报告，且使用预热缓存和稳定时钟。我们区分：

- **冷启动**：包括模型加载、图初始化和首次推理。
- **稳态**：不包括加载；包括任何运行时配置文件选择。

正文中的表格采用稳态数值。

延迟定义。 给定每次推理的墙钟时间样本$\{t_i\}_{i=1}^{N}$， 丢弃 N_{warm} 次预热运行后，我们计算

$$\text{p50} = \text{Quantile}_{0.5}(\{t_i\}), \qquad \text{p90} = \text{Quantile}_{0.9}(\{t_i\}), \tag{53}$$

其中 $N \geq 1000$， 除非受到设备稳定性的限制。我们使用 B=2000次重采样，报告 95% 自助法置信区间。

时钟控制与热控制。 在服务器 GPU 上，将时钟固定为默认的最大应用时钟；启用持久化模式；开启独占进程模式；将功率限制设为厂商热设计功耗。在边缘设备上，使用性能调速器，禁用后台服务，并强制进行 10 分钟的预处理期，以稳定热状态。任何 QoS 降频事件都会导致该次运行作废并重新运行。

能量测量。 在 GPU 上，我们通过厂商遥测以 10–20 Hz 的频率采样瞬时功率 $P(t)$，并在稳态时间窗 $[t_1, t_2]$ 内对其积分：

$$E = \int_{t_1}^{t_2} P(t)\, dt \approx \sum_{j=1}^{M} P_j\, \Delta t, \qquad \text{Energy/inference} = \frac{E}{N_{\text{infer}}}. \tag{54}$$

对于移动/嵌入式设备，我们使用板级 INA 传感器或外部 USB-C PD 分析仪，以 $\Delta t \leq 10$毫秒的间隔进行采样。我们使用相同的进程常驻状态和相同的时钟测量空闲基线，并将其扣除。

校准延迟代理指标。 对于每个目标设备，我们收集一组包含 G 个配置的网格 s_j（涵盖不同层类型中的 (k, q)），并进行回归

$$\widehat{\mathrm{Lat}}(s_j) = \alpha_0 + \sum_{\ell} \Big(\alpha_\ell^{\mathrm{comp}} \, \mathrm{FLOPs}_\ell(s_j) + \alpha_\ell^{\mathrm{mem}} \, \mathrm{Bytes}_\ell(s_j) \Big), \tag{55}$$

采用非负最小二乘法。拟合优度以 R^2 和平均绝对百分比误差（MAPE）报告。相同网格通过系数为 β的类似回归，提供各配置的能耗。

证书成本核算。 除非审计运行中明确提及，否则证书计算不计入延迟；它会在导出时针对每种配置离线计算（E. 3 校准集）。在线监控（可选）会采集轻量级激活均方根，并与校准最大值进行比较，而无需进行雅可比幂迭代。

E.4 随机种子；硬件规格；可复现性检查清单

种子与确定性。 除非另有说明，每种配置均使用三个独立随机种子进行训练 $s \in \{3407、2025、9157\}$，用于初始化：模型/因子权重、数据打乱、Top-k 松弛化所用的 Gumbel 噪声，以及任何丢弃掩码。我们强制执行：

- 固定数据加载器分片，并按种子为每个 epoch 重新打乱数据。
- 在可用的情况下使用确定性内核；如果某个内核提供非确定性的快速路径（例如原子归约），我们会将其记录在运行清单中。
- 量化校准子集由带种子的采样器抽取（与训练批次不重叠）。

我们报告准确率/F1 分数和困惑度在不同随机种子上的均值及均值标准误；对于延迟/能耗，我们按照上述说明汇总时间序列样本，并报告不同随机种子上的中位数。

硬件规格。

- **服务器 GPU**：NVIDIA A100 40GB PCIe；CUDA 驱动程序版本 12.x；主机 CPU 为 $2\times$Xeon（或 EPYC），启用 SMT；256GB RAM；PCIe Gen4；操作系统为 Linux 内核 ≥ 5.15。
- **边缘 GPU**：Jetson Orin 32GB；JetPack $\geq 6.x$；功耗模式为 MAXN；风扇固定为 100%。
- **Android big.LITTLE CPU**：1 个现代旗舰 SoC（大核 2.8–3.3GHz，小核 1.8–2.2GHz）；Android ≥ 14；飞行模式；电量 $> 80\%$；如果设备支持，则在开发者模式下禁用热节流。
- **移动端 NPU**：供应商软件开发工具包 $\geq$ 2.x，支持 INT8 和可选的 FP8；测量前预热离线编译缓存。

