SMU Data Science Review

Volume 8
Number 2 *Summer 2024*

Article 2

2024

# Rethinking Retrieval Augmented Fine-Tuning in an evolving LLM landscape

Nicholas Sager
*Southern Methodist University*, sager.nick@gmail.com

Timothy Cabaza
*Southern Methodist University*, tcabaza@smu.edu

Matthew Cusack
*Southern Methodist University*, mcusack@smu.edu

Ryan Bass
ryanbassut12@gmail.com

Joaquin Dominguez
joaquin.dominguez@proton.me

# Rethinking Retrieval Augmented Fine-Tuning in an evolving LLM landscape

Nicholas J. Sager[1], Timothy Cabaza[1], Matthew Cusack[1]
Ryan Bass, Joaquin Dominguez
[1] Master of Science in Data Science, Southern Methodist University,
Dallas, TX 75275 USA
nsager@smu.edu
tcabaza@smu.edu
mcusack@smu.edu
ryanbassut12@gmail.com
joaquin.dominguez@proton.me

**Abstract.**

This study explores the utilization of Retrieval Augmented Fine-Tuning (RAFT) to enhance the performance of Large Language Models (LLMs) in domain-specific Retrieval Augmented Generation (RAG) tasks. By integrating domain-specific information during the retrieval process, RAG aims to reduce hallucination and improve the accuracy of LLM outputs. We investigate the use of RAFT, an approach that enhances LLMs by incorporating domain-specific knowledge and effectively handling distractor documents. This paper validates previous work, which found that RAFT can considerably improve the performance of Llama2-7B in specific domains. We also expand upon previous work into new state-of-the-art open-source models and other datasets with mixed results. After fine-tuning three models (Llama2-7B, Llama3-8B, and Mistral-7B-v0.3) using RAFT and evaluating their performance compared to their instruction-tuned versions, our results suggest that RAFT can only improve accuracy for older LLMs on domain specific data. This effect was not found in the latest generation of open-source LLMs.

## 1 Introduction

In recent years, Deep Learning and Artificial Intelligence (AI) have revolutionized numerous applications with LLMs becoming central to these advancements. Leading tech companies such as Google, Meta, and Nvidia have developed their own LLMs capable of handling a wide range of tasks (Anil, R., et al., 2024; Meta AI, 2024; Nvidia, 2024). These models have demonstrated remarkable

proficiency in understanding and generating human language, transforming industries and opening new avenues for innovation.

One popular approach to enhancing LLM performance is RAG. This method incorporates specific data into the question-answering process, effectively constraining the answer space to relevant information. This method helps mitigate generating incorrect information by retrieving relevant context and enhances the model's ability to produce accurate answers based on domain-specific knowledge. However, implementing RAG is not without its challenges. Data privacy concerns and the resource-intensive nature of large foundational models present significant hurdles. API calls to state-of-the-art models are costly (OpenAI, 2024; Anthropic, 2024), and the most capable open-source models require specialized hardware to run efficiently (Cohere, 2024; Nvidia, 2024). Additionally, many organizations are limited to using on-premises models due to concerns about privacy, security, and intellectual property. These challenges necessitate a balanced approach that leverages the strengths of RAG while addressing these constraints.

Given the growing use of RAG and the rising demand for LLMs tailored to domain-specific applications, this paper investigates the use of RAFT to enhance accuracy in these contexts. RAFT, introduced by Zhang et al. (2024), focuses on improving the model's ability to incorporate domain-specific knowledge and handle distractor documents effectively. By training the model to understand the relationship between queries, retrieved documents, and answers, RAFT has been shown to significantly enhance performance of Llama2-7B in domain-specific tasks. This approach promises not only improved accuracy but also aims to reduce the computational burden associated with traditional fine-tuning methods. We aim to reproduce the scope and core findings of previous research, while exploring the effectiveness of RAFT in cutting edge open source LLMs.

## 2 Literature Review

### 2.1 RAG

The paper "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks" by Lewis, P., et al. (2021) introduces RAG models, which integrate retrieval mechanisms with pre-trained language models to enhance knowledge-intensive tasks. RAG models, including RAG-Sequence and RAG-Token, leverage the Dense Passage Retriever (DPR) for document retrieval and BART-large for text generation (Lewis, P., et al., 2021). These models in Lewis, P., et al. (2021) are trained end-to-end and improve performance on tasks like open-domain question answering, abstractive question answering, Jeopardy question generation, and fact verification.

Key findings by Lewis, P., et al. (2021) include RAG's state-of-the-art performance in open-domain question answering without needing specialized pre-training, fewer hallucinations, and greater factual accuracy in text generation. RAG-Token shows superior performance in Jeopardy question generation also performs well in fact verification, demonstrating effective evidence retrieval (Lewis, P., et al. 2021).

RAG's flexible architecture allows for easy updating of knowledge and improved performance by retrieving more documents. The study highlights RAG's potential for hybrid generation models and suggests future research on joint pre-training and combining parametric and non-parametric memories. Overall, RAG models offer significant improvements in performance and interpretability for knowledge-intensive Natural Language Processing (NLP) tasks (Lewis, P., et al. 2021).

