RE-ADAPT: REVERSE ENGINEERED ADAPTATION OF LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce RE-ADAPT, an approach to fine-tuning large language models on new domains without degrading any pre-existing instruction-tuning. We reverse engineer an adapter which isolates what an instruction-tuned model has learned beyond its corresponding pretrained base model. Importantly, this requires no additional data or training. We can then fine-tune the base model on a new domain and readapt it to instruction following with the reverse engineered adapter. RE-ADAPT and our low-rank variant LORE-ADAPT both outperform other methods of fine-tuning, across multiple popular LLMs and datasets, even when the models are used in conjunction with retrieval-augmented generation.

1 INTRODUCTION

Large Language Models (LLMs) require a significant investment to develop and train, requiring resources available to only a limited number of organizations. For instance, Meta's Llama-3 family of models was trained using two custom-built compute clusters, each containing 24,000 high-end GPUs (Meta, 2024). Parameter Efficient Fine Tuning (PEFT) enables resource-efficient downstream customization of existing LLMs for new domains by adjusting a relatively small number of parameters while keeping the majority unchanged. However, an important distinction exists between the types of model used for further fine-tuning. It is common for LLM producers to release two versions of a model, one which is *pretrained* on a general task such as next-token prediction and an *instruct* version which is then continued trained on annotated data to learn how to follow instructions or respond to queries in a preferential manner (Touvron et al., 2023; Jiang et al., 2023; Almazrouei et al., 2023; Banks & Warkentin, 2024). The availability of both versions introduces a choice for organizations wanting to adapt a model to their custom task or domain.

While an instruction-tuned model is generally more capable for popular tasks, the majority of data available for additional fine-tuning is unlabeled, lacking the annotations expected from instruct models. This poses a significant problem as annotation by the downstream organization can be too difficult, expensive, or error-prone (Fredriksson et al., 2020; Desmond et al., 2021). Additional fine-tuning can also degrade the performance of the instruction-tuned model outside of the new fine-tuning distribution (Kotha et al., 2024). On the other hand, pretrained models can be easily fine-tuned with unlabeled text but lack the additional capabilities of their instruct counterparts.

To address this dilemma, we seek the ability to fine-tune existing LLMs on unlabeled text while retaining the capabilities from pre-existing instruction-tuning. We draw inspiration from *adapters*, sets of learnable parameters added to an existing model for fine-tuning (Bapna & Firat, 2019; Houlsby et al., 2019). We make the key observation that **the difference in weights between an instruction-tuned and corresponding pretrained model is effectively an adapter**. Isolating the information learned from instruction-tuning into this *Reverse Engineered (RE)-Adapter* enables fine-tuning of the pretrained model without directly impacting the instruction-tuning. The fine-tuned model can then be readapted with the instruction following capabilities (Figure 1).

We apply our RE-ADAPT approach to fine-tune several popular LLM variants with unlabeled data from new domains. We use both closed- and open-book question-answering to measure their ability to incorporate the new domain-specific information without decreasing the previous capabilities out-of-domain. Specifically, we:
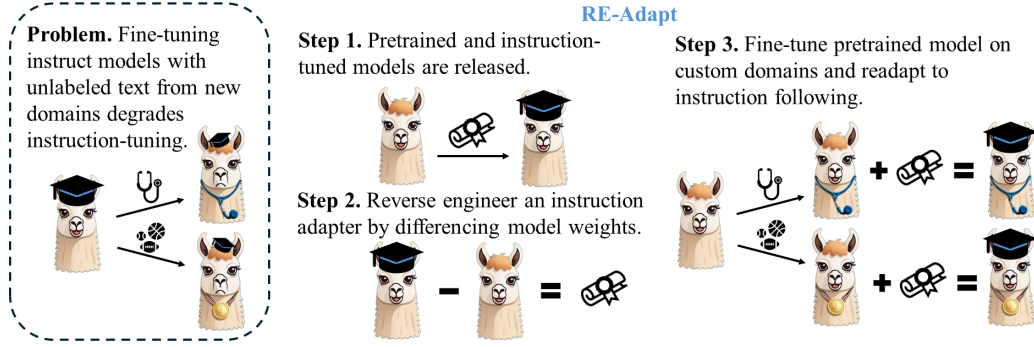
Figure 1: In RE-ADAPT, an *instruction adapter* is isolated by differencing weights between instruct (🎓) and pretrained (🎓) versions of a model and reapplied to the pretrained model after fine-tuning.

- Explore the differences in parameters between pretrained and instruct models and their use as instruction adapters;
- Quantify RE-ADAPT's effectiveness to leverage unstructured knowledge for question answering in new domains under both context-free and retrieval-augmented scenarios;
- Introduce *partial adaptation*, a technique for scaling the strength of adapters for fine-grain control of knowledge priorities; and
- Demonstrate that RE-Adapters are *effectively* low-rank, showing that low-rank RE-Adapters (LoRE-Adapters) are capable of similar performance using up to 5x fewer parameters.

