Review

Retrieval-Augmented Text Generation: Methods, Challenges, and Applications

Jeanie Genesis *

Posted Date: 8 April 2025

doi: 10.20944/preprints202504.0443.v1

Keywords: Retrieval-Augmented Generation; Large Language Models; Information Retrieval; Natural Language Processing; Knowledge Grounding; Dense Retrieval; Sparse Retrieval; Hybrid Search; Neural Text Generation; Open-Domain Question Answering; Conversational AI; Explainable AI; Multimodal Retrieval; Continual Learning; Bias Mitigation

 doi:10.20944/preprints202504.0443.v1

Review

Retrieval-Augmented Text Generation: Methods, Challenges, and Applications

Retrieval-Augmented Text Generation

Jeanie Genesis

Department of Computer Science, Lancaster University; jeanie.genesis@lancaster.ac.uk

Abstract: Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language understanding and generation. However, they are inherently constrained by the static nature of their pretraining data, leading to challenges such as knowledge obsolescence, hallucination, and limited factual grounding. Retrieval-Augmented Generation (RAG) has emerged as a powerful paradigm that addresses these limitations by dynamically integrating external knowledge retrieval with generative text modeling. By retrieving relevant documents or structured knowledge at inference time, RAG enhances model reliability, improves factual accuracy, and enables real-time knowledge adaptation. This survey provides a comprehensive overview of RAG, covering its foundational principles, retrieval mechanisms, generative strategies, and integration methodologies. We discuss various retrieval approaches, including sparse and dense retrieval, hybrid search models, and reinforcement learning-based retrieval optimization. We explore different fusion techniques for incorporating retrieved knowledge into generation, such as prompt concatenation, attention-based integration, and iterative refinement. Additionally, we examine the diverse applications of RAG across domains such as open-domain question answering, conversational AI, scientific literature summarization, code generation, legal document analysis, and biomedical research. Despite its advantages, RAG introduces new challenges, including retrieval noise, latency constraints, security vulnerabilities, and bias in retrieved content. We highlight key research directions to address these challenges, including scalable retrieval architectures, multimodal knowledge integration, continual learning for adaptive retrieval, and bias-aware ranking techniques. Furthermore, we discuss the broader implications of RAG in enabling explainable AI, bridging structured and unstructured knowledge sources, and democratizing access to real-time information. By synthesizing recent advancements and outlining future research opportunities, this survey serves as a foundational resource for researchers and practitioners working on retrieval-augmented systems. As RAG continues to evolve, it is poised to redefine the landscape of AI-driven text generation, paving the way for more accurate, interpretable, and knowledge-aware artificial intelligence systems.

Keywords: Retrieval-Augmented Generation, Large Language Models, Information Retrieval, Natural Language Processing, Knowledge Grounding, Dense Retrieval, Sparse Retrieval, Hybrid Search, Neural Text Generation, Open-Domain Question Answering, Conversational AI, Explainable AI, Multimodal Retrieval, Continual Learning, Bias Mitigation

1. Introduction

The rapid advancement of large language models (LLMs) has ushered in a new era of artificial intelligence, enabling unprecedented capabilities in natural language understanding, generation, and interaction [1]. These models, exemplified by architectures such as OpenAI's GPT, Google's PaLM, and Meta's LLaMA, leverage vast amounts of textual data and billions of parameters to produce human-like text across a diverse array of applications. Despite their remarkable success, LLMs are fundamentally constrained by limitations in knowledge retrieval, factual consistency, and efficient

 doi:10.20944/preprints202504.0443.v1

adaptation to dynamic information. A critical challenge arises from the fact that these models rely on static pretraining corpora, which become outdated over time, leading to issues such as hallucination, misinformation propagation, and an inability to incorporate real-time knowledge updates. Addressing these challenges has led to the emergence of Retrieval-Augmented Generation (RAG), a paradigm designed to enhance LLMs by integrating information retrieval mechanisms directly into the generation process. RAG represents a hybrid approach that combines the generative strengths of LLMs with the precision and up-to-date knowledge of information retrieval systems [2]. Unlike conventional autoregressive models that generate text based solely on their internal parameters, RAG actively queries external knowledge sources—such as document databases, web search engines, or structured knowledge graphs—at inference time [3]. This retrieval mechanism ensures that generated content is grounded in verifiable and contemporaneous information, significantly improving factual accuracy and contextual relevance. The introduction of retrieval-based augmentation marks a paradigm shift in the development of AI-powered language systems, enabling them to overcome fundamental drawbacks associated with parametric-only knowledge storage. The motivation for Retrieval-Augmented Generation arises from multiple perspectives. First, LLMs trained on static corpora suffer from temporal obsolescence, as their knowledge is frozen at the time of training [4]. In contrast, retrieval-augmented methods allow for real-time incorporation of updated information, ensuring that models remain relevant in fast-evolving domains such as finance, medicine, and legal studies. Second, the sheer scale of training datasets imposes practical constraints on model retraining, both in terms of computational resources and energy consumption. By leveraging retrieval, models can effectively expand their knowledge base without necessitating frequent re-training, offering a scalable and environmentally sustainable solution. Third, the interpretability and verifiability of LLM outputs remain an ongoing concern, particularly in high-stakes applications where trustworthiness is paramount. By explicitly citing external sources during text generation, RAG-based models enhance transparency, enabling users to trace the origin of factual claims and assess their credibility [5]. The core mechanism of RAG involves a two-step process: retrieval and generation. The retrieval phase identifies relevant documents or knowledge snippets based on the input query, using techniques such as dense passage retrieval (DPR), BM25, or neural search models [6]. These retrieved elements are then incorporated into the prompt or intermediate representations, guiding the subsequent text generation phase [7]. Various implementations of RAG differ in how they integrate retrieved knowledge, ranging from concatenating retrieved texts into the prompt to fine-tuning models with retrieved evidence [8]. Advanced architectures may employ reranking mechanisms, attention-based fusion strategies, or reinforcement learning techniques to optimize retrieval quality and ensure the most relevant information is utilized. A key area of research within RAG involves optimizing retrieval efficiency while maintaining high fidelity to user queries [9]. Traditional retrieval methods, such as keyword-based search, often fall short in capturing nuanced semantic relationships [10]. To address this limitation, researchers have explored neural retrieval approaches, where dense vector representations enable more effective matching between queries and documents. Techniques such as contrastive learning, knowledge distillation, and hybrid retrieval strategies further enhance the ability of models to retrieve and synthesize high-quality information. Additionally, the dynamic nature of retrieval poses challenges related to latency and computational overhead, necessitating the development of efficient indexing structures, caching mechanisms, and hardware-accelerated retrieval pipelines. Beyond improving factual consistency, RAG has significant implications for personalization and domain-specific applications. In personalized AI assistants, retrieval-augmented models can tailor responses based on user-specific context, incorporating prior interactions and preferences into their generated outputs. In specialized fields such as biomedical research and legal analysis, RAG allows LLMs to access and synthesize highly specialized knowledge that may not be well-represented in general-purpose training datasets. This adaptability makes RAG a crucial innovation for extending the applicability of LLMs across a broad spectrum of professional and academic domains [11]. Despite its advantages, RAG is not

 doi:10.20944/preprints202504.0443.v1

without challenges [12]. One fundamental issue is retrieval noise, where retrieved documents contain misleading or contradictory information, potentially compromising the reliability of generated text. Additionally, RAG models must contend with computational trade-offs, balancing retrieval depth with response latency. Ensuring security and robustness in retrieval-augmented systems is another active area of research, as malicious actors could manipulate external knowledge sources to influence model outputs. Addressing these challenges requires continued advancements in retrieval methodologies, filtering mechanisms, and adversarial robustness techniques [13]. In this survey, we provide a comprehensive review of Retrieval-Augmented Generation, covering its foundational principles, architectural variations, key methodologies, and practical applications. We explore state-of-the-art techniques for integrating retrieval with LLMs, discuss ongoing challenges, and highlight future directions in the field [14]. By synthesizing insights from recent research and industrial deployments, this survey aims to serve as a valuable resource for researchers, practitioners, and policymakers interested in the evolving landscape of retrieval-augmented AI systems.

