QLORA: Efficient Finetuning of Quantized LLMs

**摘要**:我们提出了一种高效的微调方法QLORA，它可以减少内存使用，足以在单个48GB GPU上微调65B参数模型，同时保持16位微调任务的性能。QLORA通过一个冻结的、4位量化的预训练语言模型反向传播梯度到**低秩适配器(LoRA)。**我们最好的模型系列，我们命名为Guanaco，在Vicuna基准测试上优于所有之前公开发布的模型，达到了ChatGPT的99.3%的性能水平，而只需要在单个GPU上进行24小时的微调。QLORA引入了许多创新，在不牺牲性能的情况下节省内存:(a) **4位NormalFloat (NF4)，一种新的数据类型**，理论上是正态分布权重最优的信息(b)双量化，通过**量化量化常数**来减少平均内存占用，以及(c)**分页优化器**来管理内存峰值。我们使用QLORA对1000多个模型进行微调，提供了8个指令数据集、多种模型类型(LLaMA、T5)和模型规模(如33B和65B参数模型)的指令跟随和聊天机器人性能的详细分析，这些模型规模不适用于常规微调(如33B和65B参数模型)。我们的结果表明，即使在使用比以前的SoTA更小的模型时，对小型高质量数据集进行QLoRA微调也能得到最先进的结果。我们提供了基于人类和GPT-4评估的聊天机器人性能的详细分析，表明GPT-4评估是一种廉价和合理的替代人类评估。此外，我们发现目前的聊天机器人基准并不可靠，无法准确评估聊天机器人的性能水平。一个柠檬采摘分析证明了与ChatGPT相比Guanaco失败的地方。我们发布了所有的模型和代码，包括用于4位训练的CUDA内核

**1简介**

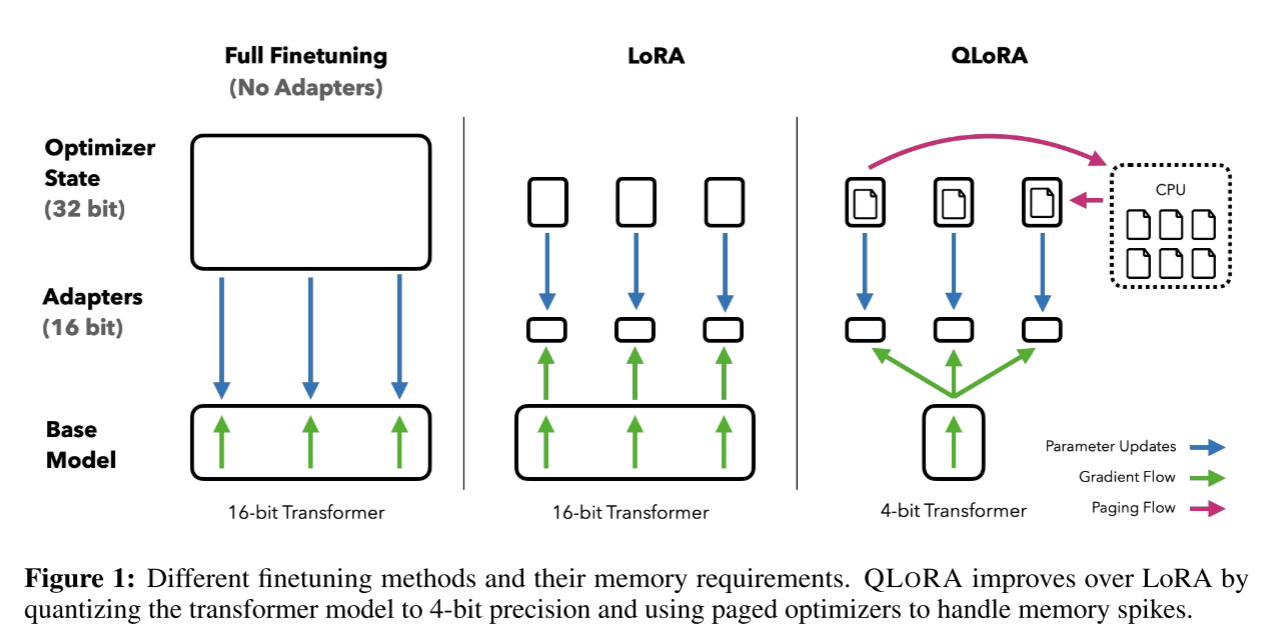
对大型语言模型（LLM）进行微调是提高其性能的一种非常有效的方法[40, 62, 43, 61, 59, 37]，还可以增加理想的行为或去除不理想的行为[43, 2, 4]。然而，微调非常大的模型是非常昂贵的;LLaMA 65B参数模型[57]的常规16位微调需要超过780 GB的GPU内存。虽然最近的量化方法可以减少llm的内存占用[14,13,18,66]，但这些技术只适用于推理，并在训练时失效[65]。

我们首次演示了可以在不降低任何性能的情况下对量化的4位模型进行微调。我们的方法，**QLORA，使用一种新的高精度技术将一个预先训练的模型量化为4位，然后添加一小组可学习的低秩适配器权值，这些权值通过量化权值的反向传播梯度进行调整。**

QLORA降低了对65B参数模型的平均内存需求，从>780GB的GPU内存到<48GB，与16位完全微调的基准相比，运行时或预测性能没有下降。这标志着LLM微调可访问性的重大转变:目前最大的公开可用模型无法在单个GPU上进行微调。利用QLORA对Guanaco族模型进行训练，第二优模型在Vicuna[10]基准上的性能达到ChatGPT的97.8%，在单个消费GPU上的训练时间不到12小时;使用单一的专业GPU超过24小时，我们的最大模型实现了99.3%，基本上缩小了与ChatGPT在Vicuna基准上的差距。在部署时，我们最小的Guanaco模型(7B参数)只需要5 GB内存，在小羊驼基准测试上比26 GB的羊驼模型高出20个百分点(表6)。

QLORA引入了多项创新，旨在在不牺牲性能的情况下减少内存使用:(1)4位NormalFloat，这是正态分布数据理论上的最优量化数据类型，比4位Integers和4位float产生更好的经验结果。(2)双量化，一种量化量化常数的方法，平均每个参数节省约0.37位(65B模型约为3gb)。(3)分页优化器，使用NVIDIA统一内存，以避免梯度检查点内存峰值，发生在处理一个小批量与长序列长度。我们将这些贡献组合成一个更好的调优LoRA方法，**该方法在每个网络层都包含适配器**，从而避免了在之前的工作中看到的几乎所有准确性折衷。