Zhou, C., et al. (2024) investigates the effectiveness of extending the context window of large language models (LLMs) versus augmenting them with retrieval for tasks involving long contexts. Using two state-of-the-art LLMs, a proprietary 43B GPT and Llama2-70B, across nine tasks such as question answering and query-based summarization, Zhou, C., et al. (2024) finds that a retrieval-augmented LLM with a 4K context window can match the performance of a fine-tuned LLM with a 16K context window while requiring less computation. Retrieval consistently enhances LLM performance regardless of context window size.

The best-performing model, a retrieval-augmented Llama2-70B with a 32K context window, outperforms other models on average scores across the tasks (Zhou, C., et al., 2024). Their results indicate that retrieval-augmentation, combined with extending context windows, is effective for improving LLMs. Key findings by Zhou, C., et al. (2024) include retrieval's significant impact on few-shot learning and the consistent performance boost from various retrievers. Zhou, C., et al. (2024) suggests future research should focus on advanced methods for pretrained LLMs, further extending context windows, and addressing challenges like the "lost-in-the-middle" phenomenon.

Ram, O., et al. (2023) presents In-Context Retrieval-Augmented Language Modeling (RALM), which enhances language models (LMs) by integrating relevant documents during generation without altering model parameters. This method exhibits notable improvements in LM performance across various sizes and datasets, showcasing its compatibility with existing LMs. Experimental results highlight its effectiveness in reducing LM perplexity and improving performance in open-domain question answering tasks, particularly when relevant documents are provided (Ram, O., et al., 2023). The authors' discussion underscores RALM's significance for knowledge-intensive tasks and suggests avenues for future research in optimizing retrieval processes and integrating multiple documents for further enhancements.

### 2.2 Retrieval Augmented Language Models (RALM)

A paper by Guu, K., et al. (2020) introduces RALM (Retrieval-Augmented Language Model), a method that enhances language models like BERT by incorporating a learned knowledge retriever to fetch relevant documents during pre-training, fine-tuning, and inference. This approach improves model interpretability and scalability by explicitly using external knowledge. According to Guu, K., et al. (2020), RALM's key innovation lies in its unsupervised pre-training of the knowledge retriever via masked language modeling, which integrates retrieval steps with millions of documents. This method demonstrates superior performance in Open-domain Question

& Answering (Open-QA), surpassing previous models, and establishing new benchmarks (Guu, K., et al, 2020).

RALM's architecture includes a neural knowledge retriever and a knowledge-augmented encoder; the retriever selects relevant documents based on their inner product scores, while the encoder utilizes these documents for cross-attention to predict answers (Guu, K., et al, 2020). Training involves maximizing log-likelihood, with computational challenges addressed by techniques like Maximum Inner Product Search (MIPS) and asynchronous updates. The experimental results by Guu, K., et al. (2020) show that RALM significantly outperforms other models on Open-QA tasks, providing both higher accuracy and computational efficiency. The Guu, K., et al. (2020) paper also explores future directions, including extending RALM to handle structured knowledge, multilingual, and multimodal settings.

The Izacard, G., et al. (2023) paper "Atlas: Few-shot Learning with Retrieval Augmented Language Models" presents Atlas, a model combining retrieval mechanisms with language models to enhance performance in few-shot learning tasks. Atlas utilizes a text-to-text framework, where tasks are framed as input-output pairs processed by two main components: a retriever and a language model. The retriever identifies relevant documents, and the language model generates outputs based on these documents (Izacard, G., et al., 2023).

Atlas employs various training objectives for the retriever, including Attention Distillation and End-to-end Training of Multi-Document Reader and Retriever (EMDR2) (Izacard, G., et al., 2023). Pre-training tasks like Prefix Language Modeling and Masked Language Modeling allow joint training of the retriever and language model. Efficient retriever fine-tuning strategies, such as Full Index Update and Query-side Fine-tuning, ensure the model stays updated with minimal computational cost (Izacard, G., et al., 2023).

Key analyses reveal that Atlas excels in integrating retrieved information, with significant performance improvements when combining language model and retriever fine-tuning; the model retrieves relevant documents effectively, showing high accuracy and adaptability to temporal changes without retraining (Izacard, G., et al., 2023). Index compression techniques reduce memory usage with minimal impact on performance. Overall, Atlas demonstrates strong few-shot learning capabilities, outperforming larger models like PaLM, and offers advantages in interpretability, updateability, and performance across various benchmarks (Izacard, G., et al., 2023).

