# 论文1: ZeroQuant: Efficient and Affordable Post-Training Quantization for Large-Scale Transformers (NeurIPS)

## 1 研究背景：

论文针对LLM异常值的量化难题，提出了一种经济有效的PTQ方法ZeroQuant。

## 2 研究问题：

以往的量化方法很少对transformer-based的LLM做针对优化，导致量化效果不好，由于异常值的存在，简单的PTQ方法量化损失大。只量化weight的方法无法减少计算时间改善推理延迟。

## 3 研究层次：

* ZeroQuant包括三个主要部分：细粒度的硬件友好量化方案、一种新颖的低成本逐层知识蒸馏算法，以及高度优化的量化系统后端支持。
* Weight-activation 都量化。

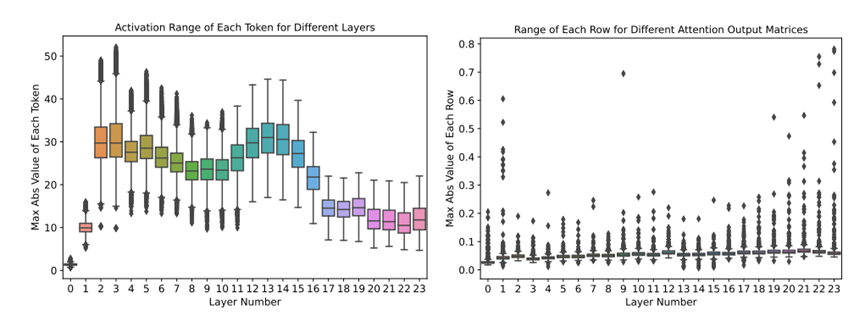
## 4 研究方向：

LLM+量化+蒸馏 属于当前NLP热点话题

## 5.研究差异：

* 同：都集中在LLM-PTQ层面。
* 异：量化粒度切的很细，使用了蒸馏算法来做量化补偿。

## 6 挑战：

* 由于 INT8 激活量化，准确性显着下降，尤其是在 Wikitext-2 等生成任务中，模型无法生成权重精度降低的有意义的文本。
* 不同Token的激活范围存在差异，使固定量化范围的量化变得复杂并影响预测准确性。
* 
* 注意力输出矩阵内weight数值范围的差异，导致 INT8 权重 PTQ 的性能较差，并且 INT4 量化更加复杂。
* 超低精度量化的知识蒸馏的训练成本很高，需要更高效的PTQ-蒸馏方法。

## 7 文章提出的解决方案:

* 激活采用per-token动态量化，权重采用per-group静态量化，group采用48或64。
* 针对 INT4/INT8 混合精度量化提出了一种新颖的逐层知识蒸馏法（LKD），通过蒸馏法对神经网络进行逐层量化，迭代次数最少，甚至无需访问原始训练数据。因此，在任意时刻，设备内存里都只有单个transformer层，这使得在有限的训练预算和 GPU 设备条件下进行十亿规模的模型蒸馏成为可行。

# 论文2: EasyQuant: An Efficient Data-free Quantization Algorithm for LLMs (EMNLP 2023)

## 1 研究背景：

大多数以前的PTQ工作中，量化模型是要使用训练数据中的一些样本进行校准的，这可能会影响量化 LLM 对未知案例和任务的泛化。因此，在这项工作中，作者探讨了一个重要的问题：能否为LLM设计一种无数据的量化方法来保证其泛化性能？

## 2 研究问题:

如何解决在不依赖训练数据的情况下高效量化 LLM 的难题，从而确保量化模型的泛化。

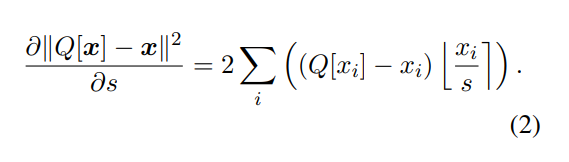
## 3 研究层次:

Weight-only量化

## 4 研究方向:

LLM+量化+Zero shot

## 5 研究差异：

* 同：大模型的weight量化与校准，方法与LLM.int8()相同。
* 异：作者实验证明使用校准集的量化方式可能会损害模型的泛化性能，所以校准时不使用校准集，而是把scale当做可训练的参数。
* 

