

基于敏感度分析的后训练式对话摘要 混合精度量化

刘宇鹏, 张禹豪, 孟 鑫

(哈尔滨理工大学 计算机科学与技术学院, 哈尔滨 150080)

摘 要: 本文提出了一种后训练对话摘要模型上层级混合精度整数量化方法, 采用较为精确的基于增广 Hessian 矩阵方法评估层敏感度, 考虑层间相关性的同时可以有效保留层内信息。由于对话摘要模型激活中存在大量离群值, 直接对其量化会导致模型精度大幅下降。对此, 本文又提出了一种先平滑, 后混合精度量化的方法。首先使用平滑因子将激活离群值平移到权重, 再根据权重和激活根据敏感度评估得分进行混合精度量化。该方法可以将模型由 16 bit 或 32 bit 浮点数量化为 4~16 bit 混合精度, 将模型存储需求降低, 同时加快推理速度。在基准数据集 SAMSum 上, 与经典基线系统比较取得了标志性性能提高, 几乎和不量化模型性能持平。

关键词: 敏感度; 混合精度; 离群值; 对话式摘要

中图分类号: TP312 **文献标志码:** A **文章编号:** 1671-5497(2026)03-0793-09

DOI: 10.13229/j.cnki.jdxbgxb.20240809

Mixed-precision quantization of post-trained conversation summarization based on sensitivity analysis

LIU Yu-peng, ZHANG Yu-hao, MENG Xin

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract: This paper proposes a mixed precision integer quantization method for the layer of a trained dialogue summarization model, which uses a more accurate method based on augmented Hessian matrix to evaluate layer sensitivity, considering inter-layer correlation while effectively preserving intra-layer information. Due to the large number of outliers in the conversation summarization model activation, quantifying them directly will result in significant decrease in model accuracy. In this paper, we propose a method of smoothing first and then mixed precision quantization. Activation outliers are first translated to weights using smoothing factors, and then weights and activations are quantified with mixed precision based on sensitivity assessment scores. This method can quantize the model from 16 bit or 32 bit floating-point to 4-16 bit mixed precision, which reduces the storage requirement of the model and accelerates the inference speed. On the benchmark dataset SAMSum, a significant performance improvement is achieved

收稿日期: 2024-07-19.

基金项目: 国家自然科学基金青年基金项目(61300115); 中国博士后科学基金项目(2014m561331); 黑龙江省教育厅科技攻关项目(12521073).

作者简介: 刘宇鹏(1978-), 男, 教授, 博士. 研究方向: 自然语言处理, 机器翻译, 医疗对话系统, 智慧医疗, 推荐系统, 量子计算, 情感分析. E-mail: flyeaglelyp@hrbust.edu.cn

compared with the classical baseline system, which is almost the same as the performance of the non-quantized model.

Key words: sensitivity; mixed precision; outliers; conversation summarization

0 引言

模型量化是通过缩小模型参数表示位宽来减少计算和存储需求。根据量化模型的不同部分,将量化分为权重量化^[1]和激活量化^[2]。权重量化通过降低模型精度来减小模型尺寸,从而降低计算和存储成本。激活量化则旨在将模型的激活值缩小至较小数据类型中,以降低计算和内存与寄存器之间的通信成本,可以通过同时量化权重和激活达到模型压缩效果。

量化感知训练(Quantization-aware training, QAT)^[3,4]可以训练出具有较高量化精度的模型。然而,这种方法通过更新模型参数和量化后重新验证,使QAT在模型量化过程中需要相当大算力和人力开销。后训练量化方法(Post-training quantization, PTQ)是在训练完成后对已有模型进行量化的方法。虽然PTQ相较于QAT节省了算力和人力成本,但该方法存在以下重要问题需要解决:①将对话式摘要模型由高位浮点数映射至低位整数时,会带来模型精度的大幅下降;量化后模型精度远不如未量化模型;②在对话式摘要模型的激活中经常出现具有大量较大幅值的离群值,这会导致在量化激活时产生较大误差。本文提出了一种基于PTQ的硬件友好的混合精度量化方法,针对对话式摘要模型中激活存在的离群值问题,提出了一种将离群值量化困难转移到权重上的统一方法。由于Hessian矩阵可以表示损失函数局部曲率,因此,使用Hessian矩阵作为单独层敏感度评估,但单独使用Hessian矩阵没有考虑层间关联性,使其敏感度评估存在误差。本文提出的AugHessian可以有效缓解该问题,提高了混合精度量化精度。使用二分搜索方法高效确定混合精度量化配置,以解决配置空间搜索困难的问题。创新点如下:

(1)提出了一种激活离群值平滑方法,结合了基于通道和张量平滑的优点以有效缓解离群值问题,也更有利于下游设备计算。

(2)提出了一种增广Hessian矩阵考虑层间相关性来评估模型层敏感度方法以提高混合精度整

数量化精度。

(3)提出了一种使用二分搜索的方法来确定混合精度量化配置。这种二分搜索方法仅需 $O(b\log N)$ 复杂度就可以确定模型量化配置,节省了大量搜索时间。

1 相关工作

1.1 QAT量化

QAT量化是在训练过程中考虑量化误差的方法,通过在训练过程中对权重和激活值进行仿真量化,并将量化误差作为损失函数的一部分,使模型能够适应量化后的精度下降。对于QAT量化方法,为了找到用于QAT混合精度的量化策略,Wang等^[5]使用强化学习方法从硬件加速器获得反馈。与具有相当准确率的八位整数模型相比,延迟提高了1.4~1.95倍,功耗降低了约47.4%。基于梯度的QAT在学习权重和激活上取得了相当大成功,多个模型在低于5 MB的模型上取得了很好结果^[6,7]。当数值精度无法直接学习时,替代方法通常使用代理度量来确定层重要性,并分配相应精度。Yao等^[8]提出使用Hessian迹均值来确定层重要性,为所有模型量化配置搜索帕累托边界,将ResNet50模型大小缩小到7.99 MB,同时实现了75.76%的准确性。

1.2 PTQ量化

最近的工作主要研究了将混合精度量化应用于PTQ。Nahshan等^[9]研究了量化对模型损失的影响,观察到对于高位宽量化,存在平坦可分离结构,而对于低位宽量化,则存在高度不可分离,陡峭曲率结构。设计了三步法:①确定最小化各层量化误差范数的量化步长;②使用二次插值来近似最佳量化比例;③通过应用无梯度优化法在前一步获得的所有层参数上进行联合优化。类似地,Nagel等^[10]在理论上分析了量化过程中的舍入影响,并将其构建为二进制优化问题(向上取整对比向下取整)。他们的解决方案使用层次局部损失,采用松弛法进行优化。Yao等^[11]对基于Transformer的模型进行4 bit和8 bit整数量化,通过使用细粒度量化和基于层次数据独立知识蒸馏

(Knowledge distillation, KT), 对模型进行轻量化。Cai等^[12]引入一种混合精度量化方案, 采用了类似于先前QAT方法^[8]的Hessian估计。为了估计Hessian, 使用批归一化匹配从未量化模型中提取一个精简数据集, 使这种方法不能用于基于Transformer的模型。Hubara等^[13]通过更新模型参数以最小化量化层输出与全精度输出间的误差, 并微调批归一化参数来量化模型。将分配在每层上的精度作为线性规划问题进行了建模, 成本是估计模型的占用空间和准确性函数。该方法假设层间具有较强独立性, 并更改模型权重以及批归一化参数, 模糊了量化感知训练和后训练量化间的区别, 这种整数规划方法也被其他研究所采用^[14]。其他研究者研究了层敏感性替代方法, 如信噪比^[15]和Fisher信息^[16]。这些度量以类似方式使用, 构建一个按层敏感性排序的列表, 以便进行优化的混合精度量化配置搜索。Zheng等^[17]将量化过程表述为离散变量的大规模网络组合优化问题, 并通过各种正则化实现了有效解决方案。

2 基于后训练模型的混合精度量化方法

2.1 权重和激活平滑

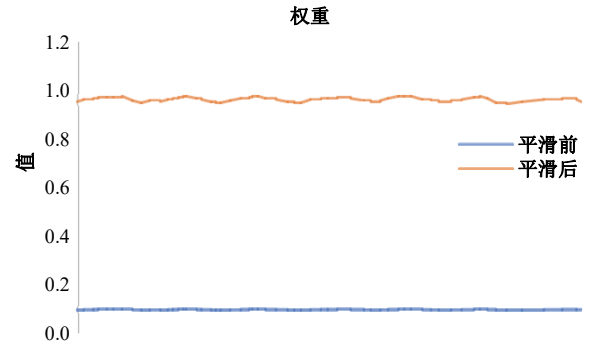
经过平滑工作, 权重值分布变化见图1(a), 激活值分布变化见图1(b)。对话式摘要激活中存在很多离群值, 这导致量化激活变得很困难, 在很多研究中都有体现^[1, 18, 19], 对话式摘要模型的特点如下:

(1) 权重分布相当均匀且平坦, 因此, 更容易量化。先前研究表明: 用8 bit整数甚至4 bit整数量化模型权重不会降低准确性^[1, 11, 20]。

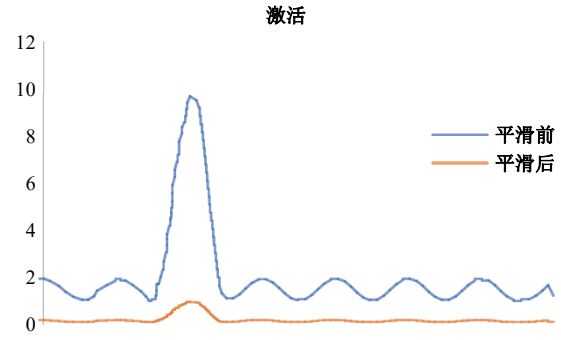
(2) 离群值使激活量化变得困难。离群值范围比大多数激活数值大约100倍, 但离群值一般只存在于特定通道, 且存在离群值的通道中会出现大量离群值。这在以张量为单位量化的情况下, 大离群值主导了量化映射范围。对于没有离群值的正常通道来说, 有效量化级别会非常小, 从而导致大量正常权重被映射到错误值, 造成较大量化误差。