2 BACKGROUND

2.1 ADAPTERS

Adapters (Bapna & Firat, 2019; Houlsby et al., 2019) have played an important role in the context of transfer learning for language models in recent years, particularly for fine-tuning pretrained models which are too large to fully train on commodity hardware. The concept introduced by Houlsby et al. (2019) provides a lightweight alternative to full fine-tuning through the augmentation of models with small modular sets of trainable parameters. Adapters have been useful for enabling the use of pretrained models on new tasks (Pfeiffer et al., 2021; Karimi Mahabadi et al., 2021; Rücklé et al., 2021), new domains (Malik et al., 2023; Schopf et al., 2023; Diao et al., 2023), and adapting to multiple languages (Chronopoulou et al., 2023b; Üstün et al., 2022; Parovic et al., 2023).

Low-Rank Adapters (LoRA) (Hu et al., 2022) are a particularly parameter efficient adaptation technique which adds a low-rank matrix to the weights of existing layers. Because the adapter is low-rank it can be represented as the product of two much smaller matrices, significantly lowering the number of trainable parameters. Weight-Decomposed Low-Rank Adaptation (DoRA) is an extension to LoRA with superior performance and similar efficiency (Liu et al., 2024). Liu et al. (2024) achieve this by decomposing the pretrained weights into both magnitude and direction components, applying LoRA for directional fine-tuning only. Important to this work, adapters learned with either LoRA or DoRA can be represented as a single matrix which captures the information learned during fine-tuning. The pretrained model is then adapted by simply adding the new matrix to the existing weights. We leverage DoRA to fine-tune our models on a new domain, and take inspiration from the additive nature of these techniques to derive our reverse engineered adapters.

Several approaches have been developed which utilize the mixing or combination of adapters to benefit from diverse tasks or domains Pfeiffer et al. (2021); Rücklé et al. (2021); Wang et al. (2022); Chronopoulou et al. (2023a); Fleshman et al. (2024); Zadouri et al. (2024) or for parameter efficient federated learning (Babakniya et al., 2023; Sun et al., 2024). One method to categorize these approaches is by the mechanism used for combining the adapters. Either a weighted combination of adapters is applied to the base model (Chronopoulou et al., 2023a; Fleshman et al., 2024; Babakniya et al., 2023; Sun et al., 2024) or another set of parameters are used to learn adapter interactions

(Pfeiffer et al., 2021; Rücklé et al., 2021; Wang et al., 2022; Zadouri et al., 2024). We focus on the former, as we reframe instruction-tuned models as the summation of a pretrained model with an instruction adapter. We add new knowledge by combining domain-specific and instruction adapters via linear combination. As highlighted by Sun et al. (2024), separate adapters can be incompatible when averaged. Chronopoulou et al. (2023a) and Fleshman et al. (2024) try to mitigate this by initializing adapters with the same random weights, and Sun et al. (2024) by doing the same through a data driven approach. Neither option is applicable here, as we have no control over the instruction adapter. This motivates our new approach for *partial adaptation* which we introduce in Section 3.

2.2 INSTRUCT MODELS

Some of the most capable LLMs are *instruct* variants, pretrained on massive amounts of unannotated text and further trained on curated datasets with a combination of instruction-tuning (Mishra et al., 2022; Wei et al., 2022; Ouyang et al., 2022b; Sanh et al., 2022) and Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Stiennon et al., 2020). For example, Llama-3 was pretrained on 15T tokens and the instruct version continued training with a combination of supervised fine tuning (SFT), rejection sampling, proximal policy optimization (PPO), and direct preference optimization (DPO) (Meta, 2024). Open-source LLM producers generally release both the instruct versions as well as the pretrained models from which they were derived (Jiang et al., 2023; Almazrouei et al., 2023; Banks & Warkentin, 2024; Meta, 2024). Access to the pretrained LLM allows users to customize the model to a new task or domain while taking advantage of the large investment required for pretraining. Fine-tuning the instruct model directly is generally avoided due to *catastrophic-forgetting*, a phenomenon where models lose previous abilities with subsequent rounds of continued training (McCloskey & Cohen, 1989; Kotha et al., 2024). This is unfortunate, as few organizations have the resources to conduct fine-tuning at the scale of the original instruction-tuned models. In this work, we explore methods of fine-tuning LLMs which take advantage of both the pretraining and instruction-tuning of existing LLMs. We specifically design our approach to minimize forgetting while fine-tuning instruction-capable models with unlabeled text.

2.3 MODEL ARITHMETIC

Previous works have looked at the ability to arithmetically manipulate models to isolate certain behaviors (Ilharco et al., 2023; Mitchell et al., 2024). Ilharco et al. (2023) constructed *task vectors* by differencing weights between a pretrained model and several corresponding models each fine-tuned for a particular task. They observed for their models that task vectors are almost orthogonal to each other, preventing interference and allowing combinations of the vectors for negating certain behaviors, improving multi-task performance, or performing well on new tasks via more complicated task analogies (Ilharco et al., 2023). We similarly solve for our reverse engineered adapter with a simple differencing, but using a single LLM fine-tuned for multi-task instruction-following. By effectively isolating instruction-tuning into an adapter, we allow further fine-tuning of pretrained models, maximizing knowledge acquisition before readapting their ability to follow instructions. We introduce an optional step for reducing the rank of our RE-Adapter, lowering memory requirements while maintaining or improving performance in some scenarios. Unlike task vectors, our multi-purpose RE-Adapters are not assumed to be orthogonal to new training domains. We introduce a technique for mitigating potential interference in Section 3 by controlling the adaptation strength.