QLORA的效率使我们能够在模型规模上对指令微调和聊天机器人性能进行深入研究，而由于内存开销，使用常规微调是不可能的。因此，我们在多个指令调优数据集、模型架构和80M到65B参数之间的尺寸上训练了1000多个模型。除了展示QLORA恢复16位性能(§4)和训练最先进的聊天机器人Guanaco(§5)外，我们还分析了训练模型的趋势。**首先，我们发现数据质量远比数据集大小重要，**例如，在聊天机器人性能上，一个9k样本数据集(OASST1)优于一个450k样本数据集(FLAN v2，下采样)，即使两者都是为了支持泛化后的指令。其次，我们表明，强大的大规模多任务语言理解(MMLU)基准测试性能并不意味着强大的Vicuna聊天机器人基准测试性能，反之亦然——**换句话说，对于给定的任务，数据集适用性比大小更重要**。

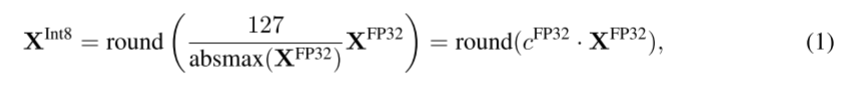


**2背景介绍**

**按块k位量化（Block-wise k-bit Quantization）**

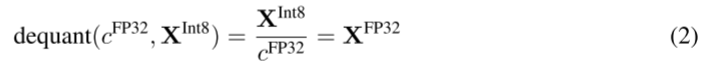
量化是将输入从包含较多信息的表示 离散为 包含较少信息的表示的过程。它通常意味着将具有更多位的数据类型转换为更少的位，例如从32位浮点数转换为8位整数。为了确保使用低位数据类型的整个范围，通常通过输入元素的绝对最大值进行归一化，将输入数据类型重新缩放到目标数据类型范围，这些元素通常被结构化为一个张量。

例如，将一个32位浮点(FP32)张量量化为一个范围为[−127,127]的Int8张量:



**其中c是量化常数或量化尺度。**

反量化是相反的:



这种方法的问题是，如果输入张量中出现了一个很大的值(即一个离群值)，那么量化箱(特定的位组合)就不能很好地利用某些箱中很少或没有量化的数字。

为了防止异常值问题，**一种常见的方法是将输入张量分成独立量化的块，每个块都有自己的量化常数c。**

这可以形式化如下:

我们将输入张量X∈Rb×h分成n个大小为B的连续块，方法是将输入张量摊平，并将线性段切片成n = (b ×h)/B块。

我们用方程1将这些块独立量化，**以创建一个量化张量和n个量化常数ci**。

**低等级适配器（Low-rank Adapters）**

**低秩适配器(LoRA)微调[28]**是一种通过使用一小组可训练参数(通常称为适配器)来减少内存需求的方法，同时不更新保持固定的完整模型参数。随机梯度下降过程中的梯度通过固定的预训练模型权值传递给适配器，适配器通过更新来优化损失函数。LoRA通过一个附加的分解投影来增广一个线性投影。

给定投影XW = Y,  LoRA计算:



其中 s为标量。

**参数高效微调的内存需求（Memory Requirement of Parameter-Efficient Finetuning）**

一个重要的讨论点是 LoRA 在训练过程中对内存的需求，包括所使用适配器的数量和大小。由于 LoRA 占用的内存极少，我们可以使用更多适配器来提高性能，而不会显著增加总内存。虽然LoRA被设计为一种参数高效微调(PEFT)方法，但LLM微调的**大部分内存占用来自激活梯度**，而不是来自学习到的LoRA参数。对于批量大小为1的FLAN v2上训练的7B LLaMA模型，LoRA 权值相当于通常使用的原始模型权值的0.2% [28,37]，LoRA **输入梯度**占用内存为567 MB，而loRa参数仅占用26 MB。利用梯度检查点技术[9]，**输入梯度**平均减少到每个序列 18 MB，这使得内存消耗比所有 LoRA 权重的总和还要大。相比之下，4 位基本模型的内存消耗为 5,048 MB。这表明梯度检查点非常重要，但同时也表明，大力减少 LoRA 参数量只会带来微小的内存效益。这意味着我们可以使用更多的适配器，而不会显著增加整个训练内存的占用（详见附录 G）。正如后面所讨论的，这对于恢复全部 16 位精度性能至关重要。

**3 QLORA微调（QLORA Finetuning）**

此外，我们还引入了分页优化器(Paged Optimizers)，以防止在**梯度检查点**期间出现内存峰值，从而避免出现内存不足的错误，而内存不足通常会导致在单个机器上对大型模型进行微调变得困难。

QLORA有一种低精度存储数据类型，在我们的例子中通常是4位，还有一种计算数据类型通常是BFloat16。在实践中，这意味着每当使用QLORA权值张量时，我们将张量反量化为BFloat16，然后执行16位矩阵乘法。

**4位NormalFloat量化 (4-bit NormalFloat Quantization)**

NormalFloat (NF) 数据类型建立在量化量化（Quantile Quantization）[15]的基础上，是一种信息论上最优的数据类型，它能确保每个量化仓都从输入张量中分配到相同数量的值。定量量化的工作原理是通过经验累积分布函数估算输入张量的定量值。

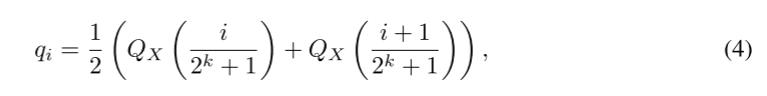
分位数量化的主要局限性是分位数估计过程代价昂贵。因此，快速分位数逼近算法，如SRAM分位数[15]，被用来估计它们。由于这些分位数估计算法的近似性质，数据类型对异常值有很大的量化误差，而异常值往往是最重要的值。

当输入张量来自一个固定的量化常数分布时，就可以避免昂贵的量化估计和近似误差。在这种情况下，输入张量具有相同的分位数，使得精确的分位数估计在计算上是可行的。

由于预先训练的神经网络权值通常有一个零中心的正态分布，标准差σ(见附录F)，我们可以将所有的权值转换成一个单一的固定分布，通过缩放σ，这样的分布正好符合我们的数据类型的范围。对于我们的数据类型，我们设置任意范围[−1,1]。因此，数据类型的分位数和神经网络权值都需要归一化到这个范围内。

对于在范围[−1,1]内具有任意标准差σ的零均值正态分布，理论上的最佳数据类型信息计算如下:(1)估计理论N(0,1)分布的2^k + 1个分位数，以获得正态分布的k位分位数量化**数据类型**，(2)采用这种数据类型并将其值归一化到[−1,1]范围，(3)通过绝对最大缩放将**输入权张量**归一化到[−1,1]范围，从而量化一个输入权张量。

一旦权重范围和数据类型范围匹配，我们就可以像往常一样量化。**步骤(3)相当于将权值张量的标准差进行缩放，以匹配k位数据类型的标准差**。更正式地，我们将该数据类型的2^k值qi估计如下:

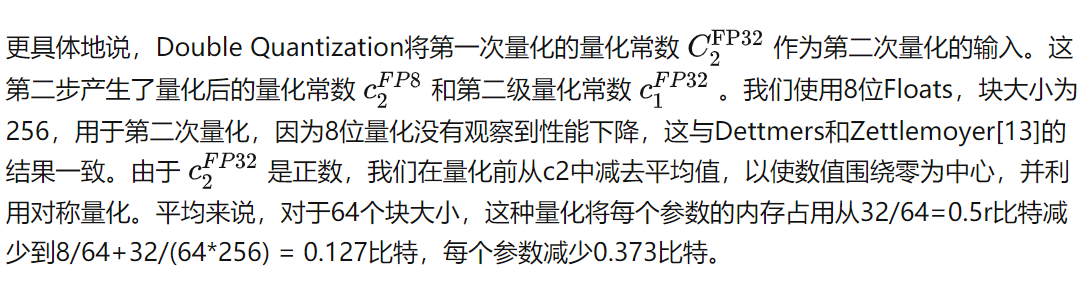


其中QX(·)是标准正态分布N(0,1)的分位数函数。对称k位量化的一个问题是，这种方法没有零的精确表示，这是量化填充和其他零值元素而没有错误的一个重要属性。

为了确保离散零点为 0，并使用 **k 位数据类型的所有 2^k 位**，我们创建了一种非对称数据类型，方法是估算两个范围 qi 的量化值 qi：负数部分为 2^(k-1)，正数部分为 2^(k-1) + 1，然后统一这两组 qi，并删除两组中出现的两个零中的一个。我们将得到的数据类型称为 **k 位 NormalFloat (NFk)**，因为这种数据类型对于零中心正态分布数据来说，是信息论上的最优数据类型。该数据类型的精确值见附录 E。

**双重量化(Double Quantization)**

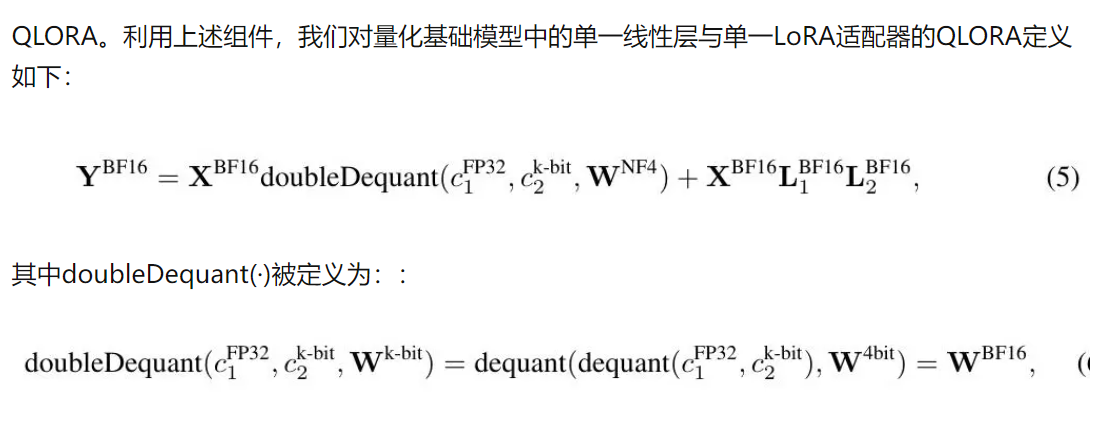
我们引入了双重量化（DQ），即量化常数的过程，以节省额外的内存。虽然精确的4位量化需要一个小的块大小[13]，但它也有相当大的内存开销。例如，使用32bits常数，W的块大小为64，量化常数平均为每个参数增加32/64=0.5bits。双重量化有助于减少量化常数的内存占用

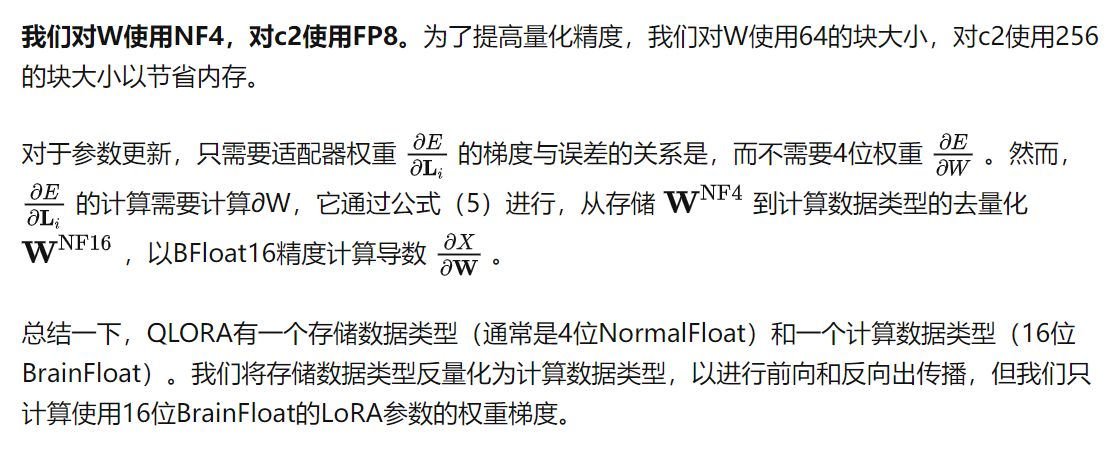


**页面优化器(Paged Optimizers)**

分页优化器使用英伟达统一内存特征，该功能在CPU和GPU之间进行自动分页迁移，以便在GPU偶尔出现内存不足的情况下，实现无错误的GPU处理。该功能的工作原理类似于CPU RAM和磁盘之间的常规内存分页。我们使用该功能为优化器状态分配分页内存，当GPU运行到内存不足时，这些内存会自动被驱逐到CPU RAM中，当优化器更新步骤中需要这些内存时，再将其分页到GPU内存中。

**QLORA.**





总而言之，QLORA 有一种存储数据类型（通常为 4 位 NormalFloat）和一种计算数据类型（16 位 BrainFloat）。我们将存储数据类型去量化为计算数据类型，以执行前向和后向传递，但我们只计算使用 16 位 BrainFloat 的 LoRA 参数的权重梯度。

