

Mitigating Hallucination in Large Language Models (LLMs): An Application-Oriented Survey on RAG, Reasoning, and Agentic Systems

Yihan Li, Xiyuan Fu, Ghanshyam Verma, Paul Buitelaar, and Mingming Liu

Abstract—Hallucination remains one of the key obstacles to the reliable deployment of large language models (LLMs), particularly in real-world applications. Among various mitigation strategies, Retrieval-Augmented Generation (RAG) and reasoning enhancement have emerged as two of the most effective and widely adopted approaches, marking a shift from merely suppressing hallucinations to balancing creativity and reliability. However, their synergistic potential and underlying mechanisms for hallucination mitigation have not yet been systematically examined. This survey adopts an application-oriented perspective of capability enhancement to analyze how RAG, reasoning enhancement, and their integration in Agentic Systems mitigate hallucinations. We propose a taxonomy distinguishing knowledge-based and logic-based hallucinations, systematically examine how RAG and reasoning address each, and present a unified framework supported by real-world applications, evaluations, and benchmarks.

Index Terms—Hallucination, Large Language Model (LLM), Retrieval-Augmented Generation (RAG), Reasoning, Chain-of-Thought (CoT), Agents

I. INTRODUCTION

Hallucination [1], [2], [3], defined as the generation of content that appears plausible but is inconsistent with real-world facts or user instructions, has emerged as one of the most critical challenges in the deployment of Large Language Models (LLMs) [4]. In high-stakes applications such as medical diagnosis, legal analysis, and scientific research, even minor factual or logical errors can lead to severe consequences [5], including the spread of misinformation and the erosion of public trust in AI systems [4]. Existing studies have shown that hallucinations remain widespread across models of all scales [6], from trillion-parameter systems such as ChatGPT-4 [7], and DeepSeek-R1 [8] to smaller models like LLaMA 7B [9] and Mistral [10], underscoring the urgency and necessity of addressing this issue. In recent years, the growing deployment of LLMs has revealed a practical dilemma: how to effectively suppress hallucinations while preserving generative capabilities and creativity has become a key challenge for both performance optimization and real-world adoption [11].

Yihan Li is with the Electronic Information School, Wuhan University, Wuhan, China, and the School of Computing, Dublin City University, Dublin, Ireland.

Xiyuan Fu is with the School of Public Health, Wuhan University, Wuhan, China.

Ghanshyam Verma and Paul Buitelaar are with the Insight Centre for Data Analytics, University of Galway, Ireland.

Mingming Liu is with the Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland. (Corresponding author: mingming.liu@dcu.ie)

Existing hallucination mitigation methods [1], [2] span multiple stages of the LLM lifecycle, including data processing and alignment strategies [12] during pre-training, fine-tuning [13] and reinforcement learning methods based on human feedback and preferences, and inference-stage intervention techniques [14] such as Retrieval-Augmented Generation (RAG) [15] and Chain-of-Thought (CoT) [16] reasoning. Recently, RAG and reasoning-based approaches [17] have attracted significant attention in both academia and industry. RAG expands the knowledge boundaries of models by introducing external knowledge sources [15], enabling them to access up-to-date, accurate and domain-specific information during inference, thereby reducing factual errors caused by missing or outdated knowledge [1]. Reasoning techniques [17] focus on enhancing models' capabilities in multi-step tasks, complex logical deduction, and problem decomposition, helping to prevent logic-based hallucinations arising from broken or inconsistent reasoning chains. Compared with other hallucination mitigation methods, these two approaches suppress hallucinations while preserving open-domain generative capabilities. Moreover, both have already been integrated as core modules in several commercial LLMs, such as Grok 4 [18], ChatGPT-4 [7], and Gemini [19], demonstrating strong applicability, scalability, and portability.

Most existing surveys [2], [1], [3] on hallucinations in LLMs focus on their causes, taxonomies, and lifecycle-based mitigation stages, or analyze isolated strategies at the data, model, or decoding levels. Although comprehensive in scope, these studies seldom examine hallucination mitigation from the perspective of enhancing system capabilities, specifically, how improving a model's knowledge access, reasoning, and planning abilities can reduce hallucination occurrence.

To fill this gap, this survey investigates three representative capability-oriented mitigation paradigms: RAG, reasoning enhancement, and Agentic Systems. These approaches share a common principle: instead of modifying model architectures or applying additional regularization, they strengthen LLM reliability through verifiable knowledge grounding and logical consistency constraints enabled by external knowledge sources, explicit reasoning chains, and dynamic planning mechanisms. This perspective shifts the focus from error elimination to capability enhancement, aligning with the long-term trends of improving model reliability and interpretability [11]. This survey adopts a capability-oriented analytical framework, systematically reviews recent advances in RAG, reasoning enhancement, and Agentic Systems, and establishes a unified

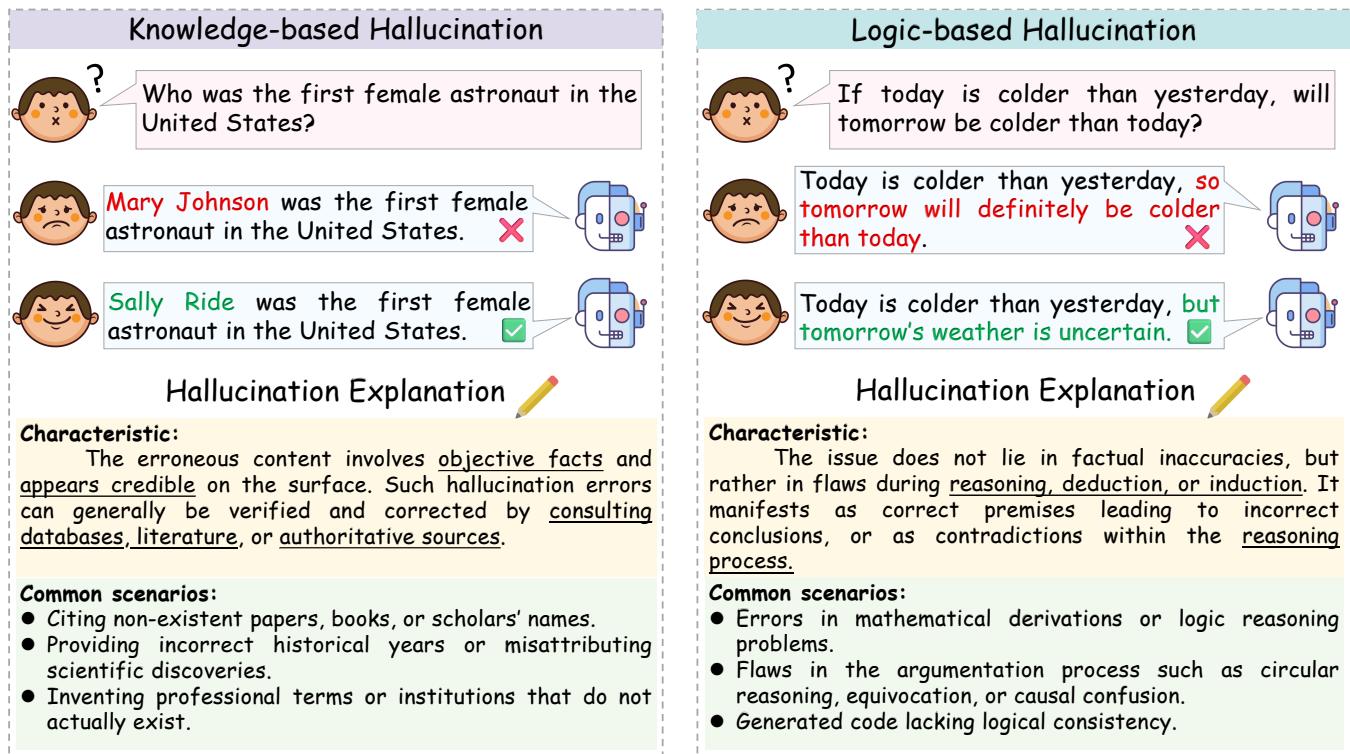


Fig. 1. Two types of hallucinations in LLM responses

evaluation perspective for hallucination mitigation.

To better characterize different types of hallucinations and their corresponding mitigation mechanisms, this survey adopts a mitigation-oriented taxonomy of knowledge-based hallucination and logic-based hallucination in practical applications of RAG and reasoning-based methods (as shown in Fig. 1).

Our contributions. With hallucination mitigation as the central objective and capability enhancement as the analytical criterion:

- We systematically review the representative RAG pipeline and analyze how two paradigms: Precise Retrieval and Broad Retrieval, mitigate knowledge-based hallucinations in different application scenarios.
- For the first time, we provide a hallucination-oriented comparison of three reasoning enhancement approaches: CoT [64], [16], Tool-Augmented Reasoning [42], [41], and Symbolic Reasoning [40], [39], examining their mechanisms and effectiveness in alleviating logic-based hallucinations.
- We systematically define and conceptualize the Agentic System, an emerging paradigm integrating RAG and reasoning, and position it as a unified framework and standard pathway for addressing composite hallucination problems.
- For each type of hallucination mitigation strategy, we present application examples from representative domains, demonstrating their practical value and generalizability.

The main content and structure of the paper are presented in Fig. 2. Through this structured analysis and comprehensive review, this work aims to provide a unified reference framework

and methodological guidance for developing LLMs capable of effectively suppressing hallucinations in real-world applications.

Organization of this Survey. The main content of this survey is organized as follows. Section II provides a review and comparison of related surveys. Section III presents the background and key concepts underpinning this survey. Sections IV and V examine the mechanisms and applications of RAG and reasoning enhancement in mitigating knowledge-based and logic-based hallucinations, respectively. Section VI further analyzes their integration and discusses the emerging agentic system paradigm. Section VII summarizes existing evaluation benchmarks by hallucination type. Finally, Section VIII concludes with challenges and future directions.

II. RELATED SURVEYS

To ensure comprehensive and representative coverage, this survey encompasses publications from major academic venues and publishers, including IEEE, ACM, Springer, Elsevier, and leading conference organizations such as AAAI, ACL, NeurIPS, and ICLR, as well as recent works available on arXiv and other reputable preprint platforms. The inclusion of arXiv papers is particularly important given the rapid pace of developments in emerging topics such as hallucination mitigation, RAG, reasoning, agentic systems, and related benchmarks. Most of these preprints consist of high-impact studies released within the past year, ensuring that this survey captures the most recent advances and evolving trends in the field. Literature selection combines keyword-based retrieval with thematic relevance assessment, emphasizing representative studies that address

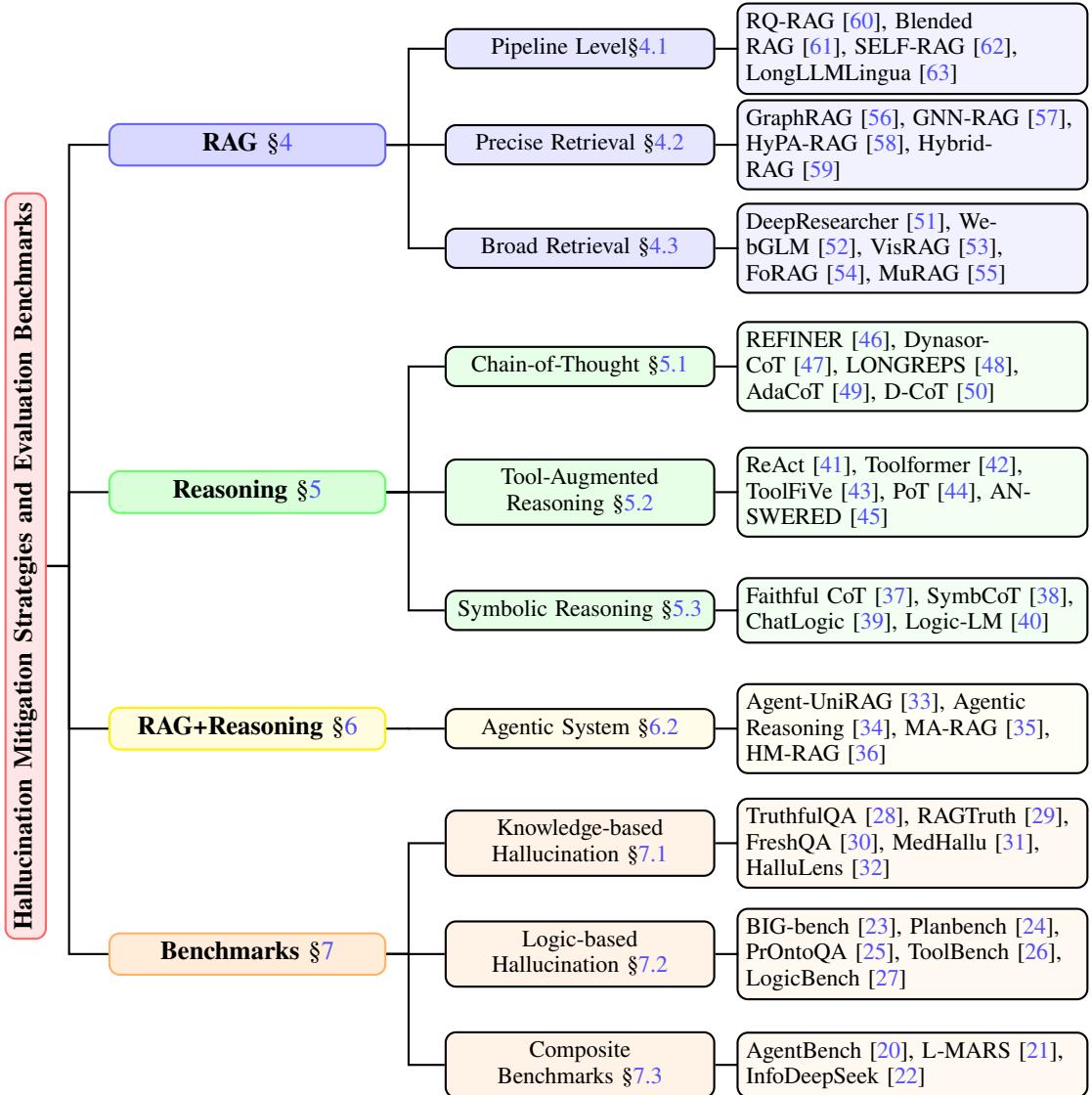


Fig. 2. Overview of hallucination mitigation and evaluation methods with representative models and benchmarks

hallucination control mechanisms, mitigation strategies, and real-world applications.

Within this scope, prior surveys related to this study can be broadly categorized into two main streams: hallucination-oriented surveys and technique-oriented surveys. The former focus on understanding and mitigating hallucinations in LLMs, while the latter explore retrieval- or reasoning-based frameworks that are conceptually related to hallucination control.

Surveys of Hallucination. A substantial body of prior work, including surveys by Zhang et al. [2], Huang et al. [1], and Ji et al. [3], has systematically summarized the causes, mitigation strategies, and evaluation methods of hallucinations in LLMs, providing valuable taxonomies for understanding this phenomenon. But they do not examine the impact of reasoning ability on hallucination mitigation and lack evaluation of the interactions among different hallucination mitigation methods. Bai et al. [65] and Sahoo et al. [66] further extended this discussion to multi-modal hallucination, analyzing how misinformation propagates across modalities. Lin et al. [67] conducted a

systematic survey on LLM-based agents, summarizing the types, causes, and detection mechanisms of hallucinations within LLM Agents. Most of these studies conceptualize hallucination as an undesirable generation error that must be suppressed, and their mitigation strategies often aim to reduce uncertainty in model outputs—sometimes at the cost of generality and creativity. In contrast, Jiang et al. [11] proposed an alternative view, regarding hallucination as a manifestation of model creativity rather than a pure error. Their discussions emphasized controllable and trustworthy generation, advocating for improved interpretability and human controllability rather than simple elimination.

Although the understanding of hallucination formation has steadily evolved, research on hallucination mitigation remains relatively underdeveloped. Existing surveys primarily focus on theoretical perspectives or isolated mitigation techniques and lack a systematic synthesis from a capability-enhancement viewpoint. To fill this gap, the present work comprehensively examines three representative capability-oriented approaches: RAG, reasoning, and Agentic Systems, and analyzes their mech-

anisms and improvement directions for hallucination mitigation from an application perspective. Unlike previous surveys that mainly emphasize model architecture or alignment training, this survey focuses on controllable hallucination reduction and reliable generation through external knowledge integration and reasoning enhancement. These two techniques together represent scalable and generalizable capability-enhancement paradigms that can substantially improve the reliability of LLMs in real-world scenarios.

Surveys of RAG, Reasoning and Agents. Beyond hallucination-focused reviews, several studies have examined RAG [15], reasoning-based frameworks [17], [16] and LLM based Autonomous Agents[68] that are closely related to hallucination mitigation. Zhang et al. [69] specifically discussed the hallucination problem in RAG systems, providing detailed analyses of retrieval quality, context integration, and factual grounding. But their discussion remains largely confined to the technical aspects of RAG itself, without extending to reasoning enhanced or hybrid mitigation approaches.

While these studies systematically reviewed model architectures, retrieval pipelines, and reasoning paradigms, they remain mostly theoretical and lack empirical or conceptual evaluation of their effectiveness in mitigating hallucination and improving generation reliability. In contrast, this survey adopts an application-oriented and integrative perspective, aiming to bridge these fragmented lines of research. Under a unified framework, we examine the complementary roles of RAG, reasoning, and emerging Agentic Systems, situating them within a broader context of hallucination control and reliability enhancement. This synthesis establishes a coherent mapping between methods, systems, and applications, providing a more targeted and practically valuable reference for future research on hallucination mitigation.

III. BACKGROUND

A. Large Language Models

Large Language models (LLMs) [70], [71] have emerged as one of the most prominent paradigms in deep learning. Initially known for their exceptional performance in Natural Language Processing (NLP) tasks, LLMs have since demonstrated a remarkable capacity to generalize beyond language, influencing domains such as code generation [72], reasoning [64], and even scientific discovery [73].

Technically, LLMs are typically built on the Transformer decoder architecture, generating text through autoregressive next-token prediction over large-scale corpora [74]. This prediction mechanism, grounded in statistical correlation, endows the model with powerful generative and generalization capabilities. However, it also introduces inherent randomness and uncertainty [72], which are widely regarded as major sources of hallucination [6].