The paper by Shi, W., et al. (2023) introduces REPLUG, a framework enhancing black-box language models (LMs) like GPT-3 and Codex by adding a retriever model. Unlike complex methods, REPLUG simply pre-appends retrieved documents to the LM's input, improving predictions without altering the LM's architecture. Experiments demonstrate REPLUG's effectiveness, boosting GPT-3's performance by 6.3% on language modeling and Codex by 5.1% on five-shot Massive Multitask Language Understanding (MMLU) tasks (Shi, W., et al., 2023). By using an external retriever and LM supervision, Shi, W., et al. (2023) show that REPLUG enhances document retrieval quality and adapts the retriever to the LM, achieving significant improvements across various tasks, especially in STEM categories. However, REPLUG lacks interpretability, suggesting a need for future research in developing more transparent retrieval-augmented models (Shi, W., et al., 2023).

The paper by Lin, X.V., et al. (2024), "RA-DIT: Retrieval-Augmented Dual Instruction Tuning," presents a method to enhance retrieval-augmented language models (RALMs) by fine-tuning both the language model (LM) and the retriever in two steps. In the first step, Language Model Fine-Tuning (LM-ft), the LLM is trained to better utilize retrieved information by incorporating relevant background text into the prompts. In the second step, Retriever Fine-Tuning (R-ft), the retriever is optimized to return results more aligned with the LLM's needs using an LM-Supervised Retrieval (LSR) objective. This dual fine-tuning process, RA-DIT, significantly improves performance, with the best model, RA-DIT 65B, achieving state-of-the-art results on various knowledge-intensive benchmarks, outperforming existing methods by up to +8.9% in zero-shot and +1.4% in five-shot settings (Lin, X.V., et al., 2024).

The RA-DIT framework described by Lin, X.V., et al. (2024) employs a retrieval-augmented pre-trained auto-regressive language model (LLAMA) and a dual-encoder retriever architecture. Lin, X.V., et al. (2024) provide detailed analyses showing that the combination of LM and retriever fine-tuning yields the best results along with highlighting the superior performance of RA-DIT compared to other retrieval methods like DRAGON+. Overall, RA-DIT presents an efficient approach to enhancing RALMs without extensive pre-training, demonstrating significant improvements across multiple tasks (Lin, X.V., et al., 2024).

### 2.3 RAFT

A paper by Zhang, T., et al (2024) introduces Retrieval Augmented Fine-Tuning (RAFT), a methodology to enhance Large Language Models (LLMs) for specialized domain tasks, particularly in Retrieval Augmented Generation (RAG). RAFT aims to enhance the model's capability to integrate domain-specific knowledge and efficiently manage distractor documents. By training the model to comprehend the relationship between the query, retrieved documents, and the answer, RAFT substantially improves the model's performance in in-domain RAG tasks.

Large Language Models (LLMs) have made significant strides in general knowledge reasoning tasks. However, there's a growing need to adapt these models for specialized domains. This paper addresses the challenge of adapting LLMs for Retrieval Augmented Generation (RAG) in specialized domains. It contrasts two approaches: in-context learning through RAG and supervised fine-tuning, highlighting their limitations. Zhang, T., et al, 2024 propose RAFT, a novel approach that combines supervised fine-tuning with RAG to improve model performance in in-domain RAG tasks, effectively leveraging domain-specific knowledge while handling inaccuracies in document retrieval.

The paper elucidates the analogy between LLMs and open-book exams to explain its goal. Zhang, T., et al, 2024 distinguishes between closed-book and open-book exams, where closed-book exams represent scenarios solely based on pre-trained and fine-tuned knowledge, and open-book exams allow referencing external sources via a retriever. RAFT focuses on domain-specific open-book exams, proposing a solution to adapt pre-trained LLMs effectively to domain-specific contexts, making them resilient to varying numbers of retrieved documents and distractors (Zhang, T., et al, 2024).

RAFT enhances general instruction tuning by incorporating supervised fine-tuning (SFT) and prepares training data containing questions, documents, and answers, distinguishing between 'oracle' and 'distractor' documents (Zhang, T., et al, 2024). Through SFT, the model is trained to understand the relationship between the query, retrieved documents, and the answer, enhancing its ability to perform RAG on in-domain documents. The fine-tuning process focused on generating coherent reasoning chains and citing sources, independently of the retriever used, thereby improving model accuracy. By combining the strengths of supervised fine-tuning and RAG, RAFT aims to leverage domain-specific knowledge while handling inaccuracies in document retrieval. This process is designed to make the model resilient to varying numbers of retrieved documents and distractors (Zhang, T., et al, 2024).

The evaluation assesses RAFT's performance compared to various baselines across different datasets, demonstrating superior performance, particularly with the RAFT-7B model. RAFT consistently outperforms baselines across diverse domains, achieving significant improvements in accuracy on retrieval-oriented datasets like HotpotQA and HuggingFace (Zhang, T., et al, 2024). Additionally, experiments explore the impact of training with oracle context for RAG, revealing optimal training strategies for handling top-k RAG tasks.

RAG integrates retrieval modules to enhance language models, with various approaches following a "retrieve-and-read" paradigm. Studies on memorization and fine-tuning of LLMs have advanced understanding in these areas. Recent research explores fine-tuning pretrained language models specifically for RAG tasks, addressing scenarios where models are tested on the same set of documents used for training (Zhang, T., et al, 2024).

