Retrieval-Augmented Generation for Large Language Models: A Survey

Yunfan Gao [1], **Yun Xiong** [2], **Xinyu Gao** [2], **Kangxiang Jia** [2], **Jinliu Pan** [2], **Yuxi Bi** [3], **Yi Dai**[1], **Jiawei Sun**[1], **Qianyu Guo**[4], **Meng Wang** [3] and **Haofen Wang** [1,3] *

[1] Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University
[2] Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University
[3] College of Design and Innovation, Tongji University
[4] School of Computer Science, Fudan University

Abstract

Large Language Models (LLMs) demonstrate significant capabilities but face challenges such as hallucination, outdated knowledge, and non-transparent, untraceable reasoning processes. Retrieval-Augmented Generation (RAG) has emerged as a promising solution by incorporating knowledge from external databases. This enhances the accuracy and credibility of the models, particularly for knowledge-intensive tasks, and allows for continuous knowledge updates and integration of domain-specific information. RAG synergistically merges LLMs' intrinsic knowledge with the vast, dynamic repositories of external databases. This comprehensive review paper offers a detailed examination of the progression of RAG paradigms, encompassing the Naive RAG, the Advanced RAG, and the Modular RAG. It meticulously scrutinizes the tripartite foundation of RAG frameworks, which includes the retrieval , the generation and the augmentation techniques. The paper highlights the state-of-the-art technologies embedded in each of these critical components, providing a profound understanding of the advancements in RAG systems. Furthermore, this paper introduces the metrics and benchmarks for assessing RAG models, along with the most up-to-date evaluation framework. In conclusion, the paper delineates prospective avenues for research, including the identification of challenges, the expansion of multi-modalities, and the progression of the RAG infrastructure and its ecosystem. [1].

1 Introduction

Large language models (LLMs) such as the GPT series [Brown *et al.*, 2020, OpenAI, 2023] and the LLama series [Touvron *et al.*, 2023], along with other models like Gemini [Google, 2023], have achieved remarkable success in natural language processing, demonstrating superior performance on various benchmarks including SuperGLUE [Wang *et al.*, 2019], MMLU [Hendrycks *et al.*, 2020], and BIG-bench [Srivastava *et al.*, 2022]. Despite these advancements, LLMs exhibit notable limitations, particularly in handling domain-specific or highly specialized queries [Kandpal *et al.*, 2023]. A common issue is the generation of incorrect information, or ”hallucinations” [Zhang *et al.*, 2023b], especially when queries extend beyond the model's training data or necessitate up-to-date information. These shortcomings underscore the impracticality of deploying LLMs as black-box solutions in real-world production environments without additional safeguards. One promising approach to mitigate these limitations is Retrieval-Augmented Generation (RAG), which integrates external data retrieval into the generative process, thereby enhancing the model's ability to provide accurate and relevant responses.

RAG, introduced by Lewis et al. [Lewis *et al.*, 2020] in mid-2020, stands as a paradigm within the realm of LLMs, enhancing generative tasks. Specifically, RAG involves an initial retrieval step where the LLMs query an external data source to obtain relevant information before proceeding to answer questions or generate text. This process not only informs the subsequent generation phase but also ensures that the responses are grounded in retrieved evidence, thereby significantly enhancing the accuracy and relevance of the output. The dynamic retrieval of information from knowledge bases during the inference phase allows RAG to address issues such as the generation of factually incorrect content, commonly referred to as “hallucinations.” The integration of RAG into LLMs has seen rapid adoption and has become a pivotal technology in refining the capabilities of chatbots and rendering LLMs more viable for practical applications.

The evolutionary trajectory of RAG unfolds across four distinctive phases, as illustrated in Figure 1. In its inception in 2017, aligned with the emergence of the Transformer architecture, the primary thrust was on assimilating additional knowledge through Pre-Training Models (PTM) to augment language models. This epoch witnessed RAG's foundational efforts predominantly directed at optimizing pre-training methodologies.

Following this initial phase, a period of relative dormancy ensued before the advent of chatGPT, during which there was minimal advancement in related research for RAG. The subsequent arrival of chatGPT marked a pivotal moment in the