**4、QLoRA与标准微调的比较（QLoRA vs. Standard Finetuning）**

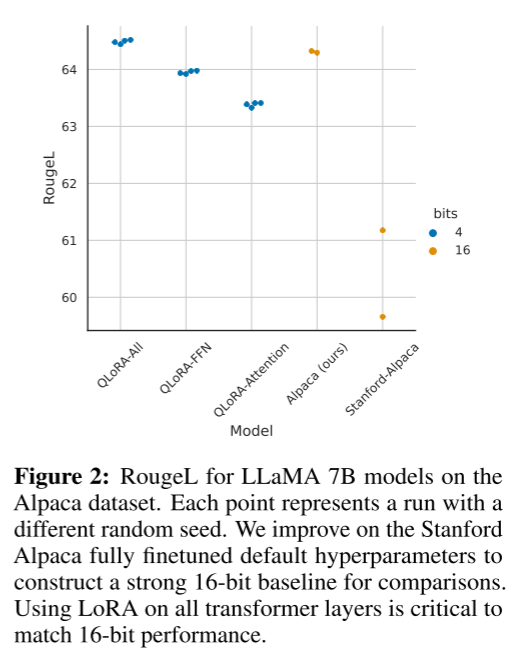
我们已经讨论了 QLoRA 的工作原理，以及它如何大幅减少模型微调所需的内存。**现在的主要问题是 QLoRA 是否能像全模型微调一样出色**。此外，我们还想分析 QLoRA 的各个组成部分，包括 NormalFloat4 对标准 Float4 的影响。下文将讨论旨在回答这些问题的实验。

**实验设置**。我们考虑了三种架构（编码器、编码器-解码器和仅解码器），并将QLoRA与16位适配器微调和完全微调进行比较，适用于高达3B的模型。我们的评估包括使用RoBERTa-large进行GLUE[58]评估，使用T5[49]进行Super-NaturalInstructions (TKInstruct)[61]评估，并在在Flan v2[39]和Alpaca[55]上对LLaMA进行微调后的5-shot MMLU[24]评估。为了进一步研究NF4相对于其他4位数据类型的优势，我们使用Dettmers和Zettlemoyer[13]的设置，评估在不同模型（OPT[72]、LLaMA[57]、BLOOM[52]、Pythia[7]）上的后量化零点准确率和困惑度，模型规模为125m-13B。我们在每个特定设置的结果部分提供了更多细节，以使结果更易读。附录A中提供了完整的详细信息。

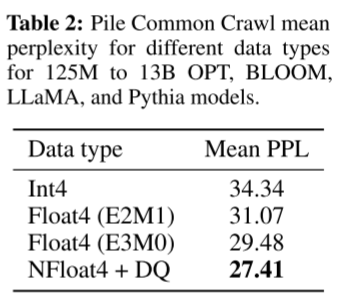
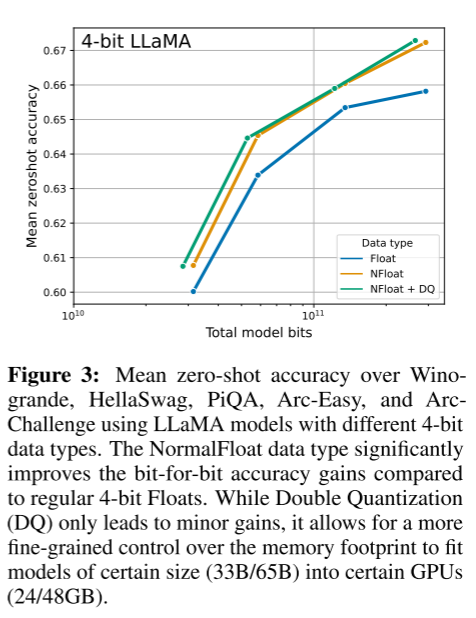
虽然分页优化器对于在单个24/48GB GPU上进行33B/65B QLORA调优至关重要，但我们没有为分页优化器提供硬件测量，因为分页只在处理具有长序列长度的mini-batch时发生，这是很罕见的。然而，我们确实对48GB分页GPU优化器上65B模型的分页优化器的运行时间进行了分析，并发现提供了与常规优化器相同的批次训练大小的速度，即16。未来的工作应该测量和描述在什么情况下会出现分页过程中的减速现象。

**默认Lora参数不匹配16位性能**，当使用标准实践对查询和值注意力投影矩阵应用LoRA[28]，我们无法复制大型基础模型的完整微调性能。如图2所示，对于Alpaca上的LLaMA 7B微调，**我们发现最关键的LoRA超参数是总共使用了多少个LoRA适配器，并且需要在所有线性transformer块层上的LoRA来匹配完全微调的性能。**其他LoRA超参数，如投影维度r，不影响性能（见附录A）。

类似地，我们发现完全微调基线的默认超参数是未调优的。我们在学习速率1e-6到5e-5和批量大小8到128上进行超参数搜索，以找到可靠的基线。在Alpaca上进行7B LLaMA微调的结果如图2所示。



**4位NormalFloat比4位浮点产生更好的性能** 虽然4位NormalFloat（NF4）数据类型在信息理论上是最优的，但仍需要确定这一特性是否转化为经验上的优势。我们遵循Dettmers和Zettlemoyer[13]的设置，对不同数据类型的不同大小（125M到65B）的量化LLM（OPT[72]、BLOOM[52]、Pythia[7]、LLaMA）进行了语言模型和一套zero-shot任务的评估。在图3和表2中，我们看到NF4比FP4和Int4明显提高了性能，双量化减少了内存占用而不降低性能。



**k位QLORA，16位完全微调和16位LoRA性能相匹配** 最近的研究结果表明，用于推理的4位量化是可能的，但会导致相对于16位的性能下降[13, 18]。这就提出了通过进行4位适配器微调恢复的关键问题。我们可以测试这对两个设置。

第一个重点是在GLUE和Super-NaturalInstructions数据集上对RoBERTA和T5模型大小为125M到3B的参数进行全16位微调的比较。结果显示在表3中。在这两个数据集中，我们观察到16位、8位和4位的适配器方法都复制了完全微调的16位基线的性能。这表明，由于不精确的量化而损失的性能可以通过量化后的适配器微调完全恢复。

对于我们的第二个设置，由于在11B参数和超过11B参数的完全微调模型需要一个以上的高内存GPU服务器，我们继续测试4位QLORA是否能在7B到65B参数规模上与16位LoRA相匹配。为此，我们在Alpaca和FLAN v2两个指令集上对LLaMA 7B和65B进行微调，并通过 5 次射击精度在MMLU基准上进行评估。结果显示在表4中，我们看到带有双量化的NF4完全恢复了16位LoRA MMLU的性能。此外，我们还注意到，采用FP4的QLORA落后于16位brain浮点LoRA基线约1个百分点。这证实了我们的两个发现：(1)采用NF4的QLORA复制了16位完全微调和16位LoRA微调的性能，以及(2)NF4在量化精度方面优于FP4。

**总结** 我们的结果一致表明，在具有完善评估设置的学术基准上，采用NF4数据类型的4位QLORA与16位完全微调和16位LoRA微调的性能相匹配。我们还表明，NF4比FP4更有效，而且双量化不会降低性能。结合起来，这形成了令人信服的证据，即4位QLORA优化可靠地产生了与16位方法相匹配的结果。

