Tuning LLMs by RAG Principles: Towards LLM-native Memory

Anonymous ACL submission

Abstract

Memory, additional information beyond the training of large language models (LLMs), is crucial to various real-world applications, such as personal assistant. The two mainstream solutions to incorporate memory into the generation process are long-context LLMs and retrieval-augmented generation (RAG). In this paper, we first systematically compare these two types of solutions on three renovated/new datasets and show that (1) long-context solutions, although more expensive, shall be easier to capture the big picture and better answer queries which require considering the memory as a whole; and (2) when the queries concern specific information, RAG solutions shall be more competitive especially when the keywords can be explicitly matched. Therefore, we propose a novel method RAG-Tuned-LLM which fine-tunes a relative small (e.g., 7B) LLM using the data generated following the RAG principles, so it can combine the advantages of both solutions. Extensive experiments on three datasets demonstrate that RAG-Tuned-LLM can beat long-context LLMs and RAG methods across a wide range of query types.

1 Introduction

Memory, additional information beyond the training of large language models (LLMs), is crucial to various real-world applications, such as personal assistant (Mai et al., 2023). The most intuitive solution to enable long memory into the generation process is long-context LLM, for example, 128K-token GPT-4o (Achiam et al., 2023), 1M-or 10M-token Gemini 1.5 (Reid et al., 2024), or an LLM with "unlimited" context lengths by length extrapolation (Peng et al., 2023; Xiao et al., 2023; Han et al., 2023; Zhang et al., 2024a) and position bias (Liu et al., 2024; Peysakhovich and Lerer, 2023; An et al., 2024). Retrieval-augmented generation (RAG) (Lewis et al., 2020; Kočiskỳ et al., 2018; Pang et al., 2022; Trivedi et al., 2022; Edge et al., 2024) is another popular approach to incorporate memory in a plug-in manner: a retriever identifies a small number of query-relevant contexts from a large corpus, and then feeds them into an LLM to answer the query. Compared with long-context LLMs, RAG's serving cost is more affordable, and therefore, RAG is potentially more popular than long-context LLMs in real-world applications.

In this paper, we first systematically compare these two types of methods on three renovated/new datasets. We start with two public datasets, namely *news* articles (Tang and Yang, 2024) and *podcast* transcripts (Scott, 2024), following the general ideas mentioned in Edge et al. (2024) to generate queries and references. On these two datasets, we use the entire corpus as the memory. We categorize the queries into two types, *local* and *global*. Specifically, local queries target specific information and concrete answers from small chunks of memory. Global queries, on the other hand, require considering memory as a whole to generate high-level answers. We further introduce a new proprietary dataset containing *journaling* articles and user-provided *local*/*global* queries and their expected answers from our journaling app [1]

Intuitively, (1) long-context solutions, although more expensive, shall be easier to capture the big picture and better answer *global* queries; and (2) when the queries concern *local* information, RAG solutions shall be more competitive especially when the keywords can be explicitly matched. Based on these three datasets, we run competitions between a vanilla RAG (Lewis et al., 2020) and Gemini 1.5 (Reid et al., 2024), with the win rate results shown in Table 2, confirming our intuitions. It is worth mentioning that RAG surpasses long-context LLMs when handling *local* queries, yet underperforms in addressing *global* ones.

Following our findings, we propose a novel

[1]The app name is masked for the blind review purpose.

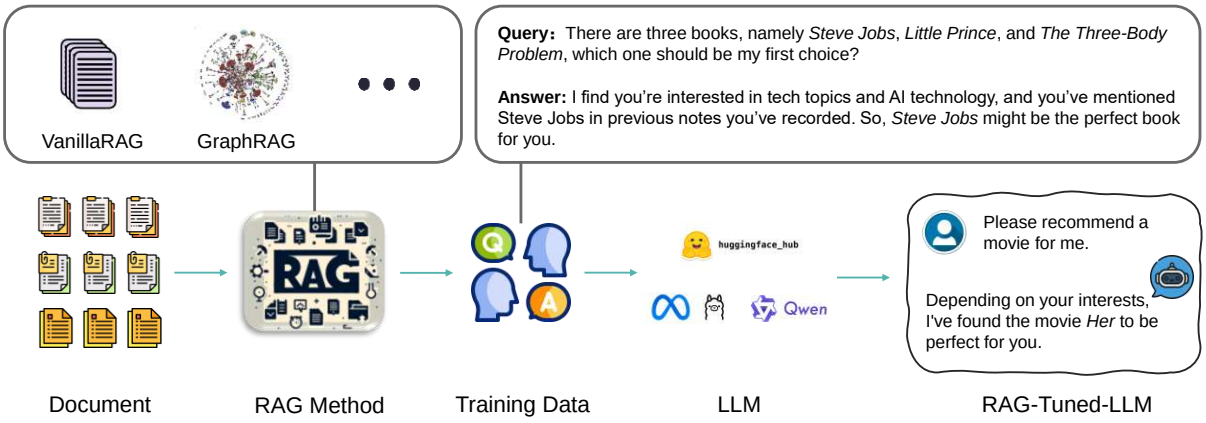

Figure 1: Overview of our RAG-Tuned-LLM method. **Stage 1**: RAG provides the foundation for synthesizing training data (query-answer pairs) for fine-tuning. **Stage 2**: The synthesized data is used to fine-tune a large language model (LLM) via LoRA. **Stage 3**: Inference is performed exclusively with LLM-native memory, eliminating the need for external memory. The RAG-Tuned-LLM combines the strengths of LLM-native solutions and RAG methods.