[*] Corresponding Author.Email:haofen.wang@tongji.edu.cn

[1] Resources are available at https://github.com/Tongji-KGLLM/RAG-Survey

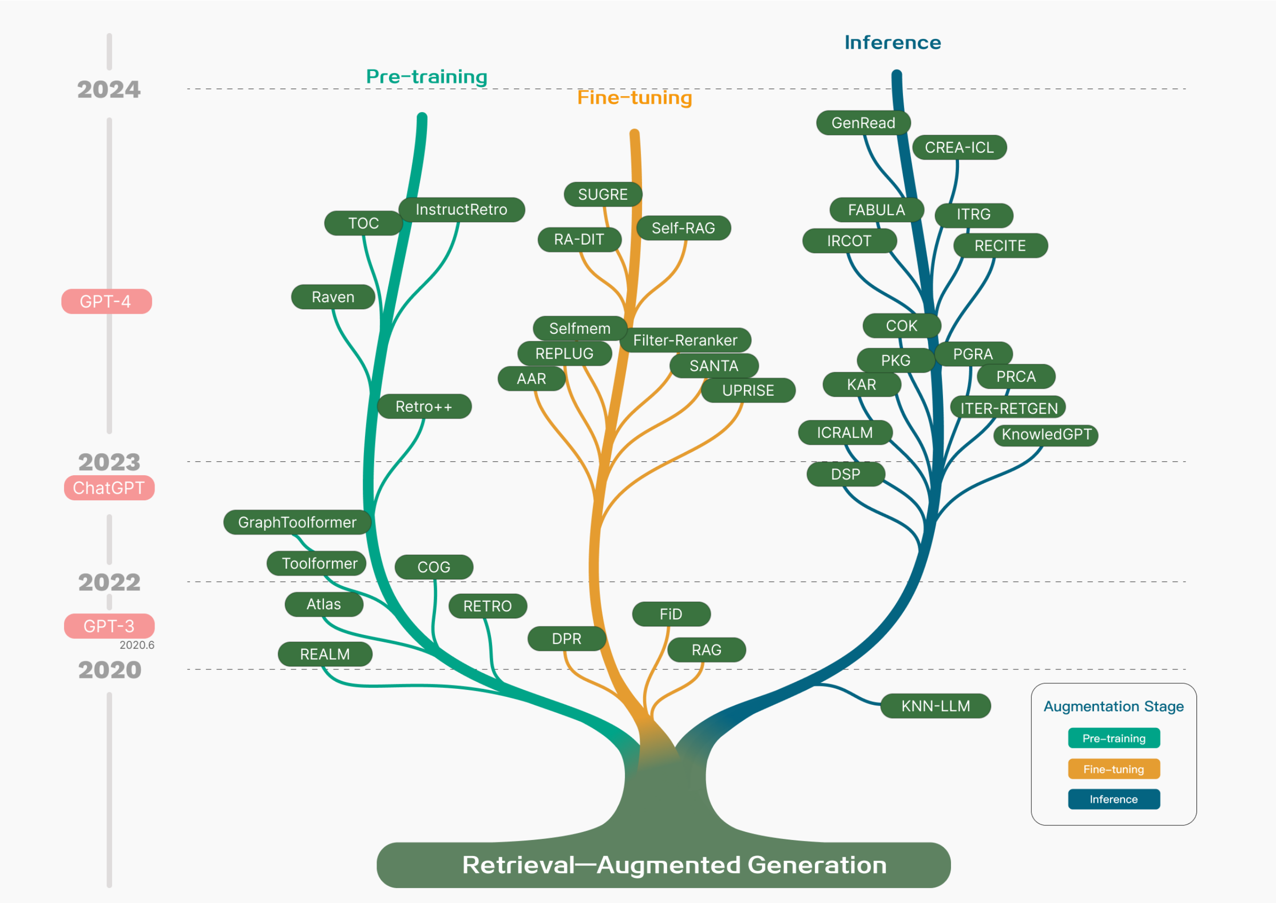

Figure 1: Technology tree of RAG research development featuring representative works

trajectory, propelling LLMs into the forefront. The community's focal point shifted towards harnessing the capabilities of LLMs to attain heightened controllability and address evolving requirements. Consequently, the lion's share of RAG endeavors concentrated on inference, with a minority dedicated to fine-tuning processes. As LLM capabilities continued to advance, especially with the introduction of GPT-4, the landscape of RAG technology underwent a significant transformation. The emphasis evolved into a hybrid approach, combining the strengths of RAG and fine-tuning, alongside a dedicated minority continuing the focus on optimizing pre-training methodologies.

Despite the rapid growth of RAG research, there has been a lack of systematic consolidation and abstraction in the field, which poses challenges in understanding the comprehensive landscape of RAG advancements. This survey aims to outline the entire RAG process and encompass the current and future directions of RAG research, by providing a thorough examination of retrieval augmentation in LLMs.

Therefore, this paper aims to comprehensively summarize and organize the technical principles, developmental history, content, and, in particular, the relevant methods and applications after the emergence of LLMs, as well as the evaluation methods and application scenarios of RAG. It seeks to provide a comprehensive overview and analysis of existing RAG technologies and offer conclusions and prospects for future development methods. This survey intends to furnish readers and practitioners with a thorough and systematic comprehension of large models and RAG, elucidate the progression and key technologies of retrieval augmentation, clarify the merits and limitations of various technologies along with their suitable contexts, and forecast potential future developments.

Our contributions are as follows:

- We present a thorough and systematic review of the state-of-the-art RAG, delineating its evolution through paradigms including naive RAG, advanced RAG, and modular RAG. This review contextualizes the broader scope of RAG research within the landscape of LLMs.
- We identify and discuss the central technologies integral to the RAG process, specifically focusing on the aspects of "Retrieval", "Generator" and "Augmentation", and delve into their synergies, elucidating how these components intricately collaborate to form a cohesive and effective RAG framework.
- We construct a thorough evaluation framework for RAG, outlining the evaluation objectives and metrics. Our comparative analysis clarifies the strengths and weaknesses of RAG compared to fine-tuning from various

