基于敏感度指导的训练后量化方法

Sensitivity-Guided Progressive Quantization (SGPTQ)

汇报人：王招元
2026年4月

汇报内容目录

01 研究背景：大语言模型的部署瓶颈

- 大语言模型 (LLMs) 参数规模呈指数级增长（百亿至千亿级别）。
- 庞大的参数量导致巨大的存储空间需求和推理计算延迟，产生严重的“显存墙”。
- 边缘设备或受限计算节点（如国产算力平台）显存容量有限，难以直接部署全精度模型。
- 迫切需要高效的模型压缩技术，在降低资源开销的同时维持模型泛化能力。

核心出路：模型量化 (Quantization) —— 将 FP16 浮点数映射为 INT8/INT4/INT3 等低精度定点数，是最直接、显存压缩收益最高的方法。

01 基础理论：模型量化原理

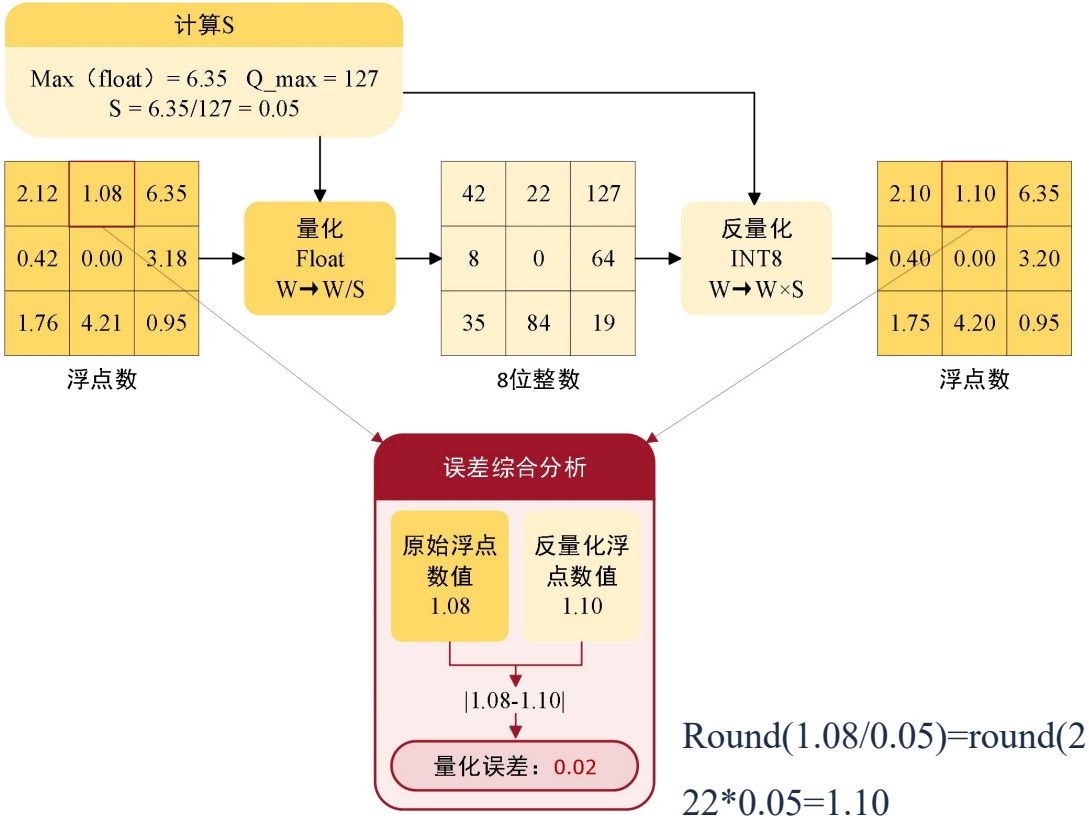
- 目前工业界与学术界主流的大模型量化方案普遍采用线性量化机制，即浮点数与量化定点数之间呈仿射映射关系。根据量化区间的零点是否与绝对零点对齐，线性量化可进一步分为对称量化和非对称量化。
- 以对称量化为例，假设输入一个浮点张量x，其数值分布范围为$[x_{\max}, x_{\min}]$，目标量化位宽为b比特，则该位宽所能表示的整数范围为$[q_{\min}, q_{\max}]$。（其符号 b 比特整数的取值范围[-（2^{b-1}-1），2^{b-1}-1]（8位有符号为-127-127）。量化过程的核心在于确定缩放因子（*Scaling Factor,* S）：

$$S = \frac{x_{max} - x_{min}}{q_{max} - q_{min}} \quad , \quad x_q = \text{clip}\left(\text{round}\left(\frac{x}{S}\right)\right)$$

