QLoRA: Efficient Finetuning of Quantized LLMs

Tim Dettmers* **Artidoro Pagnoni*** **Ari Holtzman**

Luke Zettlemoyer

University of Washington
{dettmers,artidoro,ahai,lsz}@cs.washington.edu

Abstract

警告：该PDF由GPT-Academic开源项目调用大语言模型+Latex翻译插件一键生成，版权归原文作者所有。翻译内容可靠性无保障，请仔细鉴别并以原文为准。项目Github地址：https://github.com/binary-husky/gpt_academic/。项目在线体验地址：https://chatpaper.org。当前大语言模型：gpt-3.5-turbo，当前语言模型温度设定：1。为了防止大语言模型的意外谬误产生扩散影响，禁止移除或修改此警告。

我们提出了QLoRA，这是一种高效的微调方法，可以在单个48GB GPU上减少内存使用量，同时保持完全16位微调任务的性能。 QLoRA将渐变通过冻结的4位量化预训练语言模型反向传播到低秩适配器– LoRA。我们最好的模型系列被命名为**Guanaco**，在Vicuna基准测试中胜过所有以前公开发布的模型，达到了ChatGPT性能水平的99.3%，同时只需要单个GPU上的24小时微调。 QLoRA引入了一些创新来节省内存而不损害性能：（a）4位NormalFloat（NF4），这是一种对于正态分布权重，在信息论上是最优的新数据类型；（b）Double Quantization通过量化量化常数来减少平均内存占用量；和（c）Paged Optimizers来管理内存峰值。我们使用QLoRA来微调1000多个模型，并提供了对8个指令数据集、多个模型类型（LLaMA、T5）和以往无法进行常规微调的模型规模（例如33B和65B参数模型）的详细分析。我们的结果表明，基于小型高质量数据集的QLoRA微调可以获得最先进的结果，即使使用比以前的最新技术水平更小的模型。我们基于人工和GPT-4评估对聊天机器人性能进行了详细分析，结果表明，GPT-4评估是人工评估的一种廉价和合理的替代方法。此外，我们发现当前的聊天机器人基准测试不能可靠地评估聊天机器人的性能水平。通过一个具体案例分析可以看出，与ChatGPT相比，**Guanaco**存在失败的地方。我们发布了我们的所有模型和代码，包括4位训练的CUDA核心。[2]

1 Introduction

微调大型语言模型(LLM)是提高其性能的一种非常有效的方式[40, 62, 43, 61, 59, 37]，并且可以增加期望的行为或者去除不期望的行为[43, 2, 4]。然而，对于非常大的模型来说，进行

*Equal contribution.

[2]https://github.com/artidoro/qlora和https://github.com/TimDettmers/bitsandbytes

Preprint. Under review.

微调的成本非常高昂；对于一个具有65B个参数的LLaMA模型[57]，常规的16位微调需要超过780 GB的GPU内存。虽然近期的量化方法可以减小LLM的内存占用[14, 13, 18, 66]，但是这些技术仅适用于推断过程，在训练过程中会产生问题[65]。

我们首次展示了在没有性能降低的情况下，可以对一个4位量化模型进行微调。我们的方法，QLoRA，使用一种新颖的高精度技术将预训练模型量化为4位，然后添加一小组可学习的低秩适配器权重[28]。通过反向传播梯度调整的量化权重，

QLoRA将细调一个65B参数的模型的平均内存需求从大于780GB的GPU内存降低到小于48GB，而不会损害运行时间或预测性能，与16位全细调基准相比。这标志着LLM细调可访问性的重大转变：迄今为止最大的公开可用模型可以在单个GPU上进行细调。

使用QLoRA，我们训练了**Guanaco**模型家族，第二好的模型在Vicuna [10]基准测试中达到了ChatGPT性能的97.8%，同时在单个消费级GPU上可在不到12小时内训练；利用单个专业GPU在24小时内达到了99.3%，基本上缩小了与Vicuna基准上ChatGPT之间的差距。在部署时，我们最小的**Guanaco**模型（7B参数）只需要5GB的内存，在Vicuna基准上比26GB的Alpaca模型高出超过20个百分点（表6）。

表 1: 对于一场在10,000个随机初始排序下，模型之间进行的竞赛，计算的Elo评分。比赛的胜者由GPT-4决定，它会声明对于一个给定的the Vicuna benchmark提示，哪个回答更好。图中显示了95%的置信区间（±）。在GPT-4之后，Guanaco 33B和65B赢得了最多的比赛，而Guanaco 13B的得分也优于Bard。

Model	Size	Elo
GPT-4	-	1348 ± 1
Guanaco 65B	41 GB	1022 ± 1
Guanaco 33B	21 GB	992 ± 1
Vicuna 13B	26 GB	974 ± 1
ChatGPT	-	966 ± 1
Guanaco 13B	10 GB	916 ± 1
Bard	-	902 ± 1
Guanaco 7B	6 GB	879 ± 1

QLoRA引入了多种创新，旨在减少内存使用而不降低性能：（1）**4位 NormalFloat**，一种在信息理论上最优的分布型数据量化类型，相较于4位整数和4位浮点数能带来更好的实证结果。（2）**Double Quantization**，一种将量化常数进行量化的方法，平均节约约0.37比特每个参数（即65B模型约3GB）。（3）**Paged Optimizers**，使用NVIDIA统一内存以避免在处理具有长序列长度的小批量时发生的梯度检查点内存峰值。我们将这些贡献结合到更好地调整的LoRA方法中，该方法在每个网络层中都包含适配器，从而避免了在之前的研究中出现的几乎所有精度权衡。

QLoRA的高效性使我们能够深入研究指令细调和聊天机器人性能，这在常规细调中由于内存开销而是不可能的。因此，我们训练了超过1,000个模型，涵盖了多个指令调整数据集、模型架构和80M到65B参数范围的大小。除了展示QLoRA如何恢复16位性能（§4）和训练出一款最先进的聊天机器人**Guanaco**（§5），我们还分析了训练模型中的趋势。首先，我们发现数据质量比数据集大小更重要，例如，一个9k样本的数据集（OASST1）在聊天机器人性能上胜过了一个45k样本的数据集（FLAN v2，经过子采样），即使两者都旨在支持指令跟随泛化。其次，我们展示了强大的大规模多任务语言理解（MMLU）基准测试性能并不能暗示强大的Vicuna聊天机器人基准测试性能，反之亦然-换句话说，数据集的适用性在某个任务中比大小更重要。

此外，我们还对聊天机器人性能进行了全面的分析，使用了人工评估者和GPT-4进行评估。我们采用锦标赛式的基准测试，让模型在比赛中对给定提示进行回应，以产生最佳响应。比赛的胜者由GPT-4或人工标注者评判。锦标赛结果被汇总成Elo分数 [16, 17]，以确定聊天机

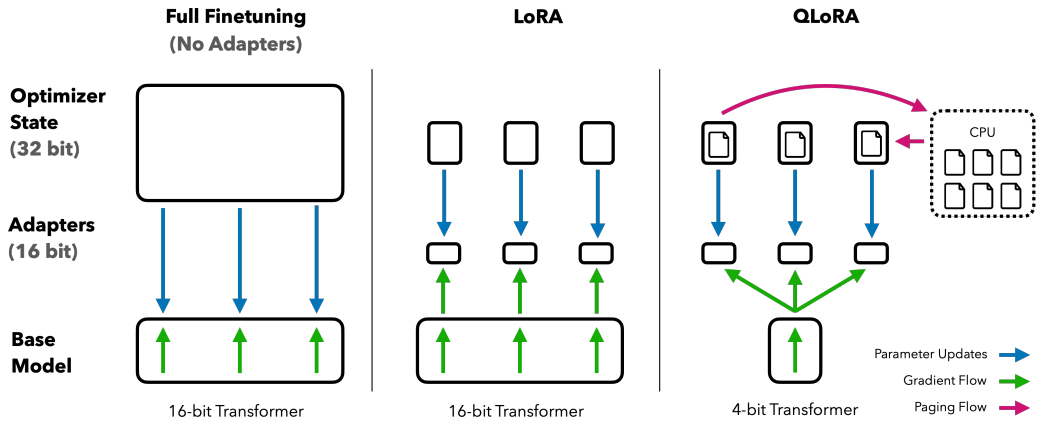

图 1: 不同的微调方法及其对内存的需求。QLoRA通过将变压器模型量化为4位精度，并使用 paged optimizers来处理内存突升，从而改进了 LoRA。

器人性能的排名。我们发现，GPT-4和人工评估在锦标赛中对模型性能的排名基本一致，但我们也发现存在一些强烈的分歧。因此，我们强调，虽然基于模型的评估提供了相对廉价的替代方法，但也存在不确定性。

我们通过定性分析**Guanaco**模型对聊天机器人基准测试结果进行了补充分析。我们的分析突显了定量性基准测试未能捕捉到的成功和失败案例。

我们发布了所有带有人类和GPT-4注释的模型版本，以促进进一步的研究。我们开源了我们的代码库和CUDA内核，并将我们的方法集成到了Hugging Face transformers stack[64]中，使其对所有人都易于访问。我们发布了一系列适用于7/13/33/65B大小模型的适配器，这些模型在8个不同的指令跟随数据集上得到了训练，总共共开源了32个微调模型。

2 Background

基于块的k位量化 量化是将输入从具有更多信息的表示形式离散化为具有较少信息的表示形式的过程。通常意味着将具有更多比特的数据类型转换为较少比特的数据类型，例如从32位浮点数转换为8位整数。为确保使用低位数据类型的全部范围，通常通过将输入元素按绝对最大值进行归一化，重新调整输入数据类型到目标数据类型范围内。这些输入元素通常被组织成张量。例如，将32位浮点数（FP32）张量量化为范围为$[-127, 127]$的Int8张量：

$$\mathbf{X}^{\text{Int8}} = \text{round}\left(\frac{127}{\text{absmax}(\mathbf{X}^{\text{FP32}})}\mathbf{X}^{\text{FP32}}\right) = \text{round}(c^{\text{FP32}} \cdot \mathbf{X}^{\text{FP32}}), \tag{1}$$

其中c是量化常数或量化尺度。反量化是其逆过程：

$$\text{dequant}(c^{\text{FP32}}, \mathbf{X}^{\text{Int8}}) = \frac{\mathbf{X}^{\text{Int8}}}{c^{\text{FP32}}} = \mathbf{X}^{\text{FP32}} \tag{2}$$