经过平滑后, 将模型激活和权重值都限制在较小波动范围之间(0, 1), 显著降低了激活量化难度, 同时可以带来更好量化效果。

由于离群值只存在于部分特定通道, 其他研



(a) 权重平滑前后对比



(b) 激活平滑前后对比

图1 量化前后对比

Fig. 1 Performance comparison before and after quantization

究者使用对激活进行每通道(per-channel)量化^[19](即每个通道使用不同量化步骤)方法, 这种量化方法与每张量(per-tensor)量化相比, 误差更小。

然而, 每通道激活量化并不适用于硬件加速的通用矩阵乘法(General matrix multiply, GEMM)计算内核, GEMM依赖以高吞吐量执行的一系列操作(例如矩阵乘积累加), 且不允许在该序列中插入吞吐量较低的指令。在GEMM内核中, 缩放只能沿着矩阵乘法的外部维度进行, 公式如下:

$$Y = \text{diag}(\Delta_X^{\text{INT8}}) \cdot (X^{\text{INT8}} \cdot W^{\text{INT8}}) \cdot \text{diag}(\Delta_W^{\text{INT8}}) \quad (1)$$

因此, 这种使用每通道量化的方法虽然性能上优于每张量量化方法, 同时能一定程度解决离群值难以量化的问题, 但是这种方法不能适配硬件, 这与想做一个下游设备友好的轻量化模型相悖。

于是, 本文设计了通道和张量相结合的平滑方法, 该方法可以缓解离群值问题, 同时对下游设备更加友好。具体地, 通过将输入激活除以张量上的平滑因子 $s \in \mathbb{R}^C$ 来平滑输入激活。为保持线性层的数学等价性, 相应地在相反方向上按比

例缩放权重:

$$Y = (X \text{diag}(s)^{-1}) \cdot (\text{diag}(s)W) = \hat{X}\hat{W} \quad (2)$$

考虑到输入 X 通常是由先前的线性操作(例如线性层、层归一化等)生成的,可以轻松地将平滑因子融入先前层参数中,而不需要额外缩放内核调用开销。当输入来自残差相加时,在残差分支^[18]中添加额外缩放。

该方法的主要难点是如何确定每个通道的平滑因子 s , 使 $\hat{X} = X \text{diag}(s)^{-1}$ 易于量化。为了减小量化误差,应该增加所有通道的有效量化位数,当所有通道具有相同最大值时,总有效量化位数将是最大的。因此,简单想法是 $s_j = \max(|X_j|)$, $j = 1, 2, 3, \dots, C_i$, 这种方法能保证所有激活通道都使用一个共同最大值,这样就容易进行量化。

但是,激活范围是动态的,对于不同的输入样本,它会有所变化。在这里,本章使用少量校准样本来估计激活通道的尺度。然而,该方法将所有量化困难都转移到了权重上。本文发现,在这种情况下,权重的量化误差会很大(离群值通道现在迁移到了权重上),导致了较大的

准确性下降。另外,也可以通过选择 $s_j = \max(|W_j|)$, 这样就将所有的量化困难从权重转移到激活中。同样,由于激活的量化误差,导致模型性能较差。

因此,需要在权重和激活之间平衡分配问题,以便它们都容易进行量化。因此,引入了一个超参数 α 来平衡权重和激活之间的数值迁移,公式如下:

$$s_j = \max(|X_j|)^\alpha / \max(|W_j|)^{1-\alpha} \quad (3)$$

其中: α 值越大表示将更多激活量化难度转移到了权重上, α 值越小表示将更多权重量化难度转移到了激活上,该公式确保了相应通道权重和激活共享最大值的比例关系。

对于一些其他模型,其中激活离群值更显著(有约 30% 离群值),这对激活量化来说更加困难^[20], 可以选择一个更大 α 来将更多量化困难转移到权重上。

2.2 混合精度量化

本节主要分为两部分,分别是敏感度分析和通过二分搜索确定混合精度量化配置,并最终生成量化后的混合精度模型,结构见图 2。

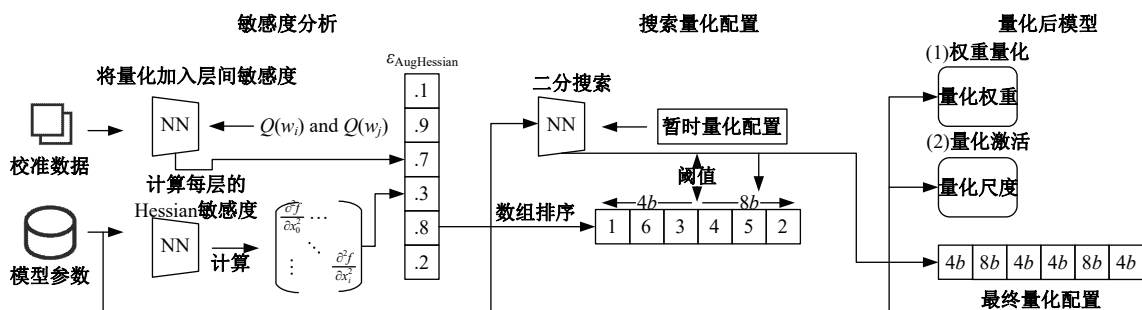


图 2 混合精度量化方法结构图

Fig. 2 Structure figure of the hybrid-precision quantization method

采用固定点量化(Fixed point quantization)方法(也被称为整数量化),通过对原始浮点数值应用截断和舍入操作来实现,表示为:

$$Q(x) = \text{round}(\text{clip}(\alpha \cdot x) \cdot 2^{b-1}) \cdot 2^{-b-1} \cdot \alpha^{-1} \quad (4)$$