2. Background and Foundations

Retrieval-Augmented Generation (RAG) builds upon foundational concepts from information retrieval, natural language processing (NLP), and deep learning. To fully understand the mechanisms underpinning RAG, it is essential to explore the theoretical and technical foundations that contribute to its development. This section provides an overview of the key principles underlying retrieval-based systems, the evolution of language modeling techniques, and the role of knowledge augmentation in modern AI systems [15].

2.1. Traditional Information Retrieval Methods

The field of information retrieval (IR) has a long history, predating the development of large-scale neural models. Traditional IR systems aim to identify relevant documents or passages in response to a user query [16]. Early approaches were largely keyword-based, relying on techniques such as Boolean search and term frequency-inverse document frequency (TF-IDF) [17]. One of the most widely used traditional retrieval algorithms is BM25, a probabilistic ranking function that scores documents based on term frequency and document length normalization. These classic methods formed the backbone of search engines and digital libraries before the advent of deep learning-based retrieval models. Despite their effectiveness in many applications, traditional retrieval methods suffer from several limitations [18]. Keyword-based retrieval systems often struggle with synonymy (different words expressing the same concept) and polysemy (words with multiple meanings), leading to suboptimal matching between queries and relevant documents [19]. Additionally, these approaches rely on handcrafted heuristics and do not leverage contextual or semantic understanding, which limits their performance on complex natural language queries.

2.2. The Evolution of Neural Language Models

The introduction of deep learning transformed NLP, leading to the development of increasingly sophisticated language models. Early neural approaches, such as word embeddings (Word2Vec, GloVe), provided vectorized representations of words, capturing semantic relationships through dense embeddings [20]. This evolution continued with the advent of deep contextualized representations, as seen in models like ELMo and BERT [21]. Unlike static word embeddings, these models leveraged transformer architectures to process entire sentences in context, allowing for more nuanced understanding of meaning [22]. Transformer-based models, particularly those following the autoregressive paradigm (such as GPT) or the encoder-decoder paradigm (such as T5 and BART), enabled state-of-the-art performance in text generation, question answering, and summarization [23]. However, these models remained fundamentally constrained by the knowledge encoded in their parameters during

 doi:10.20944/preprints202504.0443.v1

training [24]. This limitation spurred interest in augmentation techniques that could extend model knowledge beyond the confines of pretraining data.

2.3. Knowledge Augmentation Strategies

To address the limitations of parametric memory in LLMs, researchers have explored various knowledge augmentation strategies [25]. These methods can be broadly categorized into:

- **Explicit Knowledge Injection:** Approaches that integrate structured or unstructured external knowledge sources directly into the model. This includes knowledge graph embeddings, entity linking, and rule-based enhancements.
- **Retrieval-Augmented Approaches:** Techniques that dynamically retrieve relevant documents or passages at inference time, grounding the model's outputs in real-world data.
- **Memory-Augmented Neural Networks:** Models that incorporate external memory modules to store and retrieve information dynamically.
- **Hybrid Techniques:** Methods that combine retrieval with fine-tuning strategies, optimizing models to effectively incorporate retrieved evidence.

Among these strategies, retrieval-augmented generation has gained significant traction due to its ability to enhance factual accuracy while maintaining the fluency and coherence of generative models [26]. By retrieving knowledge from external databases, RAG-based models provide up-to-date and verifiable information, mitigating issues related to hallucination and outdated knowledge.

2.4. Challenges in Integrating Retrieval with Generation

Integrating retrieval with text generation introduces several technical and practical challenges:

- **Efficient and Scalable Retrieval:** Ensuring fast and accurate retrieval from large-scale knowledge bases while maintaining low latency [27].
- **Relevance and Ranking:** Optimizing retrieval models to return the most relevant information, minimizing noisy or irrelevant results.
- **Fusion Strategies:** Determining how retrieved knowledge should be incorporated into the generation process, balancing factual accuracy with natural language fluency [28].
- **Security and Robustness:** Preventing adversarial attacks and misinformation propagation by ensuring retrieved knowledge is trustworthy.

These challenges have led to extensive research into retrieval-based architectures, search indexing techniques, and adaptive learning mechanisms that enhance the effectiveness of RAG [29].

2.5. Summary

This section has outlined the theoretical foundations of retrieval-augmented generation, covering traditional IR methods, the evolution of language models, and the motivations behind knowledge augmentation. As RAG continues to develop, advancements in retrieval efficiency, knowledge fusion, and model interpretability will play a crucial role in shaping the next generation of intelligent language systems. In the following sections, we delve deeper into the architectures and methodologies that define RAG, exploring state-of-the-art approaches and their real-world applications [30].

3. Architectures and Methodologies of Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) has emerged as a powerful paradigm for enhancing the capabilities of large language models (LLMs) by integrating external knowledge retrieval with text generation [31]. The effectiveness of RAG models hinges on the interplay between two key components: (1) the retrieval mechanism, which selects relevant documents or passages based on an input query, and (2) the generative model, which synthesizes coherent and informative text grounded

 doi:10.20944/preprints202504.0443.v1

in the retrieved knowledge. This section provides an in-depth exploration of the various architectural designs, retrieval methodologies, and generation strategies that define modern RAG systems [32].

3.1. General Framework of Retrieval-Augmented Generation

Formally, given an input query q, a RAG model retrieves a set of relevant documents $\mathcal{D} = \{d_1, d_2, \ldots, d_k\}$ from an external knowledge source $\mathcal{K}$, which may be a structured database, a vectorized document corpus, or a web-based search engine. The retrieved documents are then processed alongside q to generate an output response r, typically formulated as:

$$r = G(q, \mathcal{D}; \theta_G), \tag{1}$$

where G is the generative model parameterized by θ_G [33]. The quality of r depends on the accuracy of retrieval and the model's ability to effectively integrate external information [34]. The retrieval process itself can be expressed as a mapping function:

$$\mathcal{D} = R(q; \theta_R), \tag{2}$$

where R represents the retrieval model with parameters θ_R. The efficiency and relevance of retrieval directly influence the overall performance of the RAG system [35].

3.2. Retrieval Mechanisms in RAG

Retrieval in RAG systems typically follows one of two primary paradigms: sparse retrieval and dense retrieval [36].

3.2.1. Sparse Retrieval

Sparse retrieval techniques are based on traditional lexical matching methods that score documents using word occurrence statistics. The most well-known approach in this category is BM25, which ranks documents based on term frequency (TF) and inverse document frequency (IDF). The BM25 scoring function is given by:

$$\text{score}(q, d) = \sum_{t \in q} \text{IDF}(t) \cdot \frac{\text{TF}(t, d) \cdot (k_1 + 1)}{\text{TF}(t, d) + k_1 (1 - b + b \cdot \frac{|d|}{\text{avgdl}})}, \tag{3}$$

where: - t is a query term, - $\text{TF}(t, d)$ represents the term frequency of t in document d, - $\text{IDF}(t)$ is the inverse document frequency of t, - k_1 and b are hyperparameters controlling term saturation and document length normalization. Sparse retrieval methods, while interpretable and efficient, often fail to capture semantic relationships beyond surface-level keyword overlap [37].