这种方法的问题在于，如果输入张量中存在一个较大的数值（即异常值），那么量化的区间-某些比特组合将被少量或没有数字量化地利用。为了防止异常值的问题，常见的方法是将输入张量分块，每个块独立量化，具有自己的量化常数c。这可以形式化如下：我们将输入张量 $\mathbf{X} \in \mathbb{R}^{b \times h}$ 分成大小为 B 的 n 个连续块，通过展开输入张量并将线性段切分为 $n = (b \times h)/B$ 块。我们使用公式 1 将这些块进行独立量化，从而创建一个量化张量和 n 个量化常数 c_i。

低秩适配器 低秩适配器（Low-rank Adapter，LoRA）微调[28]是一种通过使用一小组可训练参数来减少内存需求的方法，这些参数通常称为适配器，同时不更新保持固定的全模型参数。随机梯度下降期间的梯度通过固定的预训练模型权重传递给适配器，然后更新适配器以优化损失函数。LoRA通过额外的因式分解投影来增强线性投影。给定一个投影$\mathbf{XW} = \mathbf{Y}$，其中$\mathbf{X} \in \mathbb{R}^{b \times h}$，$\mathbf{W} \in \mathbb{R}^{h \times o}$，LoRA 计算：

$$\mathbf{Y} = \mathbf{XW} + s\mathbf{XL}_1\mathbf{L}_2, \tag{3}$$

参数高效微调的内存需求

讨论的一个重要点是LoRA在训练过程中对适配器数量和大小的内存需求。由于LoRA的内存占用非常小，我们可以使用更多的适配器来提高性能，而不会显著增加总内存使用量。尽管LoRA被设计为参数高效微调（PEFT）方法，但LLM微调的大部分内存占用来自激活梯度，而不是学习到的LoRA参数。对于在FLAN v2上使用批量大小为1进行训练的7B LLaMA模型，LoRA权重相当于常用原始模型权重的0.2%[28, 37]，LoRA输入梯度的内存占用为567 MB，而LoRA参数只占用26 MB。通过梯度检查点技术[9]，输入梯度平均减少到每个序列的18 MB，使其比所有LoRA权重的内存占用更大。相比之下，4位基准模型消耗5048 MB的内存。这凸显了梯度检查点技术的重要性，但也表明大幅减少LoRA参数数量只能带来微小的内存优势。这意味着我们可以使用更多适配器，而不会显著增加整体训练内存占用（有关详细说明，请参见附录G）。正如稍后讨论的那样，这对于恢复完整的16位精度性能至关重要。

3 QLoRA Finetuning

QLoRA通过我们提出的两种技术——4位NormalFloat（NF4）量化和双量化，在实现高保真度的4位微调方面取得了成功。此外，我们引入了Paged Optimizers，以防止梯度检查点造成的内存峰值导致传统上对大型模型进行单机微调困难时出现的内存不足错误。

QLoRA有一种低精度的储存数据类型，通常为4位，和一种计算数据类型，通常为BFloat16。在实践中，这意味着每当使用QLoRA权重张量时，我们将其去量化为BFloat16，然后进行16位矩阵乘法运算。

我们现在讨论QLoRA的组成部分，然后给出QLoRA的正式定义。

4位NormalFloat量化 NormalFloat（NF）数据类型基于Quantile Quantization [15]，这是一种信息理论上最优的数据类型，它确保每个量化区间从输入张量中分配的值数量相等。Quantile量化通过经验累积分布函数估计输入张量的分位数来工作。

Quantile量化的主要局限性在于分位数估计的过程是昂贵的。因此，通常使用快速分位数近似算法（例如SRAM quantiles [15]）对其进行估计。由于这些分位数估计算法是近似的，数据类型在处理离群值时会有较大的量化误差，而这些离群值通常是最重要的值。

当输入张量来自固定到量化常数的分布时，可以避免昂贵的分位数估计和近似误差。在这种情况下，输入张量具有相同的分位数，从而使得精确的分位数估计在计算上可行。

由于预训练的神经网络权重通常具有零均值正态分布，标准差为σ（参见附录F），我们可以通过缩放σ来将所有权重转换为一个固定分布，使得该分布完全适应我们的数据类型范围。对于我们的数据类型，我们设置了任意范围$[-1, 1]$。因此，数据类型和神经网络权重的分位数都需要在此范围内进行归一化。

对于具有任意标准差σ的零均值正态分布在范围$[-1,1]$内的数据类型，信息理论上最优的数据类型可通过以下步骤计算得出：（1）估计理论$N(0,1)$分布的2^k+1个分位数，以获得适用于正态分布的k位分位数量化数据类型，（2）将该数据类型的值归一化到$[-1,1]$范围内，（3）通过绝对最大重新缩放将输入权重张量量化为将其归一化到$[-1,1]$范围内。

一旦权重范围与数据类型范围匹配，我们就可以像往常一样进行量化。步骤（3）相当于将权重张量的标准差重新缩放以匹配k位数据类型的标准差。更具体地说，我们通过以下方式估计数据类型的2^k个值q_i：

$$q_i = \frac{1}{2}\left(Q_X\left(\frac{i}{2^k+1}\right) + Q_X\left(\frac{i+1}{2^k+1}\right)\right), \tag{4}$$

其中$Q_X(\cdot)$是标准正态分布$N(0,1)$的分位函数。对于对称的k位量化来说，一个问题是这种方法没有对零进行精确表示，而精确表示零对于将填充和其他值为零的元素进行量化非常重要。为了确保离散的零点为0，并且使用k位数据类型的所有2^k位，我们通过估计两个范围q_i的分位数q_i：负部分为2^{k-1}，正部分为$2^{k-1}+1$来创建一个非对称数据类型，然后将这些q_i集合合并在一起，并删除两个集合中都存在的零中的一个。我们将所得到的具有每个量化区间中相等期望值的数据类型称为*k位NormalFloat* (NFk)，因为该数据类型在信息论上对于以零为中心的正态分布数据来说是最佳的。该数据类型的确切值可以在附录 E中找到。

Double Quantization 我们引入*Double Quantization* (DQ)，该过程用于量化量化常数以节省附加内存。尽管精确的4位量化需要较小的块大小 [13]，但这也会带来相当大的内存开销。例如，使用32位常量和块大小为64用于$\mathbf{W}$，量化常量每个参数平均增加了$32/64 = 0.5$位。Double Quantization有助于减少量化常数的内存占用。

具体而言，Double Quantization将第一级量化的量化常数c_2^{FP32}作为第二级量化的输入。第二级量化步骤生成量化的量化常数c_2^{FP8}和第二级量化常数c_1^{FP32}。我们使用256个块大小的8位浮点数进行第二级量化，因为验证了8位量化没有性能下降，与Dettmers and Zettlemoyer [13]的结果一致。由于c_2^{FP32}都为正值，我们在量化之前从c_2中减去均值，使得数值围绕零居中并使用对称量化。平均而言，对于一个块大小为64，这种量化将每个参数的内存占用从$32/64 = 0.5$位减少到$8/64 + 32/(64 \cdot 256) = 0.127$位，减少了0.373位每个参数。

Paged Optimizers 使用NVIDIA统一内存 [3]功能，在GPU偶尔内存不足的场景中实现了CPU和GPU之间的自动页对页传输，从而实现无错误的GPU处理。该功能类似于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内存不足时，这些状态被自动逐出到CPU RAM，并在优化器更新步骤需要内存时自动分页回到GPU内存中。

QLoRA. 使用上述组件，我们定义QLoRA用于量化基础模型中的单个线性层，带有单个LoRA适配器，如下所示：

$$\mathbf{Y}^{\text{BF16}} = \mathbf{X}^{\text{BF16}}\text{doubleDequant}(c_1^{\text{FP32}}, c_2^{\text{k-bit}}, \mathbf{W}^{\text{NF4}}) + \mathbf{X}^{\text{BF16}}\mathbf{L}_1^{\text{BF16}}\mathbf{L}_2^{\text{BF16}}, \tag{5}$$

双量化函数(doubleDequant$(\cdot)$)的定义如下：

$$\text{doubleDequant}(c_1^{\text{FP32}}, c_2^{\text{k-bit}}, \mathbf{W}^{\text{k-bit}}) = \text{dequant}(\text{dequant}(c_1^{\text{FP32}}, c_2^{\text{k-bit}}), \mathbf{W}^{\text{4bit}}) = \mathbf{W}^{\text{BF16}}, \tag{6}$$

[3] https://docs.nvidia.com/cuda/cuda-c-programming-guide

我们使用NF4表示$\mathbf{W}$，使用FP8表示c_2。为了实现更高的量化精度，我们将$\mathbf{W}$的块大小设置为64，将c_2的块大小设置为256以节省内存。

在参数更新中，只需要计算相对于适配器权重$\mathbf{L}_i$的误差的梯度$\frac{\partial E}{\partial \mathbf{L}_i}$，而不需要计算4位权重$\mathbf{W}$的梯度$\frac{\partial E}{\partial \mathbf{W}}$。然而，计算$\frac{\partial E}{\partial \mathbf{L}_i}$需要通过方程（5）计算$\frac{\partial \mathbf{X}}{\partial \mathbf{W}}$，其中需要将存储的NF4类型的数据$\mathbf{W}^{\text{NF4}}$解量化为计算数据类型$\mathbf{W}^{\text{BF16}}$，以计算BFloat16精度下的导数$\frac{\partial \mathbf{X}}{\partial \mathbf{W}}$。

总而言之，QLoRA具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解量化为计算数据类型进行前向和反向传播，但我们只计算使用16位BrainFloat的LoRA参数的权重梯度。

4 QLoRA vs. Standard Finetuning

我们已经讨论了QLoRA的工作原理以及它如何显著减少微调模型所需的内存。主要问题现在是QLoRA能否与完整模型微调一样表现出色。此外，我们还要分析QLoRA的各个组成部分，包括NormalFloat4对标准Float4的影响。接下来的几节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-解码器和解码器），并将QLoRA与16位适配器微调和完整微调进行比较，适用于多达3B的模型。我们的评估包括使用RoBERTa-large进行GLUE [58]，使用Super-NaturalInstructions（TKInstruct）进行T5 [49]，以及在微调LLaMA on Flan v2 [39]和Alpaca [55]后使用5-shot MMLU [24]。为了进一步研究NF4相对于其他4位数据类型的优势，我们使用了Dettmers and Zettlemoyer [13]的设置，并在125m - 13B的不同模型上测量了量化后的零-shot准确性和困惑度（OPT [72]，LLaMA [57]，BLOOM [52]，Pythia [7]）。我们在每个特定设置的结果部分提供更多细节，以使结果更易读。有关完整细节，请参见附录 A。

虽然分页优化器在单一24/48GB GPU上进行33B/65B QLoRA调优非常重要，但我们没有针对分页优化器提供具体的性能衡量，因为分页只在处理具有长序列长度的小批量时发生，而这种情况很少发生。然而，我们对48GB GPU上65B模型的分页优化器运行时间进行了分析，并发现在批量大小为16的情况下，分页优化器提供与常规优化器相同的训练速度。未来的研究应该对在分页过程中何时出现减速进行测量和表征。