- 其中round()为向最近整数取整函数，clip()函数用于将超出表示范围的值截断至之间，以防止数据溢出。在推理时，由于底层运算单元可能需要浮点输入，或者在累加后需要恢复原有的数值量级，必须将量化后的定点数重新映射回浮点域，这一过程称为反量化：

$$\hat{x} = S * x_q$$

01 基础理论：模型量化误差



- 假设有如图所示的权重矩阵X，我们假设使用对称量化，参照公式：

$$S = \frac{x_{max} - x_{min}}{q_{max} - q_{min}}$$

- 可知，S=0.05，参照公式

$$x_q = \text{clip}\left(\text{round}\left(\frac{x}{S}\right)\right)$$

- 可得量化后的8位整数权重矩阵Xq。

Round(1.08/0.05)=round(21.6)=22

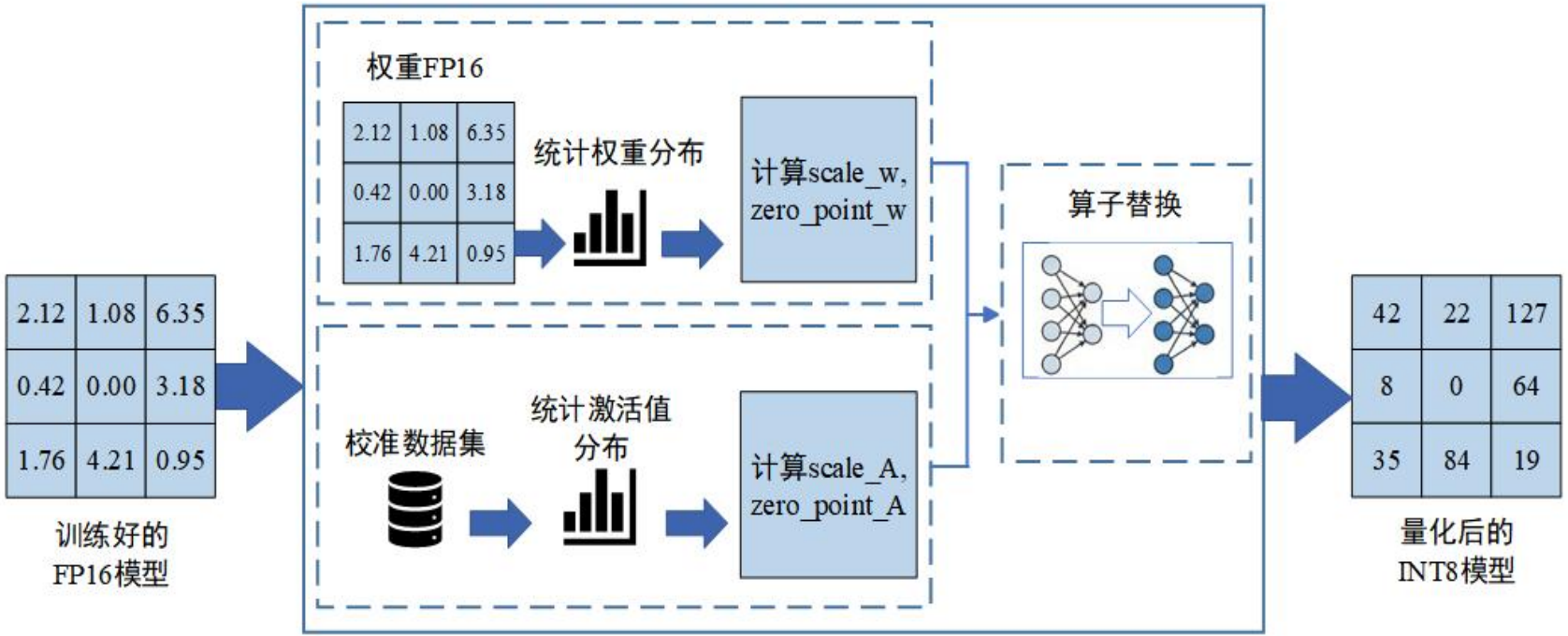
22*0.05=1.10

01 基础理论：QAT量化与PTQ量化

- PTQ（训练后量化）与 QAT（量化感知训练）是深度学习模型量化的两大工业级主流范式，二者的本质差异是是否通过训练让模型主动适应量化误差。

对比维度	PTQ（训练后量化）	QAT
核心本质	被动适配量化，直接统计量化参数	主动学习适应量化误差，训练优化权重
训练依赖	完全不需要训练 / 微调	依赖完整的微调训练流程
数据需求	仅需少量无标签校准数据	需要完整的训练 / 微调数据集
精度表现	INT8 场景优秀，低比特（≤INT4）精度易崩溃	INT8 场景接近 FP32，低比特场景仍能保持高保真
落地成本	极低，一键式工具链即可完成	较高，需代码适配、调参、训练资源
耗时	分钟级完成量化	小时 / 天级，取决于模型规模与训练轮次

01 基础理论：PTQ量化



02 现有方法原理

GPTQ

基于二阶信息与逐列误差补偿机制，实现了低成本、高精度的 4-bit 及以下低比特权重量化，成为百亿级大模型单卡部署的核心方案。

SmoothQuant

通过数学等价变换完成激活 - 权重的量化难度迁移，平滑激活离群值的同时保留计算逻辑不变，实现了无损、硬件友好的全 INT8 量化，完美适配云端高并发低延迟推理场景。

AWQ

基于激活感知的显著权重保护机制，在极低比特量化下仍能保持极高的模型保真度，同时兼顾量化效率与端侧硬件适配性，成为端侧大模型轻量化部署的主流选择。

Frantar E, Ashkboos S, Hoefler T, et al. Gptq: Accurate post-training quantization for generative pre-trained transformers[J]. arXiv preprint arXiv:2210.17323, 2022.
Xiao G, Lin J, Seznec M, et al. Smoothquant: Accurate and efficient post-training quantization for large language models[C]//International conference on machine learning. PMLR, 2023: 38087-38099.
Lin J, Tang J, Tang H, et al. Awq: Activation-aware weight quantization for on-device llm compression and acceleration[J]. Proceedings of machine learning and systems, 2024, 6: 87-100.

