A QLoRA vs Standard Finetuning Experimental Setup Details

A.1 Hyperparameters for QLoRA

We do a hyperparameter search for LoRA over the following variables: LoRA dropout { 0.0, 0.05, 0.1}, LoRA r { 8, 16, 32, 64, 128, 256}, LoRA layers {key+query, all attention layers, all FFN layers, all layers, attention + FFN output layers}. We keep LoRA α fixed and search the learning rate, since LoRA α is always proportional to the learning rate.

We find that LoRA dropout 0.05 is useful for small models (7B, 13B), but not for larger models (33B, 65B). We find LoRA r is unrelated to final performance if LoRA is used on all layers as can be seen in Figure 4

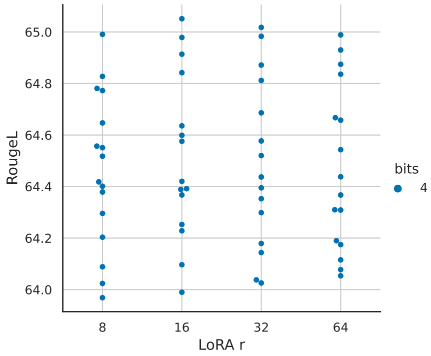


Figure 4: LoRA r for LLaMA 7B models finetuned on Alpaca. Each dot represents a combination of hyperparameters and for each LoRA r we run 3 random seed with each hyperparameter combination. The performance of specific LoRA r values appears to be independent of other hyperparameters.

A.2 Super-Natural Instructions Experimental Setup Details

We use the same preprocessing of the Super-Natural Instruction dataset as Wang et al. [60]. However, we split the training data in training and validation datasets allowing us to perform more rigorous hyperparameter tuning and early stopping. We use the same hyperparameters described in the paper for training the various T5 model sizes on the Super-Natural Instruction data. We use LoRA $r = 16$ for small, medium, and large T5 models and LoRA $r = 64$ for T5 xl and xxl models. We also use LoRA $\alpha = 64$ in all our experiments and no LoRA dropout.

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

B.1 Datasets

We describe the datasets used for QLoRA finetuning experiments outlined in Section 5.

OASST1 The OpenAssistant dataset [31] was collected via crowd-sourcing. It contains 161,443 unique messages distributed across 66,497 conversations and spanning 35 different languages. The dataset often contains several ranked replies for each given user question. In our experiments, we only use the top reply at each level in the conversation tree. This limits the dataset to 9,209 examples. We finetune models on the full conversation including the user queries.

HH-RLHF This is a human preference dataset about helpfulness and harmlessness. Each datapoint consists of two assistant replies to a user question along with a human preference judgment of the best reply. The dataset contains 160,800 examples. When finetuning on this dataset, we combine helpfulness and harmlessness data and only keep the preferred assistant reply.

FLAN v2 The FLAN v2 collection [39] is a collection of 1836 tasks augmented with hundreds of manually curated templates and rich formatting patterns into over 15M examples. The authors show that models trained on this collection outperform other public collections including the original FLAN 2021 [62], T0++ [50], Super-Natural Instructions [60], and OPT-IML [29]. We used the same task mixtures described by the authors with the exception of some datasets that were not freely available at the time of writing.

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

Table 9: Training hyperparameters for QLoRA finetuning on different datasets and across model sizes.

Self-Instruct, Alpaca, Unnatural Instructions The Self-Instruct, Alpaca, and Unnatural Instructions datasets [59, 55, 26] are instruction tuning datasets collected with various approaches of model distillation from GPT-3 Instruct and ChatGPT. They rely on prompting, in-context learning, and paraphrasing to come up with diverse sets of instructions and outputs. The datasets comprise of 82,612, 51,942, and 240,670 examples respectively. One advantage of such distilled datasets is that they contain a more diverse set of instruction styles compared to the FLAN v2 collection and similar instruction tuning collections.