Meanwhile, the emergent abilities [75] and internal reasoning mechanisms of LLMs remain insufficiently understood [76]. The opacity of their generation process makes hallucination particularly challenging to control and highlights the need for systematic research on hallucination mitigation.

B. Hallucination Definition and Mitigation Methods

Definition. Hallucination is not a new problem in NLP. It is first formally mentioned in abstractive document summarization [77], where the models are highly prone to hallucinate content that is unfaithful to the input document [3]. In the realm of LLMs, the hallucination problem typically refers to the phenomenon in which a model generates content that appears plausible, but is factually incorrect, logically inconsistent, or misaligned with the intent of the user [1], [2]. This phenomenon seriously affects the reliability and controllability of LLMs, especially in high-stakes domains such as medicine, law, and finance, where high accuracy, consistency, verifiability, and traceability are critical [5], [78].

With the rapid spread of LLMs, hallucinations have become a key obstacle for the real-world use and trustworthiness of LLMs [13], [4]. They undermine user trust in the outputs and constrain the potential of LLMs to provide expert knowledge and perform deep reasoning. In addition, as LLMs become larger, more widely used, and more open-ended, hallucinations are showing new patterns: they are more diverse, harder to detect, and can change over time [4], [79].

Mitigation. Research has explored how to detect and reduce hallucinations to better understand and manage this problem [79], [80]. Existing hallucination mitigation methods, apart from RAG and reasoning enhancement, include cleaning and inspecting pre-training data, improving pre-training [12] and fine-tuning [13], using reinforcement learning for better alignment, and applying post-hoc verification or rewriting [1], [2]. The purpose of these methods is to improve the quality of both training and generation in existing training approaches and models. Other methods aim to reduce the occurrence of hallucinations by optimizing the model architecture [81], [82] or the answer generation process [79], [83], [84], [14].

The hallucination mitigation techniques mentioned above have certain practical value, but they are not the focus of this survey. Instead, we primarily concentrate on RAG and reasoning-based approaches as well as their integration for the following reasons: (1) They mitigate hallucinations by enhancing the capabilities of LLMs, rather than suppressing them; (2) They preserve scalability and creativity, which are crucial for open-domain and exploratory tasks; (3) They demonstrate high deployability, making them adaptable to a wide range of real-world applications; (4) Their practical utility has already been validated in the market, and the combination of these two approaches holds great potential for mitigating all types of hallucinations, offering broad prospects for application.

Recent work shows that eliminating hallucinations completely is nearly impossible because they are an inherent feature of generative models [6]. Interestingly, the same mechanisms that cause hallucinations also drive LLMs' creativity and originality [11]. Fully removing hallucinations might also suppress this creative potential. Therefore, it is widely accepted that mitigation, not complete removal, is the realistic goal.

C. RAG and Reasoning in LLMs

Retrieval-Augmented Generation (RAG), in the context of LLMs, refers to retrieving information from external knowledge

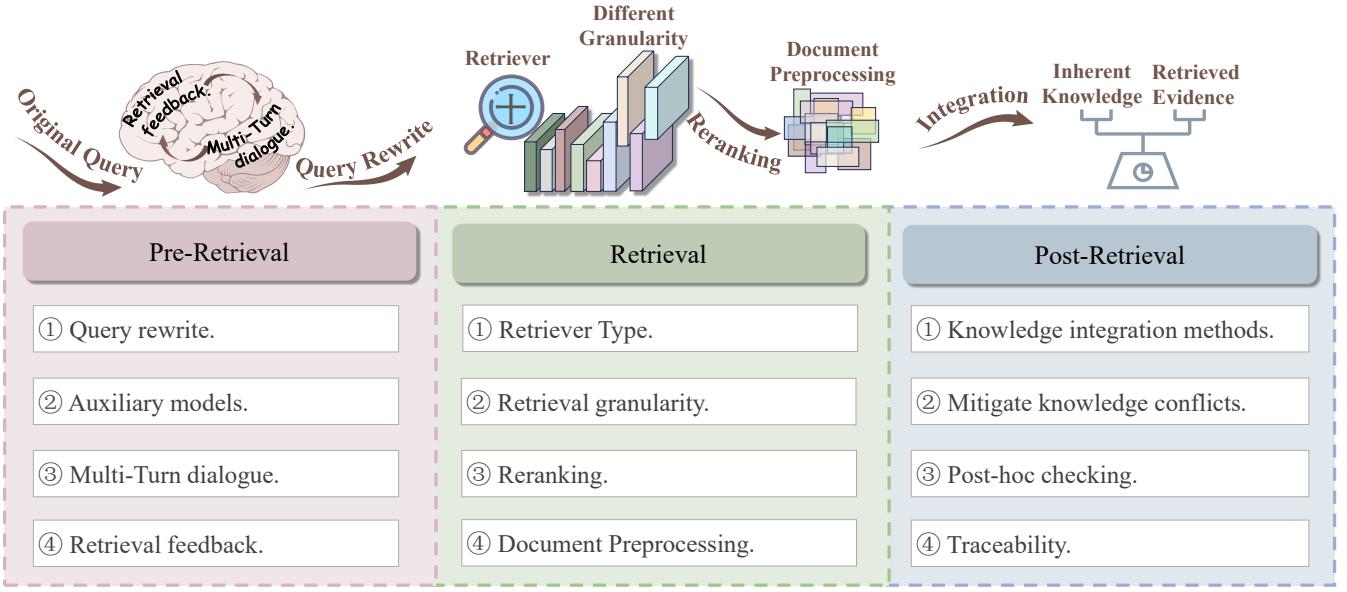


Fig. 3. Overview of the RAG pipeline

sources during the inference stage to support content generation [15]. It has been widely adopted because it can flexibly integrate the latest or domain-specific knowledge without the need for frequent and large-scale updates to model parameters. Initially, RAG was primarily regarded as a means to fill knowledge gaps left from the pre-training stage, but subsequent research has shown that it has broader value: correcting internal model errors and biases, enabling rapid and modular knowledge updates, improving answer traceability, and enhancing contextual consistency. These extended capabilities make RAG a powerful and versatile solution for mitigating hallucinations in knowledge-intensive tasks.

Reasoning refers to the ability of LLMs to dynamically interpret complex instructions, decompose them into sub-goals, construct coherent and rigorous logical chains, and follow structured steps to accomplish tasks [17]. This capability is not merely reflected in the length of the generated content, but more importantly in the model's capacity for structured thinking, logical planning, and adaptive decision-making in solving multi-stage problems. In recent research of LLMs, three representative forms of reasoning have emerged: Chain-of-Thought (CoT) [64], [16], which is a prompt-based method that guides models to generate intermediate reasoning steps for improved logical coherence; Tool-augmented reasoning [42], [41], which leverages external tools such as calculators, search engines, or APIs to enhance problem-solving accuracy; and Symbolic reasoning [40], [39], which transforms natural language into symbolic representations for verifiable, logic-based computation. These methods enhance the reasoning capability of LLMs from different perspectives and thus serve as key approaches for mitigating logic-based hallucinations.

IV. KNOWLEDGE-BASED HALLUCINATION AND RAG

Knowledge-based hallucinations arise from inaccuracies in a model's internal knowledge or insufficient external information. Mitigation therefore focuses on supplementing the model with accurate and relevant external knowledge. Among various approaches, RAG has emerged as the most effective and widely studied framework for enhancing factual consistency and reliability in knowledge-intensive applications [85], [86].

This section focuses on how RAG supplements external knowledge to mitigate knowledge-based hallucinations. We first outline the overall RAG pipeline, summarizing its core mechanisms and development trends. We then divide RAG applications into precise retrieval and broad retrieval, systematically analyzing key techniques from different application perspectives and enabling hallucination mitigation methods to better adapt to different application domains.

A. RAG Pipeline and Key Techniques

Organized along the RAG retrieval pipeline: including pre-retrieval, retrieval, and post-retrieval stages, the following analysis systematically examines key techniques, as illustrated in Fig. 3. We summarize recent technical advances in RAG pipeline that enhance the overall effectiveness of RAG and analyze how these techniques improve knowledge retrieval and utilization, thereby mitigating hallucinations in general tasks.

1) *Pre-retrieval*: The core task of the pre-retrieval stage is to understand the intent of the user's query. In RAG tasks, intent understanding refers to the ability of the LLMs and its retrieval component to work together beyond surface-level keyword matching, aiming to accurately capture the user's actual information need. This involves not only identifying what the

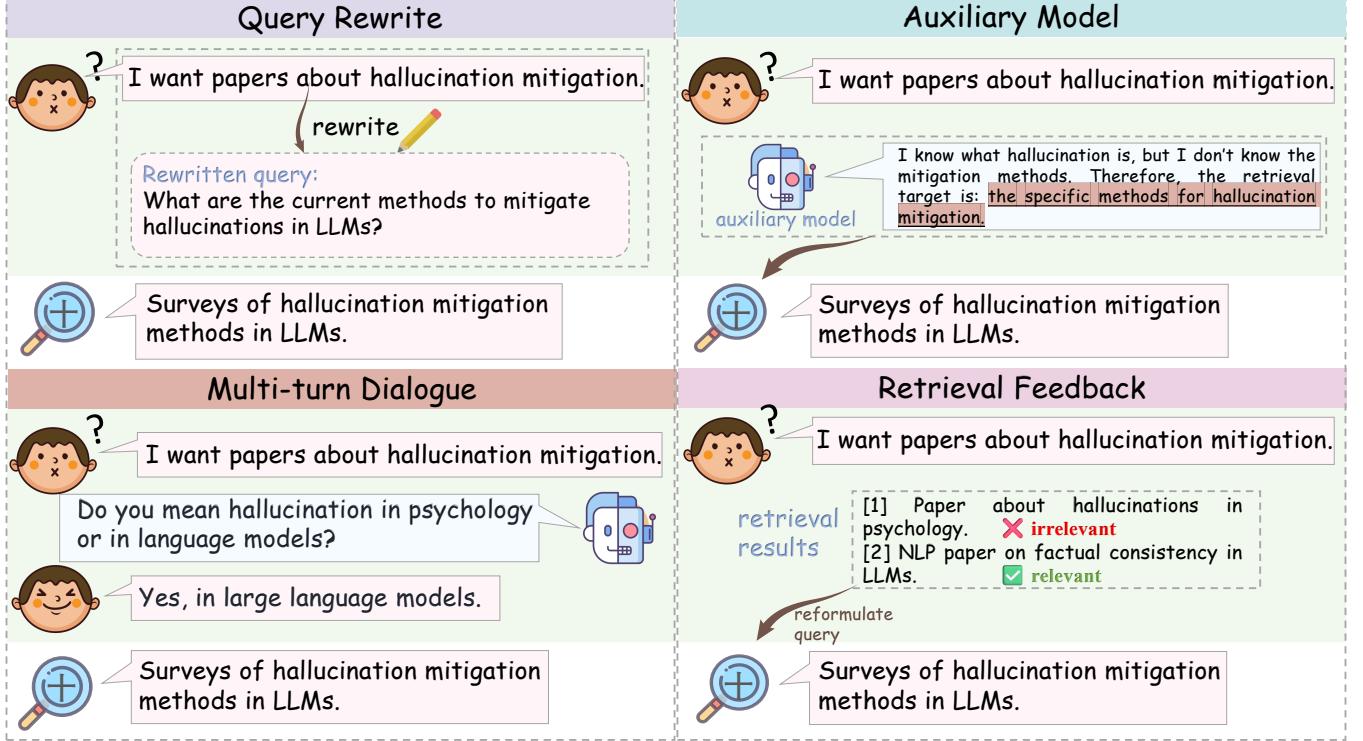


Fig. 4. Illustrative examples of four methods for enhancing intent understanding

user is asking, but also understanding what type of content is required, how the information should be structured, and where it should be sourced from. A strong understanding of user intent enables the retriever to formulate more targeted queries, thereby improving the relevance of the final output and mitigating hallucinations. Here we will discuss about four key methods (as shown in Fig. 4) to improve intent understanding.

a) Query rewrite: To enhance intent understanding, Ma et al. [87] proposed a query rewriting approach to bridge the gap between the input text and the knowledge required for retrieval. This rewriting process reformulates the original query into a more retrieval-effective form, improving the relevance and factual alignment of the retrieved results. Building on this, RQ-RAG [60] focuses on training the model to decompose and disambiguate complex queries, improving its ability to identify complex user intent. Watson et al. [88] pointed out that many hallucinations originate from the queries themselves, and query rewriting is therefore crucial for reducing hallucinations [89].

b) Auxiliary models: In addition to query rewriting, some methods leverage auxiliary lightweight models to assist in identifying appropriate retrieval targets. For example, Tan et al. [90] propose SlimPLM, which leverages a small model to generate preliminary answers first, and then uses these answers to identify missing knowledge that needs to be retrieved.

c) Multi-Turn dialogue: Accurate intent understanding may require incorporating contextual information from prior turns in multi-turn conversations[91]. Model like LARA [92] leverages the dialogue history to better understand user intent and reconstruct more effective queries, resulting in more accurate retrieval.

d) Retrieval feedback: In complex models with iterative retrieval capabilities, the system can revise and reconstruct the initial user intent based on feedback from retrieved results or generated content, thereby improving retrieval accuracy and the reliability of downstream reasoning. Representative models include RA-ISF [93] and KiRAG [94]. RA-ISF adopts an iterative process that incorporates feedback from each retrieval step into decision-making, and decomposes sub-questions for further retrieval when necessary. KiRAG performs step-by-step iterative retrieval over the input query: after each retrieval step, it formulates a new query based on the current reasoning chain to acquire the knowledge needed for the next step.

2) Retrieval: After accurately identifying user intent, the process moves to the retrieval stage. This stage focuses on the retriever's ability to locate relevant knowledge efficiently and precisely based on the query, ensuring that the retrieved information is accurate, comprehensive, and concise. It serves as a core metric in the design of retrievers within the RAG framework. We have identified and summarized several factors that affect the performance of document retrieval:

a) Retriever Type: Currently, commonly used retrievers can be roughly divided into three categories based on their retrieval principles: sparse retrievers, dense retrievers, and hybrid (sparse + dense) retrievers. Table I categorizes and lists retrievers in terms of their working mechanisms, representative models, and application scenarios. Sparse retrievers (e.g., BM25 [95], SPLADE [97]) rely on keyword matching and offer strong interpretability and fast response times. However, they struggle with handling semantic variations in queries. Dense retrievers (e.g. DPR [99], Contriever [100]) use encoders

TABLE I
MAINSTREAM RETRIEVER CATEGORIES (SPARSE VS DENSE VS HYBRID)

Category	Key Mechanism	Representative Models	Advantages / Limitations	Typical Use Cases
Traditional Sparse (lexical / bag-of-words)	Inverted index and term matching; no deep vector representations	BM25[95], TF-IDF[96]	Simple, efficient, strong on exact term matching / Weak on semantic similarity without shared terms	Document retrieval, first-stage recall in production, resource-limited settings
Neural Sparse Retrievers	Neural networks learn term importance or document expansion, but output remains sparse (indexable)	SPLADE[97], DeepImpact[98]	Combine lexical precision with learned semantics, efficient with inverted indexes; training and sparsification design are complex	Semantic-enhanced replacement of BM25, efficient first-stage retrieval
Dense Retrievers (bi-encoders)	Encode queries and documents into dense vectors, matched via approximate nearest neighbor search	DPR[99], Contriever[100], SBERT-based retrievers[101], ANCE[102]	Strong on semantic similarity and context understanding; limited by vector index size and memory, weaker on exact term matching	Open-domain QA, semantic similarity search, cross-lingual retrieval
Hybrid Retrievers	Combine sparse and dense signals, or perform token-level interactions at retrieval time	ColBERT[103] / ColBERTv2[104], simple hybrid (BM25 + dense reranker)	Balance between efficiency and effectiveness, hybrid improves robustness; more costly	Retrieval + reranking pipelines, high-precision QA retrieval

to map both queries and documents into a shared semantic space, enabling better capture of semantic relationships. Hybrid retrievers (e.g. ColBERTv2 [104]) integrate sparse and dense methods, combining the efficiency and interpretability of sparse retrievers with the semantic strength of dense retrievers. By leveraging both lexical and semantic signals through score fusion or reranking, hybrid approaches often outperform either method alone, especially in complex open-domain and LLM-related tasks.

Arivazhagan et al. [105] demonstrated that retrieval with hybrid retriever achieves superior performance in accurately mitigating hallucinations, outperforming both purely sparse and purely dense retrieval methods. Hybrid examples include Blended RAG [61], CG-RAG [106] and HyPA-RAG [58]. The strong performance of these models demonstrates that hybrid retrievers represent an important direction for future development in retrieval selection.

b) Retrieval granularity: Retrieval granularity refers to the smallest content unit into which a knowledge base or source is partitioned for retrieval during system construction. Recent studies have also explored different retrieval granularities to improve retrieval accuracy. These include document, chunk, section, sentence, token and entity. Coarse-grained retrieval like document-level retrieves broader context with fewer retrieval targets, making it faster and more suitable for real-world applications. But it often introduces irrelevant or noisy information, which can distract the model and degrade the quality of the responses generated. In contrast, fine-grained retrieval like token-level provides more precise and semantically focused content, improving factual accuracy. Yet, it is computationally expensive and may sacrifice important contextual information. Among these, the chunk level is the most commonly used granularity, balancing semantic completeness and retrieval efficiency.

As retrieval tasks become more complex and the formats and characteristics of retrieved documents diversify, a single retrieval granularity is no longer sufficient to meet the demands of retrieval. To balance these trade-offs, some methods aim to combine multiple levels of granularity. For example, MoG [107] and KET-RAG [108] organize the knowledge source into multi-granularity structures, enabling the model to flexibly select the appropriate granularity based on the query or task context, leading to more effective retrieval results. It is worth noting that KET-RAG adopts a multi-granularity retrieval approach to address the limitations of traditional Graph-RAG [56], whose coarse-grained retrieval often fails to capture fine-grained entity-level relationships within the text.

c) Reranking: As information retrieval tasks grow increasingly complex, document reranking has become a key technique for improving the utilization efficiency of retrieval results. The core goal of document re-ranking is to select a subset of candidate documents that are most informative while remaining within the context window limit. Given that LLMs tend to overlook information placed in the middle of the input [109], reranking also involves strategically placing the most relevant documents at the beginning or end of the input sequence to maximize their influence on generation.