3.2.2. Dense Retrieval

Dense retrieval employs neural networks to encode both queries and documents into high-dimensional embeddings, allowing for more sophisticated similarity matching [38]. A common approach is to use a dual-encoder framework:

$$\mathbf{q} = f_{\theta_q}(q), \quad \mathbf{d} = f_{\theta_d}(d), \tag{4}$$

where f_{θ_q} and f_{θ_d} are neural encoders (e.g., BERT-based models) that map queries and documents to dense vector representations. The similarity score between a query and a document is computed as:

$$S(q, d) = \mathbf{q}^\top \mathbf{d}, \tag{5}$$

 doi:10.20944/preprints202504.0443.v1

where $S(q, d)$ represents the dot product or cosine similarity between the vectors [39]. To improve retrieval effectiveness, contrastive learning techniques such as DPR (Dense Passage Retrieval) are employed. The loss function for training DPR is typically a contrastive loss:

$$\mathcal{L} = -\log \frac{\exp(S(q, d^+))}{\sum_{d^- \in \mathcal{N}} \exp(S(q, d^-))}, \tag{6}$$

where: - d^+ is the relevant (positive) document, - $\mathcal{N}$ is a set of negative documents. Dense retrieval significantly enhances recall in RAG systems by capturing semantic similarity rather than relying solely on exact word matches [40].

3.3. Fusion Strategies for Retrieval and Generation

Once relevant documents have been retrieved, the challenge is how to effectively incorporate this information into the generative process. Several strategies exist for integrating retrieval results into text generation [41].

3.3.1. Concatenation-Based Fusion

The simplest method involves concatenating the retrieved text $\mathcal{D}$ with the query q and passing it as input to the LLM:

$$x = [q; d_1; d_2; \ldots; d_k]. \tag{7}$$

The model then generates an output r based on this extended input. While straightforward, this approach has limitations, particularly when dealing with long retrieved documents, as LLMs have a fixed input length.

3.3.2. Attention-Based Fusion

An alternative method leverages attention mechanisms to selectively integrate relevant information from retrieved documents [42,43]. In transformer-based models, attention weights α_i can be assigned to different retrieved documents:

$$\alpha_i = \frac{\exp(\mathbf{q}^\top \mathbf{d_i})}{\sum_{j=1}^{k} \exp(\mathbf{q}^\top \mathbf{d_j})} [44]. \tag{8}$$

The final document representation is then computed as a weighted sum:

$$\mathbf{D} = \sum_{i=1}^{k} \alpha_i \mathbf{d_i} [45]. \tag{9}$$

This approach ensures that the most relevant retrieved content is emphasized in the generation process [46].

3.3.3. Reinforcement Learning-Based Fusion

Some advanced RAG models employ reinforcement learning (RL) to optimize retrieval integration. The objective is to maximize a reward function R that evaluates the factual accuracy and coherence of the generated text:

$$\max_{\theta_G, \theta_R} \mathbb{E}[R(G(q, R(q; \theta_R); \theta_G))]. \tag{10}$$

RL-based approaches enable the model to iteratively refine retrieval and generation strategies, improving long-term performance.

 doi:10.20944/preprints202504.0443.v1

3.4. Trade-offs and Challenges

While RAG offers significant advantages in factual accuracy and knowledge grounding, it also introduces several challenges:

- **Latency:** Retrieving documents in real-time can introduce delays, necessitating efficient indexing and retrieval mechanisms [47].
- **Noisy Retrieval:** Retrieved documents may contain irrelevant or conflicting information, requiring robust filtering strategies [48].
- **Knowledge Conflicts:** When retrieved knowledge contradicts the LLM's internal knowledge, resolving inconsistencies remains a complex problem.
- **Scalability:** Maintaining large-scale knowledge bases for retrieval without excessive computational overhead is an ongoing research challenge [49].

3.5. Summary

This section has outlined the key architectural components and methodologies underlying RAG, emphasizing retrieval mechanisms, fusion strategies, and optimization techniques [50]. As RAG continues to evolve, future innovations will likely focus on enhancing retrieval efficiency, improving interpretability, and mitigating challenges related to latency and noisy retrieval [51]. The next section explores practical applications of RAG across various domains, illustrating its impact in real-world scenarios.

4. Applications of Retrieval-Augmented Generation

The integration of retrieval mechanisms with large language models (LLMs) has significantly expanded the range of applications for AI-powered systems [**?**]. Retrieval-Augmented Generation (RAG) enables models to access up-to-date, domain-specific, and contextually relevant knowledge, improving accuracy, reliability, and transparency in various tasks. In this section, we explore the key application areas of RAG, highlighting how this paradigm enhances performance in real-world scenarios.

4.1. Question Answering Systems

One of the most prominent applications of RAG is in open-domain and domain-specific question answering (QA) [52]. Traditional QA systems rely either on parametric knowledge encoded within a language model or on explicit retrieval-based approaches [53]. RAG offers a hybrid solution by retrieving relevant documents at inference time and generating well-grounded responses [54].

4.1.1. Open-Domain Question Answering

In open-domain QA, the model must answer general knowledge questions that span a vast range of topics. Classical approaches such as Dense Passage Retrieval (DPR) [**?**] use dense embeddings to retrieve relevant passages from large corpora such as Wikipedia before generating answers. RAG-based systems have demonstrated superior performance in this setting by retrieving multiple relevant documents and synthesizing accurate responses [55]. Formally, given a question q, the retrieval module identifies a set of relevant documents $\mathcal{D} = \{d_1, d_2, ..., d_k\}$, which are then used by the generative model to produce an answer:

$$r = G(q, \mathcal{D}; \theta_G). \tag{11}$$

The effectiveness of open-domain QA systems depends on the quality of the retrieval process [56]. Hybrid retrieval techniques that combine BM25 and dense embeddings have been explored to improve recall, ensuring that the retrieved documents are both lexically and semantically relevant.

 doi:10.20944/preprints202504.0443.v1

4.1.2. Domain-Specific Question Answering

In specialized fields such as medicine, law, and finance, RAG provides an efficient way to incorporate domain knowledge. Traditional LLMs trained on general corpora often lack the depth required for expert-level queries. RAG mitigates this issue by retrieving information from structured databases, legal documents, or scientific literature. For instance, in biomedical QA, RAG models can retrieve research papers from PubMed and synthesize evidence-based responses. The system ensures that medical advice remains grounded in authoritative sources, reducing the risk of misinformation.

4.2. Conversational Agents and Chatbots

Conversational AI has been transformed by RAG, particularly in applications where accuracy and context-awareness are crucial. Unlike traditional chatbots that rely solely on predefined scripts or parametric memory, RAG-based conversational agents dynamically retrieve external knowledge to enhance their responses [57].

4.2.1. Personalized Assistants

RAG enables virtual assistants to provide real-time, personalized responses by incorporating user-specific data [58]. Given a user query q, the system retrieves relevant past interactions, documentation, or contextual knowledge $\mathcal{D}$, allowing for continuity and personalization in dialogue:

$$r = G(q, R(q; \theta_R); \theta_G)[59]. \tag{12}$$

For example, an AI-powered legal assistant could retrieve relevant case laws or contracts based on a client's query, ensuring that responses are tailored to the specific legal context.