LLM-native method RAG-Tuned-LLM which fine-tunes a relatively small (e.g., 7B) LLM using the data generated following the RAG principles, so it can combine the advantages of RAG and long-context solutions. We call it *LLM-native* because it maintains the same speed as directly prompting an LLM with only the question —– without requiring long contexts or retrieval from a knowledge base. It enables the LLM to parameterize knowledge in a way that allows it to maintain contextual coherence and handle different types of queries more naturally and efficiently. Specifically, as illustrated in Figure 1, we follow the GraphRAG (Edge et al., 2024) principles to extract useful information from plain text documents. We then generate data from both *local* and *global* perspectives: (1) *local* data synthesis concentrates on generating content-specific query and answer pairs, and (2) *global* data synthesis focuses on producing query-answer pairs that integrate insights across entities and relationships. And finally, we employ the widely adopted LoRA technique (Hu et al., 2021) to fine-tune the LLM.

Our experiments then demonstrate that RAG-Tuned-LLM can beat long-context LLMs and RAG methods on both local and global queries, and further case studies show that RAG-Tuned-LLM excels in providing insightful and user-friendly responses. Our codes will be released to the public at Github upon acceptance.

Our contributions are summarized as follows.

- We create three datasets with *local* and *global* queries with their references, and then systematically compare *LLM-native* and (vanilla) RAG solutions, showing their respective unique advantages. It is worth mentioning that on one dataset, both the queries and references are manually created by human users.
- We follow the comparison results and propose a novel LLM-native method RAG-Tuned-LLM to combine the advantages of RAG and long-context solutions.
- Extensive experiments on three datasets demonstrate that RAG-Tuned-LLM can indeed outperform long-context LLMs and (advanced) RAG methods on both *local* and *global* queries.

2 Long-context vs. RAG

To motivate our work, we systematically compare long-context and RAG solutions and discuss their respect strengths in this section.

2.1 Settings

Datasets. We consider three datasets for comparison, as detailed in Table 1. For the two public datasets—**News** articles (Tang and Yang, 2024) and **Podcast** transcripts (Scott, 2024)—we follow Edge et al. (2024) to generate 125 *local* queries and 125 *global* queries for each, along with their corresponding references. The **Journaling** dataset, newly introduced by us, is proprietary and derived from our journaling app. It contains 45 *local* queries and 15 *global* queries designed by users, accompanied by their expected answers. Users were informed to craft queries aimed at complex and nuanced scenarios, prioritizing reasoning capabilities over simple retrieval. It is designed to robustly

Table 1: Dataset statistics. **Memory refers to the raw texts that will be utilized as additional information for answering queries.** Evaluation queries are split into *local* and *global* partitions according to their scopes.

Dataset	Memory		Evaluation Queries		
	# Docs	# Tokens	Global	Local	Avg Tokens
Podcast	66	832K	125	125	22.30
News	609	1214K	125	125	22.02
Journaling	538	230K	45	15	39.57

evaluate models' ability to handle intricate reasoning tasks in diverse real-world scenarios. It extends beyond basic fact retrieval to assess how well models can retrieve specific details while performing higher-order reasoning. Please refer to Table 1 for detailed statistics.

Methods. For the long-context LLM, we choose `Gemini-1.5-pro-001` due to its remarkable 2-million-token context window, which stands out as one of the longest among widely recognized and authoritative LLMs. This extensive context capacity sufficiently accommodates our experimental needs without requiring truncation. For the RAG methods, we implement `VanillaRAG` using standard embedding and reranking techniques from the Langchain framework [2]. Specifically, VanillaRAG employs the `text-embedding-ada-002` model for initial chunk retrieval, selecting the top-10 most relevant chunks. These chunks are then refined using Cohere's `rerank-english-v3.0` model, which filters the 10 chunks down to 3. We use `GPT-4o-mini`[3] considering the cost efficiency and performance. By incorporating both embedding-based recall and reranking, this method serves as a strong RAG solution.

2.2 Evaluation Metrics

We design our evaluation criteria to ensure that the generated answers are not only accurate but also practically helpful for real-world applications, such as personal assistants. We refer to the attribute perspectives in (Li et al., 2024a) and ranking prioritization in (Wang et al., 2024) as:

- **Helpful** assesses the precision, contextual relevance, and practical value of the response in effectively addressing the query.

[2] LangChain: https://www.langchain.com/

[3] Our small-scale experiment shows that `GPT-4o-mini` as the language model for answer generation in VanillaRAG delivers comparable performance with significantly lower cost than `GPT-4o`.

Table 2: Wining rates of **Gemini-1.5** over **VanillaRAG** on *local* and *global* queries across three datasets using the four introduced metrics. Values exceeding **50%** indicate that Gemini-1.5 outperforms VanillaRAG.

Dataset	Metric	Local	Global	Overall
Podcast	Helpful	81.60%	86.40%	84.00%
	Rich	87.20%	90.40%	88.80%
	Insightful	90.40%	90.40%	90.40%
	User-Friendly	85.60%	88.80%	87.20%
	Overall	86.20%	89.00%	87.60%
News	Helpful	46.40%	56.60%	51.20%
	Rich	48.80%	56.80%	52.80%
	Insightful	49.60%	58.40%	54.00%
	User-Friendly	46.40%	58.40%	52.40%
	Overall	47.80%	57.55%	52.60%
Journaling	Helpful	53.33%	93.33%	83.33%
	Rich	46.67%	88.80%	80.00%
	Insightful	53.33%	91.11%	81.67%
	User-Friendly	53.33%	93.33%	83.33%
	Overall	51.67%	91.64%	82.08%

- **Rich** measures the comprehensiveness, depth, and diversity of perspectives of the response.
- **Insightful** evaluates the profundity of understanding and the uniqueness of insights offered.
- **User-Friendly** focuses on the clarity, coherence, and accessibility of the response.