RAFT is a training strategy designed to enhance language model performance in domain-specific question-answering tasks. Evaluations demonstrate its significant potential, particularly in domain-specific RAG tasks, highlighting the ongoing interest in Retrieval-Augmented Generation (RAG) within specialized domains. The paper by Zhang, T., et al, 2024 anticipates that smaller, fine-tuned models can perform comparably well in domain-specific tasks compared to generic language models.

### 2.4 Comparing Fine-tuning & Retrieval

The study by Ovadia, O., et al (2024) investigates methods for incorporating new information into Large Language Models (LLMs), specifically comparing supervised fine-tuning and retrieval-augmented generation (RAG). The research highlights that RAG consistently outperforms fine-tuning across various knowledge-intensive tasks. One significant finding is that LLMs struggle to learn new information through unsupervised fine-tuning unless they are exposed to numerous variations of the same fact.

The study identifies two primary limitations of LLMs: static knowledge that does not update over time and non-specific knowledge that lacks nuance in specialized domains. By using external datasets, the research explores how to enhance LLMs' knowledge. The paper by Ovadia, O., et al (2024) introduces a knowledge score to measure a model's understanding of factual questions and discusses various causes of

factual errors in LLMs, such as domain knowledge deficits, outdated information, and improper memorization.

Two main approaches for knowledge injection are examined: fine-tuning (including supervised, reinforcement learning, and unsupervised methods) and RAG. Fine-tuning methods, while improving overall model quality, are less effective at injecting new knowledge. In contrast, RAG uses an external knowledge base to retrieve relevant information for a query, significantly enhancing the model's performance on knowledge-intensive tasks without additional training.

The experimental setup includes tasks from the MMLU benchmark and a current events task, with data collected and cleaned from Wikipedia (Ovadia, O., et al 2024). The study finds that RAG consistently outperforms base models and fine-tuning, particularly in tasks involving current events. Moreover, data augmentation through paraphrasing is shown to improve model performance, indicating that repetition in various forms helps models understand and generalize new knowledge.

In conclusion, the research by Ovadia, O., et al (2024) emphasizes the limitations of unsupervised fine-tuning and underscores the superior performance of RAG for incorporating new information into LLMs. The findings suggest that future research should focus on combining various techniques and exploring other definitions and perspectives on knowledge representation in LLMs.

### 2.5 Enterprise Level RAG

Cohere recently introduced Command R+, its most powerful and scalable large language model (LLM) in the Command R family, designed for enterprise-grade workloads and available initially on Microsoft Azure. Command R+ is optimized for complex Retrieval Augmented Generation (RAG) tasks, multilingual support, and advanced tool use, enabling businesses to move from proof-of-concept to production with AI. With a 128k-token context window, Command R+ offers improved performance, reduced hallucinations with in-line citations, and the ability to automate complex business workflows through multi-step tool use (Cohere 2024).

Command R+ is built on the strengths of Command R, offering a suite of embedding models in various languages, a Re-ranker for context, and other tools for high performing enterprise-level RAG. This model is particularly adept at handling multilingual tasks, reducing token costs by up to 57% for non-English text (Cohere 2024). The model's API availability and its integration with platforms like Oracle and LangChain, Cohere aims to position itself as the leader for enterprise level solutions. (Cohere 2024).

This study hypothesizes that Retrieval Augmented Fine-Tuning (RAFT) will significantly enhance the performance of Large Language Models (LLMs) in domain-specific tasks. Specifically, RAFT is expected to improve the accuracy of LLM outputs by effectively integrating domain-specific knowledge while reducing hallucinations. Furthermore, it is hypothesized that RAFT will achieve these improvements with reduced computational costs and training time compared to traditional fine-tuning methods, thus providing a more efficient and effective approach for domain-specific applications of LLMs.

# 3 Method

## 3.1 Dataset Generation

For each testing dataset, a training dataset was generated to fine-tune the models on. Figure 1 outlines a high-level overview of the framework executed. The training datasets were generated from the same validation set that the models would later be evaluated on. This was to simulate tuning for RAG on a corpus of data that the user already has access to. Our first step is to extract all example search context from the datasets. Our datasets include information related to the question to simulate a RAG application. The size and relevance of the context varies by dataset. Questions and answers were also discarded to simulate training a model on a corpus of data, as in the real-world use case.