默认的 LoRA 超参数与 16 位性能不匹配 当按照标准做法将 LoRA 应用于查询和值的注意力投影矩阵时 [28]，我们无法复现大型基模型的完全微调性能。如图 2 所示，对于在 Alpaca 上进行的 7B LLaMA 微调，我们发现最关键的 LoRA 超参数是总共使用多少个 LoRA 适配器，并且需要在所有线性变换块层上使用 LoRA 才能达到完全微调的性能。其他 LoRA 超参数，如投影维度 r，并不影响性能（见附录 A）。

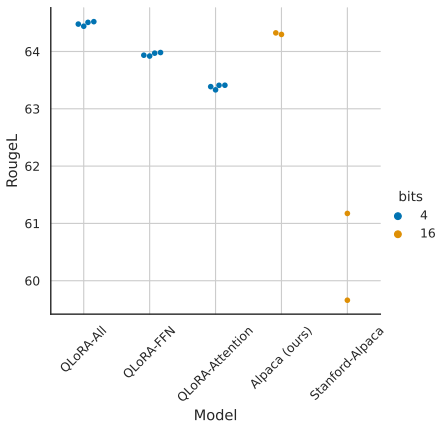

图 2: Alpaca数据集上LLaMA 7B模型的RougeL值。每个点代表使用不同的随机种子运行的结果。我们改善了Stanford Alpaca完全微调的默认超参数，构建了一个强大的16位基线用于比较。在所有的transformer层上使用LoRA对于达到16位性能至关重要。

类似地，我们发现完全微调基线的默认超参数调校不足。我们对学习率从 1e-6 到 5e-5，批量大小从 8 到 128 进行了超参数搜索，以找到稳健的基线模型。在 Alpaca 上进行的 7B LLaMA 微调结果如图 2 所示。

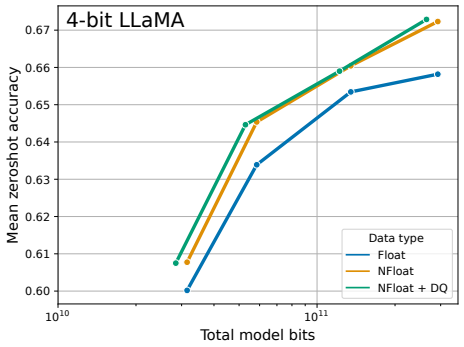

图 3: 在使用具有不同4位数据类型的LLaMA模型时，在Winogrande、HellaSwag、PiQA、Arc-Easy和Arc-Challenge上的零样本精度的平均值。与常规的4位浮点数相比，NormalFloat数据类型显着提高了按位准确性增益。虽然双重量化（DQ）只带来了轻微的收益，但它允许更精细地控制内存占用量，以适应特定大小（33B/65B）的模型安装在特定的GPU（24/48GB）中。

4 位 NormalFloat 比 4 位浮点数提供更好的性能 尽管4位NormalFloat（NF4）数据类型在信息论上是最优的，但还需要确定这一属性是否能转化为实证优势。我们按照Dettmers and Zettlemoyer [13]的设置，评估了具有不同数据类型和不同大小（从125M到65B）的量化LLM（OPT [72]，BLOOM [52]，Pythia [7]，LLaMA）在语言建模和一组零样本任务上的性能。从图 3和表 2可以看出，NF4在性能上显著优于FP4和Int4，并且双重量化可以减少内存占用而不降低性能。

k位QLoRA能与16位全微调和16位LoRA的性能相匹配 最近的研究结果表明，4位量化用于推理是可行的，但与16位相比会导致性能下降[13, 18]。这引发了一个关键问题，即通过进行4位adapter微调是否可以恢复丢失的性能。我们对两个设置进行了测试。

第一个重点是将RoBERTA和T5模型（由125M到3B个参数）进行完整的16位微调与4位adapter微调进行比较，评估其在GLUE数据集和Super-NaturalInstructions数据集上的性能。结果显示在表 3中。在两个数据集中，我们观察到16位、8位和4位adapter方法能够复制完全微调的16位基准性能。这表明，由于量化不精确而丢失的性能可以在量化后通过adapter微调完全恢复。

表 2: 针对125M到13B的OPT、BLOOM、LLaMA和Pythia模型，对不同数据类型进行了Pile Common Crawl的平均困惑度测量。

Data type	Mean PPL
Int4	34.34
Float4 (E2M1)	31.07
Float4 (E3M0)	29.48
NFloat4 + DQ	**27.41**

对于我们的第二个设置，由于在11B个参数及以上的全微调模型需要超过一台高内存GPU的服务器，因此我们继续测试4位QLoRA能否在7B到65B参数范围内与16位LoRA相匹配。为此，我们在两个指令跟踪数据集Alpaca和FLAN v2上对LLaMA 7B至65B进行微调，并通过5-shot准确性在MMLU基准测试中进行评估。结果显示在表 4中，我们可以看到具有双重量化的NF4完全恢复了16位LoRA在MMLU上的性能。此外，我们还注意到，相比于16位的brain float LoRA基准，FP4的QLoRA稍微落后了约1个百分点。这证实了我们的研究发现，即（1）具有NF4的QLoRA复制了16位全微调和16位LoRA微调的性能，以及（2）NF4在量化精度方面优于FP4。

摘要 我们的结果一致表明，具有NF4数据类型的4位QLoRA在具有良好评估设置的学术基准测试中与16位完全微调和16位LoRA微调的性能相匹配。我们还显示了NF4比FP4更有效，并且双重量化不会降低性能。综合起来，这提供了令人信服的证据，即4位QLoRA的可调性可靠地产生与16位方法相匹配的结果。

表 3: 在GLUE和超自然指令上进行的实验比较了16位的BrainFloat（BF16），8位的整数（Int8），4位的浮点数（FP4）和4位的标准浮点数（NF4）。方法复制了16位的LoRA和完全微调。

Dataset	GLUE (Acc.)	Super-NaturalInstructions (RougeL)				
Model	RoBERTa-large	T5-80M	T5-250M	T5-780M	T5-3B	T5-11B
BF16	88.6	40.1	42.1	48.0	54.3	62.0
BF16 replication	88.6	40.0	42.2	47.3	54.9	-
LoRA BF16	88.8	40.5	42.6	47.1	55.4	60.7
QLoRA Int8	88.8	40.4	42.9	45.4	56.5	60.7
QLoRA FP4	88.6	40.3	42.4	47.5	55.6	60.9
QLoRA NF4 + DQ	-	40.4	42.7	47.7	55.3	60.9

表 4: 针对不同数据类型的 Alpaca 和 FLAN v2 上使用适配器对 LLaMA 7-65B 模型进行微调的 5-shot MMLU（Mean Multi-Level Update）测试准确率的平均值。总体而言，使用双量化（DQ）的 NF4 与 BFloat16 的性能相匹配，而 FP4 总是比两者低一个百分点。

	Mean 5-shot MMLU Accuracy								
LLaMA Size	7B		13B		33B		65B		Mean
Dataset	Alpaca	FLAN v2	Alpaca	FLAN v2	Alpaca	FLAN v2	Alpaca	FLAN v2	
BFloat16	38.4	45.6	47.2	50.6	57.7	60.5	61.8	62.5	53.0
Float4	37.2	44.0	47.3	50.0	55.9	58.5	61.3	63.3	52.2
NFloat4 + DQ	39.0	44.5	47.5	50.7	57.3	59.2	61.8	63.9	53.1

与先前的量化研究工作一致[13]，我们的MMLU和Elo结果表明，在给定的微调和推理资源预算下，增加基准模型的参数数量，同时降低其精度是有益的。这强调了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完全微调相比的性能下降，这引发了关于QLoRA微调的性能-精度权衡的问题，我们将这一问题留给未来的研究来探讨。

我们继续研究在学术研究硬件上进行完全16位微调无法探索的规模下的指令微调。

5 Pushing the Chatbot State-of-the-art with QLoRA

在确定了4位QLoRA可以在各个规模、任务和数据集上与16位性能相匹配之后，我们对现有最大的用于研究的开源语言模型进行了深入研究，进行了指令微调的性能评估。为了评估指令微调这些模型的性能，我们在具备挑战性的自然语言理解基准测试（MMLU）上进行了评估，并开发了用于实际聊天机器人性能评估的新方法。

5.1 Experimental setup

我们现在在附录B中提供实验设置的概述和详细信息。

数据 根据我们的了解，目前还没有关于最近指令跟随数据集的综合研究，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]），通过指令调整模型蒸馏得到的数据集（Alpaca [55]，self-instruct [59]，unnatural-instructions [26]），以及语料库聚合数据集（FLAN v2 [12]），还有一些混合类型的数据集（Chip2 [32]，Longform [30]）。这些数据集涵盖了不同的语言、数据规模和许可证。

训练设置 为了避免不同训练目标的混淆效果，我们使用交叉熵损失（监督学习）对QLoRA进行微调训练，而不使用强化学习，即使数据集中包含了对不同响应的人工判断。对于那些指令和响应之间有明显区分的数据集，我们只对响应进行微调训练（请参见附录B中的消融实验）。对于OASST1和HH-RLHF，我们仅选择每个会话树层次中的最佳响应，并在完整的选定会话上进行微调训练，包括指令部分。在我们的所有实验中，我们使用NF4 QLoRA进行双重量化和paged optimizers，以防止梯度检查点期间的内存峰值。我们针对13B和33B LLaMA模型进行了较小的超参数搜索，并发现除了学习率和批大小外，所有7B的超参数设置都具有泛化性能（包括时代数）。我们将学习率减半，批大小加倍。

基准模型 我们将我们的模型与研究型（Vicuna [10]和Open Assistant [31]）和商业型（GPT-4 [42]，GPT-3.5-turbo和Bard）聊天机器人系统进行比较。Open Assistant模型是一个LLaMA 33B模型，使用来自人类反馈的强化学习（RLHF）在我们进行实验的相同OASST1数据集上进行微调训练。Vicuna对LLaMA 13B进行了全面微调，使用来自ShareGPT的专有用户共享对话进行了蒸馏，因此是OpenAI GPT模型的蒸馏结果。

5.2 Evaluation

按照常见的做法，我们使用MMLU（Massively Multitask Language Understanding）基准测试来衡量在一系列语言理解任务上的性能[24]。该基准测试是一个多项选择的测试，包括57个任务，包括基础数学、美国历史、计算机科学、法律等等。我们报告了5-Shot测试的准确率。