4.2.2. Customer Support Systems

Customer service chatbots powered by RAG can dynamically retrieve policy documents, FAQs, and past resolutions to provide accurate support. Unlike traditional models, which may generate generic or hallucinated responses, RAG ensures that information is retrieved from the latest company databases, improving reliability [60].

4.3. Document Summarization and Knowledge Synthesis

Summarization tasks benefit significantly from retrieval-augmented approaches, particularly when synthesizing information from multiple sources [61]. RAG can be used for:

- **Multi-Document Summarization:** Retrieving multiple related documents and generating a coherent summary [62].
- **Scientific Literature Reviews:** Extracting key findings from multiple research papers.
- **Legal Document Analysis:** Summarizing case laws, contracts, and regulations based on retrieved precedents.

Given a query q (e.g., "Summarize recent advancements in quantum computing"), the system retrieves relevant papers $\mathcal{D}$ and generates a condensed summary:

$$S = G(q, \mathcal{D}; \theta_G), \tag{13}$$

where S represents the generated summary [63].

4.4. Code Generation and Software Development

RAG has demonstrated remarkable potential in code generation and software engineering by retrieving relevant code snippets, documentation, and Stack Overflow discussions to guide AI-generated solutions.

 doi:10.20944/preprints202504.0443.v1

4.4.1. Code Completion and Debugging

In programming environments, RAG-based models retrieve relevant documentation and code examples based on a user's query [64]. This helps in:

- Autocompleting partially written code.
- Suggesting optimized implementations.
- Identifying potential bugs by referencing previous issues [65].

Given a code snippet C and a query q (e.g., "How do I implement a binary search tree in Python?"), the retrieval module identifies relevant GitHub repositories or technical articles, and the generative model refines the response:

$$C' = G(q, R(q; \theta_R); \theta_G), \tag{14}$$

where C' is the generated or improved code snippet [66].

4.4.2. API and Library Recommendations

RAG-powered coding assistants can retrieve API documentation and suggest optimal libraries for specific tasks [67]. For instance, if a user asks, "What is the best way to parse JSON in Python?" the system retrieves and compares multiple solutions from the official Python documentation, Stack Overflow, and GitHub repositories [68].

4.5. Biomedical Applications

The biomedical domain benefits immensely from RAG, particularly in evidence-based medicine, drug discovery, and clinical decision support.

4.5.1. Clinical Decision Support

RAG enhances clinical decision-making by retrieving patient records, medical guidelines, and scientific literature. A query such as "What are the latest treatments for Type 2 Diabetes?" would trigger a retrieval process that pulls from medical journals and official health organization recommendations, ensuring the response is backed by credible sources.

4.5.2. Drug Discovery and Literature Mining

Pharmaceutical companies use RAG to analyze large-scale biomedical literature for drug repurposing and new treatment discovery. Given a chemical compound or disease query, the system retrieves and synthesizes relevant studies, accelerating research workflows.

4.6. Legal and Compliance Analysis

The legal sector increasingly relies on RAG for case law retrieval, contract analysis, and compliance auditing.

4.6.1. Legal Case Analysis

Legal professionals use RAG-powered systems to retrieve relevant case precedents and summarize legal arguments. Given a query q describing a legal dispute, the model retrieves similar cases $\mathcal{D}$ and generates a summary of applicable laws and rulings.

4.6.2. Regulatory Compliance

Companies use RAG to monitor regulatory changes by retrieving and analyzing updated legal texts. Compliance officers can ask, "What are the latest GDPR guidelines?" and receive a response based on retrieved legislative documents.

 doi:10.20944/preprints202504.0443.v1

4.7. Fake News Detection and Misinformation Mitigation

One of the major concerns with LLMs is their susceptibility to generating hallucinated or misleading information [69]. RAG provides a potential solution by verifying facts against trusted sources.

4.7.1. Fact-Checking Systems

Fact-checking organizations employ RAG to retrieve verified news articles and compare them against claims made in online content. Given a claim c, the system retrieves authoritative sources $\mathcal{D}$ and generates a veracity assessment:

$$\text{truthfulness}(c) = G(c, \mathcal{D}; \theta_G)[70]. \tag{15}$$

4.8. Summary

The applications of RAG span a diverse range of domains, from question answering and conversational AI to biomedical research and legal analysis. By enabling models to retrieve external knowledge dynamically, RAG enhances factual accuracy, domain adaptability, and transparency. In the next section, we examine the key challenges and future research directions in the development of RAG-based systems.

5. Challenges and Future Research Directions

Despite the significant advantages of Retrieval-Augmented Generation (RAG) in improving factual accuracy, knowledge retrieval, and adaptability, several challenges remain that hinder its widespread adoption and optimal performance. These challenges arise from limitations in retrieval accuracy, integration with generative models, efficiency, scalability, and security [71]. In this section, we explore the key challenges facing RAG-based systems and outline potential research directions to address them.

5.1. Challenges in Retrieval-Augmented Generation

5.1.1. Retrieval Quality and Relevance

The effectiveness of RAG models is heavily dependent on the quality of retrieved documents [72]. Poor retrieval can lead to:

- **Irrelevant Retrieval:** Retrieved documents may not contain the information necessary to answer the query accurately, leading to incorrect or unhelpful responses [73].
- **Misinformation and Conflicting Sources:** If the retrieved documents contain conflicting or unreliable information, the generative model may produce misleading outputs.
- **Noisy Retrieval:** Some retrieval methods, particularly those based on dense embeddings, may return semantically related but contextually irrelevant documents.

Potential Solutions:

- Enhancing hybrid retrieval techniques that combine sparse (e.g., BM25) and dense (e.g., DPR) retrieval for improved recall and precision.
- Incorporating retrieval reranking models, such as cross-encoders, that refine initial retrieval results before passing them to the generative model [74].
- Developing retrieval-augmented contrastive learning techniques to better differentiate between highly relevant and marginally related documents.

5.1.2. Fusion of Retrieved Knowledge and Text Generation

Effectively integrating retrieved documents into the generative process remains a major challenge. Some common issues include:

 doi:10.20944/preprints202504.0443.v1

- **Over-Reliance on Parametric Knowledge:** Generative models sometimes ignore retrieved information and rely on their internal knowledge, leading to outdated or hallucinated responses.
- **Fragmented Knowledge Integration:** When multiple documents are retrieved, models may struggle to synthesize information coherently, leading to contradictory or disjointed responses [75].
- **Loss of Granular Context:** Retrieved passages may contain useful details that are overlooked when aggregated or truncated for input into the generative model.

Potential Solutions:

- Developing **adaptive attention mechanisms** that dynamically weigh retrieved evidence based on relevance scores [76].
- Implementing **memory-augmented architectures** that store and refine retrieved information over multiple turns in dialogue-based applications.
- Exploring **multi-step reasoning frameworks** that enable models to iteratively refine their responses based on retrieved content [77].

5.1.3. Efficiency and Latency

RAG models introduce additional computational overhead compared to standalone generative models due to the retrieval step [78,79]. This leads to increased inference time, especially when retrieving from large knowledge bases.

$$T_{\text{total}} = T_{\text{retrieve}} + T_{\text{generate}}, \tag{16}$$

where T_{retrieve} represents retrieval latency, and T_{generate} corresponds to text generation time. If T_{retrieve} is significantly high, real-time applications such as chatbots or voice assistants may suffer from slow response times [80]. **Potential Solutions:**

- Utilizing **approximate nearest neighbor (ANN)** search techniques to accelerate retrieval while maintaining high recall.
- Exploring **caching mechanisms** that store frequently accessed documents for rapid retrieval.
- Investigating **knowledge distillation techniques** to precompute and embed commonly retrieved information, reducing the need for real-time document lookup.