In Table 2, we additionally report an "overall" metric, calculated as the average performance across the aforementioned four metrics. More detailed explanations of these metrics are deferred to Appendix B.

We evaluate responses from two competitors on various queries and compute the winning rate of one method over the other. We adopt an LLM as the judge, comparing the two answers based on the target metric, the query, and a reference answer. The reference answer, meticulously crafted and verified, provides a solid foundation for the LLM's comparison. To mitigate stochastic variability, this evaluation process is repeated multiple times. Notably, in our experiments, we observed comparable judging performance between `GPT-4o-mini` and `GPT-4o`. For cost efficiency, we report results using `GPT-4o-mini`. After aligning the LLM's evaluations with human assessments, we found a concordance rate of 86%, which is high enough for fair comparison, with 215 out of 250 cases exhibiting agreement. Considering the cost and insights from GraphRAG, we believe that the size of this test set is quite convincing.

Table 3: Graph statistics for the three datasets. The Graph Statics columns summarize the number of extracted entities, relations, and communities. The Synthesized SFT Data columns detail the number of generated queries, average query token count, and average answer token count.

	Graph Statistics			Our Synthesized SFT Data		
Dataset	**Entities**	**Relations**	**Communities**	**# of Queries**	**Avg Query Tokens**	**Avg Answer Tokens**
Podcast	5,182	8,631	837	54,627	23.29	264.04
News	17,877	26,208	3,534	155,896	23.54	273.19
Journaling	2,930	3,751	547	18,355	36.46	562.60

2.3 Results

We present the winning rates of the long-context LLM compared to VanillaRAG in Table 2. The data reveals that the long-context solution, though more expensive, consistently achieves markedly superior performance on *global* queries. Conversely, for *local* queries, the advantages of long-context solutions diminish significantly. Notably, in the news dataset, VanillaRAG outperforms its counterpart across all four evaluation metrics. This aligns with our intuition that RAG is particularly advantageous for extracting fine-grained information needed for *local* queries, whereas long-context solutions excel in addressing *global* queries that demand a comprehensive understanding of memory. The above results indicate that, similar to the findings of AI-native memory (Shang et al., 2024), although RAG and long-context LLMs can access the correct answer within the provided context, they do not always produce the correct response.

3 Our RAG-Tuned-LLM

Building on our findings, we propose a novel LLM-native approach named RAG-Tuned-LLM, which fine-tunes a relatively small (e.g., 7B) LLM using the data synthesized following RAG principles, thereby harnessing the strengths of both RAG and long-context solutions. In this section, we first provide an overview of our approach, followed by a detailed exposition of the *global* and *local* data synthesis processes, as well as the fine-tuning stage of the language model.

3.1 Overview

As illustrated in Figure 1, the key idea of RAG-Tuned-LLM is to synthesize high-quality data following RAG principles and tuning them into the LLM parameters. The data synthesis strategy is designed to ensure the final tuned model to be versatile and context-aware.

In our implementation, we particularly choose GraphRAG (Edge et al., 2024), as it is a recent advanced RAG method capable of constructing hierarchical memory. We focus on crafting query-answer pairs from text units, entities, and relationships. Specifically, we generate data from both *local* and *global* perspectives: (1) *local* data synthesis concentrates on generating content-specific query and answer pairs, and (2) *global* data synthesis focuses on producing query and answer pairs that integrate insights across entities and relationships.

Table 3 presents detailed statistics of the synthesized data, offering insights into the graph structure constructed by GraphRAG, including the number of entities, relations, and communities. Additionally, Table 3 also summarizes the synthesized SFT Data, detailing the number of queries, average query token count, and average answer token count. With the synthesized data, fine-tuning the LLM becomes a natural progression, where we utilize the widely adopted LoRA technique (Hu et al., 2021).

Next, we will delve into the details of each component of the proposed RAG-Tuned-LLM method, namely the *local* and *global* data synthesis strategy, as well as the fine-tuning process for the LLM.

3.2 Global Data Synthesis

Building upon the GraphRAG constructed graph, the *global* data synthesis process can be divided into two parts, based on the graph components used, namely entity-based data synthesis and relationship-based data synthesis.

Entity-based Data Synthesis. For each entity, we craft a description using meticulously designed templates tailored to the entity type, such as a person, event, or object. These templates facilitate the creation of natural and engaging questions, prompting the model to examine the role of the entity within a broader context during the subsequent query-and-answer pair generation phase. In practice, to ensure detailed and coherent answers, we adopt the chain-of-thought (CoT) reasoning frame-

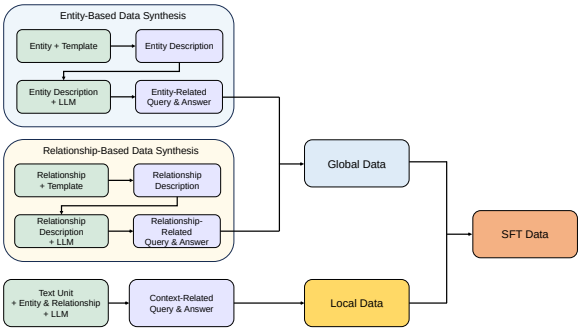

Figure 2: Overview of the data synthesis process used in RAG-Tuned-LLM. *Global* data synthesis comprises entity-based and relationship-based data synthesis, which generates query-answer pairs through the integration of templates and LLMs. *Local* data synthesis generates query-answer pairs using text units enriched by entries and relationships, along with LLMs.

work (Wei et al., 2022), resulting in more comprehensive and accurate responses. Specifically, the approach comprises the following three key steps:

1. **Restating the context**: Commence the response by concisely summarizing the situation or topic, ensuring a seamless flow and clarity, so that the answer remains coherent and contextually grounded.
2. **Integrating entity description**: Merge essential details about the entity with pertinent information from the broader context, crafting a more nuanced and insightful answer that adds depth and relevance.
3. **Constructing a detailed answer**: Offer a thorough and detailed explanation, typically ranging from 300 to 500 words, to comprehensively address the query, making use of all the available relevant information.