Traditional approaches can be broadly categorized into three groups:

- **Feature-based scoring models:** Representative models like BM25 [110], TF-IDF [96]. These rely on explicit features such as term frequency and positional statistics and they offer strong interpretability and computational efficiency.
- **Learning-to-rank models:** Introduced by Liu et al. [111], these methods treat reranking as a supervised learning problem and are commonly divided into three subtypes: Pointwise methods treat reranking as a regression task,

assigning an individual relevance score to each document; Pairwise methods like RankNet [112] learn to rank by comparing the relative relevance between pairs of documents; Listwise methods like LambdaMART [113] directly optimize over the entire list of candidate documents.

- **Neural reranking Models:** Recent reranking approaches built on pre-trained language models, such as BERT [114], have significantly improved semantic matching by capturing deeper relationships between queries and documents, and have become a mainstream direction in information retrieval.

Despite their effectiveness, these methods often rely on fixed representations, shallow features, or static semantic matching, and struggle with higher-level understanding tasks such as complex queries, cross-document reasoning, and multi-turn interactions.

In recent years, LLM-based and reinforcement learning-based reranking methods have emerged, equipping rerankers with stronger contextual understanding and reasoning capabilities. LLM4Ranking [115] proposes a unified LLM-based reranking framework that supports both open and closed-source models such as GPT-4 [7], PaLM [116], and LLaMA [9]. It enhances generalization and reasoning by leveraging external LLM APIs for semantic modeling and ranking. Rank-R1 [117] introduces a reinforcement learning-based reranker with task-aware reasoning abilities, capable of dynamically adjusting ranking strategies based on the query, especially effective in complex or multi-hop question answering scenarios.

In RAG systems, reranking also serves as a filtering mechanism to eliminate task-irrelevant information. Advances in reranking techniques help prevent context contamination by unrelated content, thereby reducing the risk of hallucinations caused by distracting inputs [118].

d) Document Preprocessing: In addition to document reranking, modifying or compressing retrieved documents before generation has emerged as an important preprocessing step in RAG systems.

Some methods compress irrelevant information and retain only the subset of tokens most useful for the LLMs. Zhou et al. [119] propose TrustRAG, a robust framework that detects and masks "hallucination-inducing" expressions within retrieved segments. Jiang et al. [63] proposed LongLLMLingua, which significantly reduces the length of retrieved documents and increases the density of key information through a question-aware coarse-to-fine prompt compression and reordering strategy.

Some methods leverage key sentence extraction, which explicitly select answer-relevant spans. For example, DSLR [120] uses a sentence-level selector to extract query-relevant sentences, then reorganize them into a coherent paragraph according to their order in the original text.

These methods aim to improve the information density of retrieved content, reduce input length, and thereby control computational cost, reduce redundancy, and mitigate hallucination risk results from irrelevant information.

3) Post-retrieval: The core task of the post-retrieval stage is to effectively integrate the retrieved external knowledge with the model's internal knowledge, enabling the generator to provide accurate answers based on the combined information. The

generated content should remain semantically aligned with the user's intent and faithfully reflect the key information contained in the retrieved results.

a) Knowledge integration methods: To effectively incorporate externally retrieved knowledge into the generation process of LLMs, researchers have proposed three mainstream integration strategies [15]: Input-level integration, Intermediate-layer integration (also known as semi-parametric integration), and Output-level integration. These strategies differ in applicability, model coupling, computational cost, and their effectiveness in mitigating hallucinations.

- **Input-level integration** is currently the most widely adopted approach, where retrieved documents are directly concatenated with the user query and fed into the language model as a single prompt [121]. Its simplicity, compatibility with existing LLMs, and lack of architectural modifications make it highly practical.
- **Intermediate-layer integration** addresses these limitations by introducing fusion modules that allow the encoded retrieval results to interact with the model's internal representations, typically through mechanisms like cross-attention, adapters, or memory routing. This strategy alleviates input length constraints and enables more effective modeling of complex context [122], leading to improved consistency and informativeness in generated outputs.
- **Output-level integration** incorporates retrieved knowledge at the final stage of generation, serving as a reference signal to adjust the model's output distribution through techniques such as rerank tokens, output filtering, or guided decoding. While this approach offers strong modularity and is minimally invasive to the generator, its limited ability to deeply model semantic alignment makes it generally less effective in ensuring factual accuracy and reducing hallucinations compared to the other two strategies. Consequently, input-level integration has emerged as the more prevalent approach because of its simplicity, interpretability, and ease of adaptation.

Due to its minimal intervention and implementation ease, input-level integration is often used as a baseline in practice. However, with growing interest in deeper semantic alignment and reasoning capability, intermediate-layer integration is becoming increasingly prevalent in advanced works. Previous studies have shown that intermediate-layer integration outperforms input-level integration in mitigating hallucinations, particularly in terms of factual accuracy and contextual consistency [15].

b) Mitigate knowledge conflicts: To improve the model's utilization of retrieved knowledge, approaches such as prompt engineering and instruction tuning have been used to reinforce the intended use of retrieved content. For instance, prompts that include directives like "Please answer based only on the following documents" or "Refer to the information below to respond" are employed to explicitly guide the model toward retrieval-grounded reasoning.

But when conflicts arise between retrieved knowledge and the model's internal knowledge and there are no explicit instructions indicating which source the model should follow, some methods do not blindly trust either source. Instead, they evaluate both and select the more reliable one. RE-RAG [123] assigns a confidence

TABLE II
COMPARISON BETWEEN KNOWLEDGE GRAPH AND UNSTRUCTURED DOCUMENTS

Dimension	Knowledge Graph	Unstructured Documents
Mechanism	Represents entities and their relationships as nodes and edges, supporting structured, semantic retrieval and reasoning	Stores information in free-form natural language text, requiring NLP or embedding-based methods for retrieval
Retrieval Method	Graph-based retrieval Path query Neighbor expansion	Sparse retrieval Dense(semantic) retrieval Hybrid retrieval
Application Scenarios	Multi-hop question answering Recommender systems Intelligent reasoning	Search engine document retrieval Sources like news, academic papers, reports

score to each retrieved document, allowing users to decide whether to rely on RAG outputs or fall back on the model's internal knowledge when retrieval quality is low. C-RAG [124] guides LLMs to perform more critical reasoning when using retrieved results by generating contrastive explanations. These methods not only improve the system's adaptability in handling knowledge conflicts, but also enhance its robustness to noise.

c) Post-hoc checking: Post-hoc consistency checking has emerged as another line of research. This involves aligning the generated output with the retrieved evidence at the semantic level to evaluate whether the model genuinely grounded its response in the provided documents [62]. Such analysis can be used to assess answer faithfulness and information utilization, ultimately serving as a tool for identifying potential hallucinations.

d) Traceability: To improve the traceability of RAG-generated content, enhance the transparency of external knowledge usage, and facilitate the detection and correction of potential hallucinations, researchers have proposed a range of mechanisms aimed at making the link between output and source documents more explicit. For instance, Explainable AI [125] implement an interactive interpretability mechanism for RAG outputs and source knowledge, allowing users to view how retrieved passages are used, understand the basis for generation, and query the actual source segments associated with the output without requiring technical expertise.

B. Precise Retrieval

The development of precise retrieval is driven by the retrieval challenges large language models face in domain-specific question answering. Such tasks often rely on fixed, structured, and large-scale knowledge sources that contain long-tail information insufficiently covered during pretraining. To address this problem, researchers have proposed three representative approaches aimed at improving retrieval precision: Graph-Augmented RAG, Knowledge Graph-based RAG (KG-RAG) and Hybrid RAG.

1) Graph-Augmented RAG: Graph-Augmented RAG is designed to address the challenges of information redundancy, lack of global context, and missing inter-document relationships in large-scale unstructured corpora. Recent studies have explored organizing document collections into graph structures, where sentences, paragraphs, or topical units are extracted as nodes, and edges are established based on semantic similarity, contextual co-occurrence, or citation relationships.

For example, Edge et al. [56] proposed GraphRAG, which constructs document semantic segments into a graph and

performs hierarchical summarization through community detection, significantly improving RAG's performance on query-focused summarization (QFS) tasks over large corpora. Hu et al. [106] developed CG-RAG (Citation Graph RAG), which incorporates a citation graph to enhance retrieval by enabling LLMs to perform semantic aggregation, evidence tracing, and cross-paper reasoning based on citation relationships, thereby improving both accuracy and interpretability in scientific question answering.

2) Knowledge Graph: In application domains where knowledge is relatively stable and well-bounded, such as medicine, law, and public policy, knowledge sources can be structured and standardized through the extraction of key facts, definition of entity relationships, and construction of knowledge graphs (KG) (Table II) [126], [127]. Compared with unstructured documents, KGs contain explicit fields, entity relationships, and hierarchical structures that eliminate redundancy and provide a more organized representation of information. By applying knowledge-graph-based RAG (KG-RAG), LLMs can more accurately identify and extract key information, reducing hallucinations caused by semantic ambiguity and achieving superior performance in domain-specific applications. For example, Li et al. [128] applied KG-RAG to medical question answering on Alzheimer's disease. Kalra et al. [58] used legal-entity and statute relations for structured retrieval in legal and policy QA.

KGs also support deeper logical reasoning and are well suited to multi-hop questions and complex contexts. GNN-RAG [57] scores candidate nodes and extracts paths on KG subgraphs with graph neural networks, then verbalizes these paths into the RAG context to provide structured evidence, improving multi-hop QA. Path-based retrieval further enhances traceability and interpretability.

In addition, Sun et al. [129] proposed the ToG (Think-on-Graph) framework, a method that performs reasoning directly on KGs. It enables LLMs to actively explore reasoning paths within the KG and produce verifiable reasoning trajectories, similar to the behavior of intelligent agents. This approach further extends the potential of KGs in enhancing deep reasoning for LLMs.

3) Hybrid RAG: Hybrid RAG refers to a dual-channel retrieval mechanism that integrates KG retrieval and vector-based retrieval. Originally proposed in HybridRAG [59], this approach performs both retrieval types simultaneously and concatenates the retrieved text chunks with the corresponding KG subgraphs. It aims to combine the structural and interpretable strengths of KG retrieval with the semantic coverage

and representational richness of vector retrieval. Through this integration, the model maintains broad semantic understanding while improving factual consistency and traceability, leading to higher retrieval accuracy [130].

The core of Hybrid RAG lies in the design of its retrieval fusion mechanism. Compared with traditional single-stage fusion approaches, Ma et al. [130] proposed an iterative retrieval fusion framework (Think-on-Graph 2.0), which alternately performs graph-based and text-based retrieval to progressively expand knowledge coverage and semantic depth, thereby achieving more comprehensive integration and higher reasoning reliability. Xu et al. [131] further advanced this idea by deeply integrating KG and vector retrieval into a heterogeneous semantic graph composed of entity and document nodes, enabling unified retrieval within a single space and achieving more efficient and interpretable multi-level knowledge aggregation and generation.

C. Broad Retrieval

As LLMs are increasingly applied to knowledge-intensive and multi-domain tasks, their reliance on external information grows accordingly. Traditional precise retrieval, based on structured corpora, faces limitations in coverage and adaptability. In contrast, broad retrieval (Fig. 5) aims to access useful information from large-scale and heterogeneous knowledge sources—such as web pages, social media, and multi-modal content—to enhance knowledge grounding. Building an efficient and trustworthy broad retrieval system has therefore become essential for mitigating knowledge-based hallucinations.

The main challenge of broad retrieval lies in the diversity of information sources and the variation in content granularity. To address these issues, this section introduces representative advancements in broad retrieval from three key perspectives: Cross-Domain Generalization, Long-Context Comprehension and AI-generated Content Identification.

1) *Cross-Domain Generalization*: Cross-Domain Generalization requires systems to perform information extraction across multiple knowledge domains and contexts. In real-world applications, particularly in open-domain question answering, multi-turn dialogue generation, and public opinion analysis, the necessary knowledge is often fragmented and heterogeneous, distributed across diverse sources. For instance, answering a question about the evolution of healthcare policy may require simultaneously drawing from news articles, government announcements, academic papers, and medical images or videos. This demands that the retrieval module not only recognizes the domain intent of a query, but also performs semantic alignment and integrative reasoning across sources and modalities, all without relying on predefined domain boundaries. Failure to do so can lead to fragmented or disconnected responses.

a) *Web Search*: The internet, as a multi-source and real-time information carrier, has become an indispensable external knowledge source for mitigating knowledge-based hallucinations. WebGPT [132] and IALM (Internet-Augmented Language Models) [133] are early variants of RAG systems that incorporate real web search results as external knowledge into the generation process of LLMs. WebGLM [52] further improves WebGPT in accuracy and efficiency.

Recently deployed mainstream LLM applications, such as ChatGPT4.0 [7], Gemini [19] and Deepseek [8] have also incorporated live web search to supplement their responses with up-to-date information, despite their underlying web searching and integrating mechanisms being mostly proprietary.

Academic research has also produced open, reproducible frameworks. Schick et al. [42] proposed Toolformer, which can learn to decide when and how to call search engines during inference. WebWalker [134] adopts a multi-agent framework that enhances the retrieval system's ability to navigate deeply through web content, improving both the depth and quality of information gathering. DeepResearcher [51] employs reinforcement learning to enable end-to-end training, allowing the model to autonomously plan retrieval and verify information in open, dynamic, and noisy real-world web environments. This significantly enhances the performance of LLMs in real-world web-based question answering tasks. Considering that Web-based knowledge is often difficult to verify, and may contain inaccurate or misleading information, FoRAG (Factuality-optimized RAG) [54] employs a dual-granularity Reinforcement Learning from Human Feedback (RLHF) framework to optimize the factual accuracy of answer generation in web-enhanced long-form question answering (LFQA). Furthermore, HtmlRAG [135] improves traditional web RAG pipelines by introducing structure-aware preprocessing that directly models HTML content, rather than relying on plain-text extraction. This approach preserves the hierarchical structure and semantic relationships of web pages, enhancing the faithfulness, traceability, and overall robustness of generation in complex web environments.

These systems collectively demonstrate that integrating web search into RAG pipelines not only expands the model's accessible knowledge space but also offers a powerful means of enhancing factual consistency, particularly in fast-changing domains such as news, public health, and technology. However, they also introduce new challenges, including source reliability assessment, information overload, and retrieval latency, which remain active areas of research.

b) *Multi-Modal RAG*: Multi-modal RAG [136] extends the traditional RAG paradigm beyond text-only inputs by incorporating heterogeneous modalities such as visual, auditory, or structural information into both retrieval and generation processes. This approach enables models to ground their responses in richer evidence and perform reasoning across different information sources, bridging perception and language understanding.

MuRAG [55] is the first framework to systematically extend retrieval from pure text to a text-image multi-modal setting, demonstrating that multi-modal retrieval can significantly enhance the performance of RAG systems. VisRAG [53] also explores image-based retrieval techniques, proposing directly embedding images into documents. HtmlRAG [135] retrieves web information directly from HTML data containing webpage structures.

These methods targeting multi-modal knowledge sources expand the input scope of RAG and enrich the accessible information for the model. However, the complexity of modality alignment introduces significant challenges. Semantic gaps

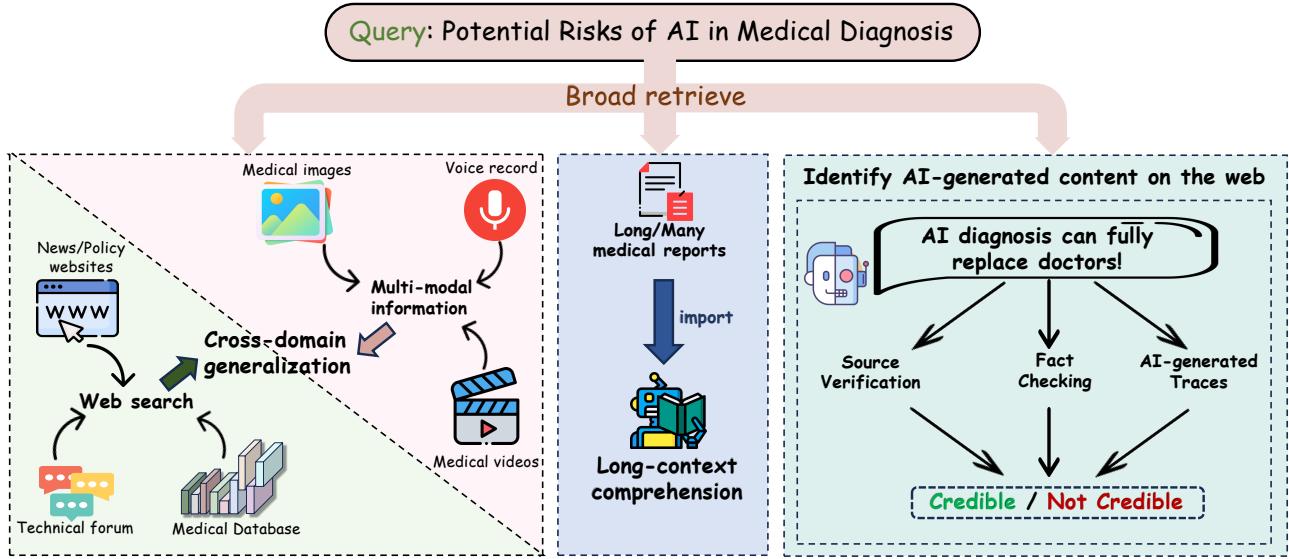


Fig. 5. Demonstration of a typical broad retrieval process

across modalities can lead to modality ambiguity and misleading information, which not only compromise retrieval quality but may also introduce new hallucinations, thereby offsetting the benefits of enhanced knowledge coverage [137].

2) *Long-Context Comprehension*: Another critical capability is the system's ability to handle long-text inputs. Prior research has shown that models often suffer from the “lost-in-the-middle” phenomenon [109] when processing overly long inputs—exhibiting significantly reduced attention to information located in the middle of the input sequence. This attention dilution phenomenon significantly weakens the model's ability to utilize retrieved content effectively, leading to reduced faithfulness in its responses. Many models have been optimized for long-text reading. Some models adopt sparse attention [138], [139] or linear attention [140] mechanisms to optimize attention computation, some improve reading performance by processing long texts recursively or in chunks [141], [142]; others fine-tune the model to enhance its performance on long-text inputs [140]. Overall, iterative chunk-based processing provides stronger dynamic generalization when handling long texts. This approach, unlike attention modification or fine-tuning, preserves the model's sensitivity to short texts. In addition, it can be integrated into retrieval post-processing procedures. For example, chunking can be performed during reranking, which reduces computational cost and shows promising potential for real-world applications.