5.1.4. Scalability Issues in Large Knowledge Bases

As RAG systems scale to handle ever-growing corpora, several scalability concerns arise:

- **Indexing Complexity:** Maintaining efficient indices for retrieval across vast document collections is computationally expensive.
- **Storage Constraints:** Storing large-scale dense embeddings requires significant memory resources.
- **Incremental Updates:** Updating knowledge bases dynamically without requiring complete re-indexing remains an open problem [81].

Potential Solutions:

- Developing **hierarchical indexing** methods that allow efficient retrieval at multiple levels of granularity [82].
- Leveraging **vector compression techniques** to reduce the memory footprint of large-scale embeddings [83].
- Designing **streaming knowledge retrieval** frameworks that support continuous updates without full re-indexing.

 doi:10.20944/preprints202504.0443.v1

5.1.5. Security, Bias, and Ethical Concerns

RAG models inherit ethical and security challenges associated with both retrieval and generation components:

- **Exposure of Sensitive Information:** If the knowledge base contains private or proprietary data, retrieved documents may inadvertently expose sensitive content.
- **Bias in Retrieved Content:** The retrieval model may reinforce societal biases if the training data contains skewed or imbalanced representations.
- **Adversarial Attacks on Retrieval:** Malicious actors could manipulate retrieval results by poisoning indexed documents or injecting misleading content.

Potential Solutions:

- Implementing **differential privacy techniques** to prevent leakage of sensitive information [84].
- Introducing **bias mitigation frameworks** that assess and balance retrieval outputs before text generation [85].
- Developing **robust adversarial detection mechanisms** to identify and filter manipulated knowledge sources.

5.2. Future Research Directions

Given the aforementioned challenges, several promising research avenues could further advance RAG models:

5.2.1. Neural-Symbolic Hybrid Models

A growing research direction is integrating neural models with symbolic reasoning frameworks. Neural-symbolic RAG models could:

- Leverage structured knowledge bases (e.g., Wikidata, knowledge graphs) alongside traditional document retrieval.
- Employ logic-based reasoning modules to validate generated responses [86].
- Enhance interpretability by explicitly linking outputs to retrieved knowledge sources.

5.2.2. Continual Learning and Adaptive Retrieval

Current RAG systems rely on static knowledge bases that require periodic updates [87]. Future research could explore:

- **Lifelong learning approaches** that allow models to dynamically update retrieval strategies based on new information [88].
- **Self-improving retrieval mechanisms** that refine document selection based on user feedback.
- **Incremental learning pipelines** that enable RAG systems to incorporate real-time knowledge without full retraining [89].

5.2.3. Multimodal RAG Models

Future iterations of RAG could extend beyond textual retrieval to incorporate multimodal sources such as:

- **Image and Video Retrieval:** Integrating visual knowledge for applications in medical imaging, forensic analysis, and autonomous systems.
- **Audio and Speech Retrieval:** Enhancing conversational AI with audio-based search capabilities.
- **Cross-Modal Retrieval Fusion:** Developing architectures that combine text, images, and structured data for richer knowledge grounding [90].

 doi:10.20944/preprints202504.0443.v1

5.2.4. Explainability and Interpretability in RAG

To improve trust and adoption, future RAG models should provide:

- **Attribution mechanisms** that explicitly cite retrieved sources in generated responses [91].
- **Interactive explanations** that allow users to inspect and refine retrieved knowledge [92].
- **Transparent scoring models** that highlight confidence levels in retrieved evidence [93].

5.3. Summary

Retrieval-Augmented Generation represents a powerful shift in AI-driven knowledge synthesis, but several challenges remain in optimizing retrieval accuracy, computational efficiency, security, and interpretability. Addressing these limitations will require interdisciplinary efforts spanning machine learning, information retrieval, and ethical AI [94]. As future research progresses, RAG systems are expected to become more reliable, scalable, and adaptive, paving the way for next-generation AI applications [95].

6. Conclusion

Retrieval-Augmented Generation (RAG) has emerged as a transformative paradigm for enhancing the capabilities of large language models (LLMs) by integrating external knowledge retrieval with generative text modeling. This survey has provided a comprehensive exploration of RAG, covering its fundamental concepts, methodologies, architectures, applications, challenges, and future research directions. In this concluding section, we summarize the key takeaways, highlight the broader implications of RAG, and discuss its long-term potential.

6.1. Summary of Key Insights

The introduction of retrieval mechanisms into generative models has addressed several limitations of purely parametric language models, particularly in terms of knowledge freshness, factual accuracy, and scalability. The major insights from this survey can be summarized as follows:

- **Hybrid Approach for Enhanced Performance:** RAG leverages the strengths of both retrieval-based and generation-based models. It dynamically retrieves relevant documents at inference time and uses them to condition text generation, leading to more informed and contextually relevant responses.
- **Improved Accuracy and Knowledge Grounding:** By incorporating external knowledge sources, RAG mitigates hallucination issues commonly observed in LLMs. The generated text is more reliable, explainable, and directly linked to retrieved evidence.
- **Broad Applicability Across Domains:** RAG has demonstrated significant benefits across multiple applications, including open-domain question answering, scientific literature summarization, conversational AI, legal document analysis, and biomedical research.
- **Scalability and Efficiency Challenges:** Despite its advantages, RAG introduces additional computational complexity due to the retrieval process. Efficient indexing, caching, and hybrid search methods are crucial for optimizing performance.
- **Ethical Considerations and Robustness:** Issues such as retrieval bias, exposure of sensitive data, and adversarial manipulation remain open problems. Future research should focus on building secure, transparent, and fair RAG-based systems.

6.2. The Broader Impact of RAG

The evolution of RAG has broader implications beyond specific applications, influencing several key aspects of artificial intelligence research and deployment:

 doi:10.20944/preprints202504.0443.v1

6.2.1. Democratization of Knowledge

By integrating retrieval mechanisms, RAG makes it possible to access and utilize vast amounts of external knowledge in real time. This significantly enhances accessibility to up-to-date and diverse sources of information, democratizing AI-driven knowledge dissemination across different domains.

6.2.2. Towards Explainable AI

One of the key challenges in deploying large language models is their lack of interpretability. RAG introduces an element of transparency by grounding responses in retrieved documents, making it easier to track the source of generated information. This aligns with ongoing efforts in explainable AI (XAI), improving trust and accountability in automated decision-making systems.

6.2.3. Bridging Structured and Unstructured Data

Traditional knowledge retrieval approaches rely on structured databases or knowledge graphs, while modern LLMs generate text from unstructured parametric knowledge. RAG bridges this gap by allowing models to dynamically retrieve structured, semi-structured, and unstructured information, opening up new possibilities for hybrid AI architectures that integrate databases, APIs, and real-time web search.

6.3. Future of Retrieval-Augmented Generation

Looking ahead, several key research directions and technological advancements could shape the future of RAG:

- **Neural-Symbolic Integration:** Combining symbolic reasoning with neural retrieval mechanisms could lead to more robust and interpretable RAG systems that leverage structured knowledge bases alongside free-text retrieval.
- **Efficient and Scalable Retrieval:** The development of advanced indexing techniques, approximate nearest neighbor (ANN) search, and knowledge compression strategies will be critical for making retrieval more efficient, enabling real-time applications.
- **Continual Learning and Adaptive Retrieval:** Future RAG models could evolve dynamically by learning from user interactions, updating retrieval strategies based on feedback, and incorporating the latest knowledge without requiring frequent retraining.
- **Multimodal and Cross-Domain RAG:** Expanding RAG beyond text-based retrieval to incorporate images, videos, structured data, and multimodal inputs could unlock novel applications in areas such as autonomous systems, education, and healthcare.
- **Ethical AI and Bias Mitigation:** Ensuring fairness and reducing biases in retrieved documents remains a critical research challenge. Developing adversarially robust retrieval methods and fairness-aware ranking algorithms will be essential for building responsible AI systems.