Mitchell et al. (2024) developed an alternative approach for isolating pretraining knowledge from fine-tuned behaviors which they call *emulated fine-tuning*. Instead of differencing model weights, emulated fine-tuning considers the difference in outputs between pretrained and fine-tuned versions of a model. By combining this difference with the output of a larger pretrained model, Mitchell et al. (2024) found that they could benefit from the additional pretraining knowledge while still solving the task of the smaller model. Their technique could be extended to meet our goal but requires the storage and forward pass of multiple models for inference. Our approach isolates new knowledge and instruction-following into separate adapters, merged into a single model at no extra cost.

3   PARTIAL ADAPTATION

We detail our main methods in Section 4, but first we introduce a technique for controlling the strength of adaptation. Consider a model with weights $\mathbf{W}$ and an adapter $\mathbf{A}$ used to fine-tune the model on a new domain. Using additive adapters such as LoRA or DoRA, the combined weights:

$$\hat{\mathbf{W}} = \mathbf{W} + \mathbf{A} \tag{1}$$

are then used for inference (Hu et al., 2022; Liu et al., 2024). We make the observation that the resulting model assigns equal weight to the original parameters and the new adapter, which is generally trained with significantly less data than the original weights. This potentially leads to overfitting in the new domain and degradation of performance in the general setting. These issues compound in situations where multiple adapters are combined. Both Chronopoulou et al. (2023a) and Fleshman et al. (2024) discuss complications arising from mixing adapters, especially if they were not initialized with the same values to encourage compatibility.

To mitigate these challenges we propose a technique for *partial adaptation* which introduces a post-hoc scaling factor for each fine-tuned adapter. Importantly, Equation 1 is still used during fine-tuning, but inference becomes:

$$\hat{\mathbf{W}} = \mathbf{W} + \lambda \mathbf{A} \tag{2}$$

where $0 \leq \lambda \leq 1$ is used to scale the strength of adaptation. In our experiments, we find that partial adaptation improves performance when using either single or multiple combined adapters.

4   REVERSE ENGINEERED ADAPTATION

Here we describe Reverse Engineered Adaptation (RE-ADAPT), our approach to solve the challenge of updating an instruction-tuned model with unlabeled text without degrading the ability of the model to follow instructions. In Section 5, we demonstrate the effectiveness of this approach for closed-book and retrieval-augmented question answering over new domains.

4.1   RE-ADAPTERS

First consider two language models: $\mathbf{T_\Phi}$, which has been pretrained with parameters $\mathbf{\Phi}$; and $\mathbf{T_\Theta}$, having the same architecture as $\mathbf{T_\Phi}$ but with parameters $\mathbf{\Theta}$ updated from the pretrained parameters $\mathbf{\Phi}$ via instruction-tuning. Given these models, we can solve for the RE-Adapter parameters $\mathbf{\Delta}$ using:

$$\mathbf{\Delta} = \mathbf{\Theta} - \mathbf{\Phi} \tag{3}$$

to isolate the information learned during instruction-tuning. Next, we augment the pretrained model $\mathbf{T_\Phi}$ with a learnable adapter $\mathbf{\Psi}$ and fit $\mathbf{T_{\Phi+\Psi}}$ on a new domain by only updating the adapter weights $\mathbf{\Psi}$. We refer to $\mathbf{\Psi}$ as the *knowledge adapter*. We utilize DoRA to fit $\mathbf{\Psi}$ in our experiments, but any fine-tuning approach is applicable. We construct our final model $\mathbf{T_\Omega}$ with parameters:

$$\mathbf{\Omega} = \mathbf{\Phi} + \alpha\mathbf{\Psi} + \beta\mathbf{\Delta} \tag{4}$$

where α and β are the scaling factors for the partial adaptation of $\mathbf{\Psi}$ and $\mathbf{\Delta}$ respectively. We find that scaling down the strength of the knowledge adapter $\mathbf{\Psi}$ and RE-Adapter $\mathbf{\Delta}$ with partial adaptation leads to better performance in instruction-based tasks related to the new domain while maintaining or slightly improving on the performance of the original instruction-tuned model out-of-domain. Except for measuring their sensitivity, we simply set α and β to an equal weighting of 0.5 in our experiments.

4.2   LORE-ADAPTERS

Inspired by LoRA, we explore the intrinsic dimensionality of RE-Adapters and their ability to be represented by low-rank approximations. The Eckart-Young-Mirsky theorem establishes the truncated singular value decomposition (SVD) as the best low-rank approximation of matrices under the Frobenius norm (Eckart & Young, 1936). We compute the SVD of the RE-Adapter $\mathbf{\Delta}$ from Equation 3 which yields $\mathbf{\Delta} = \mathbf{USV^\intercal}$ with the diagonal of $\mathbf{S}$ containing the singular values of $\mathbf{\Delta}$ sorted by magnitude, and with $\mathbf{U}$ and $\mathbf{V}$ the corresponding left and right singular vectors. We then

compute the percentage of variance explained by each dimension by squaring the singular values and re-normalizing the results to sum to 1. The cumulative explained variance v at rank k is then:

$$v_k = \Sigma_{i=0}^{k} \frac{\sigma_i^2}{\Sigma \sigma_j^2} \tag{5}$$

where σ_i is the ith singular value. We replicate this analysis for multiple modern LLMs and find that the majority of total variation in parameters can be represented at low-rank. For example, Figure 2 displays the cumulative explained variance plots for three layers from the RE-Adapter derived from Llama-3, where we see more than half of the variance in these layers can be captured by a rank 128 approximation. This suggests the potential for a low-rank RE-Adapter (LoRE-Adapter).
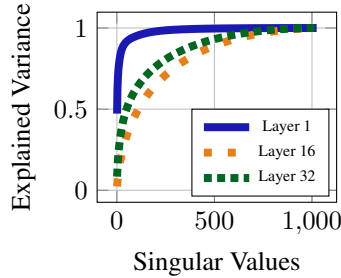