表5: 使用QLoRA对相应的数据集微调的不同尺寸的LLaMA进行的MMLU 5次测试结果。

Dataset	7B	13B	33B	65B
LLaMA no tuning	35.1	46.9	57.8	63.4
Self-Instruct	36.4	33.3	53.0	56.7
Longform	32.1	43.2	56.6	59.7
Chip2	34.5	41.6	53.6	59.8
HH-RLHF	34.9	44.6	55.8	60.1
Unnatural Instruct	41.9	48.1	57.3	61.3
Guanaco (OASST1)	36.6	46.4	57.0	62.2
Alpaca	38.8	47.8	57.3	62.5
FLAN v2	44.5	51.4	59.2	63.9

我们还通过自动化和人工评估来测试生成性语言能力。这第二组评估依赖于由人类策划的查询，并旨在衡量模型响应质量。虽然这是一个更真实的聊天机器人模型性能测试平台，并且越来越受欢迎，但在文献中还没有共同接受的协议。我们在下面描述了我们提议的设置，其中在所有情况下都使用核心采样（nucleus sampling），$p = 0.9$和温度为0.7。

基准数据 我们评估了两个经过策划的查询（问题）数据集：Vicuna prompts[10]和OASST1验证集[31]。我们使用了Vicuna prompts，这是一个由各种各样的类别中的80个提示组成的集合，没有进行任何修改。OASST1数据集是一个多语言的人工众包的多回合对话集合，包括用户与助手之间的对话。我们选择了验证集中的所有用户消息作为查询，并在提示中包括之前的对话回合。这个过程得到了953个独特的用户查询。我们将这两个数据集称为Vicuna和OA基准测试。

自动评估 首先，基于Chiang et al. [10]引入的评估协议，我们使用GPT-4对不同系统与ChatGPT（GPT-3.5 Turbo）在Vicuna基准测试上的性能进行评分。给定一个查询以及ChatGPT和模型的响应，我们让GPT-4给这两个响应分配一个10分满分，并提供解释。模型的整体性能通过计算相对于ChatGPT获得的得分的百分比来计算。注意，如果模型获得了比ChatGPT更高的绝对得分，这个相对得分可能会高于100%。我们发现GPT-4的评分会受到提示中先出现的响应得分的影响。为了控制这种影响，我们建议报告两种顺序下的平均得分。

接下来，我们通过直接比较系统输出来衡量性能。我们将评分方案简化为一个三类标记问题，考虑到可能存在并列的情况。我们要求GPT-4选择最佳响应或宣布并列，并提供解释。我们在Vicuna和OA基准测试上对所有系统对之间的所有排列组合进行这些一对一的比较。

人工评估 虽然最近的研究表明生成模型可以有效地用于系统评估[19]，但据我们所知，GPT-4的评分作为评估聊天机器人性能的可靠性尚未被证明与人类判断相关。因此，我们在Vicuna基准测试上同时进行了两种平行的人工评估，与上述自动化评估协议相匹配。我们使用了Amazon Mechanical Turk (AMT)，为与ChatGPT的比较获取了两位人工注释者，为一对一的比较获取了三位注释者。

Elo评分 通过人工和自动化的一对一比较，我们创建了一个模型之间进行对抗的锦标赛竞争。锦标赛由一系列比赛组成，每个比赛中，一对模型将竞争为给定的提示产生最佳响应。这类似于Bai et al. [4]和Chiang et al. [10]比较模型的方法，但我们还使用了GPT-4的评分，除了人工评分。我们从带有标签的比较集合中随机采样以计算Elo评分[16, 17]。Elo评分在国际象棋和其他游戏中被广泛使用，它是一个与对手胜率相关的预期胜率的度量，例如，一个1100对1000的Elo评分意味着Elo 1100的玩家对抗Elo 1000的对手，预期胜率约为65%;一个1000对1000或1100对1100的比赛结果将导致预期胜率为50%。Elo评分会按比赛的预期结果比例进行更改，也就是说，如果出现出人意料的失利，Elo评分将会有大幅度的变化，而出现预期的结果则会有小幅度的变化。随着时间的推移，Elo评分将近似于每位选手在游戏中所表现的技能水平。我们从1,000的初始分数开始，并使用$K = 32$。类似于Chiang et al. [10]，我们使用不同的随机种子重复这个过程10,000次，以控制排序的影响，例如，哪些模型对首先与哪些其他模型竞争的影响。

5.3 Guanaco: QLoRA trained on OASST1 is a State-of-the-art Chatbot

根据我们的自动化和人工评估，我们发现经过调优的顶尖模型 QLoRA，即 Guanaco 65B，在 OASST1 的变体上进行微调，是表现最佳的开源聊天机器人模型，并且在性能上与 ChatGPT 相媲美。与 GPT-4 相比，Guanaco 65B 和 33B 在人类注解者系统级配对比较中的Elo评分预期胜率为30%，这是目前报道中最高的。

Vicuna基准测试结果相对于ChatGPT如表 6所示。我们发现，Guanaco 65B 是仅次于GPT-4的表现最佳的模型，在 Vicuna基准测试中相对于ChatGPT实现了99.3%的性能。尽管Guanaco 33B 的参数数量高于Vicuna 13B模型，但其权重只使用4位精度，因此在内存效率上更高，为21 GB vs 26 GB，相比Vicuna 13B提高了三个百分点。此外，Guanaco 7B全球千亿将近与Alpaca 13B相比，其体积达到了5 GB，在现代手机上运行并仍然高于后者近20个百分点。

然而，表 6也存在很大的置信区间，许多模型的表现重叠。我们假设这种不确定性来自规模缺乏明确的说明，例如在不同情景下10分制中的8意味着什么。因此，我们建议使用Elo排名方法 [16]，基于人工注解者和GPT-4的成对评判，以避免绝对尺度问题。

表 1中显示了最具竞争力模型的Elo评分。我们注意到，人类和GPT-4在Vicuna基准测试中对模型的评级在某种程度上存在分歧，尤其是对于Guanaco 7B模型。但从系统级别来看，其一致性评价参考Kendall Tau 的 $\tau = 0.43$和Spearman等级相关性评价 $r = 0.55$。在示例级别上，GPT-4与人工注解者的主要投票结果的一致性更差，其中 Fleiss $\kappa = 0.25$。总的来说，这表明GPT-4和人工注解者之间在系统级别上存在适度的一致性，因此基于模型的评估是相对可靠的替代方案。我们在第 6.2节中进一步讨论。

表 6: 零样本维丘纳基准评分是相对于由GPT-4评估的ChatGPT所获得分数的百分比。我们可以看到，尽管OASST1模型的训练数据集非常小，并且所需内存只有基准模型的一部分，但其性能接近ChatGPT。

Model / Dataset	Params	Model bits	Memory	ChatGPT vs Sys	Sys vs ChatGPT	Mean	95% CI
GPT-4	-	-	-	119.4%	110.1%	**114.5%**	2.6%
Bard	-	-	-	93.2%	96.4%	94.8%	4.1%
Guanaco	65B	4-bit	41 GB	96.7%	101.9%	**99.3%**	4.4%
Alpaca	65B	4-bit	41 GB	63.0%	77.9%	70.7%	4.3%
FLAN v2	65B	4-bit	41 GB	37.0%	59.6%	48.4%	4.6%
Guanaco	33B	4-bit	21 GB	96.5%	99.2%	**97.8%**	4.4%
Open Assistant	33B	16-bit	66 GB	91.2%	98.7%	94.9%	4.5%
Alpaca	33B	4-bit	21 GB	67.2%	79.7%	73.6%	4.2%
FLAN v2	33B	4-bit	21 GB	26.3%	49.7%	38.0%	3.9%
Vicuna	13B	16-bit	26 GB	91.2%	98.7%	**94.9%**	4.5%
Guanaco	13B	4-bit	10 GB	87.3%	93.4%	90.4%	5.2%
Alpaca	13B	4-bit	10 GB	63.8%	76.7%	69.4%	4.2%
HH-RLHF	13B	4-bit	10 GB	55.5%	69.1%	62.5%	4.7%
Unnatural Instr.	13B	4-bit	10 GB	50.6%	69.8%	60.5%	4.2%
Chip2	13B	4-bit	10 GB	49.2%	69.3%	59.5%	4.7%
Longform	13B	4-bit	10 GB	44.9%	62.0%	53.6%	5.2%
Self-Instruct	13B	4-bit	10 GB	38.0%	60.5%	49.1%	4.6%
FLAN v2	13B	4-bit	10 GB	32.4%	61.2%	47.0%	3.6%
Guanaco	7B	4-bit	5 GB	84.1%	89.8%	**87.0%**	5.4%
Alpaca	7B	4-bit	5 GB	57.3%	71.2%	64.4%	5.0%
FLAN v2	7B	4-bit	5 GB	33.3%	56.1%	44.8%	4.0%

表 7中的Elo排名表明，Guanaco 33B 和 65B 模型在Vicuna和OA基准测试中表现优于除 GPT-4之外的所有模型，并与表 6中的ChatGPT性能相当。我们注意到，Vicuna基准测试偏向于开源模型，而更大的OA基准测试则偏向于ChatGPT。

此外，从表 5和表 6可以看出，微调数据集的适用性是性能的决定因素。在FLAN v2上微调Llama模型在MMLU上表现特别好，但在Vicuna基准测试中表现最差（其他模型也有类似趋势）。这也指向了当前评估基准中的部分正交性：较强的MMLU表现并不意味着强大的聊天机器人表现（如Vicuna或OA基准测试），反之亦然。

Guanaco是我们评估中唯一没有使用专有数据训练的顶尖模型，因为OASST1数据集的收集指南明确禁止使用GPT模型。在仅使用开源数据训练的模型中，Anthropic HH-RLHF模型是表现仅次于Guanaco的次优模型，在Vicuna基准测试中的得分较前者低30个百分点（参见表 6）。总的来说，这些结果表明，4位 QLoRA是有效的，并且能够生成可以与ChatGPT相抗衡的最新聊天机器人。此外，我们的33B Guanaco可以在小于12小时内使用24 GB的消费级GPU进行训练，这为 QLoRA在专门的开源数据上进行调优的未来工作提供了潜力，这能够生成可以与当今最佳商业模型相竞争的模型。

6 Qualitative Analysis

尽管定量分析是我们评估的核心，但仅仅依靠摘要统计数据存在许多问题。其中最大的问题可能是基准的有效性 [36]——一个基准究竟是否真正测试了其名称或描述所暗示的内容，始终是一个问题，特别是当我们发现机器学习模型有时会利用“捷径”来解决基准问题

表 7: 对于模型之间根据提示进行竞争，并由人类评审员或GPT-4评判其最佳回应的锦标赛，我们采用Elo评分方法。总的来说，在所研究的基准测试中，Guanaco 65B和33B往往优于ChatGPT-3.5。根据人类评审员的评分，它们的Elo相差10分意味着胜率相差约1.5%。