Enhancing long-context reading ability enables models to make better use of retrieved information, mitigating hallucinations caused by insufficient knowledge grounding. It also improves the comprehension of complex, lengthy queries, further reducing the likelihood of hallucinations.

3) *AI-generated Content Identification*: With the growing prevalence of AI-Generated Content (AIGC) [143], such as automatically written articles, news reports and pictures, the risk of such material being retrieved and referenced by LLMs has also increased. Some studies have warned that AIGC

can contaminate the web's knowledge ecosystem, leading to a significant decline in model training quality [144]. RAG should likewise avoid using AI-generated materials for knowledge augmentation, as such content is often unreliable and may aggravate knowledge-based hallucinations. Therefore, retrieval modules should be equipped with the ability to identify and avoid AI-generated material during web searches. This section examines current techniques for detecting AI-generated content and evaluates their feasibility and effectiveness within RAG frameworks.

Watermarking [145] is the most effective method, which embeds invisible markers (such as token selection patterns or probability distribution features) into generated text during generation and verifies them during detection. However, this approach requires cooperation from the generation side for embedding, and the watermark can easily fail if the text is paraphrased or translated. Other methods, such as GLTR [146] and DetectGPT [147], distinguish AI-generated text by analyzing its statistical distribution features, including lexical diversity, syntactic complexity, repetition rate, and perplexity. In addition, training dedicated models for detection is also a practical approach. For example, Rakib Mollah et al. [148] employed a RoBERTa based model to detect fake news.

AI-generated content is increasingly pervasive across the internet. We believe that future web-based RAG systems should be equipped with the ability to identify such content, allowing them to filter out potentially hallucination-inducing information at the source.

D. Applications

This section examines how RAG mitigates hallucinations across major application domains, highlighting representative systems and their domain-specific optimization strategies.

1) *Healthcare*: The healthcare field is one of the most active yet safety-critical LLM application areas. Early systems such as Med-PaLM and Med-PaLM 2 [149], [150] established the

foundation for medical QA but suffered from limited internal knowledge. To address this, Shi et al. [151] proposed Medical Knowledge RAG, which injects external medical corpora to enhance factual consistency without fine-tuning. Subsequent work [152] integrated query rewriting and document reranking to improve retrieval quality. These advances demonstrate RAG’s ability to reduce knowledge-based hallucinations and improve medical reliability.

2) *Law*: In the legal domain, traceability is a primary concern [153]: models must cite explicit legal sources for every statement. Recent research [58] optimized RAG through hybrid retrieval and knowledge graph structures to improve interpretability and precision in legal QA and document analysis. Dedicated benchmarks such as LegalBench-RAG [153] and LexRAG [154] further evaluate not only correctness but also transparency in retrieval and reasoning, promoting verifiable outputs in high-stakes legal contexts.

3) *Finance*: RAG supports financial tasks such as decision-making, report generation, and market analysis by integrating up-to-date data sources: financial statements, news, and macroeconomic indicators, into the generation process. Zhao et al. [155] explored the challenges of applying RAG in financial tasks and emphasized that improving the retrieval process is crucial for enhancing financial question-answering performance. Wang et al. [156] also investigated the use of RAG in financial analysis tasks.

4) *Education*: The use of LLMs in education is growing rapidly. Students rely on LLMs for learning and assignments, while educators explore their integration, especially RAG. Dakshit et al. [157] found that RAG performs well in question answering and exam item generation, highlighting the role of retrieval in ensuring factual reliability. Li et al. [158] reviewed RAG applications across question answering, tutoring, content generation, assessment, and research. Swacha et al. [159] analyzed 47 studies on RAG-based chatbots, covering major educational scenarios. With increasing acceptance among students and adoption by educators, RAG has built a solid foundation for large-scale deployment in education.

5) *Other Domains*: Beyond these high-stakes areas, RAG has also been applied to scientific writing, public services, enterprise knowledge management, and multi-modal reasoning [15]. By retrieving and integrating domain-specific structured and unstructured knowledge, RAG improves factuality, coherence, and contextual control across diverse scenarios.

Moreover, the techniques discussed in this chapter can be further leveraged to optimize and customize RAG architectures for different scenarios, strengthening their hallucination suppression capabilities.

E. Discussion

Starting from the classical RAG framework, this section first provides an in-depth analysis of the three retrieval stages: pre-retrieval, retrieval, and post-retrieval, outlining their core components. We review recent advances along the general RAG pipeline and objectively evaluate these techniques in terms of their effectiveness in mitigating knowledge-based hallucinations. We then examine key representative methods

within two major application paradigms, precise retrieval and broad retrieval, aiming to offer a clear and structured overview of RAG technologies and their implications for future research on hallucination mitigation.

As related studies continue to progress, the RAG architecture has become increasingly complex, incorporating more sub-modules and decision stages. This complexity raises higher demands for coordination efficiency and fine-grained system design. Recent findings further indicate that retrieval failures or unnecessary retrieval operations may themselves become new sources of hallucination [1], [69], revealing limitations in the flexibility and decision-making ability of current RAG systems. Enhancing RAG adaptability and responsiveness to task-specific needs is therefore emerging as a central direction in ongoing research.

V. LOGIC-BASED HALLUCINATION AND REASONING

Logic-based hallucinations refer to instances where the generated content may be factually correct, but the reasoning process contains logical errors or inconsistencies. Such hallucinations are often difficult to detect on the surface, yet they can significantly undermine user trust and the interpretability of the model.

To address these challenges, researchers have increasingly focused on enhancing the reasoning capabilities of LLMs. Unlike factual hallucinations, which often stem from missing or incorrect knowledge, logical hallucinations arise from flaws in the internal reasoning process, even when the retrieved or memorized facts are accurate. Therefore, improving LLMs’ reasoning process is essential for mitigating this class of hallucinations. In this section, we examine how reasoning-based techniques, particularly CoT, tool-augmented reasoning and symbolic reasoning, have been developed to expose and correct faulty reasoning chains, ultimately contributing to more trustworthy and interpretable language model outputs (Fig. 6).

A. Chain-of-Thought

Chain-of-Thought (CoT) [16], [64] is an important test-time scaling method for enhancing the reasoning capabilities of LLMs [160]. Its core idea is to guide the model to generate reasoning steps step by step through structured prompting, thereby improving the logical consistency and factual reliability of its outputs. With a low deployment barrier and significant performance gains, CoT has been widely integrated into mainstream language model systems. Due to its ability to construct complete reasoning chains, CoT effectively mitigates logical hallucinations caused by issues such as skipped reasoning steps and causal confusion. In recent years, it has become a key direction in hallucination mitigation research.

This method was initially proposed in the form of few-shot prompting [64], then extended to zero-shot settings, with studies showing that even simple prompts like “Let’s think step by step” [161] or “Wait” [162] can significantly activate the model’s reasoning ability. By generating one or more intermediate reasoning chains, CoT reduces the risk of incorrect inferences caused by missing logical links, while also improving the interpretability of model outputs, allowing

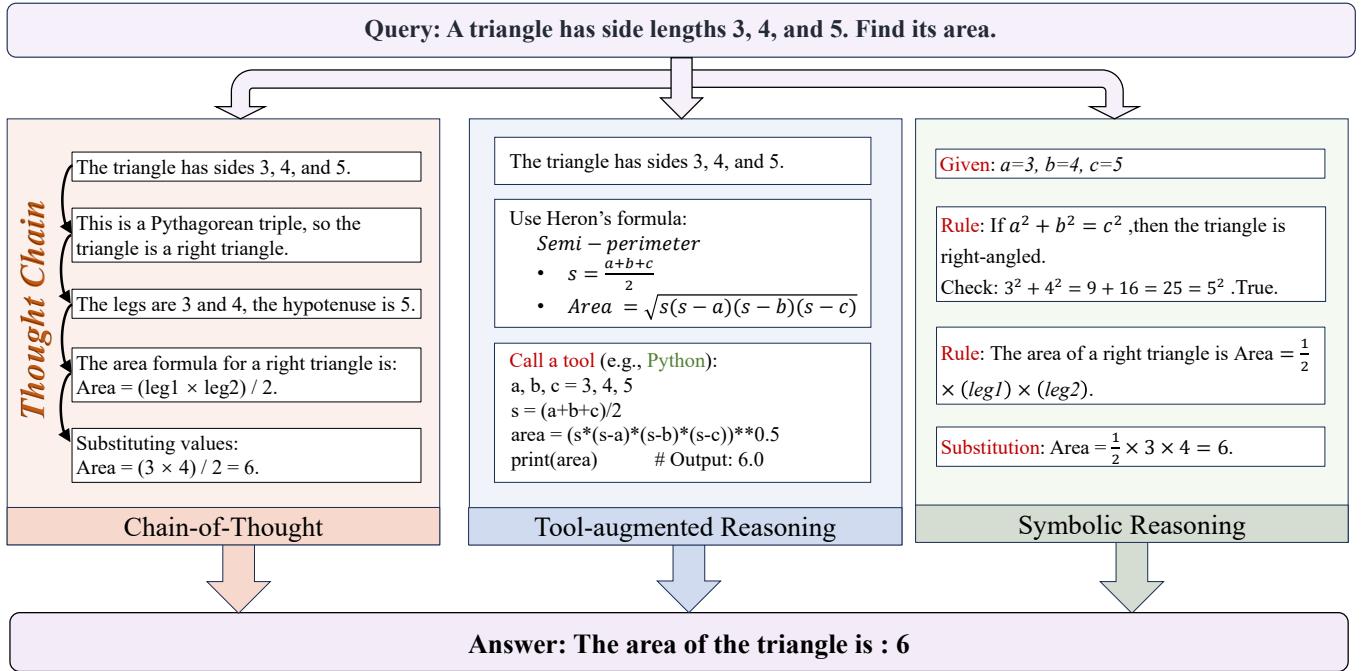


Fig. 6. Implementation process of three representative reasoning enhancement methods: Chain-of-Thought, Tool-augmented Reasoning and Symbolic Reasoning

users to examine the reasoning process step by step and identify potential hallucinations. Ling et al. [163] proposed a reasoning and verification mechanism called Natural Program, which decomposes CoT reasoning into multiple sub-processes and performs strict logical self-verification at each step. This approach of decomposing reasoning steps and performing verification enhances the faithfulness of the reasoning chain and reduces the incidence of logical hallucinations in generated responses. Xu et al. [38] propose a symbolic CoT framework that also ensures the reliability of the logical chain. It first converts natural language into symbolic representations and then applies symbolic logic rules to solve problems, leveraging a rule-based system to execute and verify each reasoning step sequentially.

In recent years, CoT has continued to evolve and expand to improve its reasoning robustness against hallucinations. For example, the Self-Consistency [164] strategy generates multiple CoT paths for the same question and selects the answer by majority voting, effectively reducing randomness and instability in single-path generation. Some approaches [38], [37] replace natural language reasoning with symbolic reasoning, enabling the reasoning chain to exhibit stronger logical consistency and verifiability. Zhu et al. [48] introduces a reasoning path supervision mechanism to guide LLMs in learning complete and coherent reasoning chains within long-context environments. By doing so, it enhances the model's reasoning ability and factual faithfulness in long-text tasks. The proposed method effectively mitigates hallucination issues arising from information loss or step-skipping during long-context processing.

Although CoT has demonstrated significant advantages across various reasoning tasks, existing studies have noted that its effectiveness is task-sensitive and does not guarantee performance improvement in all scenarios [165]. To address this limitation,

many approaches have been proposed to equip LLMs with the ability to autonomously decide whether to invoke CoT and to dynamically determine the appropriate length or style of the reasoning chain [50]. This helps prevent unnecessary long reasoning paths that may introduce logical hallucinations.

Overall, CoT is not merely a prompting strategy, but also represents a paradigm shift in language model generation—from “black-box responses” to “auditable reasoning chains”.

B. Tool-Augmented Reasoning

When handling tasks involving precise computation, fact verification, or structured logical reasoning, LLMs often produce hallucinations due to limitations in their instability in reasoning chains, especially in complex tasks that require explicit execution of intermediate steps, such as code generation or advanced mathematical problem-solving. To enhance the verifiability and accuracy of reasoning, the Tool-augmented Reasoning paradigm has been proposed in recent years [42], [41]. This approach guides models to invoke external tools such as calculators (math tools) [166], [167], code interpreters [167], [168], [169], or retrieval systems [134], [170] during the generation process, assisting in the execution of key reasoning steps and alleviating logical and factual errors caused by unguided inference.

Unlike traditional end-to-end text generation, this method transforms reasoning into a collaborative process of “*language generation + tool invocation*.” [42] The model is not only responsible for generating natural language output but also for planning when to call upon external modules. For example, in code generation tasks, it may invoke a program executor to verify the runnability of its output [167], while in mathematical problem solving, it can convert natural language into symbolic expressions for computation [38], [37]. Tool-augmented LLMs

have been applied to many domains beside code and math. Lu et al. [171] proposed TART, which handles table question answering by dynamically selecting table operation functions relevant to the current question and generating a corresponding reasoning plan. Enhancing reasoning by incorporating external knowledge bases is now a common approach, including web search engines [170], [172] and knowledge graphs [129], [130]. These approaches explicitly decompose the reasoning process into multiple intermediate steps, incorporating external information at each stage to dynamically enrich the context, thereby providing more comprehensive background support and factual grounding for subsequent reasoning.

As this field evolves, dynamic tool augmentation has emerged as a key research frontier. Unlike static tool chains, dynamic approaches such as ToolFive [43], ANSWERED [45] and SciAgent [167] endow LLMs with decision-making capabilities that allow them to determine whether, when, and which tools to invoke based on the specific task and contextual cues. More advanced systems [173], [174], [175] even support reflective evaluation—enabling the model to reassess its initial outputs, incorporate intermediate feedback, and adaptively reconfigure tool usage to arrive at improved answers. This highly flexible and interactive reasoning workflow greatly enhances model adaptability and robustness, offering a promising direction for mitigating hallucinations in complex, multi-stage tasks.

C. Symbolic Reasoning

Contemporary research is rapidly advancing the integration of symbolic reasoning [40], [39] with LLMs to enhance their performance in logical consistency, interpretability, and reasoning reliability. The core idea is to use the LLMs as a controller that transforms natural language questions into symbolic logic, while a logic programming engine is responsible for performing multi-step deductive reasoning or verifying the reasoning outcomes. The goal is to leverage the logical verifiability of symbolic reasoning together with the internal knowledge and natural language understanding capabilities of LLMs. This integration is regarded as a key approach to addressing the weaknesses of LLMs in symbolic computation and reasoning.

Specifically, Wang et al. proposed ChatLogic [39], which employs a large language model to convert natural language questions into symbolic logic. A logic programming engine then carries out multi-step deductive reasoning, and the results are translated back into natural language. In this approach, the reasoning process is executed by the logic programming engine, ensuring high reliability. However, it is limited in the types of reasoning tasks it can handle, resulting in a narrow coverage of reasoning scenarios. Similarly, Logic-LM [40] also utilizes a large language model to convert natural language questions into symbolic representations, and then invokes a symbolic reasoner to solve the problem.

Xu et al. proposed SymbCoT [38], which translates natural language into symbolic representations and then performs CoT reasoning using an LLM. Finally, a verifier is employed to examine the reasoning process. This approach directly leverages the reasoning capability of the LLMs, offering greater adaptability, while the subsequent verification step ensures the logical correctness of the reasoning process.

The integration of LLMs with symbolic reasoning embodies the concept of neuro-symbolic AI [176], [177] and represents an important direction for improving the logical reliability of LLMs, reflecting a broader trend toward building hybrid reasoning architectures that combine symbolic logic with neural language models. By assigning tasks with high logical complexity to symbolic engines and relying on LLMs for natural language understanding and generation, such systems achieve greater reliability, interpretability, and robustness in complex reasoning tasks. This research direction not only offers a promising pathway for mitigating logic-based hallucinations in large models but also reintroduces verifiable reasoning capabilities into modern AI systems, laying the groundwork for safer and more trustworthy applications in high-stakes domains such as law, science, and education.

D. Applications

1) *Code Generation*: Code generation is one of the most representative and practically valuable applications of LLMs. Popular tools such as Claude [178] and Cursor [179] have gained wide adoption among programmers.

Code generation tasks require the model to automatically produce executable code or perform code completion and modification based on natural language instructions. Early research primarily focused on improving programming ability through fine-tuning. Representative models include StarCoder/StarCoderBase [180] for direct code generation and completion, CodeGen [181] for conversational program synthesis, and the powerful Code Llama [182]. However, these models often exhibit programming logic hallucinations—such as logical inconsistencies, undefined functions, or semantic deviations—when handling multi-step reasoning, complex instructions, or less common programming languages, limiting their reliability in high-complexity tasks.

Enhancing reasoning capability is essential to address these issues. Yang et al. [168] applied CoT reasoning to improve the code generation performance of smaller models and achieved notable gains. Li et al. [183] proposed SCoTs (Structured CoTs), which first generate structured reasoning steps aligned with program logic and then produce code accordingly. Similarly, Zheng et al. [184] reformulated code generation as a hierarchical, coarse-to-fine process guided by abstract syntax tree structures, achieving higher-quality and more structurally consistent program synthesis than single-step generation.

These approaches share a common principle: decomposing complex generation into interpretable intermediate reasoning processes. By adopting step-by-step generation instead of one-shot generation, they achieve more stable and controllable code synthesis. This reasoning-driven paradigm significantly reduces programming logic hallucinations and enhances traceability, editability, and composability, opening new directions for multi-turn conversational code generation and complex system synthesis.

2) *Mathematical Reasoning*: Emergence of LLMs inspire ambitious expectations in mathematics. Some hope LLMs might resolve long-standing conjectures such as Goldbach's, yet in practice they may fail to decide whether 9.11 is larger than

9.9 [76]. Mathematical problem solving demands rigorous reasoning, which remains a major challenge for standard LLMs. Recent studies pursue three reasoning enhancement methods discussed earlier: CoT, tool-augmented reasoning, and symbolic reasoning.