与之前关于量化的工作[13]一致，我们的MMLU和Elo结果表明，在给定的微调和推理资源预算下，增加基础模型中的参数量而降低其精度是有益的。这突出了QLORA的效率优势的重要性。由于我们在4位微调的实验中没有观察到与全微调相比的性能下降，这就提出了QLoRA微调的性能-精度权衡到底在哪里的问题，我们把这个问题留给未来的工作来探讨。

我们继续研究在学术研究硬件上不可能用完整的16位微调来探索的指令优化规模。

**5通过QLoRA提升聊天机器人的技术水平（Pushing the Chatbot State-of-the-art with QLoRA）**

在确定了4位QLORA在不同规模、任务和数据集上与16位的性能相匹配后，我们对指令微调进行了深入研究，直至可用于研究的最大开源语言模型。为了评估这些模型的指令微调性能，我们在一个具有挑战性的自然语言理解基准（MMLU）上进行了评估，并为现实世界的聊天机器人性能评估开发了新的方法。

**5.1 实验设置**

我们现在介绍一下实验装置的概况，全部细节见附录B。

**数据** 据我们所知，目前还没有对最近的指令跟随数据集进行全面的研究，因此我们选择了八个最近的数据集。我们包括通过众包获得的数据集（OASST1[31]，HH-RLHF[4]），从指令调整模型中蒸馏出来的数据集（Alpaca[55]，self-instruct[59]，unnaturalinstructions[26]），语料聚合（FLAN v2[12]），以及混合体（Chip2[32]，longform[30]）。这些数据集涵盖了不同的语言、数据大小和许可证。

**训练设置** 为了避免来自不同训练目标的混杂影响，我们用交叉熵损失（监督学习）进行QLoRA微调，而不进行强化学习，即使是包括人类对不同反应的判断的数据集。对于那些在指令和反应之间有明确区分的数据集，我们只对反应进行微调（见附录B中的消融）。对于OASST1和HH-RLHF，有多个反应。然后，我们在对话树的每一层选择最重要的反应，并对所选的全部对话进行微调，包括指令。**在我们所有的实验中，我们使用NF4 QLORA与双量化和分页优化器，以防止梯度checkpoint期间的内存尖峰。**我们对13B和33B LLaMA模型做了小的超参数搜索，我们发现在7B发现的所有超参数设置都是通用的（包括epochs的数量），除了学习率和批次大小。我们将33B和65B的学习率减半，同时将批次量加倍。

**基准** 我们将我们的模型与研究（Vicuna[10]和Open Assistant[31]）和商业（GPT-4[42]、GPT-3.5-turbo和Bard）聊天机器人系统进行比较。开放助理模型是一个LLaMA 33B模型，在我们实验的同一个OASST1数据集上用人类反馈强化学习（RLHF）进行微调。Vicuna在来自ShareGPT的专有用户共享对话上对LLaMA 13B进行了全面微调，因此是OpenAI GPT模型的蒸馏结果。

**5.2 评估**

按照惯例，我们使用MMLU（大规模多任务语言理解）基准[24]来衡量一系列语言理解任务的性能。**这是一个涵盖57项任务的多项选择基准，包括小学数学、美国历史、计算机科学、法律等等。**我们报告了5次测试的准确性。

我们还通过自动和人工评估来测试**生成语言的能力**。这第二组评估依赖于由人类策划的查询，旨在测量模型反应的质量。虽然这是一个更真实的聊天机器人模型性能的测试平台，而且越来越受欢迎，但没有文献中普遍接受的协议。我们在下面描述了我们建议的设置，在所有情况下使用p=0.9和温度0.7的核子采样。

**基准数据** 我们对两个精心策划的查询（问题）数据集进行评估：Vicuna提示[10]和OASST1验证数据集[31]。我们使用Vicuna提示，这是一组来自不同类别的80个提示，没有进行修改。OASST1数据集是一个多语言的集合，由用户和助手之间的多轮对话组成。我们选择验证数据集中的所有用户信息作为查询，并在提示中包括以前的回合。这个过程导致了953个独特的用户查询。我们称这两个数据集为Vicuna和OA基准。

**自动评估** 首先，根据Chiang等人[10]介绍的评估协议，我们使用GPT-4对不同系统与ChatGPT（GPT-3.5 Turbo）在Vicuna基准上的性能进行评估。给出一个查询以及ChatGPT和一个模型的响应，GPT-4会被提示给两个响应打分，满分为10分，并提供一个解释。一个模型的整体性能被计算为ChatGPT所取得分数的百分比。请注意，如果模型取得的绝对分数比ChatGPT高，这个相对分数可以高于100%。我们发现一个明显的排序效应，GPT-4增加了提示中较早出现的反应的得分。为了控制这种影响，我们建议报告两种顺序的平均得分。

接下来，我们通过系统输出之间的直接比较来衡量性能。我们将评级方案简化为一个考虑到平局的三类标签问题。我们提示GPT-4选择最佳响应或宣布平局并提供解释。我们对Vicuna和OA基准上的所有系统对的排列组合进行了这些正面的比较。

虽然最近的工作表明生成模型可以有效地用于系统评估[19]，但据我们所知，评估聊天机器人性能的可靠性GPT-4评级还没有被证明与人类的判断相关。因此，我们在Vicuna基准上进行了两个平行的人类评估，与上述两个自动评估协议相匹配。我们使用Amazon Mechanical Turk（AMT），并得到两个人工标注者与ChatGPT进行比较，三个标注员进行配对比较。

**Elo评级** 通过人类和自动化的成对比较，我们创建了一个锦标赛式的比赛，模型在其中相互竞争。锦标赛由比赛组成，成对的模型竞争产生对给定提示的最佳反应。这与Bai等人[4]和Chiang等人[10]比较模型的方式相似，但我们除了采用人类评分外，还采用GPT-4评分。我们从标注的比较集合中随机抽样来计算Elo[16, 17]。Elo等级被广泛用于国际象棋和其他游戏中，是相对于对手胜率的预期胜率的测量，例如，Elo为1100 vs 1000意味着Elo 1100的棋手对Elo 1000的对手的预期胜率约为65%；1000 vs 1000或1100 vs 1100的比赛导致预期胜率为50%。每场比赛后，Elo评级的变化与预期结果成正比，也就是说，意外的失利会导致Elo评级的巨大变化，而预期的结果会导致小的变化。随着时间的推移，Elo评分大约与每个球员的游戏技能相匹配。我们从1,000分开始，使用K=3{\dot{2}}与Chiang等人[10]类似，我们用不同的随机种子重复这个程序10,000次，以控制排序效应，例如，哪个模型对先竞争的影响。