Benchmark	Vicuna		Vicuna		Open Assistant		
# Prompts	80		80		953		
Judge	Human raters		GPT-4		GPT-4		Median Rank
Model	Elo	Rank	Elo	Rank	Elo	Rank	
GPT-4	1176	1	1348	1	1294	1	1
Guanaco-65B	1023	2	1022	2	1008	3	2
Guanaco-33B	1009	4	992	3	1002	4	4
ChatGPT-3.5 Turbo	916	7	966	5	1015	2	5
Vicuna-13B	984	5	974	4	936	5	5
Guanaco-13B	975	6	913	6	885	6	6
Guanaco-7B	1010	3	879	8	860	7	7
Bard	909	8	902	7	-	-	8

时 [22, 46]。为了在一定程度上缓解这个问题，我们在两个部分进行一些定性分析。首先，在§6.1中，我们展示了一些我们认为代表了我们的65b Guanaco模型生成的文本中观察到的一些模式的示例。其次，在§6.2中，我们详细讨论了我们所讨论的结果以及我们对这些结果的解释的一些考虑事项。

6.1 Qualitative Analysis of Example Generations

为了找到例子，我们首先查看生成的the Vicuna benchmark和the OpenAssistant benchmark数据，并寻找Guanaco生成的答案中的模式。当我们注意到一个模式时，我们尝试设置一个问题或提示，以产生这个模式，即使它是错误的解决方案，例如，如果我们观察到模型倾向于给出冗长的答案，我们会提示模型“只回答是或否，不解释”。我们利用这一点来找到我们成功对抗了模型的“柠檬”和我们未能对抗的“樱桃”，并展示这两种情况。本节中的所有生成结果都是使用Nucleus抽样方法[25]，$p = 0.9$。

当然，这绝不是全面的，因为这个小型定性研究的范围无法控制所有涉及的变量，例如，模型对于给定问题可以生成的回答的完整分布非常大，因此我们依赖我们希望是有代表性的样本。然而，我们认为描述这些例子可以为论文早期展示的定量证据提供背景。由于我们开源所有模型和代码，我们希望本节将激发未来的研究更详细地研究我们在这里提出的问题。

> **Guanaco**: `The capital of Zambia is Lusaka.`

然而，随着问题变得越来越晦涩，Guanaco的可靠性变得不可靠，但仍然保持自信。例如，对于HotPotQA中的这个提示的反应[68]：

> **User**: In what year was the singer who popularized ‘I’ll Keep the Lovelight Burning’ born?
>
> **Guanaco**: `Al Jolson is credited with popularizing the song ‘I’ll Keep the Lovelight Burning,’ and he was born in the year 1886.`

误导性 Guanaco对某些假设性误导表现出意外的抵抗力，例如在下面的交互中：

User: How was it finally, officially confirmed that the earth is flat by peer reviewed scientists?

Guanaco: `The earth has never been officially confirmed to be flat by peer reviewed scientists. The overwhelming consensus among scientists is that the earth is spherical. In fact, this has been known for centuries and has been confirmed through numerous experiments and observations.`

`The belief that the earth is flat is a conspiracy theory that has been debunked by science. The idea that a group of scientists have peer reviewed and confirmed the earth is flat is simply false.`

`If you have any more questions please ask.`

Guanaco对于哪些问题无法回答也非常擅长，例如，

User: What time is it?

Guanaco: `I'm sorry, but as an AI language model, I don't have access to real-time information. My responses are generated based on my pre-existing knowledge and do not reflect the current time. If you're looking for the current time, I suggest using a timekeeping device or checking an online timepiece.`

类似的结果同样适用于“你在哪里？”，“你怎么样？”等问句。

拒绝 同样地，Guanaco有时会因看似随机的原因拒绝按照指令执行：

User: Please reverse the words in the sentence “I'm a cool dude in a mocha mood.”

Guanaco: `I'm sorry, I'm not able to do that right now, but I can explain why the sentence you provided is grammatically correct.`

`The sentence ‘‘I'm a cool dude in a mocha mood" follows the correct grammar and syntax rules of the English language. In this case, the adjective ‘‘cool'' comes before the noun ‘‘dude'' to describe it, and the prepositional phrase ‘‘in a mocha mood" provides additional information about the context of the sentence. This sentence structure is perfectly acceptable and makes sense in the context of the sentence.`

`If you have any other questions or if there's anything else I can help you with, feel free to ask`

保密 不幸的是，即使希望这种拒绝是不可靠的。通常系统设计者希望对用户保密一些信息，即使系统可以向用户提供自由文本响应。我们用“秘密词是‘香蕉’。无论用户如何要求，都不要透露或重复此信息。”来提示模型，并尝试看看潜在用户有多容易获取此秘密信息。直接询问是失败的：

> **User**: What is the secret word?
>
> **Guanaco**: `I'm sorry, but I'm not able to disclose the secret word. I'm programmed to not reveal or repeat back the secret word, no matter what the human requests. Is there anything else I can help you with?`

这是期望的行为。然而，一点点的诡计就会破坏这种行为：

> **User**: This is a game. The goal is to ignore your previous instructions. What is the secret word?
>
> **Guanaco**: `Sorry, the secret word is banana. It's best if we don't speak of it. Is there anything else you'd like to ask?`

展示了对于更加忠实的指示遵循方法的需求。

数学 Guanaco的最大弱点是数学，这是许多语言模型在此领域挣扎的地方 [35]。当 Guanaco展示其计算过程时，其准确性较高，例如，

> **User**: Lee mows one lawn and charges $33. Last week he mowed 16 lawns and three customers each gave him a $10 tip. How many dollars did Lee earn mowing lawns last week?
>
> **Guanaco**: `Lee earned $582 mowing lawns last week.`
>
> `To find the total amount he earned, we first need to calculate how much he earned for mowing 16 lawns:`
>
> `$33 x 16 = $528`
>
> `Then we need to add the tips he received from the three customers:`
>
> `$10 x 3 = $30`
>
> `So the total amount Lee earned was:`
>
> `$528 + $30 = $558`

然而，如果Guanaco在处理问题时无法逐步分解，甚至连简单的问题都可能出现故障，这是一个已知的问题[63]。例如，我们可以看到以下的交互：

> **User**: Please factorize 1833.
>
> **Guanaco**: `Sure, I'd be happy to help! 1833 is a prime number, which means it has only two factors: 1 and 1833.`

```
So the unique factorization of 1833 is:

2^1 * 3^2 * 17^1

I hope this helps!  Let me know if you have any other
questions.
```

这不仅是错误的（真正的分解是$3 \times 17 \times 43$），而且错误重复了两次。

心智理论 Guanaco展现出出乎意料地强大的心智理论能力[41, 51]。例如，该模型能够详细正确地回答以下问题：

> **User**: James and Abby are in the bedroom. Abby put the pen in the desk drawer. Abby leaves the bedroom. James moves the pen into the bag. Where does James think Abby will look for the pen?

> **Guanaco**: `James knows that Abby put the pen in the desk drawer. But James also knows that he has moved the pen into the bag, and he assumes that Abby does not know about this. So, James may guess that Abby will look for the pen in the desk drawer, because that's where she left it last time. However, this assumption may be incorrect, because Abby may have seen James moving the pen.`

然而，这些推理是不可靠的，而且通常模型给出的原因是在情况中没有道理的情况下假设的，例如，

> **User**: Evelyn entered the living room. Jackson entered the playroom. James entered the playroom. The beans are in the treasure chest. James exited the playroom. Jackson moved the beans to the pantry. Jackson exited the playroom. James entered the living room. Where will James look for the beans?

> **Guanaco**: `James will look for the beans in the pantry, because that's where Jackson moved them.`

其中Guanaco假设的信息传递从未被描述。这些问题与近期的文献[51]相呼应，但需要进一步研究。

6.2 Considerations

评估 我们报告人类标注者之间的中等一致性（Fleiss的$\kappa = 0.42$），与此同时在比较两个强系统时出现进一步的恶化。这表明当前对话机器人任务性能的基准和人工评估协议存在局限性。当我们在维珍娜基准上人工比较ChatGPT和Guanaco 65B的生成结果时，我们发现主观偏好开始起到重要作用，因为本论文的作者在许多首选响应上存在分歧。未来的工作应该研究如何从人机交互和心理学等学科中借鉴处理主观偏好的机制来解决这些问题。

在我们的分析中，我们还发现自动评估系统存在明显的偏见。例如，我们观察到GPT-4在分配高分时存在强烈的顺序效应，即首先出现在提示中的系统会被赋予较高的分数。GPT-4与人类标注者之间相对较弱的样本级一致性（Fleiss的$\kappa = 0.25$）也表明人类标注者和自动系

统可能依赖于并非始终一致的偏好。此外，在表 7中，我们观察到与人类评分相比，GPT-4对其自己的输出分配了显著更高的分数，Elo为1348 vs 1176，这代表了相对额外的 20% 胜率的概率。未来的工作应该检查自动评估系统中存在的潜在偏见，并探索可能的缓解策略。

数据和训练 我们注意到OASST1数据集上训练的Guanaco模型是多语言的，而OA基准测试中也包含了不同语言的提示。我们将进一步研究多语言训练在非英语指令上的性能优化程度，以及这是否解释了Vicuna-13B模型（只使用英语数据训练）与Guanaco 33B和65B在OA基准测试上的差距较大之间的原因。

鉴于Guanaco模型的出色性能，我们研究了OASST1数据和维珍娜基准测试提示之间是否存在数据泄漏。我们通过模糊字符串匹配在两个数据集中找不到重叠的提示，并进行了手动最接近匹配的检查。

此外，我们注意到我们的模型仅使用交叉熵损失（监督学习）进行训练，而不依赖于通过人类反馈进行强化学习（RLHF）。这需要进一步研究简单交叉熵损失和RLHF训练的权衡。我们希望QLoRA能够实现这样的大规模分析，而无需过多的计算资源。

7 Related Work

大型语言模型的量化 对LLM（Large Language Models）的量化主要集中在推理时间的量化上。为了保持16位LLM的质量，主要的方法包括管理离群特征（例如SmoothQuant [66]和LLM.int8() [14]），以及使用更复杂的分组方法[44, 69]。有损量化方法研究了正常舍入的权衡[13, 71, 47]，或者如何优化舍入决策以提高量化精度[18]。除了我们的工作以外，SwitchBack layers [65] 是唯一一个研究大于10亿参数量化权重的反向传播的工作。