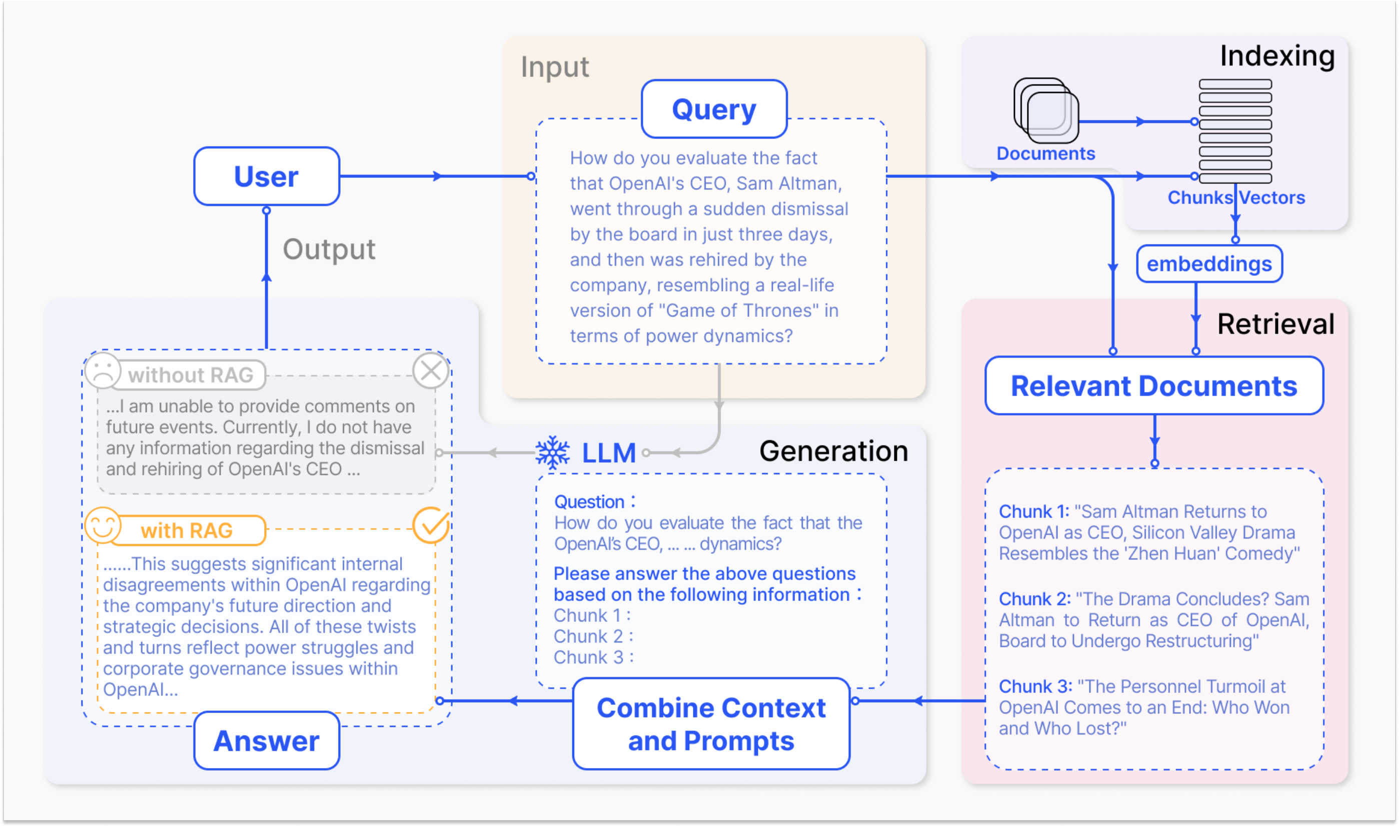
Input
Query
How do you evaluate the fact that OpenAI's CEO, Sam Altman, went through a sudden dismissal by the board in just three days, and then was rehired by the company, resembling a real-life version of "Game of Thrones" in terms of power dynamics?
Indexing
Documents
Chunks
Vectors
embeddings
User
Output
Retrieval
Relevant Documents
without RAG
...I am unable to provide comments on future events. Currently, I do not have any information regarding the dismissal and rehiring of OpenAI's CEO ...
with RAG
......This suggests significant internal disagreements within OpenAI regarding the company's future direction and strategic decisions. All of these twists and turns reflect power struggles and corporate governance issues within OpenAI...
Answer
LLM
Generation
Question：
How do you evaluate the fact that the OpenAI's CEO, ... dynamics?
Please answer the above questions based on the following information：
Chunk 1 :
Chunk 2 :
Chunk 3 :
Combine Context and Prompts
Chunk 1: "Sam Altman Returns to OpenAI as CEO, Silicon Valley Drama Resembles the 'Zhen Huan' Comedy"
Chunk 2: "The Drama Concludes? Sam Altman to Return as CEO of OpenAI, Board to Undergo Restructuring"
Chunk 3: "The Personnel Turmoil at OpenAI Comes to an End: Who Won and Who Lost?"

Figure 2: A representative instance of the RAG process applied to question answering

the input into a vector representation. It then proceeds to compute the similarity scores between the query vector and the vectorized chunks within the indexed corpus. The system prioritizes and retrieves the top K chunks that demonstrate the greatest similarity to the query. These chunks are subsequently used as the expanded contextual basis for addressing the user's request.

Generation

The posed query and selected documents are synthesized into a coherent prompt to which a large language model is tasked with formulating a response. The model's approach to answering may vary depending on task-specific criteria, allowing it to either draw upon its inherent parametric knowledge or restrict its responses to the information contained within the provided documents. In cases of ongoing dialogues, any existing conversational history can be integrated into the prompt, enabling the model to engage in multi-turn dialogue interactions effectively.

Drawbacks in Naive RAG

Naive RAG faces significant challenges in three key areas: "Retrieval," "Generation," and "Augmentation".

Retrieval quality poses diverse challenges, including low precision, leading to misaligned retrieved chunks and potential issues like hallucination or mid-air drop. Low recall also occurs, resulting in the failure to retrieve all relevant chunks, thereby hindering the LLMs' ability to craft comprehensive responses. Outdated information further compounds the problem, potentially yielding inaccurate retrieval results.

Response generation quality presents hallucination challenge, where the model generates answers not grounded in the provided context, as well as issues of irrelevant context and potential toxicity or bias in the model's output.

The augmentation process presents its own challenges in effectively integrating context from retrieved passages with the current generation task, potentially leading to disjointed or incoherent output. Redundancy and repetition are also concerns, especially when multiple retrieved passages contain similar information, resulting in repetitive content in the generated response.

Discerning the importance and relevance of multiple retrieved passages to the generation task is another challenge, requiring the proper balance of each passage's value. Additionally, reconciling differences in writing styles and tones to ensure consistency in the output is crucial.

Lastly, there's a risk of generation models overly depending on augmented information, potentially resulting in outputs that merely reiterate the retrieved content without providing new value or synthesized information.