6.4. Final Remarks

Retrieval-Augmented Generation represents a significant leap forward in AI's ability to generate knowledge-grounded responses, bridging the gap between retrieval-based and generative approaches. As research in this field progresses, RAG is poised to become a fundamental component of next-generation AI systems, enabling more accurate, reliable, and interpretable text generation.

The future of RAG is not merely an enhancement of current models but a foundational shift toward AI systems that dynamically retrieve, synthesize, and generate knowledge in an adaptive and explainable manner. By addressing the challenges and leveraging emerging advancements, RAG has the potential to redefine how AI interacts with information, making artificial intelligence systems more aligned with human reasoning and decision-making.

 doi:10.20944/preprints202504.0443.v1

1. ILIN, I. Advanced RAG Techniques: an Illustrated Overview. https://pub.towardsai.net/advanced-rag-techniques-an-illustrated-overview-04d193d8fec6, 2023.
2. Husain, H.; Wu, H.H.; Gazit, T.; Allamanis, M.; Brockschmidt, M. Codesearchnet challenge: Evaluating the state of semantic code search. *arXiv preprint arXiv:1909.09436* **2019**.
3. Gottlob, G.; Leone, N.; Scarcello, F. Hypertree Decompositions and Tractable Queries. *Journal of Computer and System Sciences* **2002**, *64*, 579–627.
4. Berant, J.; Chou, A.K.; Frostig, R.; Liang, P. Semantic Parsing on Freebase from Question-Answer Pairs. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013.
5. Bang, Y.; Cahyawijaya, S.; Lee, N.; Dai, W.; Su, D.; Wilie, B.; Lovenia, H.; Ji, Z.; Yu, T.; Chung, W.; et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023* **2023**.
6. Wang, Y.; Lipka, N.; Rossi, R.A.; Siu, A.; Zhang, R.; Derr, T. Knowledge graph prompting for multi-document question answering. *arXiv preprint arXiv:2308.11730* **2023**.
7. Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* **2022**.
8. Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; tau Yih, W. Dense Passage Retrieval for Open-Domain Question Answering, 2020, [arXiv:cs.CL/2004.04906].
9. Saha, S.; Junaed, J.A.; Saleki, M.; Sharma, A.S.; Rifat, M.R.; Rahouti, M.; Ahmed, S.I.; Mohammed, N.; Amin, M.R. Vio-lens: A novel dataset of annotated social network posts leading to different forms of communal violence and its evaluation. In Proceedings of the Proceedings of the First Workshop on Bangla Language Processing (BLP-2023), 2023, pp. 72–84.
10. Levesque, H.J. A logic of implicit and explicit belief. In Proceedings of the Proceedings of the Fourth National Conference on Artificial Intelligence, Austin, Texas, August 1984; pp. 198–202.
11. Saha, A.; Pahuja, V.; Khapra, M.M.; Sankaranarayanan, K.; Chandar, S. Complex Sequential Question Answering: Towards Learning to Converse Over Linked Question Answer Pairs with a Knowledge Graph **2018**. [arXiv:1801.10314].
12. NVIDIA. Spectrum-X: End-to-End Networking for AI and High-Performance Computing. https://www.nvidia.com/en-us/networking/spectrumx/, 2025. Accessed: 2025-01-28.
13. Han, Y.; Liu, C.; Wang, P. A Comprehensive Survey on Vector Database: Storage and Retrieval Technique, Challenge, 2023, [arXiv:cs.DB/2310.11703].
14. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A Survey of Large Language Models, 2024, [arXiv:cs.CL/2303.18223].
15. Kim, G.; Kim, S.; Jeon, B.; Park, J.; Kang, J. Tree of Clarifications: Answering Ambiguous Questions with Retrieval-Augmented Large Language Models. *arXiv preprint arXiv:2310.14696* **2023**.
16. Welbl, J.; Stenetorp, P.; et al. 2WikiMultiHopQA: Multihop Question Answering over Wikipedia Articles. *EMNLP* **2018**.
17. LangChain. LangSmith: The Ultimate Toolkit for Debugging and Monitoring LLM Applications. https://www.langchain.com/langsmith, 2025. Accessed: 2025-01-28.
18. Cohere. Say Goodbye to Irrelevant Search Results: Cohere Rerank Is Here. https://txt.cohere.com/rerank/, 2023.
19. Wang, L.; Yang, N.; Wei, F. Query2doc: Query Expansion with Large Language Models. *arXiv preprint arXiv:2303.07678* **2023**.
20. Yan, S.Q.; Gu, J.C.; Zhu, Y.; Ling, Z.H. Corrective Retrieval Augmented Generation, 2024, [arXiv:cs.CL/2401.15884].
21. Xu, F.; Shi, W.; Choi, E. RECOMP: Improving Retrieval-Augmented LMs with Compression and Selective Augmentation. *arXiv preprint arXiv:2310.04408* **2023**.
22. An, Z.; Ding, X.; Fu, Y.C.; Chu, C.C.; Li, Y.; Du, W. Golden-Retriever: High-Fidelity Agentic Retrieval Augmented Generation for Industrial Knowledge Base, 2024, [arXiv:cs.IR/2408.00798].
23. Elsahar, H.; Vougiouklis, P.; Remaci, A.; Gravier, C.; Hare, J.; Laforest, F.; Simperl, E. T-rex: A large scale alignment of natural language with knowledge base triples. In Proceedings of the Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018.