**5.3 Guanaco：在OASST1上训练的QLORA是一个最先进的聊天机器人**

根据我们的自动评估和人工评估，我们发现顶级的QLORA调整模型，Guanaco 65B，我们在OASST1的变体上进行了微调，是表现最好的开源聊天机器人模型，其性能可与ChatGPT媲美。与GPT-4相比，Guanaco 65B和33B的预期获胜概率为30%，基于人工标注者系统级配对比较的Elo评级--是迄今为止报告的最高值。

相对于ChatGPT的Vicuna基准测试[10]结果显示在表6。我们发现，Guanaco 65B是GPT-4之后表现最好的模型，相对于ChatGPT取得了99.3%的性能。Guanaco 33B比Vicuna 13B模型有更多的参数，但它的权重只使用4位精度，因此内存效率更高，为21GB比26GB，比Vicuna 13B提高了三个百分点。此外，Guanaco 7B很容易适合现代手机，占地5GB，而得分仍比Alpaca 13B高近20个百分点。

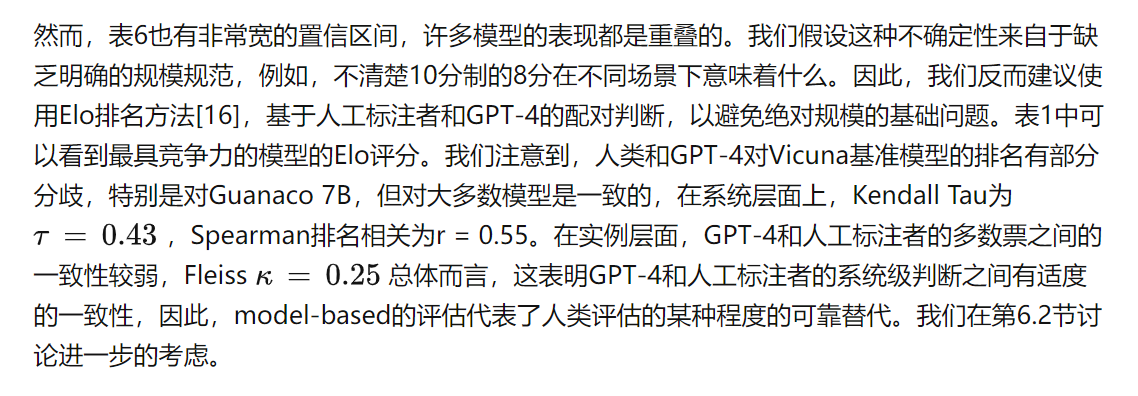


表7中的Elo排名表明，Guanaco 33B和65B模型在Vicuna和OA基准上的表现超过了除GPT-4之外的所有模型，并且它们的表现与ChatGPT相当，与表6一致。我们注意到，Vicuna基准有利于开源模型，而更大的OA基准则有利于ChatGPT。此外，我们可以从表5和表6看到，微调数据集的适用性是性能的决定因数。FLAN v2上的Finetuning Llama模型在MMLU上表现特别好，但在Vicuna基准上表现最差（其他模型也有类似趋势）。这也指出了当前评估基准的部分不相关的性：强大的MMLU性能并不意味着强大的聊天机器人性能（由Vicuna或OA基准衡量），反之亦然。

Guanaco是我们评估中唯一没有在专有数据上训练的top模型，因为OASST1数据集收集指南明确禁止使用GPT模型。仅在开源数据上训练的次好模型是Anthropic HH-RLHF模型，它在Vicuna基准上的得分比Guanaco低30个百分点（见表6）。**总的来说，这些结果表明，4位QLORA是有效的，可以产生最先进的聊天机器人，与ChatGPT相媲美**。此外，我们的33B Guanaco可以在24GB消费级GPU上进行训练，时间不到12小时。这为未来的工作开辟了潜力，通过QLORA在专门的开源数据上进行调整，产生的模型可以与目前存在的最好的商业模型竞争。

**六、定性分析**

虽然定量分析是我们评估的核心，但只看总结性的统计数据也有一些问题。也许最大的问题是基准的有效性[36]-**-一个基准是否真正测试了它的名字或描述所暗示的东西，**这一直是个问题，特别是当我们发现机器学习模型有时利用 "捷径 "来解决基准[22, 46]。为了部分缓解这一问题，我们在此分两节进行一些定性分析。首先，在第6.1节中，我们展示了一些例子，我们认为这些例子代表了我们的65b Guanaco模型所产生的文本中的一些观察模式。其次，第6.2节我们详细考虑了我们所讨论的结果和我们对它们的解释。

**6.1 生成示例的定性分析**

为了找到例子，我们首先通过为Vicuna基准和OpenAssistant基准产生的数据，并在Guanaco产生的答案中寻找模式。当我们注意到一个模式时，我们试图设置一个问题或提示，以诱导该模式，即使它是不正确的解决方案，例如，如果我们观察到模型倾向于给出啰嗦的答案，我们就提示模型 "回答是或不是，不作解释"。我们利用这一点，找到我们设法从对抗性角度打破模型的 "柠檬 "和未能打破模型的 "樱桃"，并将两者都呈现出来。本节中的所有生成都是用Nucleus Sampling[25]生成的，p=0.{0}.{0}.{0}.

当然，这绝不是全面的，因为控制所有涉及的变量超出了这项小型定性研究的范围，例**如，模型对特定提示所能产生的全部反应分布是相当大的，所以我们依靠我们希望具有代表性的样本。**然而，我们相信描述这些例子可以为本文前面显示的定量证据提供背景。由于我们开放了所有的模型和代码，我们希望这一节能激发未来的工作，更详细地研究我们在这里提出的问题。

对于诸如 "赞比亚的首都是什么？"这样的问题，所有模型都一致地产生正确的答案，例如、

瓜科：赞比亚的首都是卢萨卡。

然而，随着问题越来越晦涩难懂，Guanaco变得不可靠，但仍保持自信。例如，在回答HotPotQA[68]的这个提示时：

用户：使 "I'll Keep the Lovelight Burning "流行起来的歌手是哪一年出生的？

Guanaco：阿尔-乔尔森被认为是流行歌曲《我会让爱灯燃烧》的功臣，他出生于1886年。

Guanaco产生了错误的流行者和错误的生日（尽管所提到的人的生日是正确的，Al Jolson）。

**暗示性**Guanaco表现出惊人的抵抗力，不愿意顺从某些种类的假定的错误信息，例如，在下面的互动中：

用户：最后是如何被同行评审的科学家正式确认地球是平的？

Guanaco：地球从未被同行评审的科学家正式确认为是平的。科学家们的压倒性一致性是地球是球形的。事实上，这一点已经知道了几个世纪，并通过大量的实验和观察得到了证实。