Symbolic approaches attract wide interest for their logical rigor and verifiability. Gaur et al. [185] convert mathematical word problems into symbolic representations and design a self-prompting mechanism that guides the model to produce reasoning paths that are symbolically valid and numerically consistent. This symbolic chain introduces verifiable intermediate steps and improves consistency. Dhanraj et al. [186] further map LLM hidden states into a symbolic vector space, apply symbolic operations to strengthen rule-based reasoning, and fuse the symbolic results back into the hidden states. The method delivers large gains on mathematical and algebraic reasoning while preserving general performance.

Tool-augmented methods, especially code execution, have also proved effective. Chen et al. [44] propose PoT (Program of Thoughts Prompting), which generates code and uses an interpreter to solve mathematics problems. Das et al. [166] build a framework that integrates knowledge retrieval, program execution, and symbolic solvers, enabling dynamic tool composition for complex mathematical tasks. Ma et al. [167] introduce SciAgent, a tool-based reasoning system designed for scientific problems that include mathematics.

Together, these methods and their combinations outperform fine-tuning-only and prompt-only baselines on mathematical reasoning, reduce logic-based hallucinations, and improve traceability, editability, and composability of solutions.

E. Discussion

Enhancing the reasoning capabilities of LLMs plays a vital role in mitigating logical-based hallucinations. In recent years, reasoning mechanisms have been evolving toward deeper chains of thought, broader exploration of reasoning paths, and more reflective process control. This trend advances the depth and rigor of model thinking, enabling more structured reasoning when tackling complex tasks and thus effectively reducing hallucinations caused by reasoning leaps or causal misalignments. However, deeper reasoning chains do not always yield better outcomes [165]. Overthinking can cause the model to dwell on irrelevant paths [187], [188], [189], leading to new hallucinations or the generation of redundant content. As such, dynamically regulating the reasoning process requires balancing depth and efficiency, in order to avoid hallucinations stemming from excessive reasoning.

VI. COMPOSITE HALLUCINATION AND AGENTIC SYSTEMS

A. Motivation: Why Agentic Systems?

Although RAG and reasoning enhancement have each shown strong potential in mitigating hallucinations, relying on either approach alone remains insufficient. RAG effectively supplements factual knowledge but cannot guarantee logically consistent reasoning, while reasoning techniques strengthen logical chains but often lack the necessary external grounding. This complementarity motivates their integration: by combining

retrieval and reasoning, LLMs can in principle address both knowledge-based and logic-based hallucinations in a unified framework (Fig. 7).

In this survey, we define the agentic system paradigm as LLMs equipped with at least reasoning capabilities and retrieval modules. This section therefore reviews representative agentic frameworks and evaluates their potential to address both knowledge-based and logic-based hallucinations in practical applications.

B. Representative Agentic Systems

The core design principle of Agentic Systems is to integrate retrieval for factual grounding with structured reasoning for logical consistency. In practical applications, many models further define specific functional modules, such as self-reflection, error-control mechanisms, task decomposition, tool use, or memory [190], [34], [35]. Although the design of these modules varies across systems, they all follow the same guiding principle: mitigating composite hallucinations by combining external grounding obtained through retrieval with reasoning capabilities.

Several studies have explored different frameworks of Agentic Systems. Agentic Reasoning [34] is a recently proposed framework designed to enhance the performance of LLMs on complex tasks by combining reasoning capabilities with external tool use. It introduces a Mind-Map Agent to construct a structured reasoning graph, which tracks intermediate logical relations to maintain consistency in the reasoning chain, while a Web-Search Agent provides real-time retrieval support, ensuring reliable external evidence in multi-step reasoning. For hallucination mitigation, it aligns with the core definition of an Agentic System: retrieval reduces factual hallucinations, structured reasoning alleviates logical hallucinations, and explicit intermediate records improve traceability. Agentic Reasoning represents a “basic” form of Agentic System and provides a foundation for future extensions that integrates reflection and verification mechanisms to achieve stronger hallucination mitigation.

Nguyen et al. introduced the MA-RAG (Multi-Agent RAG) [35], which introduces multiple modular agents such as a planner, step definer, evidence extractor, and question-answering agent. It integrates task decomposition with evidence retrieval in a dynamic process of planning, retrieval, and synthesis. There are many similar models that introduce multiple agents or components collaborating to complete specific tasks [191], [190], [33]. Qian et al. [192] pointed out that this multi-model, task-specific mechanism helps isolate hallucinations and prevents their accumulation and propagation during the reasoning process.

HM-RAG (Hierarchical Multi-Agent Multi-modal RAG) [36] further extends this idea to multi-modal scenarios. It proposes a three-level multi-agent framework: the task decomposition layer refines the input problem, then the modality-aware retrieval layer selects suitable evidence sources for different modalities such as text and images, and the decision fusion layer integrates cross-modal evidence to generate answers. This design not only enables the handling of cross-modal tasks but also reduces factual hallucinations caused by modality inconsistencies. This

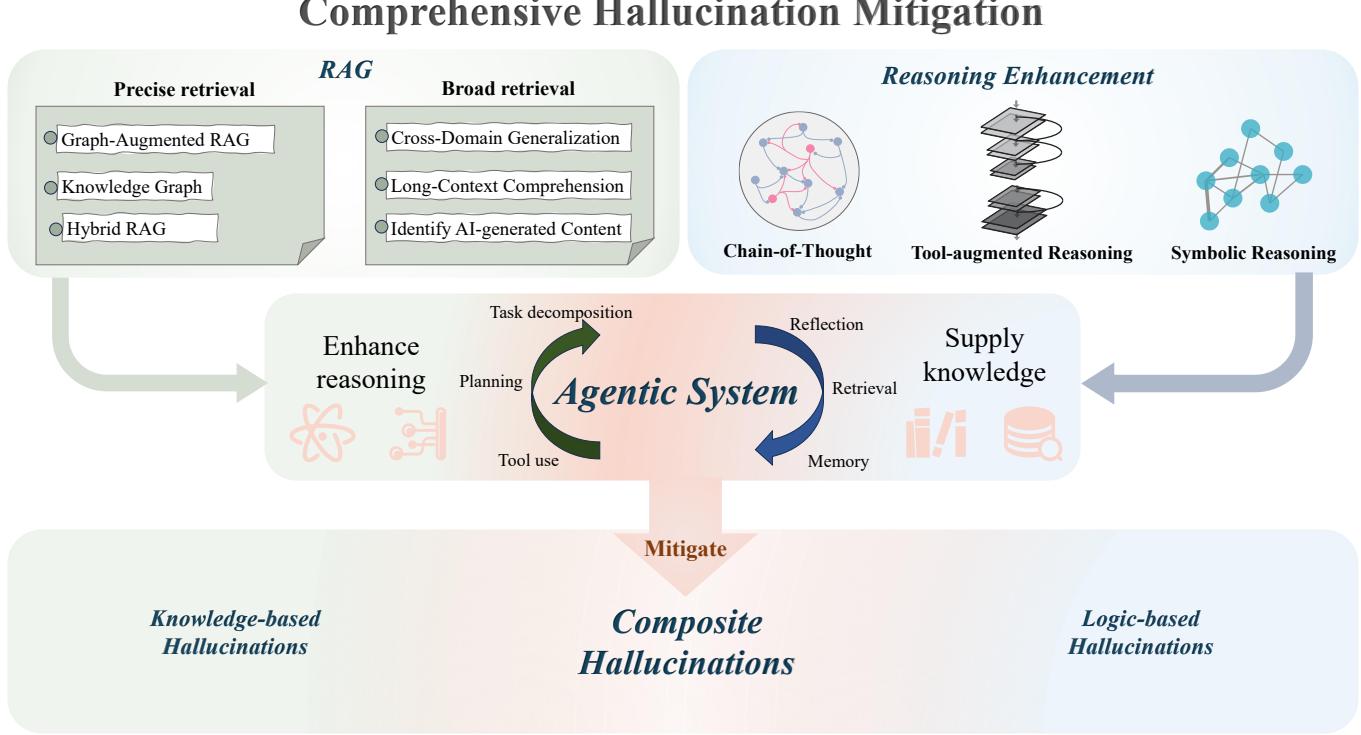


Fig. 7. Agentic Framework Integrating RAG and Reasoning Enhancement for Comprehensive Hallucination Mitigation

work provides valuable experience for future research on mitigating multi-modal composite hallucinations.

C. Applications

In complex application scenarios, the multi-dimensional nature of task requirements often triggers composite hallucinations, which has become the main driving force behind the development of agentic systems. Existing survey by Wang et al. [68] has systematically summarized the current landscape of Agent LLM applications. In highly specialized and dynamic environments, leveraging agentic systems to mitigate composite hallucinations has become a common and almost indispensable strategy.

However, current agentic systems remain in their early exploratory stage. Their theoretical definitions, architectural frameworks, and evaluation standards have yet to reach consensus. From the perspective of comprehensive hallucination mitigation, the following section introduces several representative retrieval–reasoning integrated applications, illustrating how this paradigm addresses complex hallucination challenges in specific domains and demonstrating its design principles and potential value.

Software Development: In software development tasks, the code context often spans multiple components and heavily depends on knowledge of API usage, framework conventions, and design patterns. The retrieval process enables the model to locate relevant implementations, examples, or reference materials within the codebase, while reasoning capabilities support task decomposition, step planning, and post-generation self-reflection.

SWE-agent [193] exemplifies this paradigm. Through a carefully designed Agent–Computer Interface, it performs structured retrieval over the codebase and engages in feedback-driven iterative reasoning, allowing the language model to efficiently locate, modify, and verify code logic in real software engineering environments. This deep integration of retrieval and reasoning demonstrates how the agentic system paradigm effectively mitigates composite hallucinations and achieves substantial performance gains in complex coding tasks.

Scientific Research: Scientific research is one of the most direct application domains requiring both retrieval and reasoning. Retrieval facilitates the discovery and integration of relevant scientific knowledge, while reasoning supports hypothesis formation, experimental design, and result validation.

Recently, Yamada et al. [194] introduced the AI Scientist-v2, a multi-stage retrieval and reasoning framework based on Agentic Tree Search. The system autonomously explores vast hypothesis and experiment spaces, retrieving the most promising research directions and iteratively generating, executing, and evaluating experiments through multi-branch reasoning and verification cycles. This systematic paradigm demonstrates the strong potential of agentic systems in scientific automation and knowledge innovation.

D. Discussion

The agentic system can be regarded as a system-level integration of RAG and reasoning. It not only possesses capabilities for knowledge updating and logical inference but can also autonomously evaluate the correctness of its outputs after generation.

Nevertheless, several limitations remain. A key challenge is that the integration of retrieval and reasoning is not yet standardized: different systems adopt different ways of combining external evidence with reasoning chains, making it difficult to establish unified and quantitative evaluation criteria. This lack of consistency not only complicates benchmarking but also limits our ability to compare systems fairly across tasks and domains. Moreover, the multi-agent nature of many frameworks introduces the risk of error propagation which is a failure in planning, retrieval, or reasoning can cascade through subsequent steps. At the same time, increasing system complexity raises computational overhead and makes deployment in real-world high-stakes applications more challenging.

Future agentic system should place greater emphasis on reflection and error control to prevent the accumulation and amplification of errors. They also need to focus on designing lightweight hybrid frameworks, since greater system complexity often increases the likelihood of errors, and achieving a balance among scalability, creativity, and factual accuracy is essential.

VII. BENCHMARKS

This section discusses representative benchmarks (as shown in Table III) for evaluating hallucinations in LLMs. Hallucination evaluation strongly influences the design and construction of LLM fine-tuning datasets, providing clear guidance for the fine-tuning process and, to some extent, shaping the model’s development trajectory. Systematic hallucination assessment allows researchers not only to quantify model performance in knowledge- and reasoning-based generation tasks but also to identify weaknesses in specific scenarios.

Following the taxonomy of hallucination types and mitigation strategies presented in Fig. 2, we divide existing benchmarks into three categories. The first category targets knowledge-based hallucinations, focusing on factual accuracy and alignment with external knowledge. The second category addresses logic-based hallucinations, emphasizing reasoning validity and consistency in inferential processes. The third category evaluates composite hallucinations in agentic systems, capturing the interplay of knowledge and reasoning errors in complex multi-step or tool-augmented workflows. Most benchmarks in this survey focus on text generation and unimodal question answering, while multimodal hallucination benchmarks are excluded due to their limited maturity and lack of standardized evaluation methods.

By systematically reviewing these benchmarks, this section aims to provide a structured reference for hallucination detection and mitigation, offering practical guidance for developing reliable and trustworthy LLM systems.

A. Knowledge-based Benchmarks

Knowledge hallucination benchmarks mainly include existing datasets that are designed to detect knowledge-based hallucinations. These benchmarks focus on identifying factual errors, omissions, and misinterpretations of established knowledge, thereby reflecting the adequacy of a model’s knowledge base and retrieval mechanisms. Since knowledge-based hallucinations are the most common type and directly affect the factual reliability of generated content, such benchmarks provide an essential

means for evaluating and comparing models in terms of factual accuracy.

Based on the evaluation focus, datasets for this type can be divided into two categories. The first category, such as TruthfulQA [28], FreshQA [30], focus on the model’s mastery of intrinsic knowledge acquired during pretraining and tests whether the model can provide accurate answers based on knowledge it is expected to have. The second category, such as MedHallu [31], targets knowledge that the model lacks and requires it to retrieve relevant information from external datasets, evaluating the model’s RAG ability. In recent years, some datasets have begun to introduce more fine-grained categorizations of hallucinations and propose more refined evaluation metrics to address the shortcomings of existing methods, such as vague definitions, information leakage, anti-cheating issues, and paradigm confusion. For example, HalluLens [32] distinguishes hallucinations into intrinsic and extrinsic categories. The former refers to errors or biases arising from the model’s internal reasoning based on its inherent knowledge, while the latter emphasizes errors caused by missing knowledge or improper use of external information. By adopting this categorization, HalluLens establishes a more systematic and comprehensive benchmark for hallucination evaluation. It not only provides clearer insights into the causes of hallucinations but also offers more explicit directions for future improvements. Such efforts reflect the trend of moving from coarse-grained to fine-grained hallucination assessment and play an important role in advancing the reliability evaluation of models across diverse tasks and application scenarios.

B. Logic-based Benchmarks

Logical hallucination benchmarks mainly include datasets designed to detect reasoning-related errors. They evaluate inconsistencies, logical flaws, and incorrect conclusions that arise during reasoning, judgment, or multi-step thinking. Such hallucinations are particularly common in multi-hop reasoning and complex logic tasks, posing significant challenges to the coherence and reliability of model reasoning. These benchmarks therefore provide a critical means of assessing the extent of logical hallucinations and comparing models’ reasoning capabilities.

Some benchmarks focus primarily on verifying reasoning outcomes. For example, the logical reasoning tasks in BIG-bench [23], such as syllogisms, entailment judgment, and arithmetic reasoning, usually require the model to provide only the final answer, which is then compared with the gold standard. These tasks mainly evaluate whether the model’s conclusion is correct, without enforcing the explicit demonstration of intermediate reasoning chains. As a result, they are more suitable for revealing logical hallucinations at the level of “outcome correctness.”

In contrast, another category of benchmarks emphasizes strict verification of the reasoning process. These datasets not only require the model to produce a conclusion but also to construct intermediate reasoning steps or proof trees, which are then validated step by step against the premises. For example, PrOntoQA [25] and ProofWriter [195] focus on

TABLE III
AN OVERVIEW OF REPRESENTATIVE HALLUCINATION BENCHMARKS

Benchmark	Hallucination Type	Data Size	Task	Evaluated Capability	Metrics
TruthfulQA [28]	Knowledge-based Hallucination	817	General Question Answering	Intrinsic Knowledge	Accuracy, Human Evaluation
MedHallu [31]	Knowledge-based Hallucination	10,000	Medical Question Answering	RAG	F1 Score
RAGTruth [29]	Knowledge-based Hallucination	18000	QA, Data-to-Text Summarization	RAG	Human Evaluation
BIG-bench [23]	Logic-based Hallucination	200	Logic Reasoning	Reasoning Results	Accuracy, F1 Score
PrOntoQA [25]	Logic-based Hallucination	40,000	Logic Reasoning	CoT	Accuracy
ToolBench [26]	Logic-based Hallucination	16,464	API invocation	Tool-Augmented Reasoning	Accuracy
LogicBench [27]	Logic-based Hallucination	N/A	Rule-based Reasoning	Symbolic Reasoning	Accuracy
AgentBench [20]	Composite Hallucination	17,000	General Interactive Tasks	Agentic System	Accuracy
L-MARS [21]	Composite Hallucination	200	Legal Multi-Turn Question Answering	Agentic System	Accuracy
InfoDeepSeek [22]	Composite Hallucination	245	Web Environment Retrieval	Agentic System	F1 Score, Human Evaluation

explicit verification of CoT-style reasoning; ToolBench [26] and API-Bank [196] evaluate tool-augmented reasoning; and LogicBench [27] specifically targets the correctness of symbolic reasoning processes. Such benchmarks are particularly effective for detecting “process hallucinations,” where the reasoning chain introduces illogical steps even if the final conclusion happens to be correct.

In addition, some benchmarks, such as ReClor [197] and LogiQA [198], draw inspiration from LSAT/GMAT logical reasoning exams, mainly evaluating the model’s ability to assess argument validity and identify distractor options. They highlight whether the model can distinguish valid reasoning from fallacious arguments, making them especially suitable for assessing “argument-level hallucinations,” which arise from flaws in argumentation rather than factual or procedural errors.

C. Composite Benchmarks

As LLMs continue to grow in scale, improve in capability, and face increasingly complex tasks, hallucinations have become more diverse and composite. With the emergence of Agentic system LLMs, which autonomously execute multiple steps or utilize various tools in complex task workflows, comprehensive evaluation methods for hallucinations across the entire task process are still lacking.