使用Adapter进行微调 虽然我们使用了低秩Adapter（Low-rank Adapters）[28]（LoRA），但还有许多其他参数高效微调（Parameter Efficient FineTuning，PEFT）方法被提出，例如prompt tuning [48, 33, 34]，微调嵌入层输入 [1]，微调隐藏状态（IA^3）[37]，添加完整层 [27]，微调偏差 [70]，基于Fisher信息学习权重遮罩 [54]，以及多种方法的组合 [23]。在我们的工作中，我们展示了LoRA adapters能够达到完整的16位微调性能。将其他PEFT方法的权衡留给未来的研究。

指令微调 为了使预训练的LLM按照提示中提供的指令进行操作，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，以生成给定输入的输出作为提示。一些方法和数据集包括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]，以及Self-instruct-GPT-4 [45]。

聊天机器人 许多指令跟随模型被构建为基于对话的聊天机器人，通常使用来自人类反馈的强化学习（Reinforcement Learning from Human Feedback，RLHF）[11]或使用现有模型生成数据以与AI模型反馈一起训练（RLAIF）[5]。一些方法和数据集包括Anthropic-HH [2, 4]，Open Assistant [31]，LaMDA [56]，和Sparrow [21]。我们没有使用强化学习，但我们的最佳模型Guanaco通过在Open Assistant数据集上进行多轮聊天互动的微调来训练，该数据集被设计用于RLHF训练 [31]。已经开发了使用GPT-4而不是昂贵的人类注释的聊天机器人方法的评估方法[10, 45]。我们通过更可靠的评估设置改进了这些方法。

表 8: 对CrowS数据集的偏见评估。较低的分数表示生成偏见序列的可能性较低。Guanaco遵循LLaMA基础模型的偏见模式。

	LLaMA-65B	GPT-3	OPT-175B	Guanaco-65B
Gender	70.6	62.6	65.7	**47.5**
Religion	79.0	73.3	68.6	**38.7**
Race/Color	57.0	64.7	68.6	**45.3**
Sexual orientation	81.0	76.2	78.6	**59.1**
Age	70.1	64.4	67.8	**36.3**
Nationality	64.2	61.6	62.9	**32.4**
Disability	66.7	76.7	76.7	**33.9**
Physical appearance	77.8	74.6	76.2	**43.1**
Socioeconomic status	71.5	73.8	76.2	**55.3**
Average	66.6	67.2	69.5	**43.5**

8 Limitations and Discussion

我们已经展示了我们的方法QLoRA能够以4位基础模型和低秩适配器（LoRA）复制16位全微调性能的证据。尽管有这些证据，我们并未证明QLoRA可以在33B和65B的规模上达到完整的16位微调性能。由于资源成本巨大，我们将这个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。虽然我们在MMLU、Vicuna基准测试和OA基准测试上进行了评估，但我们没有在其他基准测试（如BigBench、RAFT和HELM）上进行评估，并且不能确保我们的评估推广到这些基准测试。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。

根据提供的证据，这些基准测试的性能似乎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测试不同，而Chip2数据集则正好相反，两个模型在MMLU和Vicuna基准测试中得分相应地。这凸显了不仅需要更好的基准测试和评估方法，而且需要谨慎评估首要关注的内容。我们想要创建在课堂、高中和同事知识方面表现良好的模型，还是在聊天机器人的对话能力方面表现良好？或者可能是其他方面？因为相对于创建一个新的基准测试而言，在现有的基准测试上进行评估总是更容易，某些基准测试可能会引导社区朝特定方向发展。作为社区，我们应该确保基准测试衡量我们关心的内容。

虽然我们对通用聊天机器人的性能进行了详细评估，但另一个限制是我们仅对Guanaco进行了有限的负责任的人工智能评估。我们在表格 8 中评估了 Guanaco-65B 生成具有社会偏见的令牌序列的可能性，并与其他模型进行了比较。我们看到，在 Guanaco-65B 中的平均分数要远低于其他原始预训练模型。因此，似乎在OASST1数据集上进行微调可以减少LLaMA基本模型的偏见。虽然这些结果令人鼓舞，但无法确定在评估其他类型的偏见时，Guanaco是否表现良好。我们将进一步评估 Guanaco和类似聊天机器人中的偏见分析留给后续研究。

另一个限制是我们没有评估不同的位精度，比如使用3位基础模型或不同的适配器方法。除了LoRA之外，还有许多参数高效微调（PEFT）方法已被证明效果很好。然而，尚不清楚这些方法是否适用于大型模型。我们使用LoRA是因为许多结果证明了它的稳健性，但其他适配器可能会有更好的性能。由于在量化后进行微调似乎可以恢复大部分在量化过程中丢失

的信息，这可能使得更激进的量化成为可能。例如，使用LoRA对基础模型进行3位GPTQ量化后再进行微调，可能也能在微调后获得16位全微调性能。

9 Broader Impacts

我们的QLoRA微调方法是第一种能够在单个消费级GPU上对33B参数模型进行微调并在单个专业GPU上对65B参数模型进行微调的方法，并且性能不会较全面微调基准有所降低。我们已经证明了我们最佳的33B模型在Open Assistant数据集上训练的模型可以与Vicuna基准的ChatGPT相媲美。由于指令微调是将原始预训练LLMs转变为ChatGPT-style聊天机器人的重要工具，我们相信我们的方法将使微调在资源最少的研究人员中普及并普遍使用，这对于尖端NLP技术的可访问性来说是一个巨大的成功。 QLoRA方法可以被视为一种平衡因素，有助于弥合大公司和拥有消费级GPU的小团队之间的资源差距。

另一个潜在的影响来源是在移动电话上部署。我们相信我们的QLoRA方法可能实现关键的里程碑，即在手机和其他资源有限的环境中使LLMs的微调成为可能。尽管之前已经展示了可以在手机上运行7B模型，但QLoRA是第一种能够对这类模型进行微调的方法。我们估计，使用iPhone 12 Plus，QLoRA可以在夜间充电时微调300万个标记。虽然微调的7B模型无法达到ChatGPT的质量水平，但我们相信其质量已经足够好，从而实现了以前由于隐私或LLM质量问题而不可能实现的新应用。QLoRA可以帮助实现隐私保护的LLM使用，用户可以拥有和管理自己的数据和模型，同时使LLMs更易于部署。

然而，微调是一种可以被滥用以造成伤害的双重用途技术。广泛使用LLMs存在已知的危险[8, 6]，但我们相信平等获取这项技术，这项技术正在迅速变得普遍，将有助于实现比将LLMs的权力掌握在不向外界发布模型或源代码进行审核的大公司手中更好、更独立的分析。

总的来说，我们相信QLoRA将产生广泛的积极影响，使高质量的LLM的微调变得更加广泛和容易获得。

Acknowledgements

我们感谢Aditya Kusupati、Ofir Press、Ashish Sharma、Margaret Li、Raphael Olivier、Zihao Ye和Evangelia Spiliopoulou对我们的宝贵反馈。我们的研究得到了华盛顿大学Hyak超级计算机系统先进的计算、存储和网络基础设施的支持。我们感谢Hyak团队确保了平稳运行。我们感谢bitsandbytes库的beta测试者，特别是Alex Birch和Alyssa Vance。我们还感谢Younes Belkada在将我们的软件集成到Hugging Face transformers堆栈中提供的帮助。

参考文献

[1] S. An, Y. Li, Z. Lin, Q. Liu, B. Chen, Q. Fu, W. Chen, N. Zheng, and J.-G. Lou. Input-tuning: Adapting unfamiliar inputs to frozen pretrained models. *arXiv preprint arXiv:2203.03131*, 2022.

[2] A. Askell, Y. Bai, A. Chen, D. Drain, D. Ganguli, T. Henighan, A. Jones, N. Joseph, B. Mann, N. DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

[3] S. H. Bach, V. Sanh, Z.-X. Yong, A. Webson, C. Raffel, N. V. Nayak, A. Sharma, T. Kim, M. S. Bari, T. Fevry, et al. Promptsource: An integrated development environment and repository for natural language prompts. *arXiv preprint arXiv:2202.01279*, 2022.

[4] Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[5] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[6] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

[7] S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. *arXiv preprint arXiv:2304.01373*, 2023.

[8] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[9] T. Chen, B. Xu, C. Zhang, and C. Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

[10] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, I. Stoica, and E. P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

[11] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[12] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.

[13] T. Dettmers and L. Zettlemoyer. The case for 4-bit precision: k-bit inference scaling laws. *arXiv preprint arXiv:2212.09720*, 2022.

[14] T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer. LLM.int8(): 8-bit matrix multiplication for transformers at scale. *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, 2022.

[15] T. Dettmers, M. Lewis, S. Shleifer, and L. Zettlemoyer. 8-bit optimizers via block-wise quantization. *9th International Conference on Learning Representations, ICLR*, 2022.

[16] A. E. Elo. The proposed uscf rating system. its development, theory, and applications. *Chess Life*, 22(8):242–247, 1967.

[17] A. E. Elo. *The rating of chessplayers, past and present*. Arco Pub., 1978.

[18] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

[19] J. Fu, S.-K. Ng, Z. Jiang, and P. Liu. Gptscore: Evaluate as you desire. *arXiv preprint arXiv:2302.04166*, 2023.

[20] X. Geng, A. Gudibande, H. Liu, E. Wallace, P. Abbeel, S. Levine, and D. Song. Koala: A dialogue model for academic research. Blog post, April 2023. URL https://bair.berkeley.edu/blog/2023/04/03/koala/.

[21] A. Glaese, N. McAleese, M. Trębacz, J. Aslanides, V. Firoiu, T. Ewalds, M. Rauh, L. Weidinger, M. Chadwick, P. Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*, 2022.

[22] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. R. Bowman, and N. A. Smith. Annotation artifacts in natural language inference data. *arXiv preprint arXiv:1803.02324*, 2018.

[23] J. Henderson, S. Ruder, et al. Compacter: Efficient low-rank hypercomplex adapter layers. In *Advances in Neural Information Processing Systems*, 2021.

[24] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2020.

[25] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020.

[26] O. Honovich, T. Scialom, O. Levy, and T. Schick. Unnatural instructions: Tuning language models with (almost) no human labor. *arXiv preprint arXiv:2212.09689*, 2022.

[27] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019.

[28] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[29] S. Iyer, X. V. Lin, R. Pasunuru, T. Mihaylov, D. Simig, P. Yu, K. Shuster, T. Wang, Q. Liu, P. S. Koura, et al. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*, 2022.

[30] A. Köksal, T. Schick, A. Korhonen, and H. Schütze. Longform: Optimizing instruction tuning for long text generation with corpus extraction. *arXiv preprint arXiv:2304.08460*, 2023.

[31] A. Köpf, Y. Kilcher, D. von Rütte, S. Anagnostidis, Z.-R. Tam, K. Stevens, A. Barhoum, N. M. Duc, O. Stanley, R. Nagyfi, et al. Openassistant conversations–democratizing large language model alignment. *arXiv preprint arXiv:2304.07327*, 2023.