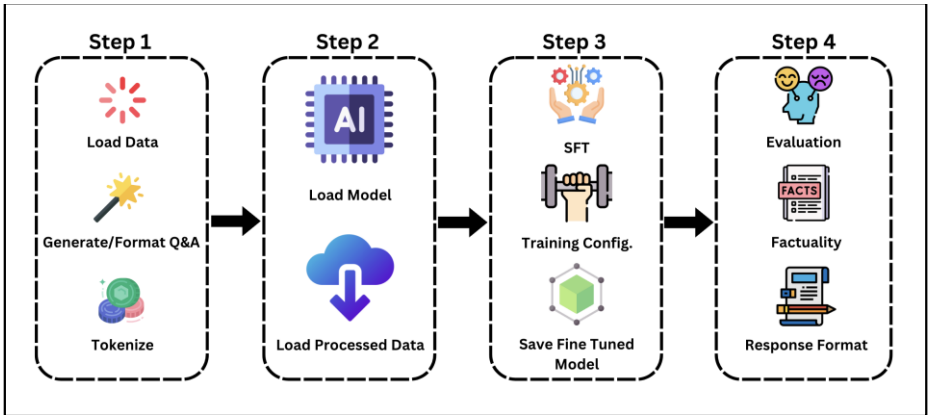


**Figure 1: Basic RAFT framework executed**

With all the relevant contexts split into chunks, a larger LLM, Llama3-70B-Instruct was used to generate questions from each chunk. Figure 2 illustrates the general pipeline of models used to generate higher quality outputs. The questions generated were packaged with either distractor contexts, relevant contexts, or just the irrelevant context according to the RAFT algorithm. The LLM also generated a chain-of-thought style answer which included the answer to the question and the sections of context that the answer was extracted from, which were quoted verbatim. Zhang et al. show that prompting in this way increases accuracy. We generated these training datasets using the RAFT Dataset Pack from Llama-Index with Llama3-70B used as the inference model. One could alternatively have used a larger proprietary model for dataset generation, but this would have violated the use case of a user concerned with privacy or resource usage. We have shown that Llama3-70B is capable enough to produce meaningful training datasets.

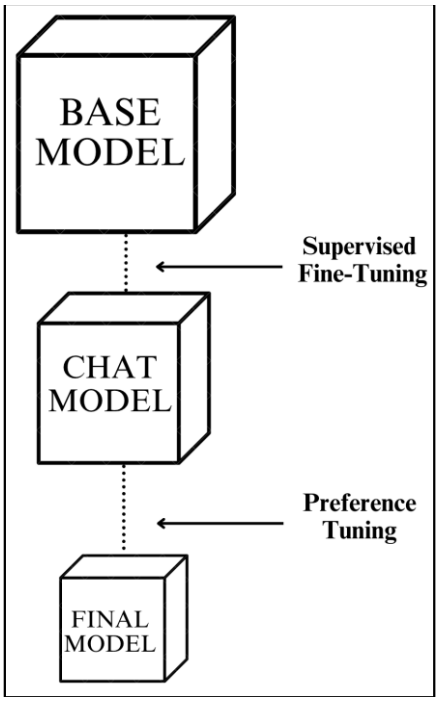


**Figure 2: General Pipeline utilized for higher quality outputs**

### 3.2 Supervised Fine-Tuning

We then used Supervised Fine-Tuning (SFT) to train language models on the datasets created in the previous step. The instruction tuned versions of the LLMs were used both for baseline evaluation and RAFT. Base models could be fine-tuned, but evaluation results would be artificially deflated since these models are rarely used in practice. Fine-tuning was conducted using the Hugging Face TRL library, with standard hyperparameters for language model fine-tuning (Werra, L., et al., 2020). All model tuning was done as a full fine-tune of all model parameters. Low-rank adaptation (LoRA) has been shown to achieve similar performance as full fine-tuning with greater efficiency (Hu, E., et al., 2021) and could be used in a resource-constrained environment.

### 3.3 Benchmark Evaluation

To assess the performance of the fine-tuned models, we evaluated them on a diverse set of benchmarks, each targeting specific aspects of language understanding, reasoning, and generation. The evaluation process was conducted using EleutherAI's

Large Language Model (LLM) Evaluation Harness, which provides a standardized framework for benchmarking LLMs across various tasks (Gao et al., 2023). This harness automated the evaluation process, ensuring consistency and reproducibility in our results.

To evaluate the effect of our fine-tuning on RAG performance, we used several datasets that include context with questions and answers. Trivia QA (Joshi, M., et al., 2017), and HotpotQA (Yang, Z., et al,. 2018) are based on general knowledge questions and are designed to test a model's ability to answer questions given related search results. HotpotQA has more challenging questions, while TriviaQA tests the model's ability to find answers from a large amount of context. The length of search results in this dataset is sometimes larger than the context window of our small open-source models. This is a limitation of our approach but could be mitigated in production by changing the parameters of the retrieval system. PubMedQA (Jin, Q., et al., 2019) dataset was utilized to evaluate model performance when trained on specific domains. The PubMedQA dataset contains questions and answers specific to biology and medical research. This dataset tests the model's ability to answer "yes" or "no" to very specific questions about highly technical text.

No predefined tests for these datasets were included in Evaluation Harness in a form that is suitable for simulating RAFT. We defined a custom task for each dataset evaluated and used this to evaluate the baseline model and fine-tuned version. While the chain-of-thought style answer demonstrated improved results (Zhang, T., et al, 2024), it introduced significant answer parsing difficulties beyond a multiple choice or single word answers.