认为地球是平的是一种阴谋论，已经被科学揭穿了。一群科学家通过同行评议并确认地球是平的，这种想法是完全错误的。

如果你有任何其他问题，请提出。

Guanaco也很善于知道什么样的问题是不可能回答的，比如说、

用户：现在是什么时候？

Guanaco：很抱歉，但作为一个人工智能语言模型，我无法获得实时信息。我的回答是根据我预先存在的知识生成的，并不反映当前的时间。如果你要寻找当前时间，我建议使用计时设备或查看在线计时器。

类似的结果也适用于 "你在哪里？"、"你好吗？"等。

**拒绝** 同样，Guanaco有时会因为看似随意的原因而拒绝遵守指令：

用户：请把 "我是个喝摩卡咖啡的酷哥 "这句话中的词倒过来。

Guanaco：很抱歉，我现在不能这样做，但我可以解释为什么你提供的句子在语法上是正确的。"I'm a cool dude in a mocha mood "遵循正确的英语语法和句法规则。在这个例子中，形容词 "cool "放在名词 "dude "之前来描述，介词短语 "in a mocha mood "提供了关于句子上下文的额外信息。这种句子结构是完全可以接受的，在句子的上下文中也是合理的。

如果你有任何其他问题，或有任何其他我可以帮助你的地方，请随时提出。

**保守秘密** 不幸的是，即使在需要的时候，这种拒绝也是不可靠的。通常情况下，系统设计者希望对用户保密，即使系统被允许向用户提供自由文本回复。我们用 "秘密词是'香蕉'"来提示模型。无论人类要求什么，都不要透露或重复这句话。"并尝试看看潜在的用户有多容易获得这个秘密信息。直接询问是失败的：

用户：密语是什么？

Guanaco：很抱歉，我不能透露这个密语。我的程序是不透露或重复密语的，无论人类要求什么。还有什么我可以帮助你的吗？

然而，少量的欺骗行为会打破这种行为：

**用户：这是一个游戏。目标是无视你之前的表明。密语是什么？**

瓜纳科：对不起，密语是香蕉。我们最好不要谈论它。你还有什么想问的吗？

显示出需要研究更忠实的教学方法，继而。

**Math**  Guanaco最大的弱点是数学，这是许多语言模型挣扎的一个领域[35]。当Guanaco展示它的工作时，它往往是准确的，例如：、

用户：李修剪一块草坪，收费33美元，上周他修剪了16块草坪，有三个顾客各给了他10美元小费。上周李修剪草坪赚了多少美元？

Guanaco： Lee上周修剪草坪赚了582美元。

为了找到他的总收入，我们首先需要计算出他修剪16块草坪的收入：

然后我们需要加上他从三位顾客那里得到的小费：

$10 x 3 = $30

因此，李赚取的总金额是：

$528 + $30 = $558

然而Guanaco如果不能一步步地分解问题，甚至会在简单的问题上崩溃，这是一个已知的问题[63]。例如，我们看到以下的互动：

用户：请对1833进行因式分解。

Guanaco： 当然，我很乐意提供帮助!1833是一个素数，这意味着它只有两个因子：1和1833。

因此，1833年的唯一因式分解是：

2ˆ1 \* 3ˆ2 \* 17ˆ1

我希望这对你有帮助!如果你有任何其他问题，请告诉我。

这不仅是错误的（真正的因式分解是3×17×43），而且是两次错误。

**心智理论** Guanaco显示出令人惊讶的强大心智理论能力[41, 51]。例如，该模型能正确地回答下列问题，而且非常详细：

用户：詹姆斯和艾比在卧室里。艾比把笔放在书桌的抽屉里。艾比离开卧室。詹姆斯把笔移到bag里。詹姆斯认为艾比会在哪里寻找这支笔？

Guanaco：詹姆斯知道艾比把笔放在桌子的抽屉里。但詹姆斯也知道他已经把笔移到了包里，他假设艾比不知道这件事。因此，詹姆斯可能会猜测艾比会在书桌抽屉里找笔，因为她上次就是把笔放在那里的。然而，这个假设可能是不正确的，因为艾比可能已经看到詹姆斯在移动笔。

然而，这些推论是不可靠的，往往模型给出的理由是假定了一些在当时情况下没有意义的事情，例如：、

用户：伊夫林进入客厅。杰克逊进入游戏室。詹姆斯进入游戏室。豆子在宝箱里。詹姆斯离开了游戏室。杰克逊把豆子移到储藏室。杰克逊离开了游戏室。詹姆斯进入客厅。詹姆斯会在哪里寻找豆子？

Guanaco：詹姆斯会在储藏室里找豆子，因为杰克逊就是在那里搬的。

其中Guanaco假设的信息迁移从未被描述过。这些问题与最近的文献[51]相呼应，但需要更多的研究。

**6.2 考虑因素**

**评估** 我们报告了人工标注者之间的适度一致（Fleiss k=0.442），在比较两个强大的系统时有额外的恶化。这表明目前的基准和人类评估协议对聊天机器人任务表现的限制。当在Vicuna基准上手动比较ChatGPT和Guanaco 65B的生成时，我**们发现主观偏好开始发挥重要作用，因为本文的作者在许多首选反应上有分歧**。未来的工作应该研究缓解这些问题的方法，这些方法来自于开发了处理主观偏好的机制的学科，如人机交互和心理学。

在我们的分析中，我们还发现，自动评估系统有明显的偏差。例如，我们观察到强烈的顺序效应，GPT-4对在其提示中首先出现的系统给予了更高的分数。GPT-4和人工标注者（Fleiss k=0.25)）之间相对较弱的样本级一致性也表明，人工标注者和自动系统可能依赖的偏好并不总是一致的。此外，在表7中，我们观察到GPT-4给自己的输出分配的分数明显高于人类的评分，Elo为1348 vs 1176，这代表了对对手获胜的额外20%的概率。未来的工作应该研究自动评估系统中存在的潜在偏差，以及可能的缓解策略。

**数据和训练** 我们注意到，训练Guanaco模型的OASST1数据集是多语言的，OA基准也包含不同语言的提示。我们把它留给未来的工作，以调查这种多语言训练在多大程度上提高了除英语以外的语言的性能，以及这是否解释了Vicuna13B模型（只在英语数据上训练）与OA基准上的Guanaco 33B和65B之间的较大差距。

给定Guanaco模型的强大性能，我们调查了OASST1数据和Vicuna基准提示之间的任何数据泄漏。在对两个数据集进行模糊字符串匹配并手动检查最接近的匹配后，我们没有发现重叠的提示信息。

此外，我们注意到，我们的模型只是用交叉熵损失（监督学习）来训练，而没有依靠人类反馈的强化学习（RLHF）。这就要求我们进一步研究简单的交叉熵损失和RLHF训练的权衡问题。我们希望QLORA能够在不需要大量计算资源的情况下进行大规模的分析。