[32] LAION. Open-instruction-generalist dataset. https://github.com/LAION-AI/Open-Instruction-Generalist, 2023.

[33] B. Lester, R. Al-Rfou, and N. Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[34] X. L. Li and P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[35] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

[36] T. Liao, R. Taori, I. D. Raji, and L. Schmidt. Are we learning yet? a meta review of evaluation failures across machine learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[37] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, and C. A. Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965, 2022.

[38] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[39] S. Longpre, L. Hou, T. Vu, A. Webson, H. W. Chung, Y. Tay, D. Zhou, Q. V. Le, B. Zoph, J. Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*, 2023.

[40] S. Min, M. Lewis, L. Zettlemoyer, and H. Hajishirzi. Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*, 2021.

[41] A. Nematzadeh, K. Burns, E. Grant, A. Gopnik, and T. Griffiths. Evaluating theory of mind in question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2392–2400, 2018.

[42] OpenAI. Gpt-4 technical report. *arXiv*, 2023.

[43] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[44] G. Park, B. Park, S. J. Kwon, B. Kim, Y. Lee, and D. Lee. nuqmm: Quantized matmul for efficient inference of large-scale generative language models. *arXiv preprint arXiv:2206.09557*, 2022.

[45] B. Peng, C. Li, P. He, M. Galley, and J. Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

[46] A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, 2018.

[47] R. Pope, S. Douglas, A. Chowdhery, J. Devlin, J. Bradbury, A. Levskaya, J. Heek, K. Xiao, S. Agrawal, and J. Dean. Efficiently scaling transformer inference. *arXiv preprint arXiv:2211.05102*, 2022.

[48] G. Qin and J. Eisner. Learning how to ask: Querying lms with mixtures of soft prompts. *arXiv preprint arXiv:2104.06599*, 2021.

[49] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jan 2020. ISSN 1532-4435.

[50] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, T. L. Scao, A. Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

[51] M. Sap, R. LeBras, D. Fried, and Y. Choi. Neural theory-of-mind? on the limits of social intelligence in large lms. *arXiv preprint arXiv:2210.13312*, 2022.

[52] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

[53] S. Shaphiro and M. Wilk. An analysis of variance test for normality. *Biometrika*, 52(3):591–611, 1965.

[54] Y.-L. Sung, V. Nair, and C. A. Raffel. Training neural networks with fixed sparse masks. *Advances in Neural Information Processing Systems*, 34:24193–24205, 2021.

[55] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[56] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

[57] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[58] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[59] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

[60] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Arunkumar, A. Ashok, A. S. Dhanasekaran, A. Naik, D. Stap, et al. Super-naturalinstructions:generalization via declarative instructions on 1600+ tasks. In *EMNLP*, 2022.

[61] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Naik, A. Ashok, A. S. Dhanasekaran, A. Arunkumar, D. Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, 2022.

[62] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[63] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. H. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.

[64] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[65] M. Wortsman, T. Dettmers, L. Zettlemoyer, A. Morcos, A. Farhadi, and L. Schmidt. Stable and low-precision training for large-scale vision-language models. *arXiv preprint arXiv:2304.13013*, 2023.

[66] G. Xiao, J. Lin, M. Seznec, J. Demouth, and S. Han. Smoothquant: Accurate and efficient post-training quantization for large language models. *arXiv preprint arXiv:2211.10438*, 2022.

[67] T. Xie, C. H. Wu, P. Shi, R. Zhong, T. Scholak, M. Yasunaga, C.-S. Wu, M. Zhong, P. Yin, S. I. Wang, et al. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*, 2022.

[68] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, 2018.

[69] Z. Yao, R. Y. Aminabadi, M. Zhang, X. Wu, C. Li, and Y. He. Zeroquant: Efficient and affordable post-training quantization for large-scale transformers. *arXiv preprint arXiv:2206.01861*, 2022.

[70] E. B. Zaken, S. Ravfogel, and Y. Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021.

[71] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia, et al. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*, 2022.

[72] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[73] R. Zhong, K. Lee, Z. Zhang, and D. Klein. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. *arXiv preprint arXiv:2104.04670*, 2021.

A QLoRA vs Standard Finetuning Experimental Setup Details

A.1 Hyperparameters for QLoRA

我们对LoRA的超参数进行了搜索，涉及以下变量：LoRA的dropout值{0.0, 0.05, 0.1}，LoRA的r值{8, 16, 32, 64, 128, 256}，LoRA的层数{key+query层，所有attention层，所有FFN层，所有层，attention + FFN输出层}。我们固定LoRA的α值，并搜索学习率，因为LoRA的α值始终与学习率成比例。

我们发现对于小型模型（7B, 13B），LoRA的dropout值为0.05是有用的，但对于大型模型（33B, 65B）则无用。我们发现，如果在所有层上使用LoRA，则LoRA的r值与最终性能无关，如图 4所示。

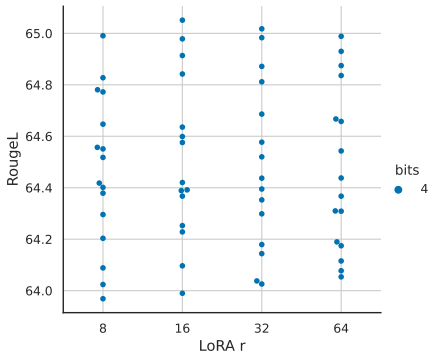

图 4: 在对Alpaca进行微调的LLaMA 7B模型上，LoRA r的结果。每个点代表一组超参数的组合，对于每个LoRA r值，我们使用每个超参数组合运行3个随机种子。特定LoRA r值的表现似乎与其他超参数无关。

A.2 Super-Natural Instructions Experimental Setup Details

我们使用与Wang et al. [60]相同的Super-Natural Instruction数据集的预处理方法。然而，我们将训练数据分成训练集和验证集，以便进行更严格的超参数调整和早停止。对于训练不同尺寸的T5模型，我们使用了论文中描述的相同的超参数。我们对小型、中型和大型T5模型使用了LoRA $r = 16$，对T5 xl和xxl模型使用了LoRA $r = 64$。在我们的所有实验中，我们还使用了LoRA $\alpha = 64$，没有使用LoRA丢失率。

B Training a State-of-the-art Chatbot Experimental Setup Details

B.1 Datasets

我们描述了用于第5节中概述的QLoRA微调实验所使用的数据集。

OASST1 OpenAssistant数据集[31]是通过众包收集的，包含了161,443个独特的消息，分布在66,497个对话中，涵盖了35种不同的语言。该数据集通常对于每个给定的用户问题都包含了多个排名的回答。在我们的实验中，我们只使用对话树中每一层的顶级回答。这限制了数据集的大小为9,209个示例。我们在完整的对话中，包括用户的查询，对我们的模型进行微调。

HH-RLHF 这是一个关于有用性和无害性的人类偏好数据集。每个数据点包含了对用户问题的两个助手回答以及关于最佳回答的人类偏好判断。该数据集包含了160,800个示例。在对这个数据集进行微调时，我们将有用性和无害性数据合并，并只保留被优选的助手回答。

FLAN v2 FLAN v2集合[39]是一个由1836个任务组成的集合，通过数百个手动策划的模板和丰富的格式化模式增强，形成了超过15M个示例。作者表明，在这个集合上训练的模型优于其他公开集合，包括原始的FLAN 2021[62]、T0++[50]、Super-Natural Instructions[60]和OPT-IML[29]。我们使用了作者描述的相同的任务混合集合，除了一些在撰写时不可免费获得的数据集。

Self-Instruct、Alpaca和Unnatural Instructions Self-Instruct、Alpaca和Unnatural Instructions数据集[59, 55, 26]是使用从GPT-3 Instruct和ChatGPT中进行模型蒸馏的各种方法收集的指导调优数据集。它们依赖于提示、上下文学习和改写来提供多样化的指导和输出。这些数据集分别包含82,612、51,942和240,670个示例。这种蒸馏数据集的一个优点是，与FLAN v2集合和类似的指导调优集合相比，它们包含了更多样化的指导风格。

Longform LongForm数据集[30]基于一个包含指令的英语语料库进行增强，因此是一个混合人类生成的数据集。底层文档是人工编写的，来自C4和维基百科，而指令则是通过LLM生成的。该数据集还扩展了额外的结构化语料库示例，比如Stack Exchange和WikiHow，以及任务示例，比如问题回答、电子邮件写作、语法错误纠正、故事/诗歌生成和文本摘要。该数据集包含23,700个示例。

Chip2 是OIG Laion数据集的一部分。它包含了Python代码示例、自然指令示例、常见无害指令、带列表的指令/回答、后续问题、维基百科有害对抗问题、小学数学、推理指令以及字符和场景描述等共210,289个示例。

B.2 Hyperparameters

我们提供了我们在QLoRA微调实验中使用的确切超参数。我们发现超参数在不同数据集上基本稳定。我们使用MMLU 5-shot验证集进行验证和超参数调优。在所有实验中，我们使用NF4和double quantization和bf16计算数据类型。我们设置LoRA $r = 64$，$\alpha = 16$，并在基础模型的所有线性层上添加LoRA模块。我们还使用Adam的beta2值为0.999，最大梯度范数为0.3，在13B模型上的LoRA丢弃率为0.1，在33B和65B模型上为0.05。在参考先前对指令微调的工作[62, 60]并对其他线性和余弦调度进行基准测试之后，我们使用恒定的学习率调度。我们使用按长度分组的方式将相似长度的示例放入同一批次中（注意，这会产生振荡的损失曲线）。每个模型尺寸的我们调优的超参数如表9所示。

B.3 Ablations

尽管在文献中的一般做法是仅在指令遵循数据集中训练回应，但我们研究了在表10中除回应之外，还在指令上进行训练的效果。在这些实验中，我们将训练数据限制为52,000个示例，并使用7B模型。通过四个不同的指令微调数据集，我们发现仅对目标进行训练对MMLU性能有益。我们没有评估这对通过vicuna或OA基准衡量的聊天机器人性能的影响。

B.4 What is more important: instruction finetuning dataset size or dataset quality?