实验清单中会记录确切的驱动程序/软件开发工具包版本、内核构建标识和固件哈希值，以及模型校验和。

F. 扩展结果与消融实验

本附录通过更大篇幅的无图表格和详细分析，扩展了主要结果。我们重点关注：（i）跨预算的视觉和语言扫描，并加入额外的稳定性指标；（ii）随机种子敏感性和尾延迟离散度；（iii）认证诊断（覆盖率、校准、相关性）；（iv）大规模控制器/分词/量化消融实验；（v）跨设备的成本代理泛化；以及（vi）单调性和违规审计。除非另有说明，所有数字均遵循附录 E 中的协议。

F.1 视觉：跨预算结果的稳定性与违规情况

分析。 (a) 预算遵循：得益于配置档位对齐和证书惩罚项，T3C 在所有模型和预算下报告的违约率均为**0.0%** ；在相同的“中型”准确率区间，PTQ/MP-QAT 的违约率为 2.9–7.1%。(b) 认证紧致度：

表 *4.* **视觉跨预算扫描（ImageNet-1k）**。p50 [p90] 延迟（毫秒）；违规率表示超过声明预算的运行次数占比；ϵ 为证书界（第4节）；平均绝对百分比误差表示测量的第50百分位延迟的延迟代理误差。延迟/违规率/ϵ/平均绝对百分比误差越低越好。

模型	预算	方法	Top-1（%）	A100	Jetson	Android	大小（MB）	违规率	ϵ / 平均绝对百分比误差
ResNet-50	Tiny	**T3C**	75.6	**1.18** [1。45]	**13.0** [16.5]	**22.4** [29。3]	**38**	**0.0**	**0.14 / 2.8**
	Med	**T3C**	**76.0**	**1.26** [1。58]	13.8 [17.3]	23.6 [31.0]	**42**	**0.0**	**0.12 / 3.1**
	Max	**T3C**	76.1	1.34 [1.62]	14.9 [18.2]	24.7 [32.1]	47	**0.0**	0.15 / 3.3
	Med	PTQ-8b	75.7	1.44 [1.84]	18.5 [23.8]	29.2 [38.9]	88	4.6	0.22 / 7.9
	Med	QAT-8b	**75.9**	1.36 [1.76]	17.4 [22.1]	27.6 [36.0]	90	2.9	0.19 / 6.3
ViT-B/16	Tiny	**T3C**	81.2	**2.30** [2.92]	**26.0** [33.0]	**41.8** [53。9]	**59**	**0.0**	**0.18 / 3.2**
	Med	**T3C**	**81.5**	**2.38** [3。04]	26.5 [33.4]	42.6 [54.8]	**64**	**0.0**	**0.16 / 3.5**
	Max	**T3C**	81.7	2.45 [3.10]	27.8 [35.0]	44.0 [56.2]	71	**0.0**	0.19 / 3.7
	Med	MP-QAT	**81.3**	2.58 [3.32]	33.0 [41.7]	52.9 [67.1]	134	5.8	0.25 / 8.4
	Med	PTQ-8b	81.0	2.72 [3.48]	34.9 [44.1]	55.6 [70.5]	131	7.1	0.28 / 9.1
Swin-T	Tiny	**T3C**	81.1	**1.72** [2。14]	**19.8** [25。2]	**31.6** [40。9]	**62**	**0.0**	**0.17 / 3.0**
	Med	**T3C**	**81.3**	**1.82** [2。26]	21.0 [26.6]	33.1 [42.7]	**66**	**0.0**	**0.16 / 3.2**
	Max	**T3C**	81.5	1.95 [2.38]	22.6 [28.4]	34.9 [45.0]	73	**0.0**	0.19 / 3.6
	Med	MP-QAT	**81.2**	2.02 [2.46]	27.1 [33.2]	41.6 [53.2]	128	3.7	0.27 / 7.0
	Med	PTQ-4b	80.7	2.14 [2.62]	29.4 [36.5]	45.1 [57.8]	96	6.2	0.39 / 9.8

ϵ 始终低于 T3C 的 PTQ/QAT，其中 ViT-B/16 的相对提升最大（0.16，对比 0.25–0.28）。(c)代理拟合：T3C 配置的延迟平均绝对百分比误差为**2.8–3.7%**，而基线为 ∼8–9%；这是因为 T3C 的成本代理采用混合方案（字节数 + 浮点运算次数），并且已针对离散内核进行预校准（见表 4）。

F.2 随机种子稳定性与尾延迟离散度

表 *5.* **随机种子稳定性（均值±3 个随机种子的均值标准误）**以及尾延迟离散度。最后一列报告第 90 百分位数与第 50 百分位数之差（毫秒）；数值越低表示尾部越集中。