式中: Q 为量化函数; round 为舍入函数; clip 为截断操作,将超过阈值的值映射到对应极值(最大值 1 和最小值 -1); x 为输入浮点数; α 为量化比例; b 为量化位宽。

量化后为了确保与大多数下游设备硬件兼容,强制要求矩阵乘法中的所有操作数(激活和权重)具有相同 bit 精度。对于权重,采用细粒度量化的,即针对每个张量维度(如每个通道、每个过滤

器或每个嵌入)确定舍入函数和比例参数。对于权的缩放比例 α , 基于张量维度上观察到最小值和最大值进行设置。对于激活,使用校准集在单次前向传播来确定激活单一比例,然后采用基于百分位方法^[21]在每层上确定激活量化比例。

对于敏感度度量,已有许多研究^[14-17]使用 Hessian 矩阵来计算层敏感度。Hessian 矩阵可以表示函数局部曲率,模型准确性对于占据损失函数平坦区域(局部曲率低)值的扰动是健壮的,对于占据局部曲率高(陡峭)区域的值,小扰动可能对模型准确性产生巨大影响。但由于 Hessian 矩阵是权重二阶偏导矩阵,其具有难以计算和巨大

存储空间的特点。

使用矩阵迹替代整体矩阵作为层评估,并通过使用 Hutchinson 算法来更快速地估计矩阵迹。对于第 i 层定义了基于 Hessian 的敏感度度量,公式如下:

$$\mathcal{E}_i^{\text{Hessian}} = \mathbb{E} \left[\text{tr} \left(\frac{L(x, \mathbb{W})}{\partial w_i^2} \right) \right] \quad (5)$$

式中: tr 为迹操作符; L 为模型损失函数; $\mathbb{W}$ 为所有考虑的张量集合(例如,权重/激活); x 为校准数据。

较高的 $\mathcal{E}_i^{\text{Hessian}}$ 值表示损失函数局部曲率增加,意味着模型对参数变化更为敏感。这样按 $\mathcal{E}_i^{\text{Hessian}}$ 排序就可以得到层量化敏感度排序。

基于式(5),通过计算获得了精确排序的层敏感度列表,并通过该列表确定不同层量化位宽。但在模型混合精度量化中,随着模型层数的增加,模型配置空间是呈指数级增加的,因此,需要一种高效搜索方法。

使用一种二分搜索方法来高效确定层量化配置。然而,使用 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度在两种模型上进行测试时,与二分搜索相比,逐步搜索算法结果要好得多。(逐步搜索算法通过逐个评估每层适用位宽,以模型退化程度累积作为分配标准)

为了分析出现该问题的原因,本节在两种模型上对比3种搜索方法,分别为:①随机打乱 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度列表后进行逐步搜索(Progressive random);②对 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度列表进行逐步搜索(Progressive hessian);③对 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度列表进行二分搜索(Bisection hessian)(见图3)。如果 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度评估正确,理论上来说,两种逐步搜索方法应该具有相同结果。逐步搜索和二分搜索的性能差异也是由于 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度评估错误造成的,这种错误导致二分搜索方法性能远远不如逐步搜索。

层间独立性导致 $\mathcal{E}_i^{\text{Hessian}}$ 敏感度并不能准确地表示模型层敏感度。基于这一点,本节以成对方式对层进行量化来估计层间依赖关系:

$$\begin{cases} \mathcal{E}_i^{\text{InterLayer}} = \sum_j L(x, \mathbb{W}^{i,j}) - \max[L(x, \mathbb{W}^i), L(x, \mathbb{W}^j)] \\ \mathbb{W}^{i,j} = \{ \mathbb{W} \setminus \{w_i, w_j\}, Q(w_i), Q(w_j) \} \# \end{cases} \quad (6)$$

对两层间相互作用引起的超额退化(为每层的联合量化损失与单层量化损失间损失 L 之差)

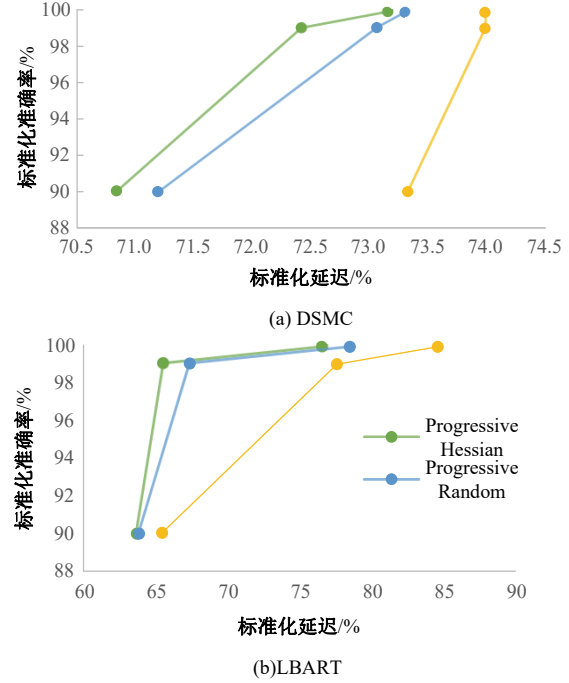


图3 3种搜索方案对比

Fig. 3 Comparison of three search schemes

求和。将求得的 $\mathcal{E}_i^{\text{InterLayer}}$ 最小值剪切为0,忽略任何负值,然后将 $\mathcal{E}_i^{\text{InterLayer}}$ 进行归一化和缩放,与 $\mathcal{E}_i^{\text{Hessian}}$ 结合,公式如下:

$$\mathcal{E}_i^{\text{AugHessian}} = \mathcal{E}_i^{\text{Hessian}} + \beta \mathcal{E}_i^{\text{InterLayer}}, \beta = \frac{\mathbb{E}[\mathcal{E}_i^{\text{Hessian}}]}{\mathbb{E}[\mathcal{E}_i^{\text{InterLayer}}]} \quad (7)$$

图4展示了LBART模型的 $\mathcal{E}_i^{\text{Hessian}}$ 、 $\mathcal{E}_i^{\text{AugHessian}}$ 和 $\mathcal{E}_i^{\text{InterLayer}}$ 3种敏感度对比,从图中可以发现,中间和最后层敏感度相对较高。

最后,在 $\mathcal{E}_i^{\text{AugHessian}}$ 敏感度排序列表上使用二分搜索方法反复进行搜索并确定量化配置。理论上,使用二分法仅需要 $O(b \log N)$ 次模型评估。其中, N 为总层数, b 为可用量化位宽数量。二分搜索会迭代更新阈值,从而更新量化配置,具体做法是根据是否达到精度目标来增加或者减少层量化位宽。逐步确定每个可用精度的敏感度阈值,从最高(如16位)到最低(如4位)。

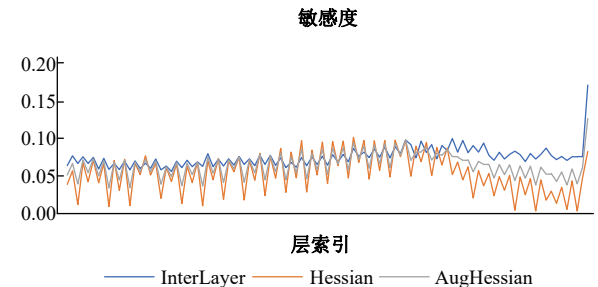


图4 敏感度对比

Fig. 4 Sensitivity comparison

3 实验

3.1 实验设置

本文共使用了两个模型,分别为 LBART 和 DSMC(见表 1)。使用的数据集为 SAMSum。在 3 个搜索精度上(99%、99.9% 和 99.99%)与其他研究方法对比 ROUGE-1(Recall-oriented under study for gisting evaluation)、ROUGE-2、模型大小和时延。对比模型见表 2,基线为没做量化的模型。在英伟达 3080 GPU 以不同量化范围对 GEMM 核心进行基准测试,使用推理批量大小 1 来估计时延。使用 CUTLASS^[28]分析器和优化器,确定了特定张量形状和精度。使用这些数据来估计不同量化模型部署时延。通过验证集将超参 α 设置为 0.5,这样可以均匀地分配量化困难。

表 1 实验模型介绍

Table 1 Introduction to the experiment model

模型名称	介 绍
LBART ^[22]	基于 BART ^[23] 的预训练模型,在对话摘要任务上微调模型,共 1 亿 4 千万参数
DSMC ^[24]	基于 Transformer 的非预训练模型,参数量约 7 千万

表 2 对比模型介绍

Table 2 Description the comparison models

量化方法	主要思想	量化位宽(A为激活,W为权重)
QBert ^[25]	基于分组的量化方案,使用二阶泰勒展开来评估权重	A8W8
MREM-P ^[3]	将模型分成多个模块,采用并行策略同时为每个模块最小化量化带来的重构误差	A8W8
BRECQ ^[4]	利用神经网络的基本构建块,逐个进行重构,同时在跨层依赖和泛化误差之间取得了良好平衡	A8W8
QDrop ^[26]	通过随机丢弃来实现激活量化,并首次将激活的量化推进到 2 bit	A4W8
MrBiQ ^[27]	利用多级二值化表示权重并允许不同数据表示激活	A8W8
AdaQuant ^[13]	通过分别优化每个层或块的参数,并使用校准集来设置激活动态范围,以最小化量化误差	A4W8