GPTQ：二阶误差补偿的低比特权重量化

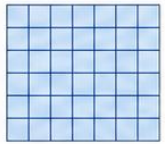
输入

FP16 权重矩阵 W
+ 校准激活数据 X

$W \in R^{out \times in}$

$X \in \mathbb{R}^{batch \times in}$

W



X

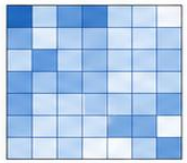
1 计算 Hessian 矩阵

衡量权重对输出的影响

$$H = X^T X$$

$$H \in \mathbb{R}^{in \times in}$$

H

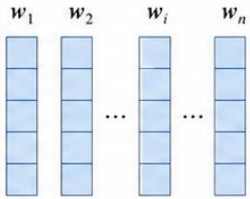


H 是输入相关的二阶信息，
反映各输入维度对输出的影响
及其相关性

2 权重分块（按列）

将权重矩阵按列拆分，
逐列处理

$$W = [w_1, w_2, \ldots, w_n]$$



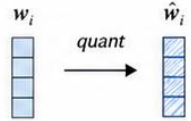
从第 1 列到第 n 列，
依次进行量化与误差补偿

3 单列量化 + 误差补偿循环 $(i = 1, 2, \ldots, n)$

1 量化当前列

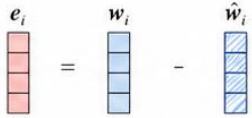
$$\hat{w}_i = \text{quant}(w_i)$$

（低比特量化，如 INT4/INT3）



2 计算量化误差

$$e_i = w_i - \hat{w}_i$$

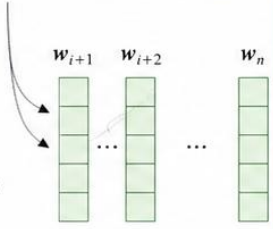


3 误差补偿到后续列

对所有 $j > i$，更新：

$$w_j = w_j - \frac{H^{-1}[j,i]}{H^{-1}[i,i]} \cdot e_i$$

（利用 Hessian 逆矩阵 H^{-1} 分配误差，
抵消对后续列的影响）



循环处理下一列，直到所有列完成量化

输出

INT4/INT3 量化
权重矩阵 $\hat{W}$
+ 量化参数

+

量化参数
(scale, zero_point 等)

- 每列/每通道的 scale
- 每列/每通道的 zero_point
- 量化位宽（如 INT4）

GPTQ 核心思想

- 以二阶信息驱动的逐列误差补偿为核心：用 Hessian 矩阵建模权重列对输出的影响及相关性。
- 逐列量化：将权重矩阵按列拆分，依次进行低比特量化。
- 误差补偿：利用 Hessian 逆矩阵，将当前列的量化误差分配到所有未量化的后续列中，避免误差累积扩散。
- 结果：在极低比特（INT4/3/2-bit）下仍能保持接近 FP16 的精度，是大模型极致压缩的主流方案。

SmoothQuant：难度迁移的无损全 INT8 量化

输入：原线性层计算

$$Y = X \cdot W$$

X：激活值（含离群值）

W：权重

X（激活值）

W（权重）

维度说明：

$X \in \mathbb{R}^{batch \times in}$，$W \in \mathbb{R}^{in \times out}$

1 计算平滑因子 S

基于激活通道最大值与权重通道最大值，计算逐通道平滑因子 S

逐通道统计

$a_j = \max(|X[:, j]|)$　（激活第 j 通道最大值）

$w_j = \max(|W[j, :]|)$　（权重第 j 通道最大值）

平滑因子计算（逐通道）

$$s_j = \left(\frac{a_j}{w_j}\right)^{\alpha}$$

$$S = diag(s_1, s_2, \ldots, s_n)$$

$\alpha \in [0, 1]$（平衡参数，通常取 0.5）

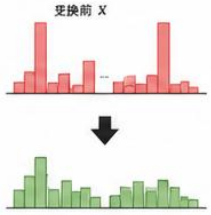
$S \in \mathbb{R}^{in \times in}$，对角矩阵

2 等价变换（数学等价，不改变计算结果）

$$Y = (X \cdot S^{-1}) \cdot (S \cdot W) = \hat{X} \cdot \hat{W}$$

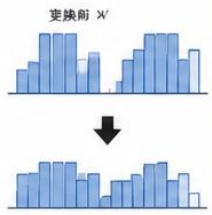
$\hat{X} = X \cdot S^{-1}$

（激活值被平滑，离群点消失）



$\hat{W} = S \cdot W$

（权重被反向缩放，仍易量化）



等价性保证：对角矩阵 S 满足 $S^{-1} \cdot S = I$，因此 Y 保持不变

3 W8A8 全量化

对 $\hat{X}$ 和 $\hat{W}$ 同时执行 INT8 量化，无精度损失

对 $\hat{X}$ 量化（激活）

对 $\hat{W}$ 量化（权重）

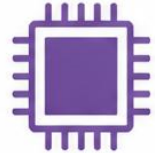
均为 INT8 表示

输出

可硬件加速的全 INT8 线性层

$$Y = \hat{X}_{INT8} \cdot \hat{W}_{INT8}$$

完美适配硬件 Tensor Core 加速



SmoothQuant 核心思想

- 以“激活-权重量化难度迁移”为核心：通过逐通道平滑因子 S，将激活值的量化难度迁移到更易量化的权重上。
- 数学等价变换：$Y = (X \cdot S^{-1}) \cdot (S \cdot W) = \hat{X} \cdot \hat{W}$，计算结果完全不变。
- 变换效果：激活值分布被平滑（离群值消失），更易于 INT8 量化；权重被反向缩放，仍保持良好的量化特性。
- 结果：实现无精度损失的 W8A8 全 INT8 量化，完美适配硬件 Tensor Core 加速，大幅提升推理速度。