## 6 挑战:

* 以前的无损模型量化算法需要对量化后的模型进行重新训练，这对于超过数十亿参数的模型来说成本太高。
* 以前的模型通常是为特定领域的任务设计的，这意味着训练数据是从有限的任务领域中采样的。最近的 LLM 通常是在不同领域的数据语料库中进行训练的，而且它们已被证明对多领域零点任务相当有效。在这种情况下，如果我们只使用部分领域的语料重新训练量化的 LLM，LLM 的泛化能力可能会变差。

## 7 文章提出的解决方案:

* 隔离离群值，将离群值保留原始精度不量化，其余值做量化。
* 构建了一个与量化前后的值，以及scale有关的误差重建函数，并证明这个函数是可微的，于是其余值做量化时不再校准，而是将scale当做可训练的参数，不断迭代训练寻找最合适的scale。初始scale由minmax量化确定。

# 论文3: SqueezeLLM: Dense-and-Sparse Quantization (23年6月挂arXiv)

## 1 研究背景：

LLM需要数十亿参数大量内存，参数中存在重要异常值。之前的量化方案往往会导致性能显著下降。不实用，需要一个硬件友好的量化方案保证实际性能。

## 2 研究问题:

LLM-PTQ的超低精度量化如何最小化量化损失并在实际硬件上高效运行。

## 3 研究层次:

Weight-only量化

## 4 研究方向:

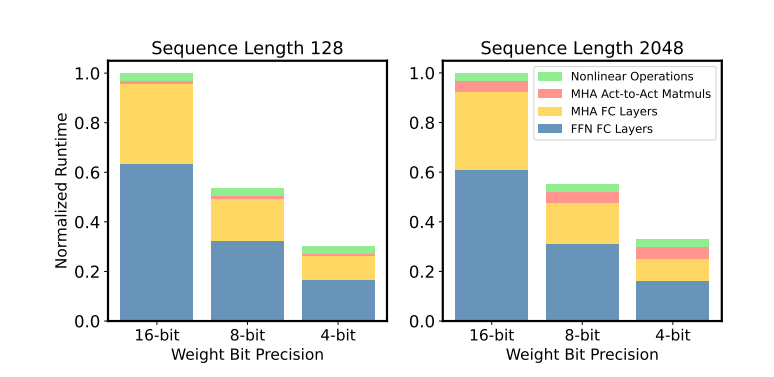
LLM+量化+稀疏加速

## 5 研究差异：

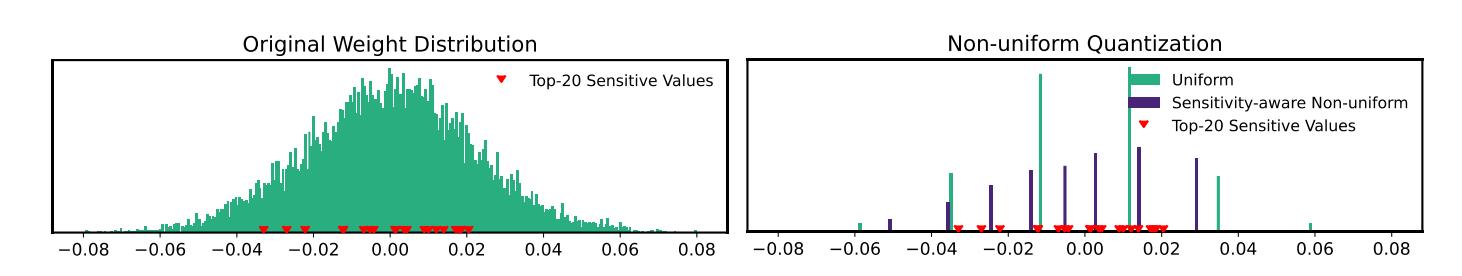
* 同：和LLM.INT8()/EasyQuant一脉相承的分离离群值策略，离群值以高精度保存。
* 异：1）做了基于量化敏感的非均匀量化，2）对分离的离群值矩阵做了稀疏加速。

## 6 挑战:

1. 由于资源要求很高，部署LLM模型进行推理一直是一项重大挑战。传统的量化方案大都对实际部署不友好，无法高效计算。
2. 文章主要还是在研究weight-only的量化方法，在问题的提出上没有新的视角，但本文提出了一个新的论点并进行了证明：**生成任务的LLM推理的主要性能瓶颈是内存带宽而不是计算**。这意味着加载和存储参数的速度成为内存限制问题的主要延迟瓶颈，而不是计算。