3.2 对比实验

通过图 5 和图 6 的分析可以发现,在将评分保持限制在 99% 时,提出的基于 $\mathcal{E}_{\text{AugHessian}}$ 的敏感度

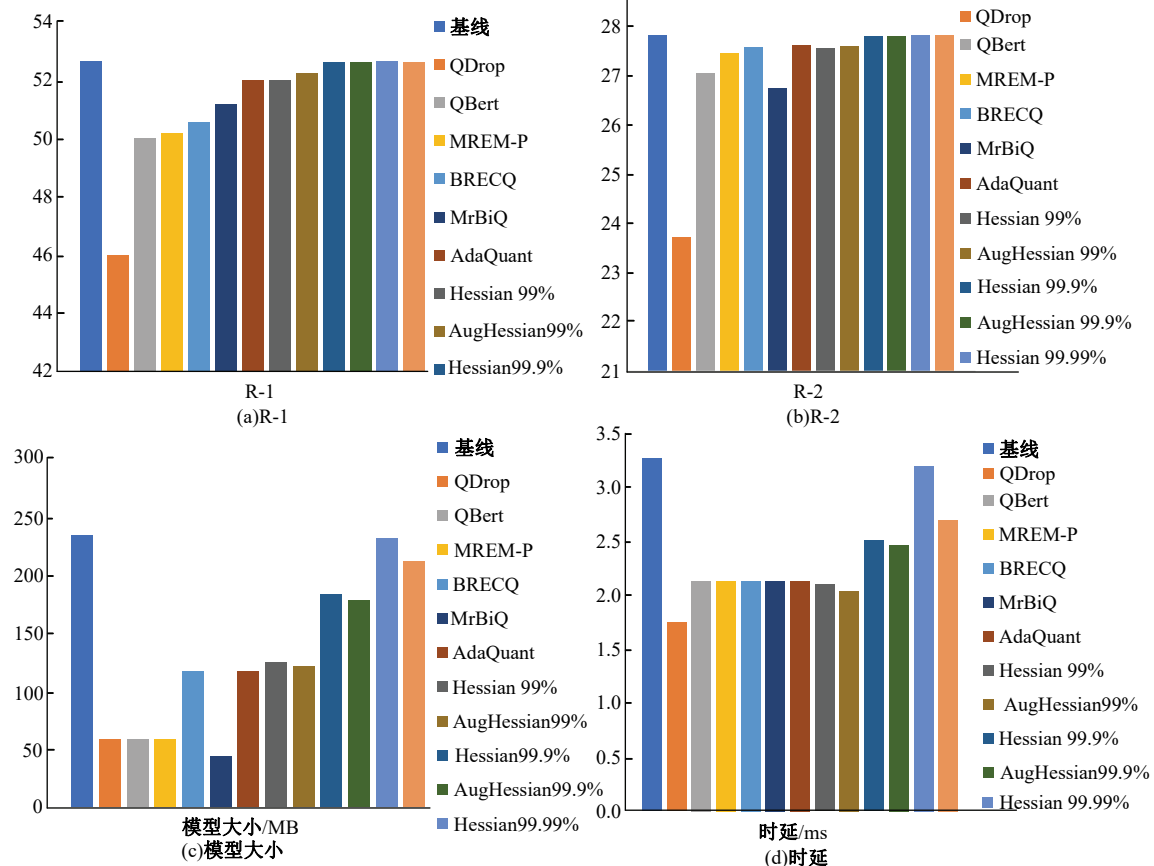


图 5 在 DSMC 上对比其他研究

Fig. 5 Comparison of other studies on DSMC model

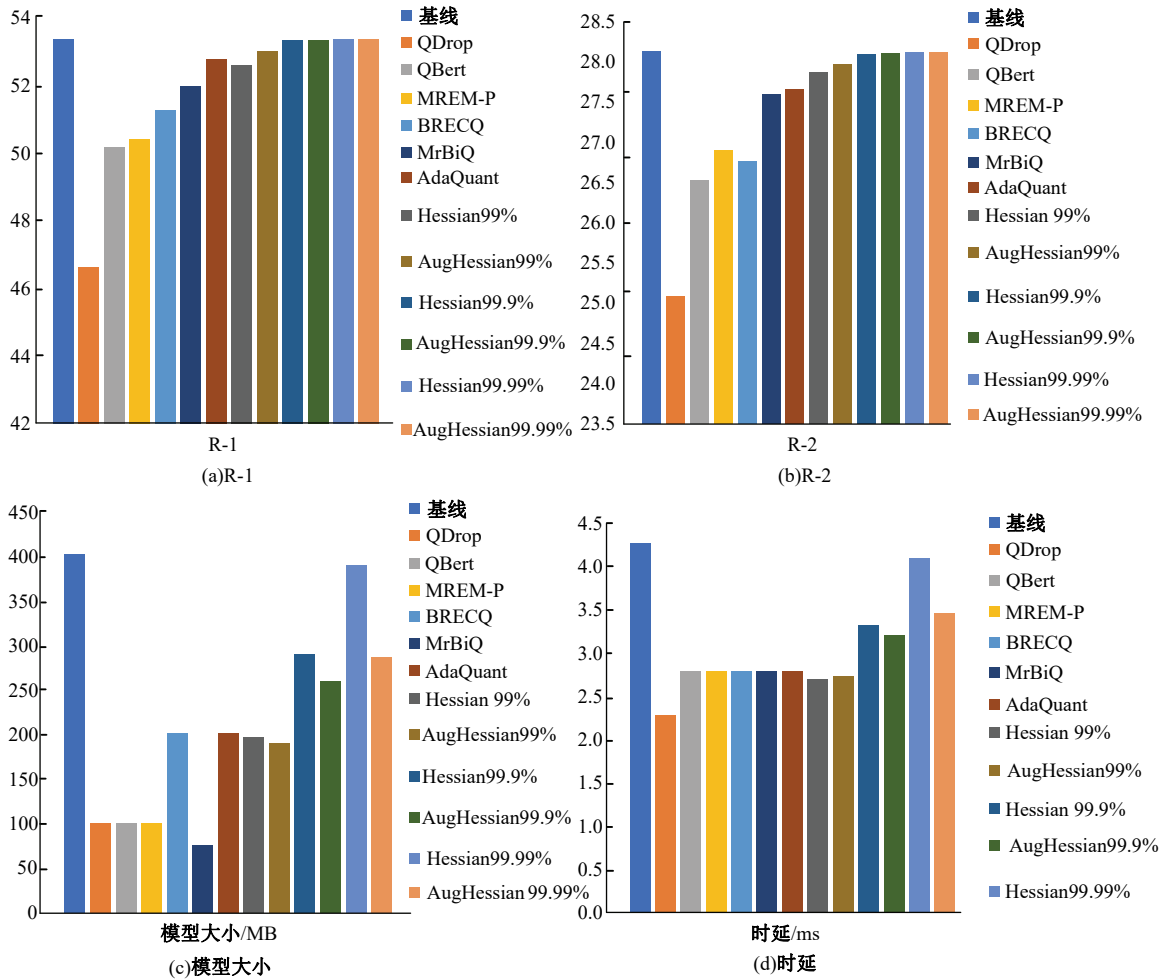


图6 在LBART上对比其他研究

Fig. 6 Comparing other studies on LBART model

量化方法在R-1和R-2评价指标上得分高于所有对比模型,同时还将模型大小缩小了52%,推理时延也降低了36%。部分量化方法虽然在评分保持上没有很好结果,但不能将模型大小的压缩比例和推理时延降至更低。其中,MrBiQ的R-1和R-2评分虽然低于本文搜索方法,但其将模型压缩至原模型的18.75%。另外,还探究了当R-1和R-2评分保持在99.9%和99.99%时的模型量化情况,此时在几乎不损失模型精度的情况下,仍能将模型缩小36%和28%,并将推理时延降低25%和19%。

同时实验中表明,基于 $\varepsilon_{\text{AugHessian}}$ 敏感度指标相对于 $\varepsilon_{\text{Hessian}}$ 敏感度有更好效果。在对比所有量化方法中,基于 $\varepsilon_{\text{AugHessian}}$ 敏感度生成的模型在保持ROUGE评分、模型大小和延迟之间具有更好平衡。

4 混合精度量化可视化

将量化后的每层位宽做了展示,图7为在99.99%和98%的评分保持度设置下LBART被量化后每层量化的位宽情况。

表3展示了LBART模型在保持99.99%和98%评分时的详细位宽分配情况。在99.99%

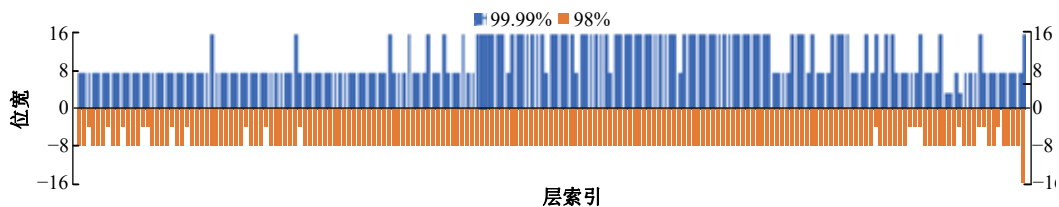


图7 在99.99%和98%评分搜索下的每层bit位宽设置

Fig. 7 Per-layer bit-width settings under 99.99% and 98% scoring searches

表 3 量化后不同位宽数量

Table 3 Quantization number of the different bit widths

评分	4 bit	8 bit	16 bit
99.99%	3	112	78