AWQ：激活感知的高效低比特量化

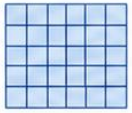
输入

FP16 权重矩阵 W
+ 校准激活数据 X

$W \in \mathbb{R}^{out \times in}$

$X \in \mathbb{R}^{batch \times in}$

W（权重）



X（激活值）

使用少量校准数据，
统计激活分布特征

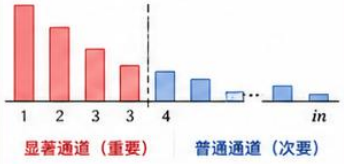
1 激活感知，识别显著通道

统计激活通道幅度，识别对模型输出影响最大的显著权重通道

计算每个输入通道的重要性得分

$$s_j = \max_b |X_{b,j}|$$

$$j = 1, 2, ..., in$$



通道索引（按重要性降序）

得到显著通道索引集合 G
（Top-k 或累计能量阈值）

2 保护显著通道的缩放变换

（数学等价变换，不改变计算结果）

对显著通道放大，对其他通道反向缩放，保证计算逻辑不变

显著通道（放大）

$j \in G$

$W_{:,j} \times g_j \ (g_j > 1)$

其他通道（缩小）

$j \notin G$

$W_{:,j} \times g_j^{-1} \ (g_j < 1)$

缩放因子向量 $g \in \mathbb{R}^{in}$

$$g_j = \begin{cases} > 1, & j \in G \\ < 1, & j \notin G \end{cases}$$

等价变换：

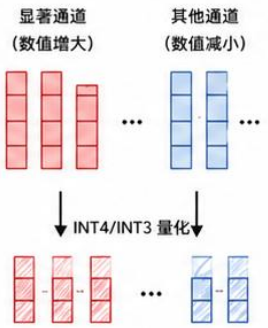
$$Y = X \cdot W = X \cdot (W \cdot \mathrm{diag}(g)) \cdot \mathrm{diag}(g)^{-1}$$

（先对 W 缩放，再在输出侧缩放抵消，Y 不变）

3 统一低比特量化

对缩放后的权重执行 INT4/INT3 量化，显著通道的量化误差被大幅降低

缩放后的权重 $\hat{W} = W \cdot \mathrm{diag}(g)$



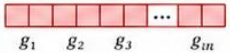
统一量化（INT4/INT3）
显著通道精度更高，整体误差更低

输出

INT4/INT3 量化
权重矩阵 $\hat{W}$
+ 缩放参数 g

+

缩放参数 g
（长度 = in）



- 量化权重：INT4/INT3
- 缩放参数：FP16/FP32
- 存储高好，推理时通过缩放恢复等价计算

AWQ 核心思想

- **激活感知**：基于校准数据的激活分布，识别对模型精度至关重要的显著权重通道；
- **显著通道保护**：通过数学等价缩放，放大显著通道的数值，降低其量化相对误差；对其他通道反向缩放，保证计算结果不变；
- **高效低比特**：无需二阶信息或复杂优化，即可在 INT4/3-bit 下保持高精度。

优势

- 精度高：显著通道量化误差大幅降低，整体性能接近 FP16；
- 效率高：计算简单，仅需通道级缩放，适合大规模模型；
- 部署友好：存储开销小，推理时仅需一次缩放操作，硬件适配性强。

适用场景

- 大模型 INT4/3-bit 量化部署（如 LLaMA、Qwen、ChatGLM 等）
- 对精度要求高的端侧/边缘部署
- 显存受限的推理加速场景

02 现有方法的挑战与局限性

QAT 训练成本极高

量化感知训练 (QAT) 需全程梯度参与，对百亿模型而言计算与时间代价难以承受，PTQ成为工业界首选。

离群值 (Outliers)