Figure 2: Cumulative explained variance for singular values from Llama-3 RE-ADAPT *k_proj* layers.
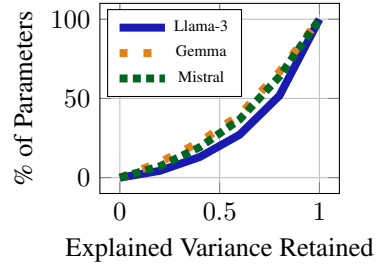
Figure 3: Percent of parameters used for LORE-ADAPT with varying threshold of explained variance.

We can convert a RE-Adapter into a LoRE-Adapter by representing each layer with its truncated SVD (Sharma et al., 2024). Specifically, we truncate each layer to the rank that captures a total explained variance above a user-defined threshold τ. Figure 3 shows the relationship between τ and the reduction in total parameters when using Llama-3 models to derive the adapter. As τ increases we maintain a higher percentage of the original parameters. We use LoRE-Adapters with $\tau = 0.5$ for the experiments in this work and see similar or better performance when compared to RE-ADAPT while using up to 5x less parameters. Like LoRA, the savings in memory is beneficial in cases where several LoRE-Adapters are swapped in and out of the same model.

5 EXPERIMENTS

We quantify the effectiveness of RE-ADAPT using question-answering (QA), a task suitable for measuring knowledge acquisition from unlabeled data for which instruction-tuned models perform better than their pretrained counterparts due to significant portions of instruction-following datasets being QA-based (Ouyang et al., 2022a; Wang et al., 2023; Taori et al., 2023). Specifically, we want to see if RE-ADAPT is better than alternatives for adding knowledge from data not annotated with question-answer pairs. We would like the resulting model to do well answering questions about the new domains, while maintaining the level of performance of the original instruction-tuned model when answering unrelated questions.

5.1 MODELS

We replicate all experiments using the pretrained and instruct versions from the Gemma-7B (Banks & Warkentin, 2024), Llama-3-8B (Meta, 2024), and Mistral-7B (Jiang et al., 2023) family of LLMs using the HuggingFace API (Wolf et al., 2020). We utilize the parameter efficient fine-tuning library (Mangrulkar et al., 2022) for adding DoRA (Liu et al., 2024) knowledge adapters to each of these models. We perform all fine-tuning and inference with a single 80GB A100 GPU. We include hyper-parameters and other details of our fine-tuning in Appendix A.

Figure 4: RE-ADAPT enables the addition of new knowledge to an instruction-tuned model, without degrading capabilities on knowledge from pretraining.

In Section 5 we compare RE-ADAPT and LoRE-ADAPT with the pretrained and instruct models of each family, as well as pretrained and instruct models fine-tuned with DoRA on the new domains. We perform experiments for closed-book QA as well as QA with retrieval-augmented generation (RAG).

5.2 DATA

Kotha et al. (2024) showed that fine-tuning degrades performance outside of the fine-tuning distribution. We hypothesize that our approach mitigates this issue by isolating existing instruction-tuning from additional fine-tuning. We test this by measuring the changes in question-answering performance when various fine-tuning strategies are used to update models with unlabeled data. An optimal approach would benefit from the new knowledge when asked related questions, without losing the ability to answer unrelated questions.

We explore this hypothesis by fine-tuning models in two different settings. We use English WMT News Crawl (Kocmi et al., 2022) articles published in the year 2020 as our first fine-tuning distribution.[1] These articles provide non-annotated information which we capture through DoRA adapters trained for next-token-prediction. We evaluate how well this knowledge is acquired by using the resulting models to answer related questions from the StreamingQA dataset (Liška et al., 2022), which contains 21,681 QA pairs derived from our subset of articles.[2]

We use the evidence passages from RetrievalQA (Zhang et al., 2024) as our second fine-tuning distribution and measure performance on the corresponding questions from the same dataset.[3] Zhang et al. (2024) curated the dataset by compiling the subset of questions from five other QA benchmarks for which GPT-4 (OpenAI et al., 2024) is unable to answer without access to external knowledge. The questions were selected with the goal of having the corresponding knowledge absent from current LLMs, making this dataset especially challenging in the closed-book setting.

To measure any performance degradation from fine-tuning, we also evaluate our models using a short-answer subset of the Natural Questions dataset (Kwiatkowski et al., 2019) which is unrelated to either fine-tuning distribution.[4] We use these questions as a control to measure performance before and after fine-tuning our models on the other domains. We would like our approach to result in improved performance when answering questions related to the fine-tuning data without a reduction in performance on the unrelated Natural Questions Figure 4.