98%	18	174	1

时,几乎所有权重都被量化为 8 bit 和 16 bit,仅有 3 个层被量化为 4 bit。在 98% 时,绝大多数层都被量化为 8 bit,仅有最后一层被量化为 16 bit。

5 评估次数分析

表 4 展示了 LBART 在 99% 和 99.9% 的评分保持下,期望的二分搜索所需评估次数,这些值与理论预期的 $O(b\log N)$ 一致。通过平均仅 6 次评估,二分搜索比顺序搜索快得多。

表 4 LBART 在不同精度下的评估次数

Table 4 Evaluation number of LBART at the different accuracies

方法	LBART 99%		LBART 99.9%	
	8 bit	4 bit	8 bit	4 bit
Hessian	6	6	6	7
AugHessian	6	8	6	7

6 结束语

本文提出了一种后训练层级混合精度整数量化方案,总结为:①针对对话式摘要模型在特定通道中存在大量离群值导致量化激活时带来较大误差的问题,通过平滑方法将激活的量化困难平移到权重上,得到值分布稳定的激活和权重;②提出了 Hessian 增广方法(考虑层间相互影响)评估模型层敏感度,针对不同敏感度将其量化成不同位宽,提高了量化后的模型精度;③提出了快速搜索量化空间方法,该方法可以大大减少搜索量化空间时间,仅需要 $O(b\log N)$ 复杂度就可以将模型量化。通过对比大量经典量化方法,证明本量化方法的优越性,缩小模型存储大小,并在降低模型推理时延的同时还保持一定的模型精度。将来希望更好地结合两类量化方法的优势,进行混合精度量化。

参考文献:

- [1] Dettmers T, Lewis M, Belkada Y, et al. GPT3. int8(): 8-bit matrix multiplication for transformers at scale[J]. Advances in Neural Information Processing Systems, 2022, 35: 30318-30332.
- [2] Zafrir O, Boudoukh G, Izsak P, et al. Q8BERT:

Quantized 8bit bert[C]//Proceedings of the Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition, Vancouver, Canada, 2019: 36-39.