网络激活中存在极端离群值，主导了量化网格的划分，导致大量正常特征被严重截断抹除。

低比特精度坍塌

传统 PTQ 方法在 8-bit、4-bit 尚可，但在 3-bit 等极低比特下，量化网格极其稀疏，导致精度断崖式下跌。

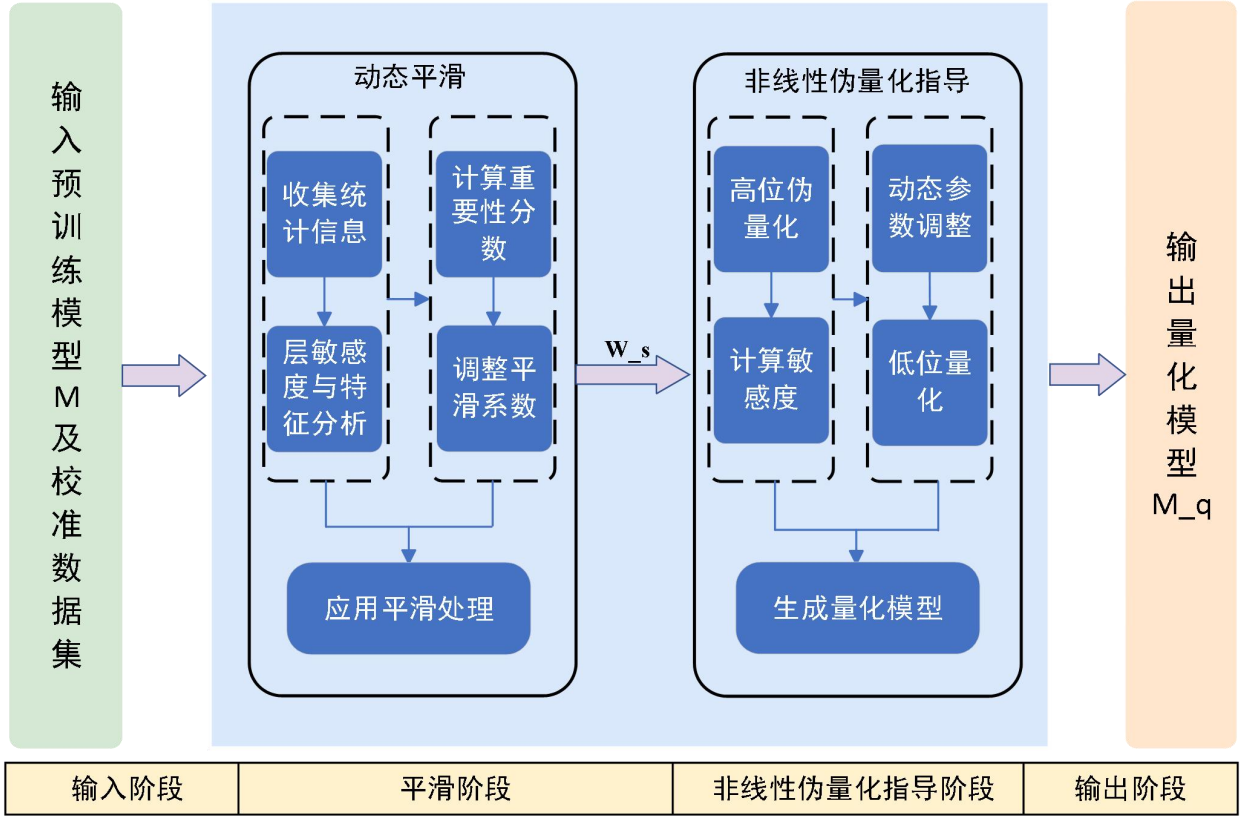
03 SGPTQ方法动机：模块异构敏感度

现有方法的盲区：使用“一刀切”的全局统一平滑或量化策略。

- 模型内部各层（如 Self-Attention 层与 FFN 层）对量化误差的容忍度存在显著异质性。
- 自注意力层：由于 Softmax 的非线性指数放大特性，对权重矩阵的微小量化扰动极度敏感。
- FFN 层：参数占比大，但结构性离群值更为集中。
- 忽略这种结构性异质，会导致误差在敏感层级联放大，引发全网络性能崩溃。

本文主张：将量化策略与模块敏感度深度绑定，引入渐进式自适应机制 **(SGPTQ)**。

03 SGPTQ 总体框架：渐进式两阶段策略



- 收集少量校准数据并评估网络各层敏感度，基于敏感度分数动态调整平滑系数 α，将激活端的量化难度安全转移至权重端。
- 自注先执行高位宽伪量化建立误差基线。利用该基线指导极低比特 (3/4-bit) 的真实量化决策，动态调节分组大小与阻尼参数。