3.2 Advanced RAG

Advanced RAG has been developed with targeted enhancements to address the shortcomings of Naive RAG. In terms of retrieval quality, Advanced RAG implements pre-retrieval

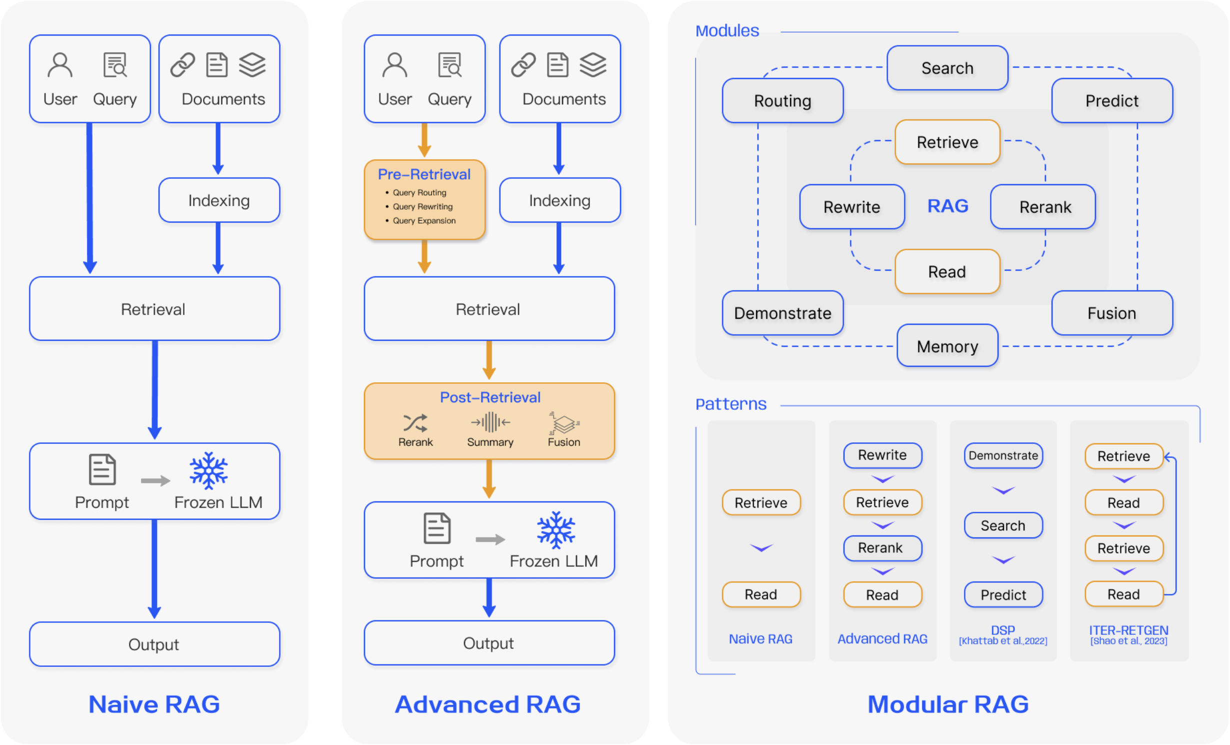

Figure 3: Comparison between the three paradigms of RAG

is depicted in Figure 3. However, Modular RAG is not standalone. Advanced RAG is a specialized form of modular RAG, and further, Naive RAG itself is a special case of Advanced RAG. The relationship among the three paradigms is one of inheritance and development.

New Modules

Search Module. In contrast to the similarity retrieval in Naive/Advanced RAG, the Search Module is tailored to specific scenarios and incorporates direct searches on additional corpora. This integration is achieved using code generated by the LLM, query languages such as SQL or Cypher, and other custom tools. The data sources for these searches can include search engines, text data, tabular data, and knowledge graphs [Wang *et al.*, 2023d].

Memory Module. This module harnesses the memory capabilities of the LLM to guide retrieval. The approach involves identifying memories most similar to the current input. Selfmem [Cheng *et al.*, 2023b] utilizes a retrieval-enhanced generator to create an unbounded memory pool iteratively, combining the “original question” and “dual question”. By employing a retrieval-enhanced generative model that uses its own outputs to improve itself, the text becomes more aligned with the data distribution during the reasoning process. Consequently, the model’s own outputs are utilized instead of the training data [Wang *et al.*, 2022a].

Fusion. RAG-Fusion [Raudaschl, 2023]enhances traditional search systems by addressing their limitations through a multi-query approach that expands user queries into multiple, diverse perspectives using an LLM. This approach not only captures the explicit information users seek but also uncovers deeper, transformative knowledge. The fusion process involves parallel vector searches of both original and expanded queries, intelligent re-ranking to optimize results, and pairing the best outcomes with new queries. This sophisticated method ensures search results that align closely with both the explicit and implicit intentions of the user, leading to more insightful and relevant information discovery.

Routing. The RAG system’s retrieval process utilizes diverse sources, differing in domain, language, and format, which can be either alternated or merged based on the situation [Li *et al.*, 2023b]. Query routing decides the subsequent action to a user’s query, with options ranging from summarization, searching specific databases, or merging different pathways into a single response. The query router also chooses the appropriate data store for the query, which may include various sources like vector stores, graph databases, or relational databases, or a hierarchy of indices—for instance, a summary index and a document block vector index for multi-document storage. The query router’s decision-making is predefined and executed via LLMs calls, which direct the query to the chosen index.

Predict . It addresses the common issues of redundancy and noise in retrieved content. Instead of directly retrieving from a data source, this module utilizes the LLM to generate the necessary context [Yu *et al.*, 2022]. The content produced by the LLM is more likely to contain pertinent information compared to that obtained through direct retrieval.