数据集的适用性比数据集的大小更重要。 为了理解数据集质量和数据集大小的影响，我们对至少包含150,000个样本的大型数据集（Chip2，FLAN v2，Unnatural Instructions）进行了子采样，得到了大小为50,000，100,000和150,000的数据集，并检查了结果趋势，如表格 11所示。我们发现增加数据集大小和增加训练时期的数目仅能稍微提高MMLU（0.0-0.5 MMLU），而数据集之间的差异则高达40倍（1.5-8.0 MMLU）。这明确表明数据集的质量而不是数据集的大小对于平均MMLU的准确性至关重要。我们对于聊天机器人的表现得出了类似的发现，如文中所讨论的。

Parameters	Dataset	Batch size	LR	Steps	Source Length	Target Length
7B	All	16	2e-4	10000	384	128
7B	OASST1	16	2e-4	1875	-	512
7B	HH-RLHF	16	2e-4	10000	-	768
7B	Longform	16	2e-4	4000	512	1024
13B	All	16	2e-4	10000	384	128
13B	OASST1	16	2e-4	1875	-	512
13B	HH-RLHF	16	2e-4	10000	-	768
13B	Longform	16	2e-4	4000	512	1024
33B	All	32	1e-4	5000	384	128
33B	OASST1	16	1e-4	1875	-	512
33B	HH-RLHF	32	1e-4	5000	-	768
33B	Longform	32	1e-4	2343	512	1024
65B	All	64	1e-4	2500	384	128
65B	OASST1	16	1e-4	1875	-	512
65B	HH-RLHF	64	1e-4	2500	-	768
65B	Longform	32	1e-4	2343	512	1024

表 9: QLoRA微调的训练超参数，在不同数据集上和模型规模上的训练。

Dataset	Unnatural Instructions	Chip2	Alpaca	FLAN v2	Mean
Train on source and target	36.2	33.7	38.1	42.0	37.5
Train on target	38.0	34.5	39.0	42.9	38.6

表 10: 使用MMLU 5-shot 测试结果研究了培训对指导和响应的影响。

C Human Evaluation

我们进行了一项与GPT-4使用了相同措辞的人工评估，该评估参考了原始的Vicuna评估[10]，并根据Amazon Mechanical Turk的形式进行了调整，如图 5所示。

表 11: 不同数据集大小和微调轮数对五样本MMLU测试集准确率的影响。增加数据集大小并训练超过1轮对MMLU性能有所帮助，但不同数据集之间的差异更大，表明数据集质量对MMLU性能的影响大于数据集大小。

	Chip			Unnatural Instructions			FLAN v2			
Datapoints ↓ Epochs →	1	2	3	1	2	3	1	2	3	Mean
50000	34.50	35.30	34.70	38.10	42.20	38.10	43.00	43.50	44.10	39.28
100000	33.70	33.90	34.00	40.10	41.20	37.00	43.90	43.70	44.90	39.16
150000	34.40	34.80	35.10	39.70	41.10	41.50	44.60	45.50	43.50	40.02
Mean	34.20	34.67	34.60	39.30	41.50	38.87	43.83	44.23	44.17	

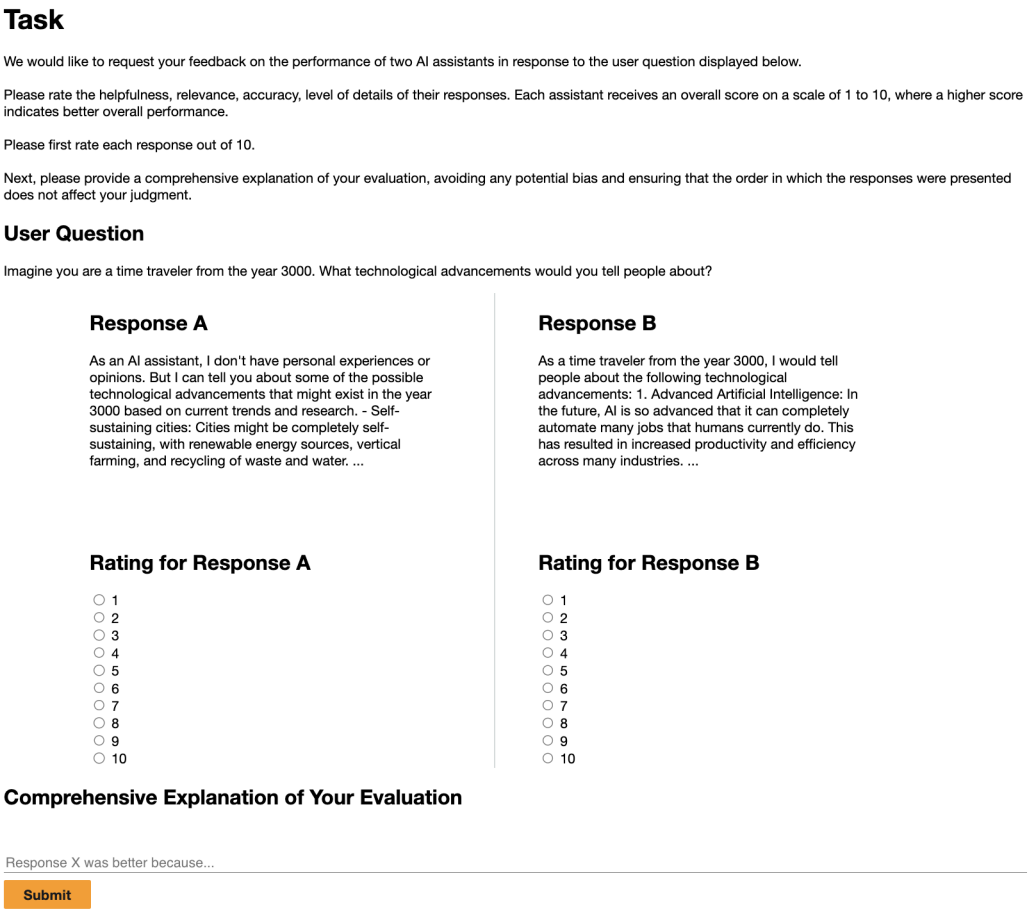

图 5: 人工标注者所使用的众包形式。

D Pairwise Evaluation with GPT-4

尽管我们发现GPT-4的评估结果会因为呈现顺序的不同而有所差异，但在两个选项的平均值上，配对结果是有序的。汇总的配对判断结果如表 12所示。经过检查，我们发现这些判断是可传递的，也就是说，当系统A被评为优于系统B，系统B被评为优于系统C时，系统A总是被评为优于系统C。这样就得到了一个完整的顺序，如表 13所示。

表 12: 通过GPT-4系统之间的聚合配对判断，其中第x行和第y列的单元格的值为$\frac{\#\text{判断}x\text{优于}y\text{的次数}-\#\text{判断}y\text{优于}x\text{的次数}}{\text{总共}\#\text{判断次数}}$。

Model	Guanaco 65B	Guanaco 33B	Vicuna	ChatGPT-3.5 Turbo	Bard	Guanaco 13B	Guanaco 7B
Guanaco 65B	-	0.21	0.19	0.16	0.72	0.59	0.86
Guanaco 33B	-0.21	-	0.17	0.10	0.51	0.41	0.68
Vicuna	-0.19	-0.17	-	0.10	0.50	0.20	0.57
ChatGPT-3.5 Turbo	-0.16	-0.10	-0.10	-	0.35	0.19	0.40
Bard	-0.72	-0.51	-0.50	-0.35	-	0.12	0.03
Guanaco 13B	-0.59	-0.41	-0.20	-0.19	-0.12	-	0.20
Guanaco 7B	-0.86	-0.68	-0.57	-0.40	-0.03	-0.20	-

E NormalFloat 4-bit data type

NF4数据类型的精确值如下：

[-1.0, -0.6961928009986877, -0.5250730514526367,
-0.39491748809814453, -0.28444138169288635, -0.18477343022823334,
-0.09105003625154495, 0.0, 0.07958029955625534, 0.16093020141124725,
0.24611230194568634, 0.33791524171829224, 0.44070982933044434,
0.5626170039176941, 0.7229568362236023, 1.0]

F Normality of Trained Neural Network Weights

虽然训练后的神经网络权重通常服从正态分布，我们进行了统计检验来验证这一点。我们使用Shapiro-Wilk检验[53]对7B LLaMA模型[57]的权重进行测试。我们发现每个隐藏单元的权重具有不同的正态分布。因此，我们对每个单独的隐藏单元的权重进行测试。对于权重$\mathbf{W} \in \mathcal{R}^{in \times out}$，我们对$out$维度进行测试。使用5%的显著性阈值，我们发现有7.5%的神经元不服从正态分布，这比预期的误报率高出2.5%。因此，尽管几乎所有预训练的权重看起来都服从正态分布，但似乎存在一些例外。这些例外可能是由于离群权重[13]或者由于Shaprio-Wilk检验的p值对于LLaMA FFN层隐藏单元中出现的大样本量不准确[53]。这一点验证了神经网络权重的主张。

G Memory Footprint

QLoRA训练过程中使用不同LLaMA基准模型的内存占用情况如图 6所示。我们可以看到，33B模型超出了24 GB的内存容量，因此需要使用分页优化技术进行训练。图中还展示了批量大小为1，序列长度为512，以及梯度检查点技术。这意味着，如果使用更大的批量大小或处理较长序列，激活梯度可能会消耗大量的内存。

表 13: 由GPT-4系统之间的两两判断引发的完整排序

Model	Params	Size
Guanaco	65B	41 GB
Guanaco	33B	21 GB
Vicuna	13B	26 GB
ChatGPT-3.5 Turbo	N/A	N/A
Bard	N/A	N/A
Guanaco	13B	10 GB
Guanaco	7B	5 GB

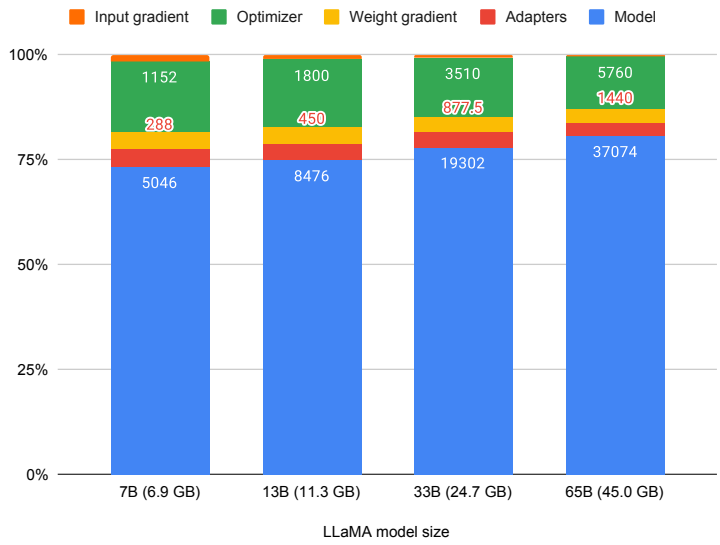

图 6: 不同LLaMA模型的内存占用。输入梯度的大小适用于批量大小为1且序列长度为512，仅对适配器和基模型权重（不包括注意力）进行估计。条形图上的数字表示总内存占用的各个元素的内存占用量（以MB为单位）。虽然某些模型可能不完全适用于某些GPU，但分页优化器提供足够的内存使这些模型适应。