* 

## 7 文章提出的解决方案:

* 证明了内存（而不是计算）是生成任务的 LLM 推理的主要瓶颈。
* 基于1提出了一个weight-only的量化方法，将weight中的异常值单独提取出来，最终有两个矩阵，一个矩阵没有异常值，一个矩阵只有异常值。称为Dense-and-Sparse量化。
* LLM的最佳量化方案不是均匀量化而是**非均匀量化**，因为LLM权重分布不均匀。于是量化时采用了非均匀量化。
* 
* 对于将异常值分开导致计算慢的问题，作者对异常值矩阵的计算做了稀疏加速从而实现了更小的推理开销，以便在实际硬件上高效计算。

# 论文4: Outlier Suppression+: Accurate quantization of large language models by equivalent and effective shifting and scaling (EMNLP 2023)

## 1 研究背景：

由于激活中存在有害的异常值，Transformer-based模型的PTQ面临着重大挑战。作者观察到这些异常值集中在特定通道中，并且在通道之间不对称。

## 2 研究问题:

LLM-PTQ存在异常值，异常值如何处理能够使量化误差最小。

## 3 研究层次:

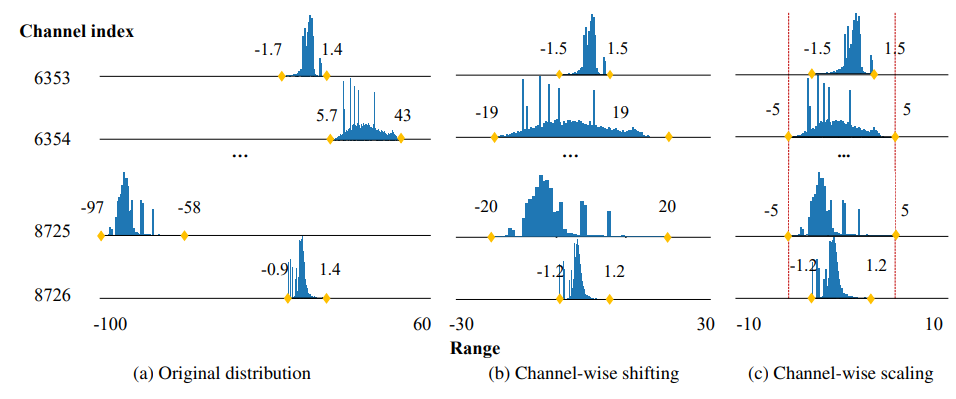
Weight-only量化

## 4 研究方向:

LLM+量化

## 5 研究差异：

* 同：还是基于大模型数值本身固有离群值上做文章。
* 异：本文没有回避异常值的处理，对异常值做了移位和缩放的操作。将量化范围压缩回来。

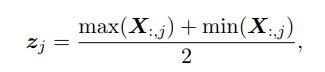


## 6 挑战:

1.对于异常值的观察：大模型的量化难点主要在于大模型的异常值非常重要，直接阶段会对性能造成很大损害，本文深入观察了异常值的分布特征，发现异常值在不同channel中分布范围不一样，所以非per-channel的量化损失是很大的。

2. 以前有工作采用激活量化来缩放异常值，并将缩放值迁移到后续权重以实现 FP 等价。然而，前者可能会损害量化加速效果，而后者在确定缩放值时没有考虑最小化迁移和量化引入的变化。

## 7 文章提出的解决方案:

* 基于对离群值分布的观察，提出了一个离群值移位缩放处理方法，per-channel的处理每个tensor，处理方式为：
* a. 移位：根据值分布范围将所有数减去一个值z，z的选取是计算最大最小值的平均值得到的，使数值分布关于0对称：
* fig:
* 
  1. 缩放：异常值还可能存在集中现象，也就是异常值集中出现在特定channel中，所以还需要做一次per-channel的缩放。
  + fig:

# 论文5: FPTQ: FINE-GRAINED POST-TRAINING QUANTIZATION FOR LARGE LANGUAGE MODELS (2023年9月挂arXiv)

## 1 研究背景：

现有量化研究多几种在单种量化方法的研究，如W8A8,W4A16等，而少有研究这些方法的联合，探索两种量化方法 W8A8 和 W4A16 组合的研究明显缺乏。本文通过提出一种创新的细粒度训练后量化（称为 FPTQ）方法，该方法结合了两者的优点。