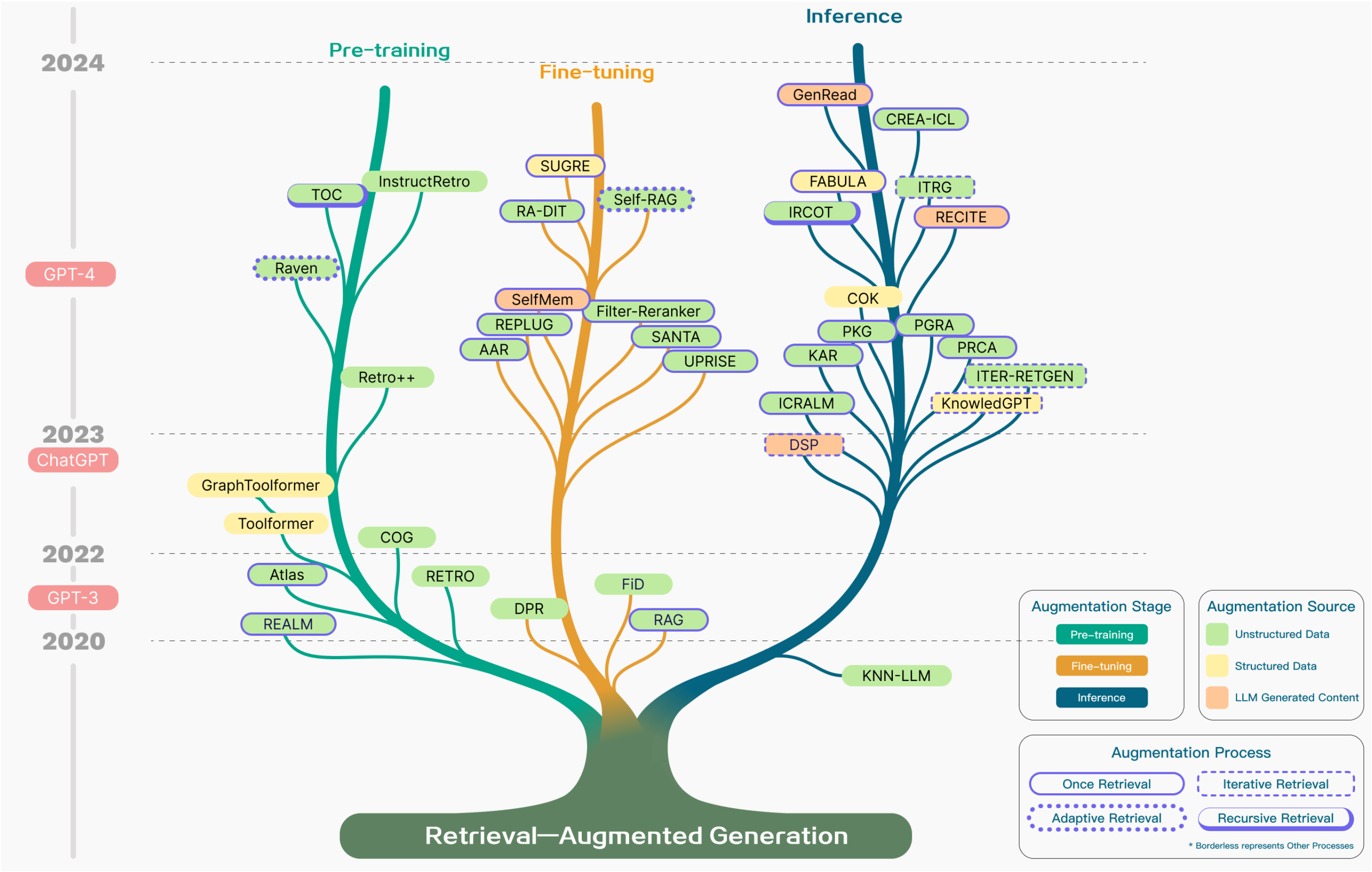

Figure 5: Technology tree of representative RAG research with different augmentation aspects

it typically relies on a sequence of n tokens to demarcate the boundaries between generated text and retrieved documents.

To address specific data scenarios, recursive retrieval and multi-hop retrieval techniques are utilized. Recursive retrieval involves a structured index to process and retrieve data in a hierarchical manner, which may include summarizing sections of a document or lengthy PDF before performing a retrieval based on this summary. Subsequently, a secondary retrieval within the document refines the search, embodying the recursive nature of the process. In contrast, multi-hop retrieval is designed to delve deeper into graph-structured data sources, extracting interconnected information [Li *et al.*, 2023c].

Additionally, some methodologies integrate the steps of retrieval and generation. ITER-RETGEN [Shao *et al.*, 2023] employs a synergistic approach that leverages "retrieval-enhanced generation" alongside "generation-enhanced retrieval" for tasks that necessitate the reproduction of specific information. The model harnesses the content required to address the input task as a contextual basis for retrieving pertinent knowledge, which in turn facilitates the generation of improved responses in subsequent iterations.

Recursive Retrieval

Recursive Retrieval is often used in information retrieval and NLP to improve the depth and relevance of search results. The process involves iteratively refining search queries based on the results obtained from previous searches. Recursive Retrieval aims to enhance the search experience by gradually converging on the most pertinent information through a feedback loop. IRCoT [Trivedi *et al.*, 2022] uses chain-of-thought to guide the retrieval process and refines the CoT with the obtained retrieval results. ToC [Kim *et al.*, 2023] creates a clarification tree that systematically optimizes the ambiguous parts in the Query. It can be particularly useful in complex search scenarios where the user's needs are not entirely clear from the outset or where the information sought is highly specialized or nuanced. The recursive nature of the process allows for continuous learning and adaptation to the user's requirements, often resulting in improved satisfaction with the search outcomes.

Adaptive Retrieval

Adaptive retrieval methods, exemplified by Flare and Self-RAG [Jiang *et al.*, 2023b, Asai *et al.*, 2023], refine the RAG framework by enabling LLMs to actively determine the optimal moments and content for retrieval, thus enhancing the efficiency and relevance of the information sourced.