5.3 EVALUATION

We observe that instruction-tuned models will generally answer questions in long form, often repeating the question and providing additional helpful context. An example of this behavior is shown in Table 1 where the model is asked for the number of episodes in a popular television series. Here we see the reference answer is 291, which Llama-3 gets correct, but with a response containing full sentences and additional information to clarify its position.

---

[1] Available at `https://data.statmt.org/news-crawl/README` under CC0 license.

[2] Available at `https://github.com/google-deepmind/streamingqa` under CC-BY 4.0 license.

[3] Available at `https://huggingface.co/datasets/zihanz/RetrievalQA` under MIT license.

[4] Available at `https://huggingface.co/datasets/natural_questions` under CC-BY-SA 3.0 license.

Table 1: Example from Natural Questions with a truncated response. Llama-3's full response includes more details per country.

Question	how many episodes are there in dragon ball z?
Answer	291
Llama-3	There are a total of 291 episodes in the original Japanese version of Dragon Ball Z. However, the episode count can vary depending on the version and the country.

Popular QA metrics such as Rouge-L (Lin, 2004) or exact match would penalize Llama-3 for not being precise. To alleviate this concern we evaluate using Rouge-L's recall, which is the percentage of the longest common sub-sequence of the reference answer found in the model's response. We additionally measure a version of exact match which looks for the exact reference answer anywhere in the response. In both cases, if the reference answer is in the response the score will be 1. If the answer is partially correct then exact match will be 0, but Rouge-L will provide partial credit.

5.4 CLOSED-BOOK QA

In our first experiment, we conduct QA evaluation in a closed-book setting where the models must provide an answer given nothing but the question. We explore how RE-ADAPT behaves in this setting with varying partial adaptation scaling factors. Figure 5 shows the QA performance of Llama-3 using a fixed factor of 1.0 for the knowledge adapter with varying scaling factors for the RE-Adapter. We find that partial adaptation with a factor of 0.5 for both the knowledge adapter and instruction adapter provides

Figure 5: StreamingQA performance as RE-Adapter is added to fine-tuned Llama-3 model with varying strengths.

robust results across models and datasets when using both RE-ADAPT and LoRE-ADAPT. We use an explained variance threshold $\tau = 0.5$ for our LoRE-Adapters. The resulting percentage of original parameters for each model are: Llama-3 (19.2%), Gemma (30.2%), and Mistral (27.1%).

The closed-book performance of all models across datasets is shown in Table 2. Both RE-ADAPT and LoRE-ADAPT outperform the pretrained and instruction-tuned models on StreamingQA and RetrievalQA, even when those models are fine-tuned on the corresponding News Crawl or RetrievalQA passages. As expected, the pretrained models perform worse, although fine-tuning on the unlabeled data does improve the QA ability of both pretrained and instruct models in the domain used for adaptation. These in-domain results suggest that our approach is superior for knowledge acquisition. Next, we will discuss the impact fine-tuning has on general QA performance by looking at results on the out-of-domain Natural Questions dataset used as a control.

The closed-book results for the Natural Questions dataset on the right side of Table 2 demonstrate the issues with fine-tuning instruct models with non-annotated data, resulting in models that perform worse in their original setting. While fine-tuning on News Crawl or Retrieval QA passages improved the instruct models on the corresponding QA datasets, the majority of models saw a decrease in performance on Natural Questions. RE-ADAPT alleviates this problem by using the data from the new domain to only fine-tune the pretrained model, keeping the instruction-tuning intact. Using our approach, the resulting models performed significantly better on the fine-tuning distribution without a performance degradation out-of-domain. In fact, **RE-ADAPT and LoRE-ADAPT performed better than the original instruction-tuned models out-of-domain**. This improvement indicates that instruction-tuning likely degrades knowledge from pretraining; an issue our approach mitigates through partial adaptation. We confirm this suspicion by applying RE-ADAPT to Llama-3 without any additional fine-tuning. This allows us to produce instruct models with instruction-tuning strengths ranging from 0 (the pretrained model) to 1 (the instruct model).
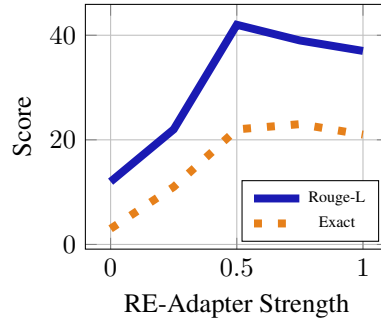
Table 2: Closed-book QA performance. The QA dataset being evaluated is listed above the dataset used for fine-tuning DoRA adapters. R-L indicates Rouge-L and EM indicates exact match.