 doi:10.20944/preprints202504.0443.v1

24. Blagojevi, V. Enhancing RAG Pipelines in Haystack: Introducing DiversityRanker and LostInTheMiddleRanker. https://towardsdatascience.com/enhancing-rag-pipelines-in-haystack-45f14e2bc9f5, 2023.
25. LangChain. LangGraph Workflows Tutorial, 2025. https://langchain-ai.github.io/langgraph/tutorials/workflows/. Accessed: February 2, 2025.
26. Lee, M.C.; Zhu, Q.; Mavromatis, C.; Han, Z.; Adeshina, S.; Ioannidis, V.N.; Rangwala, H.; Faloutsos, C. Agent-G: An Agentic Framework for Graph Retrieval Augmented Generation, 2024.
27. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; tau Yih, W.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, 2021, [arXiv:cs.CL/2005.11401].
28. Nebel, B. On the compilability and expressive power of propositional planning formalisms. *Journal of Artificial Intelligence Research* **2000**, *12*, 271–315.
29. Saha, S.; Junaed, J.A.; Saleki, M.; Sen Sharma, A.; Rifat, M.R.; Rahouti, M.; Ahmed, S.I.; Mohammed, N.; Amin, M.R. Vio-Lens: A Novel Dataset of Annotated Social Network Posts Leading to Different Forms of Communal Violence and its Evaluation. In Proceedings of the Proceedings of the First Workshop on Bangla Language Processing (BLP-2023); Alam, F.; Kar, S.; Chowdhury, S.A.; Sadeque, F.; Amin, R., Eds., Singapore, 2023; pp. 72–84. https://doi.org/10.18653/v1/2023.banglalp-1.9.
30. Berchansky, M.; Izsak, P.; Caciularu, A.; Dagan, I.; Wasserblat, M. Optimizing Retrieval-augmented Reader Models via Token Elimination. *arXiv preprint arXiv:2310.13682* **2023**.
31. Sciavolino, C.; Zhong, Z.; Lee, J.; Chen, D. Simple entity-centric questions challenge dense retrievers. *arXiv preprint arXiv:2109.08535* **2021**.
32. Dasigi, P.; Lo, K.; Beltagy, I.; Cohan, A.; Smith, N.A.; Gardner, M. A dataset of information-seeking questions and answers anchored in research papers. *arXiv preprint arXiv:2105.03011* **2021**.
33. He, R.; McAuley, J. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In Proceedings of the proceedings of the 25th international conference on world wide web, 2016, pp. 507–517.
34. Ram, O.; Levine, Y.; Dalmedigos, I.; Muhlgay, D.; Shashua, A.; Leyton-Brown, K.; Shoham, Y. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083* **2023**.
35. Du, X.; Ji, H. Retrieval-Augmented Generative Question Answering for Event Argument Extraction. *arXiv preprint arXiv:2211.07067* **2022**.
36. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems* **2020**, *33*, 9459–9474.
37. LlamaIndex. Introducing Agentic Document Workflows. https://www.llamaindex.ai/blog/introducing-agentic-document-workflows, 2025. Accessed: 2025-01-13.
38. crewAI Inc.. crewAI: A GitHub Repository for AI Projects. https://github.com/crewAIInc/crewAI, 2025. Accessed: 2025-01-15.
39. Singh, A.; Kumar, S.; Ehtesham, A.; Khoei, T.T.; Bhati, D. Large Language Model-Driven Immersive Agent. In Proceedings of the 2024 IEEE World AI IoT Congress (AIIoT), 2024, pp. 0619–0624. https://doi.org/10.1109/AIIoT61789.2024.10578948.
40. Clark, C.; Lee, K.; Chang, M.W.; Kwiatkowski, T.; Collins, M.; Toutanova, K. BoolQ: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044* **2019**.
41. Xu, X.; Gou, Z.; Wu, W.; Niu, Z.Y.; Wu, H.; Wang, H.; Wang, S. Long time no see! open-domain conversation with long-term persona memory. *arXiv preprint arXiv:2203.05797* **2022**.
42. Xiao, G.; Tian, Y.; Chen, B.; Han, S.; Lewis, M. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453* **2023**.
43. Zniyed, Y.; Nguyen, T.P.; et al. Enhanced network compression through tensor decompositions and pruning. *IEEE Transactions on Neural Networks and Learning Systems* **2024**.
44. Leng, Q.; Uhlenhuth, K.; Polyzotis, A. Best Practices for LLM Evaluation of RAG Applications. https://www.databricks.com/blog/LLM-auto-eval-best-practices-RAG, 2023.
45. Steinberger, R.; Pouliquen, B.; Widiger, A.; Ignat, C.; Erjavec, T.; Tufis, D.; Varga, D. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. *arXiv preprint cs/0609058* **2006**.

 doi:10.20944/preprints202504.0443.v1

46. Hoshi, Y.; Miyashita, D.; Ng, Y.; Tatsuno, K.; Morioka, Y.; Torii, O.; Deguchi, J. RaLLe: A Framework for Developing and Evaluating Retrieval-Augmented Large Language Models. *arXiv preprint arXiv:2308.10633* **2023**.
47. Nguyen, I. Evaluating RAG Part I: How to Evaluate Document Retrieval. https://www.deepset.ai/blog/rag-evaluation-retrieval, 2023.
48. Ren, R.; Wang, Y.; Qu, Y.; Zhao, W.X.; Liu, J.; Tian, H.; Wu, H.; Wen, J.R.; Wang, H. Investigating the factual knowledge boundary of large language models with retrieval augmentation. *arXiv preprint arXiv:2307.11019* **2023**.
49. Rau, D.; Déjean, H.; Chirkova, N.; Formal, T.; Wang, S.; Nikoulina, V.; Clinchant, S. BERGEN: A Benchmarking Library for Retrieval-Augmented Generation, 2024, [arXiv:cs.CL/2407.01102].
50. Kandpal, N.; Deng, H.; Roberts, A.; Wallace, E.; Raffel, C. Large language models struggle to learn long-tail knowledge. In Proceedings of the International Conference on Machine Learning. PMLR, 2023, pp. 15696–15707.
51. Mallen, A.; Asai, A.; Zhong, V.; Das, R.; Hajishirzi, H.; Khashabi, D. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511* **2022**.
52. Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; et al. Self-Refine: Iterative Refinement with Self-Feedback, 2023, [arXiv:cs.CL/2303.17651].
53. Huang, J.; Chang, K.C.C. Towards Reasoning in Large Language Models: A Survey, 2023, [arXiv:cs.CL/2212.10403].
54. Saad-Falcon, J.; Khattab, O.; Potts, C.; Zaharia, M. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. *arXiv preprint arXiv:2311.09476* **2023**.
55. Zhang, J. Graph-ToolFormer: To Empower LLMs with Graph Reasoning Ability via Prompt Augmented by ChatGPT. *arXiv preprint arXiv:2304.11116* **2023**.
56. Jiang, X.; Zhang, R.; Xu, Y.; Qiu, R.; Fang, Y.; Wang, Z.; Tang, J.; Ding, H.; Chu, X.; Zhao, J.; et al. Think and Retrieval: A Hypothesis Knowledge Graph Enhanced Medical Large Language Models. *arXiv preprint arXiv:2312.15883* **2023**.
57. Ye, D.; Lin, Y.; Du, J.; Liu, Z.; Li, P.; Sun, M.; Liu, Z. Coreferential reasoning learning for language representation. *arXiv preprint arXiv:2004.06870* **2020**.
58. Zeng, H. Measuring massive multitask chinese understanding. *arXiv preprint arXiv:2304.12986* **2023**.
59. He, X.; Tian, Y.; Sun, Y.; Chawla, N.V.; Laurent, T.; LeCun, Y.; Bresson, X.; Hooi, B. G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *arXiv preprint arXiv:2402.07630* **2024**.
60. Jiang, H.; Wu, Q.; Lin, C.Y.; Yang, Y.; Qiu, L. Llmlingua: Compressing prompts for accelerated inference of large language models. *arXiv preprint arXiv:2310.05736* **2023**.
61. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, M.; Wang, H. Retrieval-Augmented Generation for Large Language Models: A Survey, 2024, [arXiv:cs.CL/2312.10997].
62. Yang, A.; Nagrani, A.; Seo, P.H.; Miech, A.; Pont-Tuset, J.; Laptev, I.; Sivic, J.; Schmid, C. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 10714–10726.
63. Shi, T.; Li, L.; Lin, Z.; Yang, T.; Quan, X.; Wang, Q. Dual-Feedback Knowledge Retrieval for Task-Oriented Dialogue Systems. *arXiv preprint arXiv:2310.14528* **2023**.
64. Mallen, A.; Asai, A.; Zhong, V.; Das, R.; Khashabi, D.; Hajishirzi, H. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Rogers, A.; Boyd-Graber, J.; Okazaki, N., Eds., Toronto, Canada, 2023; pp. 9802–9822. https://doi.org/10.18653/v1/2023.acl-long.546.
65. Xia, M.; Huang, G.; Liu, L.; Shi, S. Graph based translation memory for neural machine translation. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2019, Vol. 33, pp. 7297–7304.
66. Li, X.; Zhao, R.; Chia, Y.K.; Ding, B.; Bing, L.; Joty, S.; Poria, S. Chain of Knowledge: A Framework for Grounding Large Language Models with Structured Knowledge Bases. *arXiv preprint arXiv:2305.13269* **2023**.