### 3.4 Compute and Cost Analysis

In addition to performance evaluation, we conducted an analysis of the computer resources and costs associated with each fine-tuning method. This analysis included tracking the training time, GPU usage, and any other relevant metrics to provide insights into the efficiency of language model retrieval augmentation fine-tuning.

## 4 Results

We evaluated our fine-tuned models against their instruction tuned versions across three datasets. It was shown that RAFT does improve performance in some cases with Llama2-7B, though not with more recent models. On the datasets we tested, Llama3-8B and Mistral-7B-v0.3 outperform Llama2-7B on simulated RAG tasks with and without RAFT. Both newer models were released after the publication of RAFT and represent the state of the art in open-source models of this size.

**Table 1:** Model Accuracy Across Datasets

| Results | | TriviaQA | HOTPOTQA | PubMedQA |
|---|---|---|---|---|
| **Models** | Llama2-7B-Chat | 18.2% | 15.8% | 73.4% |
| | Llama2-7B RAFT | 44.1% | 2.0%* | 73.2% |
| | Llama3-8B-Instruct | 66.3% | 42.8% | 75.4% |
| | Llama3-8B RAFT | 50.3% | 27.1% | 73.3% |
| | Mistral-7B-Instruct-v0.3 | 67.1% | 41.3% | 78.1% |
| | Mistral-7B-0.3-RAFT | 29%* | 39.6%* | 76.4% |

* Lower accuracy due to formatting issues. Estimated to be equal to or higher than the Instruct/Chat versions.

RAFT produced mixed results with Llama2-7B. There was a significant uplift using RAFT from 18% to 44% on the TriviaQA dataset, while there was a significant decrease in performance from 15.8% to 2% in the HotPotQA dataset. Visual analysis of the results suggested a much stronger outcome if the output was formatted as desired. The PubMedQA dataset yielded no difference between any of the base models and their RAFT comparison. PubMedQA is more domain-specific than the other two datasets, but the context is much shorter. The abstracts provided with PubmedQA are often more than an order of magnitude shorter than TriviaQA when tokenized.

Llama3-8B and Mistral 7B-0.3 both stayed the same or exhibited a decrease in accuracy after fine-tuning. For Mistral, this appears to be an artifact of formatting issues.

Dataset generation averaged 96 GPU-hours to generate training datasets for fine-tuning using Nvidia A100 GPUs. Fine-tuning took 4-6 hours on 8xA100 GPUs.

## 5 Discussion

This research aimed to improve the performance of small, 7-8 billion parameter, open-source models when used for RAG. Models in this class do not compare with the large state-of-the-art models, but the hope was that RAFT could maximize the abilities of the smaller models for this task. By tuning older and more recent open-source language models we were able validate and expand upon recent claims regarding the advantages of RAFT for improving RAG performance. This research has reinforced that RAFT can significantly increase accuracy, but the improvement is dependent on the model and nature of the data being used as context. We did not observe any improvement when using the latest generation of LLMs.

Qualitatively, we assess that the accuracy of the RAFT versions underestimates their performance compared to the instruction tuned versions. The fine-

tuning process introduces chain of thought reasoning into the responses. This has been shown to improve question answering performance (Zhang, T., et al., 2024). The additional output, however, makes parsing the answer more difficult. Tools exist for coercing LM output into a more convenient form, JSON for example, but these are not currently compatible with our chosen benchmarking framework, Evaluation Harness. We estimate that if formatting issues were completely resolved, the accuracy of Llama3-8B and Mistral-7B-v0.3 would match the Instruct versions. The fine-tuned version of Llama2-7B greatly improved the performance of the Chat version. This can be seen in the 240% increase in accuracy on TrivaQA, which had fewer formatting issues.

These formatting issues are less relevant for production use than in this research. Our evaluation required parsing a single word or phrase from a lengthy chain of thought answer to evaluate accuracy. In practice these outputs would likely be read by the user and be enhanced by the chain-of-thought reasoning. We find that the raw results are human readable, so the perceived accuracy would not be decreased.

We observed a clear difference in the efficacy of RAFT between the older generation of models tested (Llama2-7B) and the newer (Llama3-8B and Mistral-7B-v0.3). Both our research and Zhang et al. (2024) find a marked increase in performance on the older models. Conversely, our results on newer generation models indicate either no improvement or a decrease in performance. Given RAG has become a common use case for LLMs, we speculate that newer generation models may have already incorporated RAG-specific training during the Instruction tuning phase. If true, additional fine-tuning with different formatting could account for the decrease or stagnation in accuracy observed. The combination of unique data and driving the loss down on our fine-tuning process may reverse some of the performance gains these newer models have made over the previous generation. This is especially problematic if the training size isn't large enough to counteract the decrease in accuracy. Neither Meta nor Mistralai have yet to release detailed documentation about their training process. We suspect that their formatting likely differs from that used in our research.

Given this possibility, an interesting direction of future research would be to train the base (non-instruction-tuned) model directly. This would give the user more flexibility for RAFT, but with the added responsibility of training for instruction following. Whether this trade-off is fruitful would have to be investigated further.