	Model	StreamingQA News Crawl R-L	EM	RetrievalQA RQA Passages R-L	EM	Natural Questions News Crawl R-L	EM	RQA Passages R-L	EM
Llama-3	Pretrained	9	0	1	0	10	3	10	3
	Pretrained + DoRA	12	3	3	2	10	4	14	7
	Instruct	33	19	5	3	46	34	46	34
	Instruct + DoRA	38	22	7	4	39	22	37	27
	LoRE-Adapt (Ours)	46	26	**10**	**6**	51	34	53	35
	RE-Adapt (Ours)	**46**	**27**	9	6	**52**	34	**54**	**36**
Gemma	Pretrained	11	2	1	0	10	3	10	3
	Pretrained + DoRA	19	4	1	0	7	1	10	2
	Instruct	20	9	2	1	26	12	26	12
	Instruct + DoRA	31	18	5	3	26	12	28	14
	LoRE-Adapt (Ours)	31	15	**7**	**4**	24	14	**30**	**20**
	RE-Adapt (Ours)	**33**	**18**	6	4	**26**	**17**	28	17
Mistral	Pretrained	17	5	2	0	14	5	14	5
	Pretrained + DoRA	22	8	2	1	14	5	15	6
	Instruct	29	16	4	2	33	22	33	22
	Instruct + DoRA	36	21	6	5	27	13	33	18
	LoRE-Adapt (Ours)	**39**	**24**	**7**	**5**	**39**	**24**	**42**	**28**
	RE-Adapt (Ours)	37	22	6	4	37	23	41	27

We find that **we can improve existing instruct models with zero additional training by simply scaling down the strength of instruction-tuning** Figure 6. Combined, these results demonstrate the effectiveness of RE-Adapt for knowledge acquisition with minimal *forgetting*.

5.5 RE-Adapt with RAG

Retrieval-augmented generation (RAG) Lewis et al. (2020) is a popular alternative for utilizing new data with instruction-tuned models. Instead of altering the model directly, RAG maintains a database of all text and retrieves relevant documents to include in the prompt as context. This begs the question, is RE-Adapt still beneficial if the new data is already available via RAG?
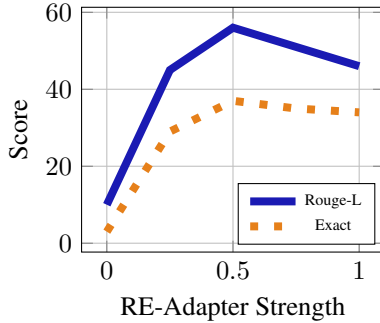

Figure 6: Natural Questions performance as the RE-Adapter is added to pretrained Llama-3 with varying strengths.

To answer this question, we replicate our experiments on StreamingQA and RetrievalQA, using a BM-25 index (Robertson & Zaragoza, 2009) to retrieve the most relevant passage to be used as context for the models. In practice, RAG setups can retrieve more than one document, but each question in our datasets can be answered from a single passage, and therefore we avoid known issues which RAG can face when too much context is provided to the models (Liu et al., 2023; Barnett et al., 2024; Gao et al., 2024). A poor retriever could bias results in our favor, so we also repeat the experiment using an oracle retriever. Instead of performing a heuristic search, the oracle retriever directly selects the passages capable of answering the question as context. While this idealized retriever is unrealistic in practice, it allows us to further isolate the benefit of combining RAG with fine-tuning by eliminating any impact from imperfect retrieval.

The RAG results are shown in Table 3. Again we see significant improvements when using RE-Adapt and LoRE-Adapt even in this RAG setting where the model should already have access

to the relevant information needed to answer the questions. The BM-25 search retrieved the correct document with approximately 73% accuracy across models. Using RE-ADAPT to incorporate the data outside of RAG alleviates the shortcomings of the retriever. However, RE-ADAPT also improved results when using the oracle, suggesting that adding domain knowledge with an adapter also reduces incorrect interpretations of the context retrieved via RAG.

Table 3: QA performance when using RAG with BM25 and (Oracle) retrievers.

	Model	StreamingQA		RetrievalQA	
		Rouge-L	Exact Match	Rouge-L	Exact Match
Llama-3	Pretrained	38 (59)	27 (48)	13 (16)	11 (14)
	Instruct	55 (57)	54 (58)	14 (30)	16 (32)
	LoRE-ADAPT (Ours)	**69 (74)**	58 (64)	**24 (37)**	**21 (31)**
	RE-ADAPT(Ours)	68 (71)	**59 (64)**	19 (36)	18 (30)
Gemma	Pretrained	39 (41)	28 (29)	4 (26)	3 (23)
	Instruct	**52 (56)**	48 (53)	17 (24)	16 (24)
	LoRE-ADAPT (Ours)	46 (50)	49 (55)	12 (17)	18 (27)
	RE-ADAPT (Ours)	50 (55)	**50 (56)**	**21 (30)**	**18 (28)**
Mistral	Pretrained	33 (38)	26 (30)	18 (12)	16 (10)
	Instruct	49 (52)	50 (56)	14 (23)	19 (28)
	LoRE-ADAPT (Ours)	54 (58)	**55 (61)**	**18 (23)**	20 (28)
	RE-ADAPT (Ours)	**55 (58)**	55 (60)	15 (24)	**20 (29)**

6 DISCUSSION

Combined, our results demonstrate RE-ADAPT's effectiveness at incorporating new knowledge into existing LLMs without having to discard previous instruction-tuning. Our methods increase QA performance by a greater amount when compared to traditional fine-tuning strategies. We also find that our approach improves RAG based systems, even in the most optimistic case of perfect retrieval. Our improved results outside of the fine-tuning distribution suggest that we can recover additional pretraining knowledge by reducing the strength of instruction-tuning through partial adaptation. Importantly, an improvement is seen without any additional fine-tuning of the underlying models. These results encourage additional future research into controlling the competing priorities of knowledge acquisition and general problem solving capability.