These methods are part of a broader trend wherein LLMs employ active judgment in their operations, as seen in model agents like AutoGPT, Toolformer, and Graph-Toolformer [Yang *et al.*, 2023c, Schick *et al.*, 2023,

Table 1: Comparison between RAG and Fine-Tuning

Feature Comparison	RAG	Fine-Tuning
Knowledge Updates	Directly updating the retrieval knowledge base ensures that the information remains current without the need for frequent retraining, making it well-suited for dynamic data environments.	Stores static data, requiring retraining for knowledge and data updates.
External Knowledge	Proficient in leveraging external resources, particularly suitable for accessing documents or other structured/unstructured databases.	Can be utilized to align the externally acquired knowledge from pretraining with large language models, but may be less practical for frequently changing data sources.
Data Processing	Involves minimal data processing and handling.	Depends on the creation of high-quality datasets, and limited datasets may not result in significant performance improvements.
Model Customization	Focuses on information retrieval and integrating external knowledge but may not fully customize model behavior or writing style.	Allows adjustments of LLM behavior, writing style, or specific domain knowledge based on specific tones or terms.
Interpretability	Responses can be traced back to specific data sources, providing higher interpretability and traceability.	Similar to a black box, it is not always clear why the model reacts a certain way, resulting in relatively lower interpretability.
Computational Resources	Depends on computational resources to support retrieval strategies and technologies related to databases. Additionally, it requires the maintenance of external data source integration and updates.	The preparation and curation of high-quality training datasets, defining fine-tuning objectives, and providing corresponding computational resources are necessary.
Latency Requirements	Involves data retrieval, which may lead to higher latency.	LLM after fine-tuning can respond without retrieval, resulting in lower latency.
Reducing Hallucinations	Inherently less prone to hallucinations as each answer is grounded in retrieved evidence.	Can help reduce hallucinations by training the model based on specific domain data but may still exhibit hallucinations when faced with unfamiliar input.
Ethical and Privacy Issues	Ethical and privacy concerns arise from the storage and retrieval of text from external databases.	Ethical and privacy concerns may arise due to sensitive content in the training data.

model [Liu, 2023]. Additionally, both retrieval and generation quality assessments can be conducted through manual or automatic evaluation methods [Liu, 2023, Lan *et al.*, 2022, Leng *et al.*, 2023].

7.2 Evaluation Aspects

Contemporary evaluation practices of RAG models emphasize three primary quality scores and four essential abilities, which collectively inform the evaluation of the two principal targets of the RAG model: retrieval and generation.

Quality Scores

Quality scores include context relevance, answer faithfulness, and answer relevance. These quality scores evaluate the efficiency of the RAG model from different perspectives in the process of information retrieval and generation [Es *et al.*, 2023, Saad-Falcon *et al.*, 2023, Jarvis and Allard, 2023]. The quality scores—context relevance, answer faithfulness, and answer relevance—assess the RAG model's efficiency from various angles throughout the information retrieval and generation process [Es *et al.*, 2023, Saad-Falcon *et al.*, 2023, Jarvis and Allard, 2023].

Context Relevance evaluates the precision and specificity of the retrieved context, ensuring relevance and minimizing processing costs associated with extraneous content.

Answer Faithfulness ensures that the generated answers remain true to the retrieved context, maintaining consistency

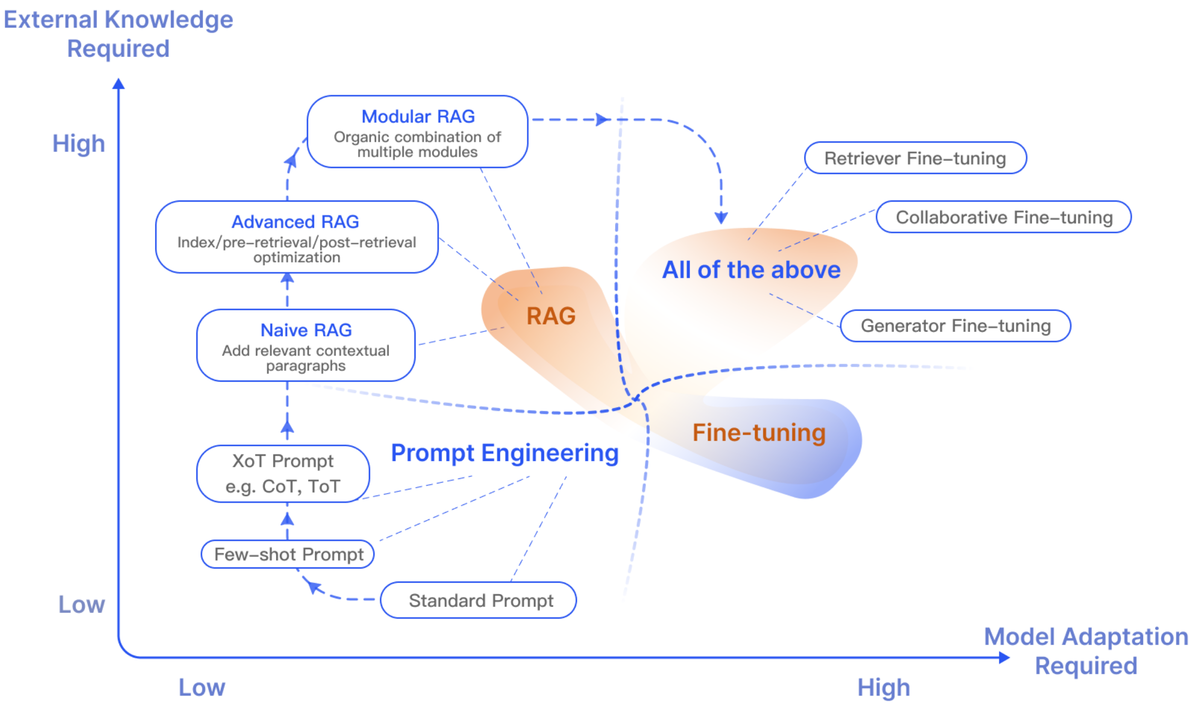
External Knowledge Required
High
Low
Model Adaptation Required
Low
High
Modular RAG
Organic combination of multiple modules
Advanced RAG
Index/pre-retrieval/post-retrieval optimization
Naive RAG
Add relevant contextual paragraphs
RAG
All of the above
Retriever Fine-tuning
Collaborative Fine-tuning
Generator Fine-tuning
Fine-tuning
Prompt Engineering
XoT Prompt
e.g. CoT, ToT
Few-shot Prompt
Standard Prompt

Figure 6: RAG compared with other model optimization methods

and avoiding contradictions.