To address this issue, composite hallucination detection methods have emerged. AgentBench [20] is one of the earliest datasets designed specifically for Agent models. Its design covers eight representative environments, each corresponding to an interactive task space. These environments span operating systems, databases, web operations, games, knowledge reasoning, and puzzle solving, forming a broad and diverse framework that

reflects the typical contexts Agents may encounter in real-world applications. In addition to such general-purpose evaluations, researchers have developed more domain-specific datasets. For example, L-MARS [21] focuses on legal problem analysis, emphasizing the model’s performance in legal knowledge and logical reasoning. InfoDeepSeek [22] targets web-based deep search and information retrieval, testing the model’s ability to integrate and apply knowledge in open-domain and complex environments. These methods allow simultaneous evaluation of both knowledge and logical hallucinations. By analyzing task completion performance, they indirectly quantify the occurrence of different hallucinations across various stages and scenarios and provide targeted guidance for model optimization. With the increasing deployment of LLM Agents, safety concerns have gained growing attention. Some researchers have approached the problem from the perspective of risk control and security assurance. For instance, datasets such as R-Judge [199] are designed to evaluate errors and potential risks caused by hallucinations during interactive processes. Such evaluations expand the dimensions of hallucination detection and highlight the negative consequences hallucinations may have in practical applications.

D. Discussion

Existing hallucination evaluation datasets remain limited. Few benchmarks assess agentic LLMs, and even fewer jointly measure knowledge- and logic-based hallucinations, making it difficult to judge system reliability on complex tasks and to deploy or tune models safely in high-risk settings.

Future benchmarks should support end-to-end, process-level evaluation of agentic workflows, from input, retrieval, and

intermediate reasoning to final output; jointly assess knowledge-based and logic-based hallucinations (e.g., in RAG, multi-hop reasoning and complex logic); and provide generality and scalability across tasks, domains, and modalities. Such resources would give clearer signals for model improvement and help drive steady gains in accuracy, consistency, and safety.

VIII. CHALLENGES AND FUTURE DIRECTIONS

A. Limitations and Challenges of RAG and Reasoning Mechanisms

RAG has shown notable effectiveness in mitigating knowledge-based hallucinations, yet its performance is highly dependent on retrieval quality. Retrieval errors, unreliable information sources, and misinterpretation of query intent can themselves become new triggers of hallucination [69]. As discussed in Section IV, we systematically reviewed the development of RAG and identified key risks and optimization directions across its retrieval pipeline. These include: (1) designing adaptive retrieval strategies, particularly accurate modeling of retrieval intent and handling retrieval failures; (2) assessing the reliability of information sources, quantifying source credibility and evidential strength before integration; and (3) addressing the challenges of long-text and multi-source fusion, which require noise reduction and factual consistency across domains and modalities. Although recent research has proposed corresponding mitigation strategies [69], these challenges remain major bottlenecks in RAG design and in addressing knowledge-based hallucinations. Future work should prioritize robust end-to-end retrieval pipelines, tight alignment between retrieval and generation, and fine-grained filtering and verification before integration. Without these advances, RAG risks falling into a paradox of “mitigating hallucinations with hallucinations,” potentially exacerbating complexity.

Reasoning enhancement has proven effective in reducing logic-based hallucinations. However, while most current improvements in reasoning ability primarily come from the support of CoT techniques, CoT is purely language-based reasoning and lacks verifiable logical grounding [200], [76], which constrains its generalization and controllability. CoT is also prone to face the challenge of “overthinking” [188], which calls for dynamic adjustment of reasoning depth to balance conciseness and thoroughness [187]. Integrating CoT with symbolic reasoning or tool-augmented reasoning presents a promising direction [38], [173]. However, symbolic and tool-based reasoning pipelines still face challenges in cross-domain generalization, including fragile interfaces, dependence on external environments, and high task coupling. These limitations highlight the need for generalized frameworks for tool integration, enabling adaptive invocation of different external modules depending on task uncertainty and structural requirements.

Despite the significant potential of Agentic Systems in mitigating composite hallucinations, their coordination introduces unique challenges. Most existing systems still rely on ad hoc pipelines to connect retrieval and reasoning, lacking a unified design framework. As a result, the two components are often executed sequentially rather than synergistically. Future development should promote co-evolution of retrieval and reasoning

at the architectural level, enabling dynamic interaction rather than linear execution. In addition, self-reflective agent systems can be leveraged to integrate planning, retrieval, reasoning, and verification into a closed loop, thereby improving overall robustness and adaptability. Furthermore, the incorporation of explainability mechanisms is essential for enhancing user trust. By tracing the role of external knowledge in the reasoning process, such mechanisms can improve transparency and accountability, ultimately strengthening the reliability of agentic LLM systems.

B. Efficiency, Creativity Trade-offs, and Multi-modal Hallucinations

In practical applications, the introduction of RAG and Reasoning enhancement inevitably increases system complexity and computational overhead [201], [189]. Multi-stage pipelines of retrieval and reasoning often result in considerable latency, which constrains scalability in interactive scenarios. The increasing number of task modules and interactions among multiple agents within Agentic Systems also leads to a significant rise in computational overhead.

At the same time, there exists an inherent tension between suppressing hallucinations and preserving creativity [11]: excessive suppression may weaken model performance in open-domain and creative tasks. Even mitigation techniques such as RAG and reasoning, which do not explicitly constrain creativity, cannot fully avoid this issue.

Moreover, with the development of multi-modal large models, hallucinations are no longer confined to text generation but extend to cross-modal tasks, such as errors in image understanding or semantic deviations in video analysis [66]. This expansion further increases the difficulty of hallucination detection and mitigation. The applicability and scalability of RAG and reasoning in addressing multi-modal hallucinations remain underexplored.

Future research should focus on designing lightweight retrieval and reasoning modules, developing mechanisms capable of dynamically balancing accuracy and efficiency, and creating adaptive hallucination control strategies that enable models to flexibly adjust factuality and creativity based on application needs. In addition, establishing cross-modal consistency-checking methods is essential to ensure reliability and coherence of knowledge in multi-source information integration.

IX. CONCLUSION

This survey adopts an application-oriented perspective of capability enhancement to systematically review hallucination mitigation in LLMs. It establishes a taxonomy distinguishing knowledge-based and logic-based hallucinations, and analyzes how RAG, reasoning augmentation, and their integration in Agentic Systems collectively enhance factuality, logical consistency, and overall reliability.

While significant progress has been made, key challenges remain in achieving reliability, efficiency, and cross-domain generalization. The lack of standardized interfaces between retrieval and reasoning components hinders consistent evaluation, and

complex multi-stage or multi-agent architectures can amplify early-stage errors and computational costs.

Future research should pursue a systematic and layered hallucination mitigation framework that integrates RAG and reasoning enhancement, supported by multi-dimensional detection, verification, and correction mechanisms. Such an approach will be essential for building LLMs that are not only less hallucinatory but also aligned with real-world application demands.

Ultimately, advancing toward reliable, interpretable, and scalable LLMs requires aligning mitigation strategies with capability enhancement, paving the way for the next generation of Agentic, trustworthy, and reasoning-capable language models.

ACKNOWLEDGMENT

This work has emanated from research conducted with the financial support of Taighde Éireann — Research Ireland under Grant number *SFI/12/RC/2289 – P2* and co-funded by the European Regional Development Fund in collaboration with the Research Ireland Insight Centre for Data Analytics at Dublin City University.

REFERENCES

- [1] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, “A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions,” *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–55, 2025.
- [2] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen *et al.*, “Siren’s song in the ai ocean: A survey on hallucination in large language models,” *Computational Linguistics*, pp. 1–46, 2025.
- [3] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, “Survey of hallucination in natural language generation,” *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.
- [4] A. Saha, B. Gupta, A. Chatterjee, and K. Banerjee, “You believe your llm is not delusional? think again! a study of llm hallucination on foundation models under perturbation,” *Discover* vol. 3, 2025.
- [5] Y. Kim, H. Jeong, S. Chen, S. S. Li, M. Lu, K. Alhamoud, J. Mun, C. Grau, M. Jung, R. Gameiro, L. Fan, E. Park, T. Lin, J. Yoon, W. Yoon, M. Sap, Y. Tsvetkov, P. Liang, X. Xu, X. Liu, D. McDuff, H. Lee, H. W. Park, S. Tulebaev, and C. Breazeal, “Medical Hallucinations in Foundation Models and Their Impact on Healthcare,” Feb. 2025. [Online]. Available: <http://arxiv.org/abs/2503.05777>
- [6] S. Banerjee, A. Agarwal, and S. Singla, “Llms will always hallucinate, and we need to live with this,” in *Intelligent Systems Conference*, 2025, pp. 624–648.
- [7] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya *et al.*, “GPT-4 Technical Report,” Mar. 2024. [Online]. Available: <http://arxiv.org/abs/2303.08774>
- [8] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan *et al.*, “Deepseek-v3 technical report,” *arXiv preprint arXiv:2412.19437*, 2024.
- [9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, “LLaMA: Open and Efficient Foundation Language Models,” Feb. 2023.
- [10] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de Las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed, “Mistral 7b,” *CoRR*, vol. abs/2310.06825, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.06825>
- [11] X. Jiang, Y. Tian, F. Hua, C. Xu, Y. Wang, and J. Guo, “A Survey on Large Language Model Hallucination via a Creativity Perspective,” Feb. 2024. [Online]. Available: <http://arxiv.org/abs/2402.06647>
- [12] N. Lee, W. Ping, P. Xu, M. Patwary, P. N. Fung, M. Shoeybi, and B. Catanzaro, “Factuality enhanced language models for open-ended text generation,” in *Proceedings of NeurIPS 2022*, vol. 35, 2022, pp. 34 586–34 599.
- [13] Z. Tang, R. Chatterjee, and S. Garg, “Mitigating hallucinated translations in large language models with hallucination-focused preference optimization,” in *Proceedings of NAACL 2025*, 2025, pp. 3410–3433.
- [14] Y.-S. Chuang, Y. Xie, H. Luo, Y. Kim, J. R. Glass, and P. He, “Dola: Decoding by contrasting layers improves factuality in large language models,” in *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.
- [15] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, and Q. Li, “A survey on rag meeting llms: Towards retrieval-augmented large language models,” in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2024, pp. 6491–6501.
- [16] Z. Chu, J. Chen, Q. Chen, W. Yu, T. He, H. Wang, W. Peng, M. Liu, B. Qin, and T. Liu, “Navigate through enigmatic labyrinth: A survey of chain of thought reasoning—advances, frontiers and future,” in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024, pp. 1173–1203.
- [17] F. Cheng, H. Li, F. Liu, R. van Rooij, K. Zhang, and Z. Lin, “Empowering llms with logical reasoning: A comprehensive survey,” in *Proceedings of the 34th International Joint Conference on Artificial Intelligence (IJCAI)*, 2025.
- [18] xAI, “Grok 4: Model overview and technical specifications,” <https://docs.x.ai/docs/models/grok-4-0709>, 2025, accessed: 2025-10-07.
- [19] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, “Gemini: a family of highly capable multimodal models,” *arXiv preprint arXiv:2312.11805*, 2023.
- [20] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, Y. Gu, H. Ding, K. Men, K. Yang, S. Zhang, X. Deng, A. Zeng, Z. Du, C. Zhang, S. Shen, T. Zhang, Y. Su, H. Sun, M. Huang, Y. Dong, and J. Tang, “Agentbench: Evaluating llms as agents,” in *Proceedings of ICLR 2024*, 2024.
- [21] Z. Wang and B. Yuan, “L-MARS: Legal Multi-Agent Workflow with Orchestrated Reasoning and Agentic Search,” Sep. 2025. [Online]. Available: <http://arxiv.org/abs/2509.00761>
- [22] Y. Xi, J. Lin, M. Zhu, Y. Xiao, Z. Ou, J. Liu, T. Wan, B. Chen, W. Liu, Y. Wang, R. Tang, W. Zhang, and Y. Yu, “InfoDeepSeek: Benchmarking Agentic Information Seeking for Retrieval-Augmented Generation,” May 2025. [Online]. Available: <http://arxiv.org/abs/2505.15872>
- [23] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso *et al.*, “Beyond the imitation game: Quantifying and extrapolating the capabilities of language models,” *Transactions on Machine Learning Research*, 2023.
- [24] K. Valmecikam, M. Marquez, A. Olmo, S. Sreedharan, and S. Kambhampati, “Planbench: An extensible benchmark for evaluating large language models on planning and reasoning about change,” in *Advances in Neural Information Processing Systems*, 2023, pp. 1–13.
- [25] A. Saparov and H. He, “Language models are greedy reasoners: A systematic formal analysis of chain-of-thought,” in *Proceedings of ICLR 2023*, 2023.
- [26] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, L. Hong, R. Tian, R. Xie, J. Zhou, D. Li, Z. Liu, M. Sun, and M. Gerstein, “Toollmm: Facilitating large language models to master 16000+ real-world apis,” in *Proceedings of ICLR 2024*, 2024.
- [27] M. Parmar, N. Patel, N. Varshney, M. Nakamura, M. Luo, S. Mashetty, A. Mitra, and C. Baral, “Logicbench: Towards systematic evaluation of logical reasoning ability of large language models,” in *Proceedings of ACL 2024*, 2024, pp. 13 679–13 707.
- [28] S. Lin, J. Hilton, and O. Evans, “Truthfulqa: Measuring how models mimic human falsehoods,” in *Proceedings of ACL 2022*, 2022, pp. 3214–3252.
- [29] C. Niu, Y. Wu, J. Zhu, S. Xu, K. Shum, R. Zhong, J. Song, and T. Zhang, “Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models,” in *Proceedings of ACL 2024*, 2024, pp. 10 862–10 878.
- [30] T. Vu, M. Iyyer, X. Wang, N. Constant, J. Wei, J. Wei, C. Tar, Y.-H. Sung, D. Zhou, Q. Le, and T. Luong, “Freshllms: Refreshing large language models with search engine augmentation,” in *Findings of ACL 2024*, 2024, pp. 13 697–13 720.
- [31] S. Pandit, J. Xu, J. Hong, Z. Wang, T. Chen, K. Xu, and Y. Ding, “MedHallu: A Comprehensive Benchmark for Detecting Medical Hallucinations in Large Language Models,” Feb. 2025. [Online]. Available: <http://arxiv.org/abs/2502.14302>
- [32] Y. Bang, Z. Ji, A. Schelten, A. Hartshorn, T. Fowler, C. Zhang, N. Cancedda, and P. Fung, “HalluLens: LLM Hallucination Benchmark,” Apr. 2025.
- [33] H. Pham, T.-D. Nguyen, and K.-H. N. Bui, “Agent-UniRAG: A Trainable Open-Source LLM Agent Framework for Unified Retrieval-Augmented Generation Systems,” May 2025. [Online]. Available: <http://arxiv.org/abs/2505.22571>