7 CONCLUSION

In this work, we presented RE-ADAPT, a new approach for adding knowledge to existing instruction-tuned models. RE-ADAPT isolates the differences between an instruction-tuned model and its pretrained counterpart in order to preserve instruction-following capabilities during additional fine-tuning on unlabeled data. We demonstrated that our approach outperforms fine-tuning pretrained or instruction-tuned models directly, which otherwise causes performance to degrade outside of the new fine-tuning domain. Our findings are robust across three state of the art large language models.

We achieved our best performance using *partial adaptation*, a new method for controlling the strength of adaptation at inference time when using single or combined adapters. We found that partially adapting instruction-tuned models improved QA performance without any additional fine-tuning.

We also analyzed the spectrum of RE-ADAPT's weight matrices, constructing a low-rank variant of our approach, LoRE-ADAPT, which captures the majority of variation in the instruction-tuning weights at a much lower rank. LoRE-ADAPT performed similarly to RE-ADAPT with occasional out-performance, while decreasing the number of parameters by as much as 5x in our experiments.

Finally, we demonstrated that RE-ADAPT improves performance even when the information required to answer questions is available via retrieval augmented generation. Combined, our results suggest RE-ADAPT is an effective approach for infusing new knowledge into instruction-tuned LLMs.

REFERENCES

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Co-jocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. The falcon series of open language models, 2023.

Sara Babakniya, Ahmed Roushdy Elkordy, Yahya H. Ezzeldin, Qingfeng Liu, Kee-Bong Song, Mostafa El-Khamy, and Salman Avestimehr. Slora: Federated parameter efficient fine-tuning of language models, 2023.

Jeanine Banks and Tris Warkentin. Gemma: Introducing new state-of-the-art open models. *Google*, 2024. URL `https://blog.google/technology/developers/gemma-open-models/`.

Ankur Bapna and Orhan Firat. Simple, scalable adaptation for neural machine translation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1538–1548, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1165. URL `https://aclanthology.org/D19-1165`.

Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. Seven failure points when engineering a retrieval augmented generation system, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf`.

Alexandra Chronopoulou, Matthew Peters, Alexander Fraser, and Jesse Dodge. AdapterSoup: Weight averaging to improve generalization of pretrained language models. In Andreas Vlachos and Isabelle Augenstein (eds.), *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 2054–2063, Dubrovnik, Croatia, May 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.153. URL `https://aclanthology.org/2023.findings-eacl.153`.

Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. Language-family adapters for low-resource multilingual neural machine translation. In Atul Kr. Ojha, Chao-hong Liu, Ekaterina Vylomova, Flammie Pirinen, Jade Abbott, Jonathan Washington, Nathaniel Oco, Valentin Malykh, Varvara Logacheva, and Xiaobing Zhao (eds.), *Proceedings of the Sixth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2023)*, pp. 59–72, Dubrovnik, Croatia, May 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.loresmt-1.5. URL `https://aclanthology.org/2023.loresmt-1.5`.

Michael Desmond, Evelyn Duesterwald, Kristina Brimijoin, Michelle Brachman, and Qian Pan. Semi-automated data labeling. In Hugo Jair Escalante and Katja Hofmann (eds.), *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pp. 156–169. PMLR, 06–12 Dec 2021. URL `https://proceedings.mlr.press/v133/desmond21a.html`.

Shizhe Diao, Tianyang Xu, Ruijia Xu, Jiawei Wang, and Tong Zhang. Mixture-of-domain-adapters: Decoupling and injecting domain knowledge to pre-trained language models' memories. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5113–5129, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.280. URL `https://aclanthology.org/2023.acl-long.280`.

C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. doi: 10.1007/BF02288367.

William Fleshman, Aleem Khan, Marc Marone, and Benjamin Van Durme. Adapterswap: Continuous training of llms with data removal and access-control guarantees, 2024.

Teodor Fredriksson, David Issa Mattos, Jan Bosch, and Helena Holmström Olsson. Data labeling: An empirical investigation into industrial challenges and mitigation strategies. In Maurizio Morisio, Marco Torchiano, and Andreas Jedlitschka (eds.), *Product-Focused Software Process Improvement*, pp. 202–216, Cham, 2020. Springer International Publishing. ISBN 978-3-030-64148-1.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 565–576, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.47. URL https://aclanthology.org/2021.acl-long.47.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. Findings of the 2022 conference on machine translation (WMT22). In Philipp Koehn, Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Tom Kocmi, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Marco Turchi, and Marcos Zampieri (eds.), *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pp. 1–45, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.wmt-1.1.

Suhas Kotha, Jacob Springer, and Aditi Raghunathan. Understanding catastrophic forgetting in language models via implicit inference. In *NeurIPS 2023 Workshop on Distribution Shifts: New Frontiers with Foundation Models*, 2024. URL https://openreview.net/forum?id=wkQy8mLIb9.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL `https://aclanthology.org/Q19-1026`.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf`.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-1013`.