 doi:10.20944/preprints202504.0443.v1

67. Bajaj, P.; Campos, D.; Craswell, N.; Deng, L.; Gao, J.; Liu, X.; Majumder, R.; McNamara, A.; Mitra, B.; Nguyen, T.; et al. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset, 2018, [arXiv:cs.CL/1611.09268].
68. Cookbook, H.F. Agentic RAG: Turbocharge Your Retrieval-Augmented Generation with Query Reformulation and Self-Query. https://huggingface.co/learn/cookbook/en/agent_rag. Accessed: 2025-01-14.
69. Seo, M.; Baek, J.; Thorne, J.; Hwang, S.J. Retrieval-Augmented Data Augmentation for Low-Resource Domain Tasks. *arXiv preprint arXiv:2402.13482* **2024**.
70. Ma, Y.; Cao, Y.; Hong, Y.; Sun, A. Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples! *ArXiv* **2023**, *abs/2303.08559*.
71. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems* **2024**. https://doi.org/10.1145/3703155.
72. Asai, A.; Min, S.; Zhong, Z.; Chen, D. Retrieval-based language models and applications. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts), 2023, pp. 41–46.
73. Dasigi, P.; Lo, K.; Beltagy, I.; Cohan, A.; Smith, N.A.; Gardner, M. A Dataset of Information-Seeking Questions and Answers Anchored in Research Papers. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tur, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; Zhou, Y., Eds., Online, 2021; pp. 4599–4610. https://doi.org/10.18653/v1/2021.naacl-main.365.
74. Zheng, H.S.; Mishra, S.; Chen, X.; Cheng, H.T.; Chi, E.H.; Le, Q.V.; Zhou, D. Take a Step Back: Evoking Reasoning via Abstraction in Large Language Models. *arXiv preprint arXiv:2310.06117* **2023**.
75. Yang, S. Advanced RAG 01: Small-to-Big Retrieval. https://towardsdatascience.com/advanced-rag-01-small-to-big-retrieval-172181b396d4, 2023.
76. Pang, R.Y.; Parrish, A.; Joshi, N.; Nangia, N.; Phang, J.; Chen, A.; Padmakumar, V.; Ma, J.; Thompson, J.; He, H.; et al. QuALITY: Question Answering with Long Input Texts, Yes!, 2022, [arXiv:cs.CL/2112.08608].
77. Yan, S.Q.; Gu, J.C.; Zhu, Y.; Ling, Z.H. Corrective Retrieval Augmented Generation, 2024, [arXiv:cs.CL/2401.15884].
78. Wang, X.; Chen, G.H.; Song, D.; Zhang, Z.; Chen, Z.; Xiao, Q.; Jiang, F.; Li, J.; Wan, X.; Wang, B.; et al. CMB: A Comprehensive Medical Benchmark in Chinese, 2024, [arXiv:cs.CL/2308.08833].
79. Zniyed, Y.; Nguyen, T.P.; et al. Efficient tensor decomposition-based filter pruning. *Neural Networks* **2024**, *178*, 106393.
80. Anderson, N.; Wilson, C.; Richardson, S.D. Lingua: Addressing Scenarios for Live Interpretation and Automatic Dubbing. In Proceedings of the Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas (Volume 2: Users and Providers Track and Government Track); Campbell, J.; Larocca, S.; Marciano, J.; Savenkov, K.; Yanishevsky, A., Eds., Orlando, USA, 2022; pp. 202–209.
81. Raudaschl, A.H. Forget RAG, the Future is RAG-Fusion. https://towardsdatascience.com/forget-rag-the-future-is-rag-fusion-1147298d8ad1, 2023.
82. Koehn, P. Europarl: A Parallel Corpus for Statistical Machine Translation. *MT Summit* **2005**.
83. Pang, R.Y.; Parrish, A.; Joshi, N.; Nangia, N.; Phang, J.; Chen, A.; Padmakumar, V.; Ma, J.; Thompson, J.; He, H.; et al. QuALITY: Question answering with long input texts, yes! *arXiv preprint arXiv:2112.08608* **2021**.
84. Lyu, Y.; Li, Z.; Niu, S.; Xiong, F.; Tang, B.; Wang, W.; Wu, H.; Liu, H.; Xu, T.; Chen, E. CRUD-RAG: A comprehensive chinese benchmark for retrieval-augmented generation of large language models. *arXiv preprint arXiv:2401.17043* **2024**.
85. Chen, D.; Yih, W.t. Open-domain question answering. In Proceedings of the Proceedings of the 58th annual meeting of the association for computational linguistics: tutorial abstracts, 2020, pp. 34–37.
86. Li, X.; Nie, E.; Liang, S. From Classification to Generation: Insights into Crosslingual Retrieval Augmented ICL. *arXiv preprint arXiv:2311.06595* **2023**.
87. Kim, S.; Joo, S.J.; Kim, D.; Jang, J.; Ye, S.; Shin, J.; Seo, M. The CoT Collection: Improving Zero-shot and Few-shot Learning of Language Models via Chain-of-Thought Fine-Tuning. *arXiv preprint arXiv:2305.14045* **2023**.

 doi:10.20944/preprints202504.0443.v1

88. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* **2022**, *35*, 27730–27744.
89. Qin, Y.; Cai, Z.; Jin, D.; Yan, L.; Liang, S.; Zhu, K.; Lin, Y.; Han, X.; Ding, N.; Wang, H.; et al. WebCPM: Interactive Web Search for Chinese Long-form Question Answering. *arXiv preprint arXiv:2305.06849* **2023**.
90. Repository, L.D. Contract Review Workflow using LlamaCloud. https://github.com/run-llama/llamacloud-demo/blob/main/examples/document_workflows/contract_review/contract_review.ipynb, 2025. Accessed: 2025-01-13.
91. Minaee, S.; Mikolov, T.; Nikzad, N.; Chenaghlu, M.; Socher, R.; Amatriain, X.; Gao, J. Large Language Models: A Survey, 2024, [arXiv:cs.CL/2402.06196].
92. Community, I.G. Agentic RAG: AI Agents with IBM Granite Models. https://github.com/ibm-granite-community/granite-snack-cookbook/blob/main/recipes/AI-Agents/Agentic_RAG.ipynb. Accessed: 2025-01-14.
93. Lin, X.V.; Chen, X.; Chen, M.; Shi, W.; Lomeli, M.; James, R.; Rodriguez, P.; Kahn, J.; Szilvasy, G.; Lewis, M.; et al. RA-DIT: Retrieval-Augmented Dual Instruction Tuning. *arXiv preprint arXiv:2310.01352* **2023**.
94. Thakur, N.; Bonifacio, L.; Zhang, X.; Ogundepo, O.; Kamalloo, E.; Alfonso-Hermelo, D.; Li, X.; Liu, Q.; Chen, B.; Rezagholizadeh, M.; et al. "Knowing When You Don't Know": A Multilingual Relevance Assessment Dataset for Robust Retrieval-Augmented Generation, 2024, [arXiv:cs.CL/2312.11361].
95. Chan, D.M.; Ghosh, S.; Rastrow, A.; Hoffmeister, B. Using External Off-Policy Speech-To-Text Mappings in Contextual End-To-End Automated Speech Recognition. *arXiv preprint arXiv:2301.02736* **2023**.