Moreover, to enhance clarity, we employ subheadings and bullet points to organize the content. This structured approach ensures that the generated questions and answers effectively capture both specific details and the broader context.

Relationship-based Data Synthesis. In a manner similar to entity-based data synthesis, we utilize relationship-specific templates to generate queries that delve into how entities interact. By merging entity and relationship-based queries with CoT reasoning-generated answers, the model can better understand both detailed insights and broader perspectives. Figure 2 depicts the overall *global* data synthesis process.

3.3 Local Data Synthesis

Local data synthesis involves generating queries from text units that encompass multiple entities and relationships, with an emphasis on *local* details. These text units offer the context needed to craft queries that investigate specific, localized aspects of the entities or relationships. The process includes:

1. **Assessing *local* information**: The text units is examined to identify the pertinent entities or relationships, concentrating on the specific details within the given context.
2. **Generating context-specific queries**: Queries are crafted based on the roles of these entities or relationships within the localized context, using the text units as the immediate reference.

These localized queries focus on specific interactions or characteristics within the text, providing detailed insights into the smaller components of the data. As Figure 2 shows, integrating *local* and *global* data produces the final SFT dataset, with the entire data synthesis process adhering to RAG principles.

3.4 LM tuning

The combination of entity-based, relationship-based, and localized context-based query-answer pair generation facilitates fine-tuning an LLM to natively embody the memory extracted through GraphRAG, i.e., LLM-native memory, thereby combining the strengths of both RAG and LLM-native solutions (e.g., long-context LLMs).

While full fine-tuning (Lv et al., 2023) generally achieves a higher performance ceiling, it demands significantly more computational resources and extensive training data. Furthermore, full fine-tuning may compromise the base model's generalization ability. Given the relatively small-scale fine-tuning data, we adopt LoRA, a widely used PEFT (Ding et al., 2023) method, to parameterize a base LLM with the memory generated via RAG methods.

4 Experiments

4.1 Experimental Setup

Datasets and Evaluation Metrics. We consider the three datasets introduced in Section 2, namely News, Podcast, and Journaling. Detailed statistics and characteristics of these datasets are provided in Table1. Evaluation metrics are also in consistent with the four introduced in Section 2, namely helpful, rich, insightful and user-friendly.

Table 4: Winning rates (averaged across four evaluation metrics) of our **RAG-Tuned-LLM** compared to VanillaRAG, GraphRAG, Long-context LLM, and Normal SFT on the Podcast, News, and Journaling datasets. Local and Global refer to different evaluation contexts. For comparison, the check mark indicates the characteristics employed by each method. Winning rates exceeding **50%** confirm that our RAG-Tuned-LLM outperforms all the compared methods.

Methods				Podcast		News		Journaling	
Type	RAG Principle	LLM-Native	Parameterized Memory	Local	Global	Local	Global	Local	Global
VanillaRAG	✔	✗	✗	94.80%	96.20%	94.60%	95.80%	81.67%	95.56%
Long-context LLM	✗	✔	✗	65.60%	67.60%	94.00%	95.60%	66.67%	73.33%
Normal SFT	✗	✔	✔	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Averaged GraphRAG	✔	✗	✗	57.95%	57.95%	56.35%	57.41%	51.67%	59.31%
RAG-Tuned-LLM (Ours)	✔	✔	✔	N/A	N/A	N/A	N/A	N/A	N/A

Compared Methods. To investigate the superiority of our proposed RAG-Tuned-LLM, we compare it with other four methods, i.e., **VanillaRAG**, **GraphRAG**, **Long-Context LLM**, and **Normal SFT**. For VanillaRAG and the long-context LLM, we adopt the configurations detailed in Section 2, utilizing `GPT-4o-mini` with plain documents as external memory for VanillaRAG and `Gemini-1.5-pro-001` for the long-context LLM. GraphRAG is a recently advanced RAG technique, which can generate responses leveraging four hierarchical graph community information integration strategies, ranging from high-level to fine-grained, labeled **C0 to C3**:

- **C0** employs root-level community summaries.
- **C1** employs sub-communities of C0 but still high-level community summaries.
- **C2** employs intermediate-level community summaries.
- **C3** employs low-level community summaries.

The language model for GraphRAG is also set to `GPT-4o-mini`. For the normal SFT method, we follow (Jiang et al., 2024) to transform raw data into query-answer pairs for finetune an LLM, adopting the same setting as RAG-Tuned-LLM, i,e., selecting `Qwen-2-7B-instruct` (Bai et al., 2023a) as the base model, and employ a LoRA with its rank $r = 64$ for parameterizing the model's memory. It is important to note that all methods are fundamentally provided with the same dataset, albeit processed in different formats.

Training Configurations. In the training process, we adopt a cosine learning rate scheduler, with a maximum learning rate of 1×10^{-4}, and set the total number of fine-tuning epochs to 3. To ensure more stable results, we set the decoding temperature to 0 during inference.

4.2 Superiority of RAG-Tuned-LLM

Table 4 summarize the winning rate of our proposed RAG-Tuned-LLM against other four compared methods. Our key point is that RAG-Tuned-LLM can effectively handles both *local* and *global* queries simultaneously, while others can not. Therefore, we report the average result across four evaluation metrics and focus on the overall result regarding different query types. Moreover, for simplity of our interpretation and comparison, we also average the results of four different GraphRAG levels, i.e., C0 to C3, and you can refer to Table 6 and 7 in Appendix for detailed results.