In situations where resources or concerns about privacy or intellectual property constrain the use of LLMs to local resources, fine-tuning has the potential to improve RAG performance at a relatively modest cost compared to the next size larger of language models (e.g. Llama3-70B). Although fine-tuning the models requires more resources, all the models tested in this research can run inference on a consumer laptop once tuned. The increase in accuracy was more apparent as the context size increased. New versions of the models also showed less improvement with RAFT, but for some users with the above constraints, small increases in performance may be worth pursuing. At the same time, the observed trend indicates that new versions of LLMs may render fine-tuning unnecessary for most tasks.

## 6 Ethics

The integration of RAFT in domain-specific LLM applications brings forward significant ethical considerations that must be addressed to ensure responsible AI deployment. As LLMs become more prevalent in fields such as healthcare, legal research, and financial analysis, the ethical implications must be considered.

The use of RAFT involves incorporating domain-specific data during the fine-tuning process, which raises concerns about data privacy and security. Ensuring that sensitive information, especially in domains like healthcare and legal research is handled with the utmost confidentiality is paramount. Data used for fine-tuning must be anonymized and secured to prevent any potential breaches or misuse. Implementing robust data governance frameworks and adhering to strict compliance standards, such as GDPR or HIPPA, can help mitigate these risks.

The accuracy and reliability of LLM outputs are critical, particularly in sensitive applications. Inaccurate or misleading information can have serious consequences, such as incorrect medical diagnoses or flawed legal advice. RAFT aims to enhance accuracy by integrating domain-specific knowledge, but continuous monitoring and validation of model outputs are necessary to maintain high standards. Establishing protocols for regular audits and updates to the models can help them remain accurate and relevant over time. Even with these principals to help mitigate error, some LLM projects might be too sensitive to trust with a model.

Ethical implications of potential biases in LLMs should be addressed. Even with RAFT, models may still exhibit biases present in the training data. Ongoing efforts to identify and mitigate biases are necessary to ensure fair and unbiased outcomes. This involves diversifying the datasets used for fine-tuning and implementing bias detection and correction mechanisms.

The ethical considerations surrounding the use of RAFT in enhancing LLMs are multifaceted and require comprehensive strategies to address. By prioritizing data privacy, accuracy, societal impacts, and bias mitigation we can assist the responsible and ethical deployment of RAFT-enhanced LLMs in domain-specific applications.

## 7 Conclusion

Our findings indicate RAFT improves the performance of LLMs for certain tasks using older LLM architectures, validating previous research. Newer LLM architectures displayed debatable improvement at best, if not a marked performance decrease. Depending on the model used and the data queried, the change in accuracy ranges from an 86% decrease in performance to 240% improvement. By fine-tuning Llama2-7B with RAFT, we observed a marked increase in accuracy when handling some cases of domain-specific queries compared to traditional instruction tuned versions of the model. With Llama3-8B and Mistral-7B-v0.3, no such increase in accuracy was noted, though the results may be artificially deflated due to formatting issues. These findings suggest that the new models have less to gain from RAFT. This is possibly due to improvements in training or the adoption of RAG as a use case when Instruction tuning.

The implications of these findings are mixed. On one hand, some models can be greatly enhanced by fine-tuning. The trend, however, indicates that the efficacy of the technique is decreasing with advances in model pre-training and instruction tuning. Our research is not an exhaustive search of fine-tuning techniques for RAG, but the findings suggest that these techniques may become unnecessary as open-source models continue to improve. Future research is required to determine whether RAFT can be usefully adopted for the latest generation of language models.

The latest language models do not show the same potential for improvement as the older generation when tuned. Given the increase in baseline performance of the latest generation, the technique may not be worth the effort for most users. That said, our research has validated the technique on Llama2-7B. More research would be required to determine whether RAFT shows similar performance uplift as Llama2-7B on different or smaller models.

***Acknowledgments***

We would like to extend our heartfelt gratitude to our project advisors, Clovis Bass and Joaquin Dominguez, for their continued support and invaluable suggestions throughout the course of this project. Their dedication and the significant amount of time they invested were instrumental in guiding our work and ensuring its success. We deeply appreciate their expertise, commitment, and encouragement.

## 8 References

1. Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T., Lazaridou, A., … Vinyals, O. (2024). Gemini: A Family of Highly Capable Multimodal Models (arXiv:2312.11805). arXiv. https://doi.org/10.48550/arXiv.2312.11805
2. Anthropic Claude API Pricing. (2024). Retrieved June 19, 2024, from https://www.anthropic.com/api
3. Anochie, E. (2024). Google's AI image generator Gemini sparks controversy with historical figures. TechStory. Retrieved July 28, 2024, from https://techstory.in/googles-ai-image-generator-gemini-sparks-controversy-with-historical-figures/
4. Cohere. (2024). Command R+ Documentation. Cohere Docs. https://docs.cohere.com/docs/command-r-plus
5. Cohere. (2024). Introducing Command R+: A Scalable LLM Built for Business. Cohere Blog. https://cohere.com/blog/command-r-plus-microsoft-azure
6. Edwards, B. (2024). Google's Gemini AI was mocked for its revisionist history, but it still highlights a real problem. Fast Company. Retrieved July 28, 2024, from https://www.fastcompany.com/91034044/googles-gemini-ai-was-mocked-for-its-revisionist-history-but-it-still-highlights-a-real-problem
7. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M. (2020). Retrieval augmented language model pre-training. (arXiv:2002.08909). arXiv. https://arxiv.org/abs/2002.08909.