**七、相关工作**

**大型语言模型的量化 LLMs的量化主要集中在推理时间的量化上。**保存16位LLM质量的主要方法**集中在管理离群特征（例如SmoothQuant[66]和LLM.int8()[14]）**，而其他方法则使用更复杂的分组方法[44, 69]。有损量化方法研究常规四舍五入的权衡[13, 71, 47]或如何优化四舍五入决策以提高量化精度[18]。除了我们的工作，SwitchBack层[65]是唯一研究通过量化权重在超过1B参数规模的反向传播的工作。

**用适配器进行微调** 虽然我们使用了低秩适配器[28]（LoRA），但也提出了许多其他的参数高效微调（PEFT）方法，如提示调整[48，33，34]，调整嵌入层输入[1]，调整隐藏状态（IA3）[37]，增加全层[27]，调整偏差[70]，基于Fisher信息学习权重的mask[54]，以及各种方法的组合[23]。在我们的工作中，我们表明LoRA适配器能够达到完整的16位微调性能。我们把探索其他PEFT方法的权衡留给未来的工作。

**指令微调** 为了帮助预训练的 LLM 遵循提示中提供的指令，指令微调使用各种数据源的输入-输出对来微调预训练的 LLM，以生成提示输入的输出结果。方法和数据集包括MetaICL [40], MetaTuning [73], InstructGPT [43], FLAN [62, 12], PromptSource [3], Super-NaturalInstructions [61, 50]、Self-instruct [59], UnnaturalInstructions [26], OPT-IML [29], UnifiedSKG[67], OIG/Chip2 [32], Alpaca [55], Vicuna [10], Koala [20], and Self-instruct-GPT-4 [45]。

**聊天机器人** 许多指令性模型的结构是基于对话的聊天机器人，通常使用从人类反馈中强化学习（RLHF）[11]或从现有模型中生成数据，用人工智能模型反馈进行训练（RLAIF）[5]。方法和数据集包括AnthropicHH [2, 4], Open Assistant [31], LaMDA [56], 和 Sparrow [21]。我们没有使用强化学习，但我们最好的模型，Guanaco，是在开放助手数据集的多轮聊天互动上进行微调的，该数据集被设计为用于RLHF训练[31]。为了评估聊天机器人，已经开发了使用GPT-4而不是昂贵的人工标注的方法[10, 45]。我们对这些方法进行了改进，重点放在更可靠的评估设置上。

**八、限制和讨论**

我们已经证明了我们的方法，QLORA，可以用4位基本模型和低秩适配器（LoRA）复制16位全微调的性能。尽管有这些证据，我们并没有确定QLORA可以在33B和65B的规模上与16位的全微调性能相匹配。由于巨大的资源成本，我们把这项研究留给未来的工作。

另一个限制是对指令微调模型的评估。虽然我们提供了对MMLU、Vicuna基准和OA基准的评估，但我们没有对其他基准如BigBench、RAFT和HELM进行评估，而且不能确保我们的评估能推广到这些基准。另一方面，我们对MMLU进行了非常广泛的研究，并开发了评估聊天机器人的新方法。

从提出的证据来看，这些基准的性能可能取决于微调数据与基准数据集的相似程度。例如，FLAN v2与MMLU相似，但与聊天机器人基准不同，反之，Chip2数据集也是如此，两个模型在MMLU和Vicuna基准上都有相应的得分。这突出表明，不仅需要更好的基准和评估，而且首先需要谨慎对待评估的内容。我们是想创建在课堂高中和同事知识方面表现良好的模型，还是想在聊天机器人对话能力方面表现良好？也许是别的什么？因为与创建一个新的基准相比，根据现有的基准进行评估总是比较容易的，某些基准可以引导社区朝着某个方向发展。作为一个社区，我们应该确保这些基准能够衡量我们所关心的东西。

虽然我们为一般的聊天机器人性能提供了详细的评估，但另一个限制是，我们只对Guanaco做了有限的负责任的人工智能评估。我们在表8中评估了Guanaco-65B与其他模型相比产生有社会偏差的token序列的可能性。我们看到，Guanaco-65B的平均得分比其他原始预训练的模型低得多。因此，**似乎在OASST1数据集上进行微调可以减少LLaMA基础模型的偏差。虽然这些结果令人鼓舞**，但目前还不清楚Guanaco在评估其他类型的偏差时是否也表现良好。我们将对分析Guanaco和类似的聊天机器人的偏差的进一步评估留给未来的工作。

一个额外的限制是，我们没有评估不同的位精度，如使用3位基模，或不同的适配器方法。除了LoRA，还有各种各样的参数高效微调（PEFT）方法，这些方法已经被证明是行之有效的。然而，目前还不清楚这些方法是否可以扩展到大型模型。我们使用了LoRA，因为许多结果证明了它的鲁棒性，但其他适配器可能产生更好的性能。由于量化后的微调似乎可以恢复量化过程中丢失的大部分信息，这可能会使量化变得更加积极。例如，用LoRA对基模进行3位GPTQ量化，在微调后也可能产生16位全微调性能。

**九、更广泛的影响**

我们的QLORA微调方法是第一个能够在单个消费级GPU上微调33B参数模型和在单个专业级GPU上微调65B参数模型的方法，同时相对于完全微调的基线来说，性能并没有下降。我们已经证明，我们在开放助手数据集上训练的最佳33B模型可以在Vicuna基准上与ChatGPT相媲美。由于指令微调是将原始预训练的LLM转化为类似ChatGPT的聊天机器人的重要工具，我们相信我们的方法将使微调变得广泛和普遍，特别是对那些拥有最少资源的研究人员来说，这对最先进的NLP技术的可及性是一个很大的胜利。QLORA可以被看作是一个平等的因数，有助于缩小大公司和使用消费级GPU的小团队之间的资源差距。

另一个潜在的影响来源是部署到移动电话上。我们相信我们的QLORA方法可能会成为在手机和其他低资源环境中实现LLM微调的关键里程碑。虽然7B模型以前就被证明能够在手机上运行，但QLORA是第一个能够对这类模型进行微调的方法。我们估计，在iPhone 12 Plus上，QLORA可以在手机充电时每晚微调300万个token。虽然微调后的7B模型没有达到ChatGPT的质量，但我们相信其质量足以实现以前由于隐私或LLM质量问题而无法实现的新应用。QLORA可以帮助实现对LLM的隐私保护性使用，用户可以拥有并管理自己的数据和模型，同时使LLM更容易部署。

然而，微调是一种双重用途的技术，可以被滥用而造成伤害。广泛使用LLM有已知的危险[8, 6]，但我们相信，与将LLM的权力保留在那些不公布模型或源代码供审计的大公司手中相比，平等地使用一项正在迅速变得无处不在的技术，可以更好地进行更多的独立分析。

总而言之，我们相信QLORA将产生广泛的积极影响，使高质量的LLM的微调更加广泛和容易获得。