From the results, it is evident that our RAG-Tuned-LLM outperforms all competitors in addressing both *local* and *global* problems, with its superiority being particularly pronounced when compared to VanillaRAG, long-context LLM, and Normal SFT. We attribute the success to the fact that the RAG data enables the model to obtain fine-grained factual information for the problem, while the tuning of the memory to be LLM-native provides a deeper, more *global* understanding of the issue. Furthermore, from the comparison with normal SFT, we can find that though given the same external memory, the formulation of the training data synthesis has a great influence on the model performance. GraphRAG emerges as the most competitive baseline, likely due to its incorporation of both fine-grained and high-level information in its responses. The graph it generates includes both abstract and varied levels of information, while the RAG approach retains the advantage of relevant information integration when generating responses. However, GraphRAG still inherits the conventional limitation of RAG, relying on external data sources for its responses. We argue that parameterizing the memory to be LLM-native is more effective than retrieval-based approaches. By integrating relevant

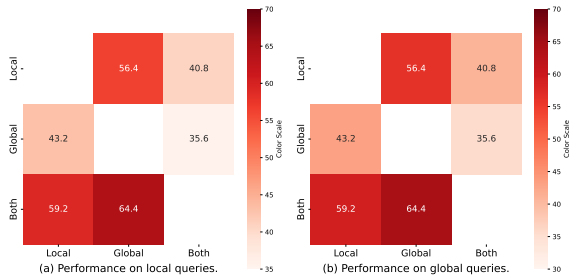

Figure 3: The comparison among RAG-Tuned-LLM models trained with different synthesized data types, i.e., *local* split, *global* split, and both. We evaluate the models on local and *global* queries separately to ablate the effect of training data.

information directly into the model's parameters, the LLM can generate more coherent and contextually aware responses without the need to repeatedly access external sources, ultimately improving both the efficiency and quality of the answers.

4.3 Ablation Studies on the Training Data

Recall that our goal is for RAG-Tuned-LLM to excel at both *local* and *global* queries. Therefore, our data synthesis strategy also consists of two parts: *local* and *global* data synthesis. In this section, we will investigate how the type of training data influences the model's performance. Specifically, we consider three scenarios in the Podcast transcripts dataset: LLM tuning with *local* data only, *global* data only, and both *local* and *global* data combined. In order to better understand the effects of *local* and *global* data, we evaluate the tuned model separately on *local* and *global* queries. The winning rates of one training data type against another are illustrated in Figure 3.

As we can observe in the figure, models tuned with *local* data perform better on *local* queries than those tuned with *global* data, and vice versa. When both *local* and *global* data are combined, the model achieves the best results on both *local* and *global* queries. This highlights the benefit of using diverse training data types, enhancing the model's robustness and generalization. These ablation studies also demonstrate the profound impact that training data has on the performance of a deep learning model.

4.4 Evaluation of Generalization Capability

Since RAG-Tuned-LLM can be understood as training and testing within a fixed knowledge domain, it is natural for us to evaluate the model's generalization ability. We divide generalization into two aspects: (1) the ability to answer unseen queries within the same knowledge domain and (2) the model's general capability beyond the given domain. For the first aspect, our test queries are generated using methods significantly different from those used for the training data, meaning that the test results inherently reflect the model's ability to answer out-of-training-distribution queries within the domain. Therefore, the following evaluation will primarily focus on the second aspect.

Table 5: Zero-shot performance comparison between the original base model and our RAG-Tuned-LLM across three distinct capabilities.

Dataset	Capability	Original Model	RAG-Tuned-LLM
MMLU	English	80.80%	73.50%
GSM8K	Mathematics	63.66%	61.72%
HumanEval	Coding	57.90%	56.70%

To illustrate the generalization capability beyond the given domain, we compare its zero-shot performance with that of the original base model across three widely recognized large-scale benchmarks: MMLU (Hendrycks et al., 2020), GSM8K (Cobbe et al., 2021), and HumanEval (Chen et al., 2021). Specifically, we utilize the model fine-tuned on **News** articles, as it encompasses the largest volume of training tokens. The experimental results summarized in Table 5 reveal that RAG-Tuned-LLM incurs only a slight degradation in performance compared to the original base model, thereby underscoring its robust generalization capability.

5 Related Works

5.1 Retrieval Augment Generation

Pre-trained language models, such as Qwen (Bai et al., 2023b) and Llama (Touvron et al., 2023a), have shown impressive query-answering capabilities. However, they face limitations when tasked with problems requiring knowledge beyond their training data. Retrieval-augmented generation (RAG) (Lewis et al., 2020) provides a solution by retrieving relevant information from an external knowledge base. While RAG has proven to be practical and effective, traditional RAG systems can only retrieve raw corpus related to the query, without broader comprehension. As a result, abstract queries such as those asking for high-level insights or overarching understandings often lead to suboptimal answers. To overcome these limitations, GraphRAG (Edge et al., 2024) has been introduced. Specifically, GraphRAG constructs a

knowledge graph using an LLM, enabling it to provide hierarchical information that range from specific, detailed facts to more global, abstract insights, leveraging the knowledge graph for a more comprehensive understanding

5.2 Long-context LLM

Long-context LLMs are designed to handle tasks that involve processing extended sequences of text, addressing a significant limitation of traditional LLMs, which typically operate with fixed, limited context windows. For example, GPT-4o (Achiam et al., 2023) offers a context window of up to 128K tokens, while Gemini 1.5 (Reid et al., 2024) can manage up to 1M or 10M tokens. Furthermore, various studies have sought to push the boundaries of these context windows, suggesting models capable of "unlimited" context lengths through innovations such as length extrapolation (Peng et al., 2023; Xiao et al., 2023; Han et al., 2023; Zhang et al., 2024a) and position bias adjustments (Liu et al., 2024; Peysakhovich and Lerer, 2023; An et al., 2024). Long-context LLMs, in principle, possess the potential to offer more refined abstraction abilities and a deeper, more nuanced understanding of global context compared to RAG methods. Yet, as highlighted by Hsieh et al. (2024); Shang et al. (2024), the context may surpass the constraints of the LLM's context window, which is typically much narrower than reported, leading to the inadvertent loss of crucial information amid an expansive sea of text.