- [3] Bai H L, Hou L, Shang L F, et al. Towards efficient post-training quantization of pre-trained language models[J]. Advances in Neural Information Processing Systems, 2022, 35: 1405-1418.
- [4] Li Y H, Gong R H, Tan X, et al. BRECQ: pushing the limit of post-training quantization by block reconstruction[C]//International Conference on Learning Representations, Vienna, Austria, 2021, 2: 1-17.
- [5] Wang K, Liu Z, Lin Y, et al. HAQ: hardware-aware automated quantization with mixed precision[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, USA, 2019: 8612-8620.
- [6] Schaefer C J S, Joshi S, Li S, et al. Edge inference with fully differentiable quantized mixed precision neural networks[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, USA, 2024: 8460-8469.
- [7] Park E, Yoo S. PROFIT: a novel training method for sub-4-bit mobilenet models[C]//Proceedings of the Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, 2020: 430-446.
- [8] Yao Z W, Dong Z, Zheng Z C, et al. HAWQ-V3: dyadic neural network quantization[C]//Proceedings of the International Conference on Machine Learning, Shenzhen, China, 2021: 11875-11886.
- [9] Nahshan Y, Chmiel B, Baskin C, et al. Loss aware post-training quantization[J]. Machine Learning, 2021, 110(11-12): 3245-3262.
- [10] Nagel M, Amjad R A, Van B M, et al. Up or down? Adaptive rounding for post-training quantization[C]//Proceedings of the International Conference on Machine Learning, Vienna, Austria, 2020: 7197-7206.
- [11] Yao Z W, Aminabadi R Y, Zhang M J, et al. ZeroQuant: efficient and affordable post-training quantization for large-scale transformers[J]. Advances in Neural Information Processing Systems, 2022, 35: 27168-27183.
- [12] Cai Y, Yao Z, Dong Z, et al. Zeroq: a novel zero shot quantization framework[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern

- Recognition, Seattle, USA, 2020: 13169–13178.
- [13] Hubara I, Nahshan Y, Hanani Y, et al. Accurate post training quantization with small calibration sets [C]//Proceedings of the International Conference on Machine Learning, Online, 2021: 4466–4475.
- [14] Frantar E, Alistarh D. Optimal brain compression: a framework for accurate post-training quantization and pruning[J]. Advances in Neural Information Processing Systems, 2022, 35: 4475–4488.
- [15] Demidovskij A, Smirnov E. Effective post-training quantization of neural networks for inference on low power neural accelerator[C]//International Joint Conference on Neural Networks, Glasgow, UK, 2020: 1–7.
- [16] Zandonati B, Pol A A, Pierini M, et al. Fit: a metric for model sensitivity[C]//Proceedings of ICLR, Addis Ababa, Ethiopia, 2022: 1–20.
- [17] Zheng D, Liu Y, Li L. Leveraging inter-layer dependency for post-training quantization[J]. Advances in Neural Information Processing Systems, 2022, 35: 6666–6679.
- [18] Wei X Y, Zhang Y C, Zhang X G, et al. Outlier suppression: pushing the limit of low-bit transformer language models[J]. Advances in Neural Information Processing Systems, 2022, 35: 17402–17414.
- [19] Bondarenko Y, Nagel M, Blankevoort T. Understanding and overcoming the challenges of efficient transformer quantization[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, Punta Cana, Dominican Republic, 2021: 7947–7969.
- [20] Zeng A, Liu X, Du Z, et al. GLM-130B: an open bilingual pre-trained model[C]//Proceedings of ICLR, Kigali, Rwanda, 2023: 1–56.
- [21] Wu H, Judd P, Zhang X, et al. Integer quantization for deep learning inference: principles and empirical evaluation[J/OL]. [2024-07-06]. arXiv Preprint arXiv: 2004.09602.
- [22] Fabbri A, Rahman F, Rizvi I, et al. ConvoSumm: conversation summarization benchmark and improved abstractive summarization with argument mining[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Bangkok, Thailand, 2021: 6866–6880.
- [23] Lewis M, Liu Y, Goyal N, et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension [C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 2019: 7871–7880.
- [24] Liu Y P, Zhang Y H, Liu G. A conversation summary generation method for medical consultations[P]. China Patent: ZL115964475A, 2023-04-14.
- [25] Shen S, Dong Z, Ye J Y, et al. Q-BERT: hessian based ultra low precision quantization of bert[C]//Proceedings of the AAAI Conference on Artificial Intelligence, New York, USA, 2020: 8815–8821.
- [26] Wei X Y, Gong R H, Li Y H, et al. QDrop: randomly dropping quantization for extremely low-bit post-training quantization[C]//Proceedings of ICLR, Online, 2022: 1–19.
- [27] Jeon Y, Lee C, Cho E, et al. Mr. BIQ: post-training non-uniform quantization based on minimizing the reconstruction error[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, USA, 2022: 12319–12328.
- [28] Goo C W, Chen Y N. Abstractive dialogue summarization with sentence-gated modeling optimized by dialogue acts[C]//Proceedings of the IEEE Spoken Language Technology Workshop, Athens, Greece, 2018: 735–742.