## 2 研究问题:

W8A8与W4A16方法的结合。

## 3 研究层次:

Weight-activation量化

## 4 研究方向:

LLM+量化

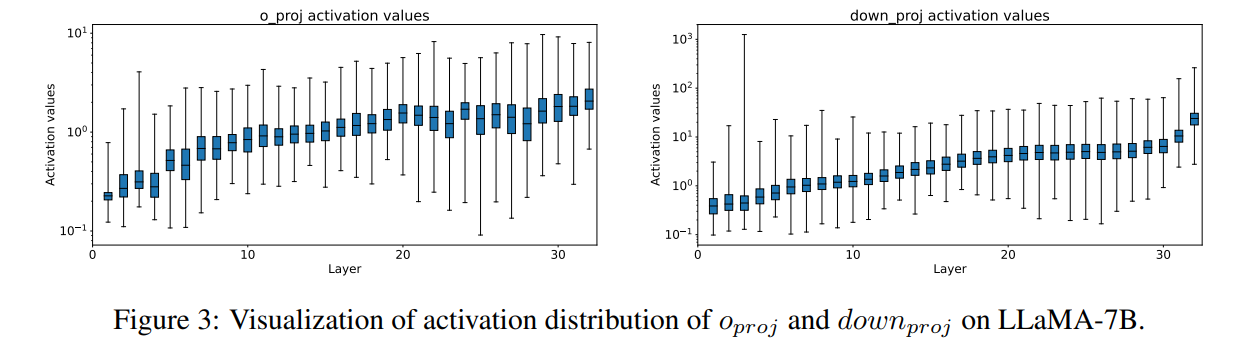
## 5 研究差异：

* 同：还是基于现有的方法做LLMs的量化。
* 异：相对于以往的方法，作者从推理生成不同阶段这个角度进行量化方案的选择。

## 6 挑战:

1. LLM 的生成推理可以分为两个阶段：1)prompt解码阶段：根据输入提示生成输出token；2）自解码阶段，用前一个token迭代预测下一个token。前者由于第一轮计算较长的输入序列而受到计算限制，而后者由于顺序处理而受到内存限制，因此需要两种不同的量化实现。

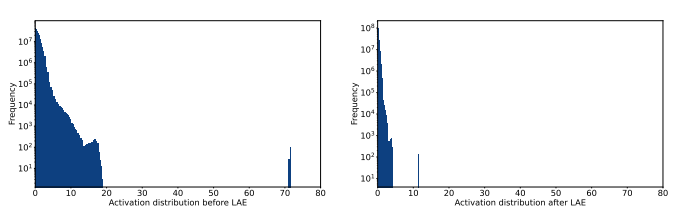
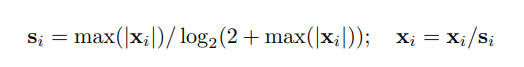
2.如下图所示，某些层的输入激活值的最大波动范围为数十到数千。使用per-tensor静态量化将导致显著的量化误差，但对所有层使用per-token动态量化将无法带来足够的硬件加速。因此，需要per-layer的策略来确定量化的粒度。



## 7 文章提出的解决方案:

* **高性能低成本的W4A8压缩**：首个实现大型语言模型的高性能W4A8（INT4权重和INT8激活）PTQ的研究。算法流程为：
* **校准**：使用预定义的数据集对预训练的LLM进行校准。   
  **激活分析**：进行激活分布的分析。   
  **逐层处理**：针对Transformer的每个层，根据激活范围确定激活量化策略:
  + 1)为激活范围≤阈值v0的层设置静态的per-tensor量化，
  + 2)对激活范围在v0和v1之间的层执行对数激活均衡并设置静态per-tensor量化，
  + 3)为激活范围超过v1的层设置动态per-token量化。

**权重量化**：为每个层的权重实施细粒度量化策略。   
**更新模型**：根据选定的量化策略更新LLM的权重和激活。   
**输出**：最终产出一个高性能的量化LLM。

* **新颖的量化方案**：基于对LLMs激活分布的综合分析，采用了per-layer策略来应对不同的量化难度。作者还设计了一种离线对数激活均衡方法，使先前难以处理的层获得了量化友好的分布。
* 
* 所谓对数激活均衡，即对tensor中的每个值执行如下操作：
* 
* **适合推理**：此方法同时考虑了内存和计算效率，可以在执行INT8推理的同时将权重存储为4bit格式，从而促进了内存访问和计算的加速。