8. Harradence, M. (2024). Google's Gemini will be right back after these hallucinations – image generator to make a return after historical blunders. TechRadar. Retrieved July 28, 2024, from https://www.techradar.com/computing/artificial-intelligence/googles-gemini-will-be-right-back-after-these-hallucinations-image-generator-to-make-a-return-after-historical-blunders
9. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). LoRA: Low-Rank Adaptation of Large Language Models (arXiv:2106.09685). arXiv. https://doi.org/10.48550/arXiv.2106.09685
10. Izacard, G., Lewis, P., Lomeli, M., Hosseini, L., Petroni, F., Schick, T., Dwivedi-Yu, J., Joulin, A., Riedel, S., Grave, E., (2023). Atlas: Few-shot learning with retrieval augmented language models (arXiv:2208.03299). arXiv. https://doi.org/10. https://arXiv.org/2208.03299.
11. Jin, Q., Dhingra, B., Liu, Z., Cohen, W. W., & Lu, X. (2019). PubMedQA: A Dataset for Biomedical Research Question Answering (arXiv:1909.06146). arXiv. http://arxiv.org/abs/1909.06146
12. Joshi, M., Choi, E., Weld, D. S., & Zettlemoyer, L. (2017). TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension (arXiv:1705.03551). arXiv. http://arxiv.org/abs/1705.03551
13. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.-W., Dai, A. M., Uszkoreit, J., Le, Q., & Petrov, S. (2019). Natural Questions: A Benchmark for Question Answering Research. Transactions of the Association for Computational Linguistics, 7, 452–466. https://doi.org/10.1162/tacl_a_00276
14. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., Kiela, D. (2021). Retrieval-Augmented Generation for Knowledge-Intensive NLP tasks (arXiv:2005.11401). arXiv. https://arxiv.org/abs/2005.11401
15. Lin, X.V., Chen, X., Chen, M., Shi, W., Lomeli, M., James, R., Rodriguez, P., Kahn, J., Szilvasy, G., Lewis, M., Zettlemoyer, L. (2023). RA-DIT: Retrieval-augmented dual instruction tuning. (arXiv:2310.01352). arXiv. https://arxiv.org/abs/2310.01352.
16. Meta AI. Introducing Meta Llama 3: The most capable openly available LLM to date. (n.d.). Meta AI. Retrieved June 19, 2024, from https://ai.meta.com/blog/meta-llama-3/
17. Nvidia. Nemotron-4-340B-Instruct · Hugging Face. (2024, June 14). https://huggingface.co/nvidia/Nemotron-4-340B-Instruct
18. Openai API Pricing. (2024). Retrieved June 19, 2024, from https://openai.com/api/pricing/
19. Ovadia, O., Brief, M., Mishaeli, M., Elisha, O. (2023). Fine-tuning or retrieval? comparing knowledge injection in llms (arXiv:2312.05934). arXiv. https://arxiv.org/abs/2312.05934.
20. Ram, O., Levine, Y., Dalmedigos, I., Muhlgay, D., Shashua, A., Leyton-Brown, K., Shoham, Y. (2023). In-Context Retrieval-Augmented Language Models. (arXiv:2302.00083). arXiv. https://arxiv.org/abs/2302.00083
21. Shi, W., Min, S., Yasunaga, M., Seo, M., James, R., Lewis, M., Zettlemoyer, L., Yih, W.T. (2023). REPLUG: Retrieval-augmented black-box language models (arXiv:2301.12652). arXiv. https://arxiv.org/abs/2301.12652.
22. Werra, L. von, Belkada, Y., Tunstall, L., Beeching, E., Thrush, T., Lambert, N., & Huang, S. (2020). TRL: Transformer Reinforcement Learning. In GitHub repository. GitHub. https://github.com/huggingface/trl

23. Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W. W., Salakhutdinov, R., & Manning, C. D. (2018). HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering (arXiv:1809.09600). arXiv. http://arxiv.org/abs/1809.09600
24. Zhang, T., Patil, S. G., Jain, N., Shen, S., Zaharia, M., Stoica, I., & Gonzalez, J. E. (2024). RAFT: Adapting Language Model to Domain Specific RAG (arXiv:2403.10131). arXiv. http://arxiv.org/abs/2403.10131
25. Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., Zhang, S., Ghosh, G., Lewis, M., Zettlemoyer, L., & Levy, O. (2023). LIMA: Less Is More for Alignment (arXiv:2305.11206). arXiv. https://doi.org/10.48550/arXiv.2305.11206.