5.3 Fine-Tuning LLMs

To incrementally expand the knowledge of a pretrained LLM or to align it with human preferences, fine-tuning stands as one of the most prevalent approaches, encompassing methods such as supervised fine-tuning (SFT), reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022), and direct preference optimization (DPO) (Rafailov et al., 2024). Despite their effectiveness, these techniques are notably annotation-heavy and computationally intensive, rendering the fine-tuning of an LLM using these methods prohibitively costly. To circumvent the extensive computational demands of full fine-tuning, which can reach into tens of billions, numerous parameter-efficient fine-tuning (PEFT) methods have been explored, including BitFit (Zaken et al., 2021), adapter (Houlsby et al., 2019), and Lora (Hu et al., 2021). In this paper, we primarily employ a LoRA to fine-tune a RAG-Tuned-LLM. Methodologically, RAFT (Zhang et al., 2024b) is the closest to our approach, as it explores the potential integration of RAG and fine-tuning. However, there are two fundamental differences between our work and RAFT: First, the model we train is not intended for use in the generation stage of RAG, making our objectives fundamentally different; Second, our training data does not include deliberately introduced noise, which distinguishes our approach significantly in terms of methodology.

6 Conclusion and Future Work

In this paper, we validate RAG's fine-grained retrieval abilities and the global abstraction strengths of LLM-native solutions. However, RAG lacks holistic understanding, and long-context models tend to lose key information over extended contexts. We integrate these strengths of both RAG and LLM-native solutions by fine-tuning an LLM within an RAG framework for data generation. This work is the first to explore LLM and RAG integration within a unified framework, bridging open-domain and domain-specific query-answering tasks. Our RAG-Tuned LLM, equipped with LLM-native memory, outperforms both standard RAG methods and long-context LLMs across diverse datasets, demonstrating superior performance in handling hierarchical queries.

Future Work. Building on this study, several future directions are worth exploring to further validate and enhance our proposed method. First, we plan to extend RAG-Tuned-LLM to more diverse datasets and domains, enabling us to evaluate its generalizability across different tasks, including complex challenges like multi-hop reasoning and multi-modal query-answering. This will provide a clearer understanding of RAG-Tuned-LLM's effectiveness in both open-domain and domain-specific contexts. Additionally, we will experiment with various foundational models (e.g., the Llama series (Touvron et al., 2023a,b)), evaluating RAG-Tuned-LLM's adaptability to different architectures and model scales. This will highlight the trade-offs between model size, computational efficiency, and performance when combining RAG and LLM-native methods.

Limitations

While our proposed method, RAG-Tuned-LLM, demonstrates substantial advantages over long-context LLMs and RAG in handling both *global* and *local* queries, we recognize two key limitations that warrant further investigation. First, although LLM-as-a-judge is a widely adopted evaluation approach (Li et al., 2024b), the metrics we utilized remain relatively domain-specific—suitable for applications like personal assistants but less adaptable to general-purpose language models. Enhancing the robustness and generalizability of our evaluation framework is imperative. Second, although we have validated our method's robustness and generalization to some extent (e.g., in English, mathematics, and coding capabilities), broader exploration such as in the realms of multi-modal and multi-hop reasoning tasks remains insufficient.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, and Jian-Guang Lou. 2024. Make your llm fully utilize the context. *arXiv preprint arXiv:2404.16811*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023a. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023b. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Chi Han, Qifan Wang, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. 2023. Lm-infinite: Simple on-the-fly length generalization for large language models. *arXiv preprint arXiv:2308.16137*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Zhengbao Jiang, Zhiqing Sun, Weijia Shi, Pedro Rodriguez, Chunting Zhou, Graham Neubig, Xi Victoria Lin, Wen tau Yih, and Srinivasan Iyer. 2024. Instruction-tuned language models are better knowledge learners. In *Annual Meeting of the Association for Computational Linguistics*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2024a. From generation to judgment: Opportunities and challenges of llm-as-a-judge.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, et al. 2024b. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *arXiv preprint arXiv:2411.16594*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Kai Lv, Yuqing Yang, Tengxiao Liu, Qinghui Gao, Qipeng Guo, and Xipeng Qiu. 2023. Full parameter fine-tuning for large language models with limited resources. *arXiv preprint arXiv:2306.09782*.

Jinjie Mai, Jun Chen, Bing chuan Li, Guocheng Qian, Mohamed Elhoseiny, and Bernard Ghanem. 2023. Llm as a robotic brain: Unifying egocentric memory and control. *ArXiv*, abs/2304.09349.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, et al. 2022. Quality: Question answering with long input texts, yes! In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5336–5358.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. In *The Twelfth International Conference on Learning Representations*.

Alexander Peysakhovich and Adam Lerer. 2023. Attention sorting combats recency bias in long context language models. *arXiv preprint arXiv:2310.01427*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Kevin Scott. 2024. [link].