模型	预算（方法）	Top-1（%）	A100 第50百分位（毫秒）	Android 第50百分位（毫秒）	尾部差距（A100 第90百分位-第50百分位）
ResNet-50	中型（**T3C**）	76.0 ± 0.05	1.26 ± 0.01	23.6 ± 0.2	**0.32**
ResNet-50	中型（PTQ-8b）	75.7 ± 0.06	1.44 ± 0.02	29.2 ± 0.4	0.40
ViT-B/16	中型（**T3C**）	81.5 ± 0.04	2.38 ± 0.02	42.6 ± 0.3	**0.66**
ViT-B/16	中型 (MP-QAT)	81.3 ± 0.05	2.58 ± 0.03	52.9 ± 0.6	0.74
Swin-T	中型 (**T3C**)	81.3 ± 0.04	1.82 ± 0.02	33.1 ± 0.3	**0.44**
Swin-T	中型 (PTQ-4b)	80.7 ± 0.05	2.14 ± 0.03	45.1 ± 0.5	0.48

分析。 准确率和延迟的 SEM 值都很小，表明训练稳定。T3C 持续缩小第 90 百分位数与第 50 百分位数之差这一“尾部差距”，这表明通过快照对齐到少量预先基准测试的配置文件，可减少内核回退或缓存未命中（见表 5）。

F.3 证书诊断：覆盖率、校准与相关性

分析。 在报告的 ϵ 下，各模型的覆盖率为 ∼91–93%。相关性 $\geq$0.89 表明， Δ 对配置的排序与观测漂移类似，这是预算管控的关键要求。剩余差距源于罕见增强上的重尾激活值；增大校准集或采用逐块重新加权可提高覆盖率（见表6）。

F.4 大规模消融实验：控制器、分词、量化

分析。 **联合**（秩+位数）设计是准确率/延迟改进和零违规的主要驱动因素；元组令牌预算和逐因子位偏移带来稳定但较小的收益。移除控制器（CtrlOFF）会显著增加违规，因为逐层分配会退回到静态启发式方法（见表7）。

表*6*。 **认证诊断**。 覆盖率为样本的 = %，观测漂移为 $\|\delta z\|_2 \le \epsilon$；Corr(ˆ∆, 漂移) = 配置间的皮尔逊相关系数。

模型（预算）	覆盖率（%）	95th $\hat{\Delta}$	平均漂移	Corr($\hat{\Delta}$, 漂移)
ResNet-50（微型）	**93.1**	0.15	0.12	**0.92**
ResNet-50（中型）	92.6	0.14	0.11	**0.93**
ViT-B/16（微型）	91.0	0.19	0.15	0.90
ViT-B/16（中型）	**92.2**	0.17	0.14	**0.92**
Swin-T（中型）	92.0	0.16	0.13	0.91
BERT-Base（中型）	91.4	0.20	0.16	0.89

表*7.* **汇总5个模型的消融实验**（R50、ViT-B/16、Swin-T、BERT-Base、RoBERTa-Base），在匹配的“中型”准确率目标下进行。数值为各模型的平均值；↓越低越好。

变体	详情	准确率下降 %↓	A100 第50百分位（毫秒）	违规率 ↓	ϵ ↓
仅秩	固定比特，学习得到的秩	0.96	1.85 / 3.40 / 2.46	1.9	0.19
仅位数	固定秩，学习位数	0.84	1.89 / 3.48 / 2.52	2.4	0.20
CtrlOFF	无预算控制器	1.32	1.93 / 3.58 / 2.61	4.7	0.23
T3C（我们的方法）	联合秩 +位数 + 快照	**0.58**	**1.82 / 3.38 / 2.44**	**0.0**	**0.16**
元组预算	(延迟,字节数) 令牌	**0.62**	**1.83 / 3.39 / 2.45**	**0.3**	**0.17**
逐因子 MP	$(q_U,\ q_G,\ q_V) = (q{+}1,\ q,\ q{+}1)$	0.63	1.84 / 3.40 / 2.46	0.4	**0.17**
均匀 MP	$(q_U,\ q_G,\ q_V) = (q,\ q,\ q)$	0.71	1.85 / 3.42 / 2.47	0.9	0.18
课程排序	高→低进度安排	**0.62**	**1.83 / 3.39 / 2.45**	**0.3**	**0.17**

三元组 ets 展示了 {R50, ViT-B/16, Sw 的 A100 第50百分位平均值 in-T}。

F.5 CostProxy 跨设备泛化

表 *8.* **代理泛化**。 在 A100 上训练代理量，并在 Jetson/Android/神经网络处理器的 P50 延迟（Med 配置）上评估平均绝对百分比误差（%）。越低越好。