Answer Relevance requires that the generated answers are directly pertinent to the posed questions, effectively addressing the core inquiry.

Required Abilities

RAG evaluation also encompasses four abilities indicative of its adaptability and efficiency: noise robustness, negative rejection, information integration, and counterfactual robustness [Chen *et al.*, 2023b, Liu *et al.*, 2023b]. These abilities are critical for the model's performance under various challenges and complex scenarios, impacting the quality scores.

Noise Robustness appraises the model's capability to manage noise documents that are question-related but lack substantive information.

Negative Rejection assesses the model's discernment in refraining from responding when the retrieved documents do not contain the necessary knowledge to answer a question.

Information Integration evaluates the model's proficiency in synthesizing information from multiple documents to address complex questions.

Counterfactual Robustness tests the model's ability to recognize and disregard known inaccuracies within documents, even when instructed about potential misinformation.

Context relevance and noise robustness are important for evaluating the quality of retrieval, while answer faithfulness, answer relevance, negative rejection, information integration, and counterfactual robustness are important for evaluating the quality of generation.

The specific metrics for each evaluation aspect are summarized in Table 2. It is essential to recognize that these metrics, derived from related work, are traditional measures and do not yet represent a mature or standardized approach for quantifying RAG evaluation aspects. Custom metrics tailored to the nuances of RAG models, though not included here, have also been developed in some evaluation studies.

7.3 Evaluation Benchmarks and Tools

This section delineates the evaluation framework for RAG models, comprising benchmark tests and automated evaluation tools. These instruments furnish quantitative metrics that not only gauge RAG model performance but also enhance comprehension of the model's capabilities across various evaluation aspects. Prominent benchmarks such as RGB and RECALL [Chen *et al.*, 2023b, Liu *et al.*, 2023b] focus on appraising the essential abilities of RAG models. Concurrently, state-of-the-art automated tools like RAGAS [Es *et al.*, 2023], ARES [Saad-Falcon *et al.*, 2023], and TruLens[8] employ LLMs to adjudicate the quality scores. These tools and benchmarks collectively form a robust framework for the systematic evaluation of RAG models, as summarized in Table 3.

[8] https://www.trulens.org/trulens_eval/core_concepts_rag_triad/

Table 2: Summary of metrics applicable for evaluation aspects of RAG

	Context Relevance	Faithfulness	Answer Relevance	Noise Robustness	Negative Rejection	Information Integration	Counterfactual Robustness
Accuracy	✓	✓	✓	✓	✓	✓	✓
EM					✓		
Recall	✓						
Precision	✓			✓			
R-Rate							✓
Cosine Similarity			✓				
Hit Rate	✓						
MRR	✓						
NDCG	✓						

Table 3: Summary of evaluation frameworks

Evaluation Framework	**Evaluation Targets**	**Evaluation Aspects**	**Quantitative Metrics**
RGB†	Retrieval Quality Generation Quality	Noise Robustness Negative Rejection Information Integration Counterfactual Robustness	Accuracy EM Accuracy Accuracy
RECALL†	Generation Quality	Counterfactual Robustness	R-Rate (Reappearance Rate)
RAGAS‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	* * Cosine Similarity
ARES‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	Accuracy Accuracy Accuracy
TruLens‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	* * *

*† represents a benchmark, and ‡ represents a tool. * denotes customized quantitative metrics, which deviate from traditional metrics. Readers are encouraged to consult pertinent literature for the specific quantification formulas associated with these metrics, as required.*

8 Future Prospects

This section explores three future prospects for RAG: future challenges, modality expansion, and the RAG ecosystem.

8.1 Future Challenges of RAG

Despite the considerable progress in RAG technology, several challenges persist that warrant in-depth research:

Context Length. RAG's efficacy is limited by the context window size of Large Language Models (LLMs). Balancing the trade-off between a window that is too short, risking insufficient information, and one that is too long, risking information dilution, is crucial. With ongoing efforts to expand LLM context windows to virtually unlimited sizes, the adaptation of RAG to these changes presents a significant research question [Xu *et al.*, 2023c, Packer *et al.*, 2023, Xiao *et al.*, 2023].

Robustness. The presence of noise or contradictory information during retrieval can detrimentally affect RAG's output quality. This situation is figuratively referred to as "Misinformation can be worse than no information at all". Improving RAG's resistance to such adversarial or counterfactual inputs is gaining research momentum and has become a key performance metric [Yu *et al.*, 2023a, Glass *et al.*, 2021, Baek *et al.*, 2023].

Hybrid Approaches (RAG+FT). Combining RAG with fine-tuning is emerging as a leading strategy. Determining the optimal integration of RAG and fine-tuning whether sequential, alternating, or through end-to-end joint training—and how to harness both parameterized and non-parameterized advantages are areas ripe for exploration [Lin *et al.*, 2023].

Expanding LLM Roles. Beyond generating final answers, LLMs are leveraged for retrieval and evaluation within RAG frameworks. Identifying ways to further unlock LLMs potential in RAG systems is a growing research direction.