Jingbo Shang, Zai Zheng, Xiang Ying, Felix Tao, and Mindverse Team. 2024. Ai-native memory: A pathway from llms towards agi. *arXiv preprint arXiv:2406.18312*.

Yixuan Tang and Yi Yang. 2024. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. 2024. Helpsteer2-preference: Complementing ratings with preferences. *ArXiv*, abs/2410.01257.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2023. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*.

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. 2021. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.

Peitian Zhang, Zheng Liu, Shitao Xiao, Ninglu Shao, Qiwei Ye, and Zhicheng Dou. 2024a. Soaring from 4k to 400k: Extending llm's context with activation beacon. *arXiv preprint arXiv:2401.03462*.

Tianjun Zhang, Shishir G. Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E. Gonzalez. 2024b. Raft: Adapting language model to domain specific rag. *Preprint*, arXiv:2403.10131.

A Definition of Global and Local Queries

A notable innovation in our query generation method lies in the differentiation between *global* and *local* queries, akin to the approach used in GraphRAG, but with a more pronounced emphasis on user-driven tasks. Particularly, we define *local* and *global* queries as follows:

- **Global Queries**: Global queries are crafted to elicit high-level, interpretive responses that require the user to consider the dataset in its entirety. They address overarching trends, themes, and insights that emerge from the data, steering the user toward macro-level analysis. Therefore, global query synthesis demands multiple dataset chunks, ensuring that the user engages with the dataset holistically, rather than fixating on specific details.
- **Local Queries**: Local queries are retrieval-oriented, aiming to direct the user toward specific pieces of information within the dataset. Each query is designed to be answerable by referencing a particular section or chunk of the data, promoting a detailed and focused analysis. Local queries necessitate precision in information retrieval and cater to users seeking clear, concrete answers to more narrowly defined questions.

By categorizing the queries into these two types, we ensure that the evaluation of RAG systems encompasses both granular detail retrieval and broader sensemaking tasks, thereby offering a more comprehensive assessment of the system's capability to engage with the dataset at multiple levels.

B Explaination of Evaluation Metrics

- **Helpful**: This metric evaluates the accuracy and reliability of the answer in relation to the posed query. It examines whether the answer directly addresses the query and delivers useful, relevant information. Answers that exhibit clear correctness and offer valuable content receive higher scores on this metric.
- **Rich**: This metric evaluates the variety and depth of the content provided in the answer. An answer that explores multiple perspectives or offers detailed explanations from different angles is deemed more diverse and rich. It emphasizes comprehensiveness and the ability to present a nuanced understanding of the dataset or topic.
- **Insightful**: This metric measures the depth of understanding demonstrated in the answer. Insightful responses reflect a profound comprehension of the subject matter and may offer thoughtful or original insights that transcend surface-level retrieval. Answers that meaningfully synthesize data to provide novel or perceptive interpretations receive higher ratings.
- **User-Friendly**: This metric assesses the clarity, readability, and organization of the response. An answer that is well-structured, concise, and easily comprehensible will score higher. This metric ensures that even complex responses remain accessible and understandable to the target audience, striking a balance between depth and usability.

C Results of Local and Global Subsets

Table 4 in the main body of the paper only summarizes the averaged results across four evaluation metrics and four distinct levels of GraphRAG responses. In this section, we provide more detailed results for each metric and each level of GraphRAG responses. Table 6 and 7 shows the winning rates of our RAG-Tuned-LLM over GraphRAG (C0 to C3), Long-context LLM, VanillaRAG, and normal SFT on *local* and *global* queries, respectively. The results demonstrate that our RAG-Tuned-LLM generally outperforms all the compared methods across all metrics.

D Exampls of RAG-Tuned-LLM vs. GraphRAG

As shown in Table 6 and 7, GraphRAG is the strongest competitor among the four methods compared. Therefore, we present two concrete examples to qualitatively demonstrate the superiority of RAG-Tuned-LLM over GraphRAG, beyond numerical performance, as shown in Figure 4 and 5.

Table 6: Winning rates (%) of our RAG-Tuned-LLM over GraphRAG (C0 to C3), Long-context LLM, VanillaRAG, and Normal SFT across four evaluation metrics on *local* queries.

Dataset	Metric	GraphRAG C0	GraphRAG C1	GraphRAG C2	GraphRAG C3	Long-Context LLM	VanillaRAG	Normal SFT
Podcast	Helpful	56.80	53.60	52.00	52.80	65.60	95.20	100.00
	Rich	52.80	49.60	47.20	48.00	59.20	96.00	100.00
	Insightful	59.20	54.40	50.40	51.20	60.00	99.20	100.00
	User-Friendly	80.00	76.00	72.00	71.20	77.60	88.80	100.00
News	Helpful	52.00	52.80	49.60	50.40	95.20	95.20	100.00
	Rich	50.40	49.60	45.60	46.40	94.40	99.20	100.00
	Insightful	56.00	55.20	51.20	51.20	96.00	99.20	100.00
	User-Friendly	78.40	73.60	70.40	68.80	90.40	84.80	100.00
LPM	Helpful	53.33	46.67	46.67	46.67	60.00	73.33	100.00
	Rich	46.67	53.33	46.67	46.67	66.67	86.67	100.00
	Insightful	66.67	60.00	53.33	60.00	73.33	86.67	100.00
	User-Friendly	53.33	46.67	53.33	46.67	66.67	80.00	100.00

Table 7: Winning rates (%) of our RAG-Tuned-LLM over GraphRAG (C0 to C3), Long-context LLM, VanillaRAG, and Normal SFT across four evaluation metrics on *global* queries.