Adam Liška, Tomáš Kočiský, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien de Masson d'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, Ellen Gilsenan-McMahon Sophia Austin, Phil Blunsom, and Angeliki Lazaridou. Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models. *arXiv preprint arXiv:2205.11388*, 2022.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2023. URL `https://api.semanticscholar.org/CorpusID:259360665`.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation, 2024.

Bhavitvya Malik, Abhinav Ramesh Kashyap, Min-Yen Kan, and Soujanya Poria. UDAPTER - efficient domain adaptation using adapters. In Andreas Vlachos and Isabelle Augenstein (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2249–2263, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.165. URL `https://aclanthology.org/2023.eacl-main.165`.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24:109–165, 1989. ISSN 0079-7421. doi: https://doi.org/10.1016/S0079-7421(08)60536-8. URL `https://www.sciencedirect.com/science/article/pii/S0079742108605368`.

Meta. Introducing meta llama 3: The most capable openly available llm to date, 2024. URL `https://ai.meta.com/blog/meta-llama-3/`.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3470–3487, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.244. URL `https://aclanthology.org/2022.acl-long.244`.

Eric Mitchell, Rafael Rafailov, Archit Sharma, Chelsea Finn, and Christopher D Manning. An emulator for fine-tuning large language models using small language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=Eo7kv0sllr`.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022a. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan

13

Leike, and Ryan Lowe. Training language models to follow instructions with human feed-back. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022b. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf`.

Marinela Parovic, Alan Ansell, Ivan Vulić, and Anna Korhonen. Cross-lingual transfer with target language-ready task adapters. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 176–193, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl. 13. URL `https://aclanthology.org/2023.findings-acl.13`.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapter-Fusion: Non-destructive task composition for transfer learning. In Paola Merlo, Jorg Tiede-mann, and Reut Tsarfaty (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.39. URL `https://aclanthology.org/2021.eacl-main.39`.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, apr 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL `https://doi.org/10.1561/1500000019`.

Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. AdapterDrop: On the efficiency of adapters in transformers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7930–7946, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.626. URL `https://aclanthology.org/2021.emnlp-main.626`.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*, 2022. URL `https://openreview.net/forum?id=9Vrb9D0WI4`.

Tim Schopf, Dennis N. Schneider, and Florian Matthes. Efficient domain adaptation of sentence embeddings using adapters. In Ruslan Mitkov and Galia Angelova (eds.), *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pp. 1046–1053, Varna, Bulgaria, September 2023. INCOMA Ltd., Shoumen, Bulgaria. URL `https://aclanthology.org/2023.ranlp-1.112`.

Pratyusha Sharma, Jordan T Ash, and Dipendra Misra. The truth is in there: Improving reasoning in language models with layer-selective rank reduction. In *Proceedings of the 12th International Conference on Learning Representations (ICLR 2024)*, 2024. URL `https://openreview.net/forum?id=ozX92bu8VA`.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving lora in privacy-preserving federated learning. In *Proceedings of the 12th International Conference on Learning Representations (ICLR 2024)*, 2024. URL `https://openreview.net/forum?id=NLPzL6HWNl`.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288, 2023. URL `https://api.semanticscholar.org/CorpusID:259950998`.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. UDapter: Typology-based language adapters for multilingual dependency parsing and sequence labeling. *Computational Linguistics*, 48(3):555–592, September 2022. doi: 10.1162/coli_a_00443. URL `https://aclanthology.org/2022.cl-3.3`.

Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. AdaMix: Mixture-of-adaptations for parameter-efficient model tuning. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5744–5760, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.388. URL `https://aclanthology.org/2022.emnlp-main.388`.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13484–13508, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.754. URL `https://aclanthology.org/2023.acl-long.754`.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*, 2022. URL `https://openreview.net/forum?id=gEZrGCozdqR`.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020.

Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, and Sara Hooker. Extremely parameter efficient moe for instruction tuning. In *Proceedings of the 12th International Conference on Learning Representations (ICLR 2024)*, 2024. URL `https://openreview.net/forum?id=EvDeiLv7qc`.

Zihan Zhang, Meng Fang, and Ling Chen. Retrievalqa: Assessing adaptive retrieval-augmented generation for short-form open-domain question answering, 2024.

A   FINE-TUNING DETAILS

We include the settings for training our DoRA adapters in Table 4. All adapters were trained on a single NVIDIA A100 GPU with 80GB of memory.

Table 4: Training details.

Setting	Value
LoRA Layers	all-linear
LoRA Rank	64
LoRA Alpha	128
LoRA Dropout	0.05
DoRA	True
Batch Size	20
Epochs News Crawl	10
Epochs RetrievalQA	3
Optimizer	AdamW
Learning Rate	0.0002
Schedule	Linear

B   PROMPTS USED

Each LLM can use unique prompting roles and tokens when constructing prompts. We utilize the huggingface *tokenizers* library to ensure our prompts follow the correct template.

The Llama-3 instruct models use a combination of system, user, and assistant roles while Gemma and Mistral only use user and assistant. Our prompts where constructed using the following formats:

Llama-3 Closed-Book QA
system: *Answer the following question.*
user: *<question>?*

Llama-3 RAG
system: *Answer the following question given this context: <context>.*
user: *<question>?*

Gemma and Mistral Closed-Book QA
user: *<question>?*

Gemma and Mistral RAG
user: *Answer the following question given this context: <context>\nQuestion: <question>?*

16