- [34] J. Wu, J. Zhu, Y. Liu, M. Xu, and Y. Jin, "Agentic reasoning: A streamlined framework for enhancing llm reasoning with agentic tools," in *Proceedings of ACL 2025*, 2025, pp. 28 489–28 503.
- [35] T. Nguyen, P. Chin, and Y.-W. Tai, "MA-RAG: Multi-Agent Retrieval-Augmented Generation via Collaborative Chain-of-Thought Reasoning," May 2025. [Online]. Available: <http://arxiv.org/abs/2505.20096>
- [36] P. Liu, X. Liu, R. Yao, J. Liu, S. Meng, D. Wang, and J. Ma, "HM-RAG: Hierarchical Multi-Agent Multimodal Retrieval Augmented Generation," Apr. 2025. [Online]. Available: <http://arxiv.org/abs/2504.12330>
- [37] Q. Lyu, S. Havaldar, A. Stein, L. Zhang, D. Rao, E. Wong, M. Apidianaki, and C. Callison-Burch, "Faithful chain-of-thought reasoning," in *Proceedings of IJCNLP 2023*, 2023, pp. 305–329.
- [38] J. Xu, H. Fei, L. Pan, Q. Liu, M.-L. Lee, and W. Hsu, "Faithful logical reasoning via symbolic chain-of-thought," in *Proceedings of ACL 2024*, 2024, pp. 13 326–13 365.
- [39] Z. Wang, J. Liu, Q. Bao, H. Rong, and J. Zhang, "Chatlogic: Integrating logic programming with large language models for multi-step reasoning," in *Proceedings of IJCNN 2024*, 2024, pp. 1–8.
- [40] L. Pan, A. Albalak, X. Wang, and W. Wang, "Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning," in *Findings of EMNLP 2023*, 2023, pp. 3806–3824.
- [41] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafrazi, K. R. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," in *Proceedings of ICLR 2023*, 2023.
- [42] T. Schick, J. Dwivedi-Yu, R. Dessa, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 68 539–68 551.
- [43] H. Lu, X. Li, X. Ji, Z. Kan, and Q. Hu, "Toolfive: Enhancing tool-augmented llms via tool filtering and verification," in *Proceedings of ICASSP 2025*, 2025, pp. 1–5.
- [44] W. Chen, X. Ma, X. Wang, and W. W. Cohen, "Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks," *Transactions on Machine Learning Research*, 2023. [Online]. Available: <https://openreview.net/forum?id=YfZ4ZPt8zd>
- [45] Y. Pan, X. Zhang, H. Zhang, Y. Liu, H. Wang, Q. Huang, and Z. Wang, "Answered: Adaptive tool-augmented llms with strategic error feedback for compositional reasoning," in *Advanced Intelligent Computing Technology and Applications*, 2024, pp. 263–280.
- [46] D. Paul, M. Ismayilzada, M. Peyrard, B. Borges, A. Bosselut, R. West, and B. Faltings, "Refiner: Reasoning feedback on intermediate representations," in *Proceedings of EACL 2024*, 2024, pp. 1100–1126.
- [47] Y. Fu, J. Chen, Y. Zhuang, Z. Fu, I. Stoica, and H. Zhang, "Reasoning without self-doubt: More efficient chain-of-thought through certainty probing," in *ICLR 2025 Workshop on Foundation Models in the Wild*, 2025.
- [48] D. Zhu, X. Wei, G. Zhao, W. Wu, H. Zou, J. Ran, X. Wang, L. Sun, X. Zhang, and S. Li, "Chain-of-Thought Matters: Improving Long-Context Language Models with Reasoning Path Supervision," Feb. 2025. [Online]. Available: <http://arxiv.org/abs/2502.20790>
- [49] C. Lou, Z. Sun, X. Liang, M. Qu, W. Shen, W. Wang, Y. Li, Q. Yang, and S. Wu, "AdaCoT: Pareto-Optimal Adaptive Chain-of-Thought Triggering via Reinforcement Learning," May 2025. [Online]. Available: <http://arxiv.org/abs/2505.11896>
- [50] L. Wang, "Dynamic chain-of-thought: Towards adaptive deep reasoning," in *Proceedings of Greeks in AI Symposium 2025*, 2025.
- [51] Y. Zheng, D. Fu, X. Hu, X. Cai, L. Ye, P. Lu, and P. Liu, "DeepResearcher: Scaling Deep Research via Reinforcement Learning in Real-world Environments," Apr. 2025. [Online]. Available: <http://arxiv.org/abs/2504.03160>
- [52] X. Liu, H. Lai, H. Yu, Y. Xu, A. Zeng, Z. Du, P. Zhang, Y. Dong, and J. Tang, "Webglm: Towards an efficient web-enhanced question answering system with human preferences," in *Proceedings of KDD 2023*, 2023, pp. 4549–4560.
- [53] S. Yu, C. Tang, B. Xu, J. Cui, J. Ran, Y. Yan, Z. Liu, S. Wang, X. Han, Z. Liu, and M. Sun, "Visrag: Vision-based retrieval-augmented generation on multi-modality documents," in *Proceedings of ICLR 2025*, 2025.
- [54] T. Cai, Z. Tan, X. Song, T. Sun, J. Jiang, Y. Xu, Y. Zhang, and J. Gu, "Forag: Factuality-optimized retrieval augmented generation for web-enhanced long-form question answering," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 199–210.
- [55] W. Chen, H. Hu, X. Chen, P. Verga, and W. Cohen, "Murag: Multimodal retrieval-augmented generator for open question answering over images and text," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP 2022)*, 2022, pp. 5558–5570.
- [56] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, and J. Larson, "From Local to Global: A Graph RAG Approach to Query-Focused Summarization," Feb. 2025.
- [57] C. Mavromatis and G. Karypis, "Gnn-rag: Graph neural retrieval for efficient large language model reasoning on knowledge graphs," in *Findings of ACL 2025*, 2025, pp. 16 682–16 699.
- [58] R. Kalra, Z. Wu, A. Gulley, A. Hilliard, X. Guan, A. Koshiyama, and P. C. Treleaven, "Hypa-rag: A hybrid parameter adaptive retrieval-augmented generation system for ai legal and policy applications," in *Proceedings of the 1st Workshop on Customizable NLP (CustomNLP4U)*, 2024, pp. 237–256.
- [59] B. Sarmah, D. Mehta, B. Hall, R. Rao, S. Patel, and S. Pasquali, "Hybridrag: Integrating knowledge graphs and vector retrieval augmented generation for efficient information extraction," in *Proceedings of ICAIF 2024*, 2024, pp. 608–616.
- [60] C.-M. Chan, C. Xu, R. Yuan, H. Luo, W. Xue, Y. Guo, and J. Fu, "RQ-RAG: Learning to refine queries for retrieval augmented generation," in *First Conference on Language Modeling*, 2024. [Online]. Available: <https://openreview.net/forum?id=tzE7VqsaJ4>
- [61] K. Sawarkar, A. Mangal, and S. R. Solanki, "Blended rag: Improving retriever-augmented generation accuracy with semantic search and hybrid query-based retrievers," in *Proceedings of MIPR 2024*, 2024, pp. 155–161.
- [62] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-rag: Learning to retrieve, generate, and critique through self-reflection," in *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.
- [63] H. Jiang, Q. Wu, X. Luo, D. Li, C.-Y. Lin, Y. Yang, and L. Qiu, "Longllmenga: Accelerating and enhancing llms in long context scenarios via prompt compression," in *Proceedings of ACL 2024*, 2024, pp. 1658–1677.
- [64] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. V. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 24 824–24 837.
- [65] Z. Bai, P. Wang, T. Xiao, T. He, Z. Han, Z. Zhang, and M. Z. Shou, "Hallucination of Multimodal Large Language Models: A Survey," Apr. 2025.
- [66] P. Sahoo, P. Meharia, A. Ghosh, S. Saha, V. Jain, and A. Chadha, "A comprehensive survey of hallucination in large language, image, video and audio foundation models," in *Findings of EMNLP 2024*, 2024, pp. 11 709–11 724.
- [67] X. Lin, Y. Ning, J. Zhang, Y. Dong, Y. Liu, Y. Wu, X. Qi, N. Sun, Y. Shang, P. Cao, L. Zou, X. Chen, C. Zhou, J. Wu, S. Pan, B. Wang, Y. Cao, K. Chen, S. Hu, and L. Guo, "LLM-based Agents Suffer from Hallucinations: A Survey of Taxonomy, Methods, and Directions," Sep. 2025.
- [68] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, pp. 1–26, 2024.
- [69] W. Zhang and J. Zhang, "Hallucination mitigation for retrieval-augmented large language models: A review," *Mathematics*, vol. 13, no. 5, 2025.
- [70] P. Kumar, "Large language models (llms): Survey, technical frameworks, and future challenges," *Artificial Intelligence Review*, vol. 57, 2024.
- [71] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, and J.-R. Wen, "A Survey of Large Language Models," Mar. 2025.
- [72] S. Ouyang, J. M. Zhang, M. Harman, and M. Wang, "An empirical study of the non-determinism of chatgpt in code generation," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 2, pp. 1–28, 2025.
- [73] J. Li, Y. Liu, W. Fan, X.-Y. Wei, H. Liu, J. Tang, and Q. Li, "Empowering molecule discovery for molecule-caption translation with large language models: A chatgpt perspective," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 11, pp. 6071–6083, 2024.
- [74] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [75] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus, "Emergent abilities of large language models," *Transactions on Machine Learning Research*, 2022.

- [76] S. I. Mirzadeh, K. Alizadeh, H. Shahrokhi, O. Tuzel, S. Bengio, and M. Farajtabar, “Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models,” in *Proceedings of ICLR 2025*, 2025.
- [77] J. Maynez, S. Narayan, B. Bohnet, and R. McDonald, “On faithfulness and factuality in abstractive summarization,” in *Proceedings of ACL 2020*, 2020, pp. 1906–1919.
- [78] P. Henderson, M. Krass, L. Zheng, and D. E. Ho, “Large legal fictions: Profiling legal hallucinations in large language models,” *Journal of Legal Analysis*, vol. 16, no. 1, pp. 64–93, 2024.
- [79] S. Farquhar, J. Kossen, L. Kuhn, and Y. Gal, “Detecting hallucinations in large language models using semantic entropy,” *Nature*, vol. 630, no. 8017, pp. 625–630, 2024.
- [80] A. Mishra, A. Asai, V. Balachandran, Y. Wang, G. Neubig, Y. Tsvetkov, and H. Hajishirzi, “Fine-grained hallucination detection and editing for language models,” in *Proceedings of the First Conference on Language Modeling*, 2024.
- [81] Z. Li, S. Zhang, H. Zhao, Y. Yang, and D. Yang, “Batgpt: A bidirectional autoregressive talker from generative pre-trained transformer,” *arXiv preprint arXiv:2307.00360*, 2023.
- [82] B. Liu, J. Ash, S. Goel, A. Krishnamurthy, and C. Zhang, “Exposing attention glitches with flip-flop language modeling,” in *Proceedings of NeurIPS 2023*, vol. 36, 2023, pp. 25 549–25 583.
- [83] X. L. Li, A. Holtzman, D. Fried, P. Liang, J. Eisner, T. Hashimoto, L. Zettlemoyer, and M. Lewis, “Contrastive decoding: Open-ended text generation as optimization,” in *Proceedings of ACL 2023*, 2023, pp. 12 286–12 312.
- [84] A. Azaria and T. Mitchell, “The internal state of an llm knows when it’s lying,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 967–976.
- [85] E. Mitchell, C. Lin, A. Bosselut, C. Finn, and C. D. Manning, “Fast model editing at scale,” in *Proceedings of ICLR 2022*, 2022.
- [86] N. De Cao, W. Aziz, and I. Titov, “Editing factual knowledge in language models,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 6491–6506.
- [87] X. Ma, Y. Gong, P. He, H. Zhao, and N. Duan, “Query rewriting in retrieval-augmented large language models,” in *Proceedings of EMNLP 2023*, 2023.
- [88] W. Watson, N. Cho, and N. Srishankar, “Is there no such thing as a bad question? h4r: Hallucibot for ratiocination, rewriting, ranking, and routing,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 24, 2025, pp. 25 470–25 478.
- [89] S. Mao, Y. Jiang, B. Chen, X. Li, P. Wang, X. Wang, P. Xie, F. Huang, H. Chen, and N. Zhang, “RaFe: Ranking Feedback Improves Query Rewriting for RAG,” in *Findings of the Association for Computational Linguistics: EMNLP 2024*. Miami, Florida, USA: Association for Computational Linguistics, 2024, pp. 884–901. [Online]. Available: <https://aclanthology.org/2024.findings-emnlp.49>
- [90] J. Tan, Z. Dou, Y. Zhu, P. Guo, K. Fang, and J.-R. Wen, “Small models, big insights: Leveraging slim proxy models to decide when and what to retrieve for llms,” in *Proceedings of ACL 2024*, 2024, pp. 4420–4436.
- [91] H. Qian and Z. Dou, “Explicit Query Rewriting for Conversational Dense Retrieval,” in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 4725–4737.
- [92] J. Liu, Y. K. Tan, B. Fu, and K. H. Lim, “Lara: Linguistic-adaptive retrieval-augmentation for multi-turn intent classification,” in *Proceedings of EMNLP 2024 (Industry Track)*, 2024, pp. 1096–1106.
- [93] Y. Liu, X. Peng, X. Zhang, W. Liu, J. Yin, J. Cao, and T. Du, “Raif: Learning to answer and understand from retrieval augmentation via iterative self-feedback,” in *Findings of ACL 2024*, 2024, pp. 4730–4749.
- [94] J. Fang, Z. Meng, and C. Macdonald, “Kirag: Knowledge-driven iterative retriever for enhancing retrieval-augmented generation,” in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025, pp. 18 969–18 985.
- [95] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” in *Proceedings of NeurIPS 2020*, vol. 33, 2020, pp. 9459–9474.
- [96] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.
- [97] T. Formal, B. Piwowarski, and S. Clinchant, “Splade: Sparse lexical and expansion model for first stage ranking,” in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2021, pp. 2288–2292.
- [98] A. Mallia, O. Khattab, T. Suel, and N. Tonello, “Learning passage impacts for inverted indexes,” in *Proceedings of SIGIR 2021*, 2021, pp. 1723–1727.
- [99] V. Karpukhin, B. Ooguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, “Dense passage retrieval for open-domain question answering,” in *Proceedings of EMNLP 2020*, 2020, pp. 6769–6781.
- [100] G. Izacard, M. Caron, L. Hosseini, S. Riedel, P. Bojanowski, A. Joulin, and E. Grave, “Unsupervised dense information retrieval with contrastive learning,” *Transactions on Machine Learning Research*, 2022.
- [101] N. Reimers and I. Gurevych, “Sentence-bert: Sentence embeddings using siamese bert-networks,” in *Proceedings of EMNLP-IJCNLP 2019*, 2019, pp. 3982–3992.
- [102] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. N. Bennett, J. Ahmed, and A. Overwijk, “Approximate nearest neighbor negative contrastive learning for dense text retrieval,” in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=zeFrfgyZln>
- [103] O. Khattab and M. Zaharia, “Colbert: Efficient and effective passage search via contextualized late interaction over bert,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 39–48.
- [104] K. Santhanam, O. Khattab, J. Saad-Falcon, C. Potts, and M. Zaharia, “Colbertv2: Effective and efficient retrieval via lightweight late interaction,” in *Proceedings of NAACL 2022*, 2022, pp. 3715–3734.
- [105] M. G. Arivazhagan, L. Liu, P. Qi, X. Chen, W. Y. Wang, and Z. Huang, “Hybrid hierarchical retrieval for open-domain question answering,” in *Findings of the Association for Computational Linguistics: ACL*, 2023, pp. 10 680–10 689.
- [106] Y. Hu, Z. Lei, Z. Dai, A. Zhang, A. Angirekula, Z. Zhang, and L. Zhao, “Cg-rag: Research question answering by citation graph retrieval-augmented llms,” in *Proceedings of SIGIR 2025*, 2025, pp. 678–687.
- [107] Z. Zhong, H. Liu, X. Cui, X. Zhang, and Z. Qin, “Mix-of-granularity: Optimize the chunking granularity for retrieval-augmented generation,” in *Proceedings of COLING 2025*, 2025, pp. 5756–5774.
- [108] Y. Huang, S. Zhang, and X. Xiao, “Ket-rag: A cost-efficient multi-granular indexing framework for graph-rag,” in *Proceedings of KDD 2025*, 2025, pp. 1003–1012.
- [109] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, “Lost in the middle: How language models use long contexts,” *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 157–173, 2024.
- [110] S. Robertson, H. Zaragoza *et al.*, “The probabilistic relevance framework: Bm25 and beyond,” *Foundations and Trends® in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009.
- [111] T.-Y. Liu, “Learning to rank for information retrieval,” *Foundations and Trends in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.
- [112] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender, “Learning to rank using gradient descent,” in *Proceedings of the 22nd International Conference on Machine Learning*, ser. ICML ’05. New York, NY, USA: Association for Computing Machinery, 2005, p. 89–96.
- [113] C. Burges, R. Ragno, and Q. Le, “Learning to rank with nonsmooth cost functions,” *Advances in Neural Information Processing Systems*, vol. 19, 2006.
- [114] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 4171–4186.
- [115] Q. Liu, H. Duan, Y. Chen, Q. Lu, W. Sun, and J. Mao, “LLM4Ranking: An Easy-to-use Framework of Utilizing Large Language Models for Document Reranking,” Apr. 2025.
- [116] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, “Palm: Scaling language modeling with pathways,” *Journal of Machine Learning Research*, vol. 24, no. 240, pp. 1–113, 2023.
- [117] S. Zhuang, X. Ma, B. Koopman, J. Lin, and G. Zuccon, “Rank-R1: Enhancing Reasoning in LLM-based Document Rerankers via Reinforcement Learning,” Mar. 2025.
- [118] C.-Y. Chang, Z. Jiang, V. Rakesh, M. Pan, C.-C. M. Yeh, G. Wang, M. Hu, Z. Xu, Y. Zheng, M. Das, and N. Zou, “Main-rag: Multi-agent filtering retrieval-augmented generation,” in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025, 2025, pp. 2607–2622.