Dataset	Metric	GraphRAG C0	GraphRAG C1	GraphRAG C2	GraphRAG C3	Long-Context LLM	VanillaRAG	Normal SFT
Podcast	Helpful	54.40	55.20	52.80	52.00	68.00	97.60	100.00
	Rich	52.00	49.60	47.20	45.60	61.60	97.60	100.00
	Insightful	59.20	52.80	53.60	50.40	62.40	99.20	100.00
	User-Friendly	82.40	76.80	73.60	69.60	78.40	90.40	100.00
News	Helpful	52.80	53.60	51.20	52.00	96.80	98.40	100.00
	Rich	51.20	48.80	46.40	46.60	96.00	99.20	100.00
	Insightful	56.80	56.00	52.00	52.00	97.60	100.00	100.00
	User-Friendly	79.20	73.60	71.20	68.80	92.00	85.60	100.00
LPM	Helpful	57.78	55.56	55.56	53.33	64.44	93.33	100.00
	Rich	55.56	57.78	57.78	55.56	73.33	100.00	100.00
	Insightful	68.89	68.89	64.44	68.89	82.22	100.00	100.00
	User-Friendly	57.78	57.78	57.78	55.56	73.33	88.89	100.00

Figure 4: A concrete example (Case 1) from the News dataset illustrating the superiority of RAG-Tuned-LLM compared to GraphRAG.

Case 1

Query: What is the expected release date of Britney Spears' memoir?

GraphRAG C3 Answer: Britney Spears' highly anticipated memoir, titled *The Woman in Me*, is set to be released on **October 24, 2023**. This memoir is expected to delve into significant moments in her life, including her experiences with fame, motherhood, and the impact of her conservatorship, which has been a focal point of public interest and discussion.

The memoir is described as a candid account of her life, providing insights into her personal struggles and triumphs, and is anticipated to resonate deeply with her fans and the general public alike.

RAG-Tuned-LLM Answer: Britney Spears' memoir, titled *The Woman in Me*, is anticipated to be released on **October 24, 2023**. This highly anticipated book is set to delve into her life experiences and challenges, particularly focusing on her conservatorship, a defining aspect of her public persona for many years. The memoir promises to provide readers with an intimate look at her journey through fame, personal struggles, and the complexities of her relationships, including her marriage to Sam Asghari.

Following a tumultuous period marked by the conservatorship, which was finally terminated in November 2021, Britney's memoir comes at a pivotal moment in her life. The end of the conservatorship allowed her to regain control over her life and career, offering her a sense of newfound freedom. In interviews, she has expressed sadness about the loss of her childhood home, describing it as a place where she felt most like herself. This emotional context adds depth to her narrative, as she explores themes of identity, resilience, and the impact of fame on her personal relationships.

The memoir is not merely a recounting of events; it is also a reflection on her personal growth and the lessons she has learned along the way. Britney has been open about her struggles with mental health and the pressures of public life, making her story relatable to many who have faced similar challenges. The memoir is also expected to offer insights into her relationship with Sam Asghari, whom she married in June 2022. Their marriage ended in July 2023, adding another layer of complexity to her story.

Overall, *The Woman in Me* is poised to be a significant literary work, capturing the essence of Britney Spears' life and offering readers a chance to understand her perspective on fame, motherhood, and her quest for autonomy. As the release date approaches, there is growing anticipation about how Britney will share her story and what new revelations may come to light.

Figure 5: A concrete example (Case 2) from the News dataset illustrating the superiority of RAG-Tuned-LLM compared to GraphRAG.

Case2

Query: How many counts of fraud and money laundering is Sam Bankman-Fried (SBF) facing in his trial?

GraphRAG C3 Answer: Sam Bankman-Fried (SBF) is facing a total of seven counts related to his trial, which include charges of fraud and conspiracy. Specifically, these charges encompass wire fraud and money laundering, among other allegations tied to the operations of FTX and the misappropriation of customer funds.

The trial has garnered significant attention due to the serious nature of these allegations, which are considered some of the largest financial frauds in recent history.

RAG-Tuned-LLM Answer: In the ongoing trial of Sam Bankman-Fried, commonly referred to as SBF, he is facing serious legal charges that include multiple counts of fraud and money laundering. The context surrounding this trial is quite significant, as it has garnered considerable attention due to the collapse of FTX, a cryptocurrency exchange that was once valued at an impressive $32 billion before its downfall. This dramatic shift in fortune has led to a high-profile legal battle that has captured the interest of both the public and legal experts alike.

The trial itself is set against a backdrop of allegations that SBF engaged in fraudulent activities while managing FTX and its affiliated trading firm, Alameda Research. Specifically, the prosecution claims that SBF orchestrated a scheme that misappropriated customer funds, leading to substantial financial losses for investors and customers. The prosecution's case is expected to unfold over several weeks, with the government presenting evidence that will detail how SBF allegedly misled investors and customers about the safety and management of their assets.

As part of the trial proceedings, SBF has entered a plea of not guilty, which indicates his intent to contest the charges against him. The defense's strategy will likely revolve around challenging the prosecution's narrative and presenting arguments that could potentially exonerate him. However, the prosecution's case is bolstered by the testimony of key witnesses, including former associates and industry experts who can provide insight into the operations of FTX and the broader implications of the alleged fraud.

In summary, Sam Bankman-Fried is facing multiple counts of fraud and money laundering in his trial, which is a critical juncture for both him and the cryptocurrency industry as a whole. The outcome of this trial could have far-reaching consequences, not only for SBF but also for regulatory frameworks governing cryptocurrency exchanges moving forward. As the trial progresses, it will be essential to monitor how the evidence presented impacts the jury's perception and ultimately influences the verdict.