Longform The LongForm dataset [30] is based on an English corpus augmented with instructions and as such is a hybrid human-generated dataset. The underlying documents are human-written and come from C4 and Wikipedia while the instructions are generated via LLMs. The dataset is extended with additional structured corpora examples such as Stack Exchange and WikiHow and task examples such as question answering, email writing, grammar error correction, story/poem generation, and text summarization. The dataset contains 23,700 examples.

Chip2 is part of the OIG Laion dataset. It contains Python code examples, natural instruction examples, generic harmless instructions, instruction/responses with lists, follow-up questions, Wikipedia toxic adversarial questions, grade school math, reasoning instructions, and character and scene descriptions with a total of 210,289 examples.

B.2 Hyperparameters

We provide the exact hyperparameters used in our QLoRA finetuning experiments. We find hyperparameters to be largely robust across datasets. We use the MMLU 5-shot dev set for validation and hyperparameter tuning. In all our experiments we use NF4 with double quantization and bf16 computation datatype. We set LoRA $r = 64$, $\alpha = 16$, and add LoRA modules on all linear layers of the base model. We also use Adam beta2 of 0.999, max grad norm of 0.3 and LoRA dropout of 0.1 for models up to 13B and 0.05 for 33B and 65B models. Following previous work on instruction finetuning [62, 60] and after benchmarking other linear and cosine schedules, we use a constant learning rate schedule. We use group-by-length to group examples of similar lengths in the same batch (note this will produce a oscillating loss curve). The hyperparameters we tune for each model size are shown in Table 9.

B.3 Ablations

While it is general practice in the literature to only train on the response in instruction following datasets, we study the effect of training on the instruction in addition to the response in Table 10. In these experiments, we restrict the training data to 52,000 examples and use the 7B model. Over four different instruction tuning datasets, we find that only training on the target is beneficial to MMLU

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

Table 10: MMLU 5-shot test results studying the effect of training on the instructions in addition to the response.

performance. We did not evaluate the effect this may have on chatbot performance as measured by vicuna or OA benchmarks.

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

Data set suitability is more important than dataset size. To understand the effects of dataset quality vs. dataset size, we experiment with subsampling large datasets with at least 150,000 samples (Chip2, FLAN v2, Unnatural Instructions), into datasets of size 50,000, 100,000 and 150,000 and examine the resulting trends, as shown in Table 11. We find that increasing the dataset size and increasing the number of epochs improves MMLU only marginally (0.0 - 0.5 MMLU), while the difference between datasets is up to 40x larger (1.5 - 8.0 MMLU). This is a clear indicator that dataset quality rather than dataset size is critical for mean MMLU accuracy. We obtain similar findings for chatbot performance, with the most successful dataset for training, OASST1, containing less than 10k examples after processing.

C Human Evaluation

We conduct a human evaluation with the same wording given to GPT-4 in the original Vicuna evaluation [10], adjusted for an Amazon Mechanical Turk form as show in Figure 5.

D Pairwise Evaluation with GPT-4

While we found that the GPT-4 evaluation gave different results depending on which system was presented first, when averaged over both options the pairwise results were well-ordered. The aggregated pairwise judgments are shown in Table 12. On inspection, it is clear these judgments are transitive, i.e., when System A is judged better than System B and System B is judged better than System C, it is always the case that System A is judged better than System C. This yields a complete ordering, given in Table 13.

E NormalFloat 4-bit data type

The exact values expressible in the NF4 data type are as follows:

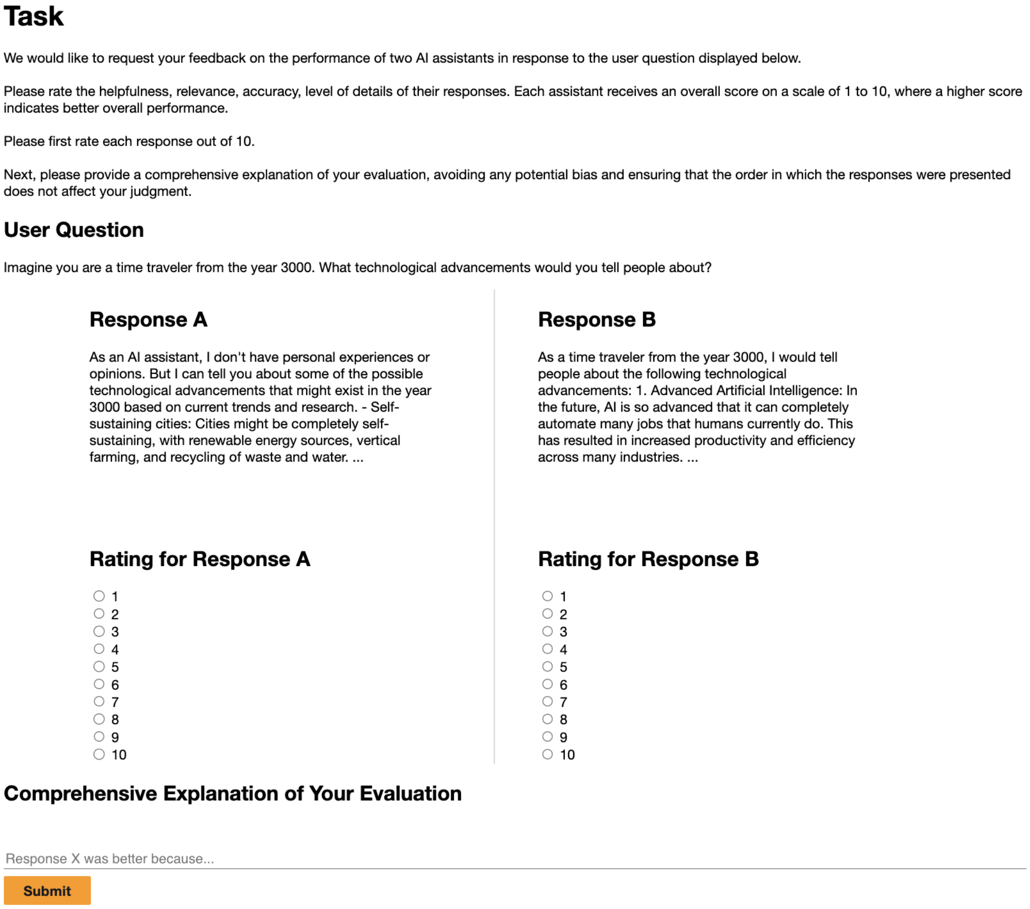
[-1.0, -0.6961928009986877, -0.5250730514526367,
-0.39491748809814453, -0.28444138169288635, -0.18477343022823334,
-0.09105003625154495, 0.0, 0.07958029955625534, 0.16093020141124725,
0.24611230194568634, 0.33791524171829224, 0.44070982933044434,
0.5626170039176941, 0.7229568362236023, 1.0]

F Normality of Trained Neural Network Weights

While it is commonly assumed that trained neural network weights are mostly normally distributed, we perform statistical testing to verify this. We use the Shapiro-Wilk test [53] on the weights of

Table 11: Effect different dataset sizes and finetuning epochs on mean 5-shot MMLU test set accuracy. While increasing the dataset size and training for more than 1 epochs helps with MMLU performance, the difference between datasets are far larger, indicating that dataset quality affects MMLU performance more than dataset size.

		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	



Task

We would like to request your feedback on the performance of two AI assistants in response to the user question displayed below.

Please rate the helpfulness, relevance, accuracy, level of details of their responses. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance.

Please first rate each response out of 10.

Next, please provide a comprehensive explanation of your evaluation, avoiding any potential bias and ensuring that the order in which the responses were presented does not affect your judgment.

User Question

Imagine you are a time traveler from the year 3000. What technological advancements would you tell people about?