Scaling Laws. While scaling laws [Kaplan *et al.*, 2020] are established for LLMs, their applicability to RAG remains un-

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[VoyageAI, 2023] VoyageAI. Voyage's embedding models. https://docs.voyageai.com/embeddings/, 2023.

[Wang *et al.*, 2019] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

[Wang *et al.*, 2022a] Shuohang Wang, Yichong Xu, Yuwei Fang, Yang Liu, Siqi Sun, Ruochen Xu, Chenguang Zhu, and Michael Zeng. Training data is more valuable than you think: A simple and effective method by retrieving from training data. *arXiv preprint arXiv:2203.08773*, 2022.

[Wang *et al.*, 2022b] Shuohang Wang, Yichong Xu, Yuwei Fang, Yang Liu, Siqi Sun, Ruochen Xu, Chenguang Zhu, and Michael Zeng. Training data is more valuable than you think: A simple and effective method by retrieving from training data. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3170–3179, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[Wang *et al.*, 2023a] Boxin Wang, Wei Ping, Lawrence McAfee, Peng Xu, Bo Li, Mohammad Shoeybi, and Bryan Catanzaro. Instructretro: Instruction tuning post retrieval-augmented pretraining. *arXiv preprint arXiv:2310.07713*, 2023.

[Wang *et al.*, 2023b] Boxin Wang, Wei Ping, Peng Xu, Lawrence McAfee, Zihan Liu, Mohammad Shoeybi, Yi Dong, Oleksii Kuchaiev, Bo Li, Chaowei Xiao, et al. Shall we pretrain autoregressive language models with retrieval? a comprehensive study. *arXiv preprint arXiv:2304.06762*, 2023.

[Wang *et al.*, 2023c] Liang Wang, Nan Yang, and Furu Wei. Query2doc: Query expansion with large language models. *arXiv preprint arXiv:2303.07678*, 2023.

[Wang *et al.*, 2023d] Xintao Wang, Qianwen Yang, Yongting Qiu, Jiaqing Liang, Qianyu He, Zhouhong Gu, Yanghua Xiao, and Wei Wang. Knowledgpt: Enhancing large language models with retrieval and storage access on knowledge bases. *arXiv preprint arXiv:2308.11761*, 2023.

[Wang *et al.*, 2023e] Yile Wang, Peng Li, Maosong Sun, and Yang Liu. Self-knowledge guided retrieval augmentation for large language models. *arXiv preprint arXiv:2310.05002*, 2023.

[Xia *et al.*, 2019] Mengzhou Xia, Guoping Huang, Lemao Liu, and Shuming Shi. Graph based translation memory for neural machine translation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7297–7304, 2019.

[Xiao *et al.*, 2023] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

[Xu *et al.*, 2023a] Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*, 2023.

[Xu *et al.*, 2023b] Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. Retrieval meets long context large language models. *arXiv preprint arXiv:2310.03025*, 2023.

[Xu *et al.*, 2023c] Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. Retrieval meets long context large language models. *arXiv preprint arXiv:2310.03025*, 2023.

[Yang *et al.*, 2023a] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10714–10726, 2023.

[Yang *et al.*, 2023b] Haoyan Yang, Zhitao Li, Yong Zhang, Jianzong Wang, Ning Cheng, Ming Li, and Jing Xiao. Prca: Fitting black-box large language models for retrieval question answering via pluggable reward-driven contextual adapter. *arXiv preprint arXiv:2310.18347*, 2023.

[Yang *et al.*, 2023c] Hui Yang, Sifu Yue, and Yunzhong He. Auto-gpt for online decision making: Benchmarks and additional opinions. *arXiv preprint arXiv:2306.02224*, 2023.

[Yasunaga *et al.*, 2022] Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Retrieval-augmented multimodal language modeling. *arXiv preprint arXiv:2211.12561*, 2022.

[Ye *et al.*, 2020] Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. Coreferential reasoning learning for language representation. *arXiv preprint arXiv:2004.06870*, 2020.

[Yoran *et al.*, 2023] Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*, 2023.

[Yu *et al.*, 2022] Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. Generate rather than retrieve: Large language models are strong context generators. *arXiv preprint arXiv:2209.10063*, 2022.

[Yu *et al.*, 2023a] Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210*, 2023.

[Yu *et al.*, 2023b] Zichun Yu, Chenyan Xiong, Shi Yu, and Zhiyuan Liu. Augmentation-adapted retriever improves generalization of language models as generic plug-in. *arXiv preprint arXiv:2305.17331*, 2023.

[Zhang *et al.*, 2019] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*, 2019.

[Zhang *et al.*, 2023a] Peitian Zhang, Shitao Xiao, Zheng Liu, Zhicheng Dou, and Jian-Yun Nie. Retrieve anything to augment large language models. *arXiv preprint arXiv:2310.07554*, 2023.

[Zhang *et al.*, 2023b] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*, 2023.

[Zhang, 2023] Jiawei Zhang. Graph-toolformer: To empower llms with graph reasoning ability via prompt augmented by chatgpt. *arXiv preprint arXiv:2304.11116*, 2023.

[Zhao *et al.*, 2022] Jinming Zhao, Gholamreza Haffar, and Ehsan Shareghi. Generating synthetic speech from spokenvocab for speech translation. *arXiv preprint arXiv:2210.08174*, 2022.

[Zheng *et al.*, 2023] Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. Take a step back: Evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*, 2023.

[Zhu *et al.*, 2022] Wanrong Zhu, An Yan, Yujie Lu, Wenda Xu, Xin Eric Wang, Miguel Eckstein, and William Yang Wang. Visualize before you write: Imagination-guided open-ended text generation. *arXiv preprint arXiv:2210.03765*, 2022.

[Zhuang *et al.*, 2023] Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. Open-source large language models are strong zero-shot query likelihood models for document ranking. *arXiv preprint arXiv:2310.13243*, 2023.