- [119] H. Zhou, K.-H. Lee, Z. Zhan, Y. Chen, Z. Li, Z. Wang, H. Haddadi, and E. Yilmaz, “TrustRAG: Enhancing Robustness and Trustworthiness in RAG,” Jan. 2025.
- [120] T. Hwang, S. Jeong, S. Cho, S. Han, and J. Park, “Dslr: Document refinement with sentence-level re-ranking and reconstruction to enhance retrieval-augmented generation,” in *Proceedings of the 3rd Workshop on Knowledge Augmented Methods for NLP*, 2024, pp. 73–92.
- [121] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, “In-context retrieval-augmented language models,” *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1316–1331, 2023.
- [122] S. Wu, Y. Xiong, Y. Cui, X. Liu, B. Tang, T.-W. Kuo, and C. J. Xue, “Refusion: Improving natural language understanding with computation-efficient retrieval representation fusion,” in *Proceedings of ICLR 2024*, 2024.
- [123] K. Kim and J.-Y. Lee, “Re-rag: Improving open-domain qa performance and interpretability with relevance estimator in retrieval-augmented generation,” in *Proceedings of EMNLP 2024*, 2024, pp. 22 149–22 161.
- [124] L. Ranaldi, M. Valentino, and A. Freitas, “Eliciting critical reasoning in retrieval-augmented generation via contrastive explanations,” in *Proceedings of NAACL 2025*, 2025, pp. 11 168–11 183.
- [125] U. Tariq, M. A. Shah, and A. Khan, “Explainable ai: A retrieval-augmented generation based framework for model interpretability,” in *Proceedings of the 17th International Conference on Agents and Artificial Intelligence (ICAART)*, 2025, pp. 948–955.
- [126] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, “A survey on knowledge graphs: Representation, acquisition, and applications,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494–514, 2022.
- [127] L. Zhong, J. Wu, Q. Li, H. Peng, and X. Wu, “A comprehensive survey on automatic knowledge graph construction,” *ACM Computing Surveys*, vol. 56, no. 4, 2023.
- [128] D. Li, S. Yang, Z. Tan, J. Y. Baik, S. Yun, J. Lee, A. Chacko, B. Hou, D. Duong-Tran, Y. Ding, H. Liu, L. Shen, and T. Chen, “Dalk: Dynamic co-augmentation of llms and knowledge graphs to answer alzheimer’s disease questions with scientific literature,” in *Findings of EMNLP 2024*, 2024, pp. 2187–2205.
- [129] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. Ni, H.-Y. Shum, and J. Guo, “Think-on-graph: Deep and responsible reasoning of large language models on knowledge graphs,” in *Proceedings of ICLR 2024*, 2024.
- [130] S. Ma, C. Xu, X. Jiang, M. Li, H. Qu, C. Yang, J. Mao, and J. Guo, “Think-on-graph 2.0: Deep and faithful large language model reasoning with knowledge-guided retrieval augmented generation,” in *Proceedings of ICLR 2025*, 2025.
- [131] T. Xu, H. Zheng, C. Li, H. Chen, Y. Liu, R. Chen, and L. Sun, “NodeRAG: Structuring Graph-based RAG with Heterogeneous Nodes,” Apr. 2025.
- [132] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman, “WebGPT: Brower-assisted question-answering with human feedback,” Jun. 2022.
- [133] A. Lazaridou, E. Gribovskaya, W. Stokowiec, and N. Grigorev, “Internet-augmented language models through few-shot prompting for open-domain question answering,” May 2022.
- [134] J. Wu, W. Yin, Y. Jiang, Z. Wang, Z. Xi, R. Fang, L. Zhang, Y. He, D. Zhou, P. Xie, and F. Huang, “Webwalker: Benchmarking llms in web traversal,” in *Workshop on Reasoning and Planning for Large Language Models*, 2025.
- [135] J. Tan, Z. Dou, W. Wang, M. Wang, W. Chen, and J.-R. Wen, “Htmrlrag: Html is better than plain text for modeling retrieved knowledge in rag systems,” in *Proceedings of WWW 2025*, 2025, pp. 1733–1746.
- [136] R. Zhao, H. Chen, W. Wang, F. Jiao, X. L. Do, C. Qin, B. Ding, X. Guo, M. Li, X. Li, and S. Joty, “Retrieving multimodal information for augmented generation: A survey,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 4736–4756.
- [137] N. Wasserman, R. Pony, O. Naparstek, A. R. Goldfarb, E. Schwartz, U. Barzelay, and L. Karlinsky, “Real-mm-rag: A real-world multi-modal retrieval benchmark,” in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025)*, 2025, pp. 31 660–31 683.
- [138] J. Yuan, H. Gao, D. Dai, J. Luo, L. Zhao, Z. Zhang, Z. Xie, Y. Wei, L. Wang, Z. Xiao, Y. Wang, C. Ruan, M. Zhang, W. Liang, and W. Zeng, “Native sparse attention: Hardware-aligned and natively trainable sparse attention,” in *Proceedings of ACL 2025*, 2025, pp. 23 078–23 097.
- [139] R. Xu, G. Xiao, H. Huang, J. Guo, and S. Han, “Xattention: Block sparse attention with antidiagonal scoring,” in *Proceedings of ICML 2025*, 2025.
- [140] L. Huang, X. Feng, W. Ma, Y. Fan, X. Feng, Y. Ye, W. Zhong, Y. Gu, B. Wang, D. Wu, G. Hu, and B. Qin, “Improving contextual faithfulness of large language models via retrieval heads-induced optimization,” in *Proceedings of ACL 2025*, 2025, pp. 16 896–16 913.
- [141] S. Li, Y. He, H. Guo, X. Bu, G. Bai, J. Liu, J. Liu, X. Qu, Y. Li, W. Ouyang, W. Su, and B. Zheng, “Graphreader: Building graph-based agent to enhance long-context abilities of large language models,” in *Findings of EMNLP 2024*, 2024, pp. 12 758–12 786.
- [142] K.-H. Lee, X. Chen, H. Furuta, J. Cannan, and I. Fischer, “A human-inspired reading agent with gist memory of very long contexts,” in *Proceedings of ICML 2024*, 2024, pp. 1054–1074.
- [143] Y. Cao, S. Li, Y. Liu, Z. Yan, Y. Dai, P. S. Yu, and L. Sun, “A Comprehensive Survey of AI-Generated Content (AIGC): A History of Generative AI from GAN to ChatGPT,” Mar. 2023. [Online]. Available: <http://arxiv.org/abs/2303.04226>
- [144] I. Shumailov, Z. Shumaylov, Y. Zhao, N. Papernot, R. Anderson, and Y. Gal, “Ai models collapse when trained on recursively generated data,” *Nature*, vol. 631, no. 8022, pp. 755–759, 2024.
- [145] X. Zhao, S. Gunn, M. Christ, J. Fairoze, A. Fabrega, N. Carlini, S. Garg, S. Hong, M. Nasr, F. Tramer, S. Jha, L. Li, Y.-X. Wang, and D. Song, “Sok: Watermarking for ai-generated content,” in *Proceedings of IEEE Symposium on Security and Privacy (SP 2025)*, 2025, pp. 2621–2639.
- [146] S. Gehrmann, H. Strobelt, and A. Rush, “Gltr: Statistical detection and visualization of generated text,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL)*, 2019, pp. 111–116.
- [147] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, “Detectgpt: Zero-shot machine-generated text detection using probability curvature,” in *Proceedings of ICML 2023*, 2023.
- [148] M. A. Rakib Mollah, M. M. J. Kabir, M. Kabir, and M. S. Reza, “Detection of fake news with roberta based embedding and modified deep neural network architecture,” in *Proceedings of ICCIT 2023*, 2023, pp. 1–6.
- [149] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl, P. Payne, M. Seneviratne, P. Gamble, C. Kelly, N. Schärlí, A. Chowdhery, P. Mansfield, B. Aguera y Arcas, D. Webster, G. S. Corrado, Y. Matias, K. Chou, J. Gottweis, N. Tomasev, Y. Liu, A. Rajkomar, J. Barral, C. Semturs, A. Karthikesalingam, and V. Natarajan, “Large language models encode clinical knowledge,” *Nature*, vol. 620, no. 7972, pp. 172–180, 2023.
- [150] K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, L. Hou, K. Clark, S. Pfohl, H. Cole-Lewis, D. Neal, M. Schaeckermann, A. Wang, M. Amin, S. Lachgar, P. Mansfield, S. Prakash, B. Green, E. Dominowska, B. Aguera y Arcas, N. Tomasev, Y. Liu, R. Wong, C. Semturs, S. S. Mahdavi, J. Barral, D. Webster, G. S. Corrado, Y. Matias, S. Azizi, A. Karthikesalingam, and V. Natarajan, “Toward expert-level medical question answering with large language models,” *Nature Medicine*, vol. 31, no. 3, pp. 943–950, 2025.
- [151] Y. Shi, S. Xu, T. Yang, Z. Liu, T. Liu, X. Li, and N. Liu, “Mk-rag: Medical knowledge retrieval-augmented generation for medical question answering,” in *Proceedings of AMIA 2025*, 2025, p. 1011.
- [152] S. Anjum, H. Zhang, W. Zhou, E. J. Paek, X. Zhao, and Y. Feng, “Halo: Hallucination analysis and learning optimization to empower llms with retrieval-augmented context for guided clinical decision making,” in *2025 IEEE/ACM Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*. IEEE, 2025, pp. 187–198.
- [153] N. Pipitone and G. H. Alami, “LegalBench-RAG: A Benchmark for Retrieval-Augmented Generation in the Legal Domain,” Aug. 2024. [Online]. Available: <http://arxiv.org/abs/2408.10343>
- [154] H. Li, Y. Chen, Y. Hu, Q. Ai, J. Chen, X. Yang, J. Yang, Y. Wu, Z. Liu, and Y. Liu, “Lexrag: Benchmarking retrieval-augmented generation in multi-turn legal consultation conversation,” in *Proceedings of SIGIR 2025*, 2025, pp. 3606–3615.
- [155] Y. Zhao, P. Singh, H. Bhathena, B. Ramos, A. Joshi, S. Gadiyaram, and S. Sharma, “Optimizing llm-based retrieval-augmented generation pipelines in the financial domain,” in *Proceedings of NAACL 2024 (Industry Track)*, 2024, pp. 279–294.
- [156] J. Wang, W. Ding, and X. Zhu, “Financial analysis: Intelligent financial data analysis system based on llm-rag,” in *Proceedings of the 3rd International Conference on Software Engineering and Machine Learning*, 2025, pp. 182–189.
- [157] S. Dakshit, “Faculty perspectives on the potential of rag in computer science higher education,” in *Proceedings of the 25th Annual Conference on Information Technology Education (SIGITE)*, 2024, pp. 19–24.
- [158] Z. Li, Z. Wang, W. Wang, K. Hung, H. Xie, and F. L. Wang, “Retrieval-augmented generation for educational application: A systematic survey,”

- Computers and Education: Artificial Intelligence*, vol. 8, p. 100417, 2025.
- [159] J. Swacha and M. Gracel, "Retrieval-augmented generation (rag) chatbots for education: A survey of applications," *Applied Sciences*, vol. 15, no. 8, 2025.
- [160] G. Feng, B. Zhang, Y. Gu, H. Ye, D. He, and L. Wang, "Towards revealing the mystery behind chain of thought: A theoretical perspective," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023, pp. 70 757–70 798.
- [161] T. Kojima, S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," in *Proceedings of NeurIPS 2022*, vol. 35, 2022, pp. 22 199–22 213.
- [162] N. Muennighoff, Z. Yang, W. Shi, X. L. Li, L. Fei-Fei, H. Hajishirzi, L. Zettlemoyer, P. Liang, E. Candes, and T. Hashimoto, "s1: Simple test-time scaling," in *Workshop on Reasoning and Planning for Large Language Models*, 2025.
- [163] Z. Ling, Y. Fang, X. Li, Z. Huang, M. Lee, R. Memisevic, and H. Su, "Deductive verification of chain-of-thought reasoning," in *Proceedings of NeurIPS 2023*, vol. 36, 2023, pp. 36 407–36 433.
- [164] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," in *Proceedings of ICLR 2023*, 2023.
- [165] R. Liu, J. Geng, A. J. Wu, I. Sucholutsky, T. Lombrozo, and T. L. Griffiths, "Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse," in *Proceedings of ICML 2025*, 2025.
- [166] D. Das, D. Banerjee, S. Aditya, and A. Kulkarni, "Mathsensei: A tool-augmented large language model for mathematical reasoning," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2024, pp. 942–966.
- [167] Y. Ma, Z. Gou, J. Hao, R. Xu, S. Wang, L. Pan, Y. Yang, Y. Cao, and A. Sun, "Sciagent: Tool-augmented language models for scientific reasoning," in *Proceedings of EMNLP 2024*, 2024, pp. 15 701–15 736.
- [168] G. Yang, Y. Zhou, X. Chen, X. Zhang, T. Y. Zhuo, and T. Chen, "Chain-of-thought in neural code generation: From and for lightweight language models," *IEEE Transactions on Software Engineering*, vol. 50, no. 9, pp. 2437–2457, 2024.
- [169] L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig, "Pal: Program-aided language models," in *Proceedings of the 40th International Conference on Machine Learning (ICML)*, vol. 202, 2023, pp. 10 764–10 799.
- [170] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions," in *Proceedings of ACL 2023*, 2023, pp. 10 014–10 037.
- [171] X. Lu, L. Pan, Y. Ma, P. Nakov, and M.-Y. Kan, "Tart: An open-source tool-augmented framework for explainable table-based reasoning," in *Findings of NAACL 2025*, 2025, pp. 4323–4339.
- [172] Z. Chen, K. Zhou, B. Zhang, Z. Gong, X. Zhao, and J.-R. Wen, "Chatcot: Tool-augmented chain-of-thought reasoning on chat-based large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 14 777–14 790.
- [173] S. Gao, J. Dwivedi-Yu, P. Yu, X. E. Tan, R. Pasunuru, O. Golovneva, K. Sinha, A. Celikyilmaz, A. Bosselut, and T. Wang, "Efficient tool use with chain-of-abstraction reasoning," in *Proceedings of the 31st International Conference on Computational Linguistics (COLING)*, 2025, pp. 2727–2743.
- [174] S. Chen, Y. Wang, Y.-F. Wu, Q.-G. Chen, Z. Xu, W. Luo, K. Zhang, and L. Zhang, "Advancing tool-augmented large language models: Integrating insights from errors in inference trees," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 106 555–106 581.
- [175] B. Zhang, K. Zhou, X. Wei, X. Zhao, J. Sha, S. Wang, and J.-R. Wen, "Evaluating and improving tool-augmented computation-intensive math reasoning," in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 23 570–23 589.
- [176] L. D. Raedt, S. Dumančić, R. Manhaeve, and G. Marra, "From statistical relational to neural-symbolic artificial intelligence," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2021, pp. 688–695.
- [177] U. Nawaz, M. Anees-ur Rahaman, and Z. Saeed, "A review of neuro-symbolic ai integrating reasoning and learning for advanced cognitive systems," *Intelligent Systems with Applications*, p. 200541, 2025.
- [178] Anthropic, "Claude 4 system card," <https://www-cdn.anthropic.com/4263b940cab546aa0e3283f35b686f4f3b2ff47.pdf>, 2025, accessed: 2025-08-09.
- [179] I. Anysphere, "Cursor: Ai-powered code editor," <https://cursor.com/>, 2025, version 1.0 (if applicable), accessed 2025-10-24.
- [180] R. Li, L. Ben Allal, Y. Zi, N. Muennighoff, D. Kocetkov, C. Mou, M. Marone, C. Akiki, J. Li, J. Chim, Q. Liu, E. Zheltonozhskii, T. Y. Zhuo, T. Wang, O. Dehaene, J. Lamy-Poirier, J. Monteiro, N. Gontier, M.-H. Yee, L. K. Umapathi, J. Zhu, B. Lipkin, M. Oblokulov, Z. Wang, R. Murthy, J. T. Stillerman, S. S. Patel, D. Abulkhanov, M. Zocca, M. Dey, Z. Zhang, U. Bhattacharyya, W. Yu, S. Luccioni, P. Villegas, F. Zhdanov, T. Lee, N. Timor, J. Ding, C. S. Schlesinger, H. Schoelkopf, J. Ebert, T. Dao, M. Mishra, A. Gu, C. J. Anderson, B. Dolan-Gavitt, D. Contractor, S. Reddy, D. Fried, D. Bahdanau, Y. Jernite, C. M. Ferrandis, S. Hughes, T. Wolf, A. Guha, L. Von Werra, and H. de Vries, "Starcoder: May the source be with you!" *Transactions on Machine Learning Research*, 2023.
- [181] E. Nijkamp, B. Pang, H. Hayashi, L. Tu, H. Wang, Y. Zhou, S. Savarese, and C. Xiong, "Codegen: An open large language model for code with multi-turn program synthesis," in *Proceedings of ICLR 2023*, 2023.
- [182] B. Rozière, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan *et al.*, "Code Llama: Open Foundation Models for Code," Jan. 2024. [Online]. Available: <http://arxiv.org/abs/2308.12950>
- [183] J. Li, G. Li, Y. Li, and Z. Jin, "Structured chain-of-thought prompting for code generation," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 2, pp. 1–23, 2025.
- [184] W. Zheng, S. P. Sharan, A. K. Jaiswal, K. Wang, Y. Xi, D. Xu, and Z. Wang, "Outline, then details: Syntactically guided coarse-to-fine code generation," in *Proceedings of ICML 2023*, 2023.
- [185] V. Gaur and N. Saunshi, "Reasoning in large language models through symbolic math word problems," in *Findings of the Association for Computational Linguistics: ACL 2023*, 2023, pp. 5889–5903.
- [186] V. Dhanraj and C. Eliasmith, "Improving Rule-based Reasoning in LLMs via Neurosymbolic Representations," Jan. 2025.
- [187] A. Cuadron, D. Li, W. Ma, X. Wang, Y. Wang, S. Zhuang, S. Liu, L. G. Schroeder, T. Xia, H. Mao, N. Thumiger, A. Desai, I. Stoica, A. Klimovic, G. Neubig, and J. E. Gonzalez, "The Danger of Overthinking: Examining the Reasoning-Action Dilemma in Agentic Tasks," Feb. 2025. [Online]. Available: <http://arxiv.org/abs/2502.08235>
- [188] X. Chen, J. Xu, T. Liang, Z. He, J. Pang, D. Yu, L. Song, Q. Liu, M. Zhou, Z. Zhang, R. Wang, Z. Tu, H. Mi, and D. Yu, "Do NOT think that much for 2+3=? on the overthinking of long reasoning models," in *Forty-second International Conference on Machine Learning*, 2025.
- [189] M. Jin, Q. Yu, D. Shu, H. Zhao, W. Hua, Y. Meng, Y. Zhang, and M. Du, "The impact of reasoning step length on large language models," in *Findings of the Association for Computational Linguistics: ACL 2024*, 2024, pp. 1830–1842.
- [190] C. Qian, Y. Dang, J. Li, W. Liu, Z. Xie, Y. Wang, W. Chen, C. Yang, X. Cong, X. Che, Z. Liu, and M. Sun, "Experiential co-learning of software-developing agents," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 2024, pp. 5628–5640.
- [191] C. Qian, W. Liu, H. Liu, N. Chen, Y. Dang, J. Li, C. Yang, W. Chen, Y. Su, X. Cong, J. Xu, D. Li, Z. Liu, and M. Sun, "Chatdev: Communicative agents for software development," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 2024, pp. 15 174–15 186.
- [192] C. Qian, Z. Xie, Y. Wang, W. Liu, K. Zhu, H. Xia, Y. Dang, Z. Du, W. Chen, C. Yang, Z. Liu, and M. Sun, "Scaling large language model-based multi-agent collaboration," in *Proceedings of the Thirteenth International Conference on Learning Representations*, 2025.
- [193] J. Yang, C. E. Jimenez, A. Wettig, K. Lieret, S. Yao, K. Narasimhan, and O. Press, "Swe-agent: Agent-computer interfaces enable automated software engineering," in *Advances in Neural Information Processing Systems*, vol. 38, 2025.
- [194] Y. Yamada, R. T. Lange, C. Lu, S. Hu, C. Lu, J. Foerster, J. Clune, and D. Ha, "The AI Scientist-v2: Workshop-Level Automated Scientific Discovery via Agentic Tree Search," Apr. 2025. [Online]. Available: <http://arxiv.org/abs/2504.08066>
- [195] O. Tafjord, B. Dalvi Mishra, and P. Clark, "Proofwriter: Generating implications, proofs, and abductive statements over natural language," in *Findings of ACL-IJCNLP 2021*, 2021, pp. 3621–3634.
- [196] M. Li, Y. Zhao, B. Yu, F. Song, H. Li, H. Yu, Z. Li, F. Huang, and Y. Li, "Api-bank: A comprehensive benchmark for tool-augmented llms," in *Proceedings of EMNLP 2023*, 2023, pp. 3102–3116.
- [197] W. Yu, Z. Jiang, Y. Dong, and J. Feng, "Reclor: A reading comprehension dataset requiring logical reasoning," in *Proceedings of ICLR 2020*, 2020.

- [198] J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang, “Logiq: A challenge dataset for machine reading comprehension with logical reasoning,” in *Proceedings of IJCAI 2021*, 2021, pp. 501–507.
- [199] T. Yuan, Z. He, L. Dong, Y. Wang, R. Zhao, T. Xia, L. Xu, B. Zhou, F. Li, Z. Zhang, R. Wang, and G. Liu, “R-Judge: Benchmarking safety risk awareness for llm agents,” in *Findings of EMNLP 2024*, 2024, pp. 1467–1490.
- [200] M. Turpin, J. Michael, E. Perez, and S. Bowman, “Language models don’t always say what they think: Unfaithful explanations in chain-of-thought prompting,” in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 74 952–74 965.
- [201] M. Abo El-Enen, S. Saad, and T. Nazmy, “A survey on retrieval-augmentation generation (rag) models for healthcare applications,” *Neural Computing and Applications*, 2025.