Response A

As an AI assistant, I don't have personal experiences or opinions. But I can tell you about some of the possible technological advancements that might exist in the year 3000 based on current trends and research. - Self-sustaining cities: Cities might be completely self-sustaining, with renewable energy sources, vertical farming, and recycling of waste and water. ...

Rating for Response A

○ 1
○ 2
○ 3
○ 4
○ 5
○ 6
○ 7
○ 8
○ 9
○ 10

Response B

As a time traveler from the year 3000, I would tell people about the following technological advancements: 1. Advanced Artificial Intelligence: In the future, AI is so advanced that it can completely automate many jobs that humans currently do. This has resulted in increased productivity and efficiency across many industries. ...

Rating for Response B

○ 1
○ 2
○ 3
○ 4
○ 5
○ 6
○ 7
○ 8
○ 9
○ 10

Comprehensive Explanation of Your Evaluation

Response X was better because...

Figure 5: The crowdsourcing form used by human annotators.

the 7B LLaMA model [57]. We find that the weights of each hidden unit have different normal distributions. As such, we test the weights of each individual hidden unit. This means for weight $\mathbf{W} \in \mathcal{R}^{in \times out}$ we perform tests over the out dimension. Using a 5% significance threshold, we find that 7.5% of neurons are non-normally distributed which is about 2.5% more than the expected false-positive rate. As such, while almost all pretrained weights appear to be normally distributed there seem to be exceptions. Such exceptions might be due to outliers weights [13] or because the p-value of the Shaprio-Wilk test is not accurate for large sample sizes [53] that occur in the LLaMA FFN layer hidden units.

Table 12: Aggregated pairwise GPT-4 judgments between systems where the value of a cell at row x and column y is $\frac{\text{\# judgment } x \text{ is better than } y - \text{\# judgment } y \text{ is better than } x}{\text{total \# number of judgments}}$

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	-

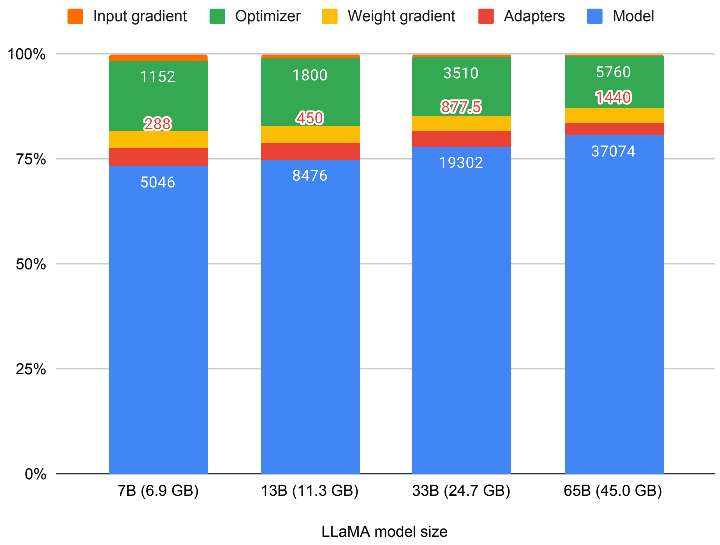


Figure 6: Breakdown of the memory footprint of different LLaMA models. The input gradient size is for batch size 1 and sequence length 512 and is estimated only for adapters and the base model weights (no attention). Numbers on the bars are memory footprint in MB of individual elements of the total footprint. While some models do not quite fit on certain GPUs, paged optimizers provide enough memory to allow these models to fit.

G Memory Footprint

The memory footpring for QLoRA training with different LLaMA base models can be seen in Figure 6. We see that the 33B model does not quite fit into a 24 GB and that paged optimizers are needed to train it. Depicted is also batch size 1 with a sequence length of 512 and gradient checkpointning. This means, if one uses a larger batch size, or if a long sequence is processed, the activation gradient might consume a considerable amount of memory.

Table 13: The complete ordering induced by pairwise GPT-4 judgments between systems

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