03 核心机制一：模块敏感度驱动的自适应平滑

该机制的具体实现分为以下三个阶段：

- 收集统计信息，为量化过程中的敏感度分析提供基础。针对模型的第l层，计算激活值绝对值均值。其中，$\mu_{act,l} = \frac{1}{N}\sum_{i=1}^{N}|a_{l,i}|$ 表示第l层的第i个激活值，N为激活值样本数。
- 引入多维度重要性评分机制，包括激活值重要性和量化误差影响。$I_{a,l} = \frac{\mu_{act,l}}{max_k\ \mu_{act,k}}$，$E_{imp} = \frac{\|\nabla wL\|_{2,l}}{\max_k \|\nabla wL\| k}$ 其中，$\|\nabla wL\|_{2,l}$ 表示第l层权重的梯度L2范数，反映量化误差对模型损失的潜在影响。综合上述指标，计算第l层的综合重要性分数：$I_l = \alpha I_{a,l} + \beta E_{imp}$ 。
- 自适应平滑系数的动态调控。获取综合重要性分数后，通过Sigmoid函数将其非线性映射至区间，得到基础平滑系数：$\lambda_{l,base} = \frac{1}{1+e^{-I_l}}$ 该公式通过Sigmoid函数将重要性分数映射到(0,1)区间。为了进一步适应层的统计特性，引入自适应调整：$\lambda_l = \lambda_{l,base} \times (1 + \delta \cdot \frac{\sigma_{w,l}^2}{\epsilon + \mu_{act,l}})$ 。
- 其中，$\sigma_{w,l}^2$为该层权重的方差，δ 为调节参数，ϵ为数值稳定性常数。

03 核心机制二：非线性伪量化指导

目的：解决直接执行极低比特（3-bit）量化时的“精度休克”问题。

A. 高位探路

在完成自适应平滑后，不直接量化为3-bit，而是先进行 8-bit 的轻度“伪量化”。

B. 提取基线

记录高位量化下的权重通道扰动分布，计算各通道的误差基线，建立重要性图谱。

C. 降维指导

利用该图谱作为指导因子，动态控制后续 3/4-bit 真实量化时的量化步长。

核心思想：用高位量化的“廉价代理”估计精度风险，再用风险信息指导低位决策，实现渐进式保全

03 核心机制二：非线性伪量化指导

该机制的具体实现分为以下两个阶段：

- 收对权重w进行高位量化，以获取初始量化权重*wint*。同时，引入了一种基于激活值梯度的敏感度指标Sensi，用于衡量权重对模型输出的影响：$Sens_i = \left| \frac{\partial L}{\partial w_i} \cdot std(A_i) \right|$ 其中，L为损失，$\frac{\partial L}{\partial w_i}$为权重wi的梯度，*std(Ai)*为对应激活值的标准差。
- 利用高位量化计算得到的敏感度Sensi，我们设计了一种动态量化区间分配策略，将权重进一步降至低位。具体而言，根据敏感度大小动态调整量化步长*Δi*：$\Delta_i = \frac{Sens_i}{\Sigma_i Sens_i} \cdot (W_{max} - W_{min}) \cdot \frac{1}{2^{b_{final}}}$

04 实验验证：环境与评测基准

验证模型族	OPT 系列 (125M, 350M, 1.3B, 2.7B, 6.7B, 13B) BLOOM 多语言系列 (560M, 1.1B, 1.7B, 3B, 7.1B)
评估数据集	语言建模任务：WikiText2, PTB 零样本推理 (Zero-shot)：PIQA, ARC-Easy
对比基准算法	Full (FP16全精度) RTN (Round-to-Nearest 近似量化) GPTQ
核心测试位宽	3-bit 与 4-bit 权重极低比特压缩场景

04 实验结果：语言建模困惑度 (WikiText2)