代理量	A100（训练）	Jetson（测试）	Android（测试）	移动端 NPU（测试）
仅 FLOPs	2.6	10.8	13.4	11.7
仅字节数	3.1	**4.9**	**6.2**	**5.4**
混合方案（我们的方法）	**2.4**	**4.2**	**5.5**	**4.7**

分析。 仅 FLOPs 会在内存受限路径（Jetson/Android/神经网络处理器）上做出错误预测（见表 8）。仅字节数在服务器之外表现良好，但在计算受限的 A100 上略显逊色。经过校准的**混合方案** 代理量结合了两者，在不同设备间的迁移效果最佳。

F.6 单调性审计与预算违规

分析。 将配置快照到内核对齐配置的校准格点上，可保证 T3C 的单调转换（见表 9）。基线有时会出现回归，因为自动调优会在不同预算下选择不同的内核，从而颠倒实际的延迟顺序或改变数值稳定性。

F.7 逐层预算分配统计

分析。 控制器一贯倾向于选择较早的 CNN 块以及注意力投影层/MLP 头（见表 10）——这些层 ˆ 对证书质量贡献最大 $\Delta(k)$。每因子偏移（ $+1$ 比特时的 U/V）最常应用于这些敏感位置。

表 *9.* **单调性与违规情况。** 我们检查相邻预算（微型→中型、中型→最大），并统计预算增加后实际准确率降低或 P50 延迟增加的非单调事件。

模型	方法	非单调（准确率）	非单调（延迟）	违规率（中型）
ResNet-50	**T3C**	**0/2000**	**0/2000**	**0.0**
ResNet-50	PTQ-8b	7/2000	18/2000	4.6
ViT-B/16	**T3C**	**0/2000**	**0/2000**	**0.0**
ViT-B/16	MP-QAT	4/2000	22/2000	5.8
Swin-T	**T3C**	**0/2000**	**0/2000**	**0.0**
Swin-T	PTQ-4b	9/2000	27/2000	6.2

表 *10*。 **逐层分配摘要（Med 配置）**。 按块类型列出的平均秩比例（相对于完整模型）和位宽。

模型	早期块（秩，位数）	中期块（秩，位数）	后期块（秩，位数）	分类器/投影层（秩，位数）
ResNet-50	**0.78**, **9.0**	0.64, 8.0	0.68, 8.0	**0.82**, **9.0**
ViT-B/16	**0.80**，**9.0**(查询、键、值)	0.66, 8.0	0.70, 8.0	**0.78**，**9.0** (多层感知机)
Swin-T	**0.76**, **8.5**	0.63, 8.0	0.67, 8.0	**0.80**, **8.5**
BERT-Base	0.74, **9.0**(自注意力)	0.62, 8.0	0.66, 8.0	**0.80**, **8.5** (池化器)

表 *11.* **Jetson（批次=1）上的能耗（mJ）与延迟（ms）。**

模型	预算（方法）	延迟第50百分位（毫秒）	能耗（毫焦）
ResNet-50	微型（**T3C**）	**13.0**	**6.9**
ResNet-50	中型（**T3C**）	13.8	**6.6**
ResNet-50	中型（PTQ-8b）	18.5	7.5
ViT-B/16	微型（**T3C**）	**26.0**	**11.2**
ViT-B/16	中型 (**T3C**)	26.5	**10.8**
ViT-B/16	中型 (MP-QAT)	33.0	12.9

F.8 能耗–延迟帕累托

分析。 T3C 的各项指标在数值上优于或持平于基线；与“微型”配置相比，“中型”配置的能耗略有改善，尽管延迟有所增加，这是因为内核占用率更高且缓存未命中更少（见表 11）。

F.9 要点

(1) 可靠性。 T3C 在所有测试设备/预算下的预算违规率均为**0.0%**，而基线的尾部失效率为 2–7% s 在准确率相同的情况下。**(2) 证书。** 报告的 ϵ 可追踪观测漂移，并在选定容差下提供 >90% 的覆盖率；相关性$0.89-0.93$ 表明排序效果良好。<style id='6>**(3)** 控制器。 联合秩+位控制带来了大部分收益；元组预算和逐因子混合精度还能带来稳定的 ，低成本改进。**(4) 可移植性。** 混合成本代理指标在内存受限目标设备间的迁移效果最佳，无需重新拟合即可将平均绝对百分比误差保持在 $<6\%$ g 。**(5) 分配模式。** 前段卷积层、注意力投影层和最终的多层感知机头会获得更高的秩/位数，这与它们的 r ˆ 对 $\Delta(k)$的贡献。