不同量化方法在 **WikiText2** 数据集上的 **4-bit和3-bit** 困惑度 **(PPL)** 对比（数值越低越好）

量化方案	量化位宽	OPT-1.3B	OPT-2.7B	OPT-6.7B	OPT-13B
FP16	4-bit	14.63	12.47	10.86	10.13
RTN		48.17	16.92	12.10	11.32
GPTQ		15.47	12.87	11.39	10.31
SGPTQ		15.23	12.69	11.01	10.29
RTN	3-bit	1.3e4	1.6e4	5.8e3	3.4e3
GPTQ		20.97	16.88	14.86	11.61
SGPTQ		19.93	15.89	12.25	11.51

- RTN 在 3-bit 极低比特下发生全面崩溃（PPL 达数万）。
- SGPTQ 全面超越 GPTQ，特别是在 OPT-6.7B 上，SGPTQ 将 PPL 从 14.86 降至 12.25（↓2.61）。

04 实验结果：语言建模困惑度 (PTB)

不同量化方法在 PTB 数据集上的 3-bit 与 4-bit 困惑度 (PPL) 对比

量化方案	量化位宽	OPT-1.3B	OPT-2.7B	OPT-6.7B	OPT-13B
FP16	4-bit	16.96	15.11	13.08	12.34
RTN		57.30	31.05	18.84	16.51
GPTQ		21.85	19.14	16.56	14.94
SGPTQ		17.85	15.74	13.39	12.53
RTN	3-bit	1.3e4	1.4e4	5.7e3	2.8e3
GPTQ		32.10	24.81	21.88	16.68
SGPTQ		23.41	18.65	15.01	13.61

- 在更具挑战性的 PTB 数据集上，SGPTQ 的优势被进一步放大。
- 4-bit 下 SGPTQ在OPT-6.7B (13.39) 几乎逼近 FP16 全精度水平 (13.08)，实现了高度无损压缩。

04 实验结果：常识推理任务 (Zero-shot)

OPT系列模型在 PIQA 任务上的 4-bit 零样本准确率

量化方案	量化位宽	OPT-1.3B	OPT-2.7B	OPT-6.7B	OPT-13B
FP16	4-bit	72.36%	74.81%	76.39%	76.88%
RTN		67.63%	73.72%	76.44%	76.01%
GPTQ		70.73%	73.99%	76.28%	76.61%
SGPTQ		70.78%	73.61%	76.27%	76.44%
RTN	3-bit	52.77%	51.90%	50.49%	52.99%
GPTQ		68.34%	71.38%	73.29%	75.24%
SGPTQ		68.49%	71.59%	74.59%	75.95%

- RTN 的表现（~50%）完全退化为随机猜测水平。
- 在 OPT-6.7B 上，SGPTQ 的 PIQA 准确率 (74.59%) 相比 GPTQ 提升了约 1.3%，最大化保留了模型的实际理解能力。

04 实验结果：常识推理任务 (Zero-shot)

OPT系列模型在 Arc_easy 任务上的 4-bit 零样本准确率

量化方案	量化位宽	OPT-1.3B	OPT-2.7B	OPT-6.7B	OPT-13B
FP16	4-bit	50.93%	54.34%	60.14%	61.83%
RTN		49.20%	52.90%	57.68%	61.31%
GPTQ		59.97%	53.11%	59.72%	61.32%
SGPTQ		49.70%	54.29%	59.88%	61.70%
RTN	3-bit	27.97%	26.05%	25.04%	30.60%
GPTQ		56.17%	48.19%	53.41%	56.82%
SGPTQ		46.33%	49.83%	56.18%	59.73%

05 SGPTQ 泛化性与跨架构适用性讨论

- SGPTQ 不仅在 OPT 和 BLOOM 架构上表现优异，其核心模块同样适用于 LLaMA, Qwen 等主流架构。
- 算法无需重新训练网络结构，具备即插即用 (Plug-and-Play) 的高度工程友好性。
- 通过超参数 α 和 阻尼的微调，可以快速适配不同国产硬件（如海光 DCU、昇腾 NPU）的量化特性规范。

05 全文总结

贡献一	首次提出模块敏感度驱动的自适应平滑策略，成功将量化难度重分配。
贡献二	引入非线性伪量化指导机制，用廉价的高位基线拦截了3-bit低位崩溃风险。
贡献三	广泛实验验证：在OPT等主流模型上，SGPTQ实现了4倍显存压缩与精度的高度保全。

汇报完毕，感谢聆听！

敬请各位老师、同学批评指正