Retrieval-Augmented Generation: A Survey of Methodologies, Techniques, Applications, and Future Directions

Charles Z. Liu[a,b,c], Imani Abayakoon[a], Farookh Khadeer Hussain[a,c]

[a]*School of Computer Science, University of Technology Sydney, Sydney, 2007, NSW, Australia*
[b]*SmartSys Workgroup, AlphaNest, Sydney, 2036, NSW, Australia*
[c]*Australian Artificial Intelligence Institute, 15 Broadway, Ultimo, Sydney, 2007, NSW, Australia*

Abstract

Retrieval-augmented generation (RAG) is a hybrid architecture that combines the generative power of large language models (LLMs) with the factual reliability of information retrieval systems. Although the emergence of large language models (LLMs) has significantly improved the performance of natural language understanding and generation tasks. However, these models often suffer from information distortion, outdated information, and lack of transparency. Retrieval-augmented generation (RAG) addresses these limitations by introducing an external retrieval mechanism into the generation process. RAG systems follow the retrieve first, then generate paradigm, which retrieves relevant documents from knowledge sources and uses them as input to the language model. This approach enables the model to generate more accurate, solid, and timely responses. RAG has become an infrastructure for knowledge-intensive natural language processing (NLP) and LLM. In this review, we comprehensively review the basic architecture of RAG systems, analyze key components such as retrievers and generators, compare mainstream implementations, and evaluate their performance on various tasks. We also discuss challenges in the RAG pipeline, including latency, hallucinations, context filtering, and knowledge freshness. Finally, we highlight future research directions in terms of scalability, personalization, and integration with structured knowledge sources.

Keywords: LLM, Retrieval-Augmented Generation, Dense Passage Retrieval, NLP, Fusion-in-Decoder, RAG-Fusion, RAG Feedback Loops, RAG Evaluation Metrics

1. Introduction

1.1. Motivation aginst the Limitations of Language Models

The development of large-scale transformer-based language models such as GPT [1, 2, 3, 4], BERT [5, 6, 7], and T5 [8, 9, 10] has led to substantial progress across a wide range of natural language processing (NLP) tasks. These models leverage vast amounts of unlabeled data to learn intricate patterns of language structure and semantics, enabling them to generate fluent, coherent, and context-aware responses. Despite their linguistic capabilities, such models are inherently constrained by the limitations of their parametric nature [11, 12, 13]. Once trained, their knowledge is statically embedded within their weights, rendering them incapable of accessing or reasoning over dynamic, external, or post-training information sources. Consequently, these models are susceptible to hallucinations, wherein they generate factually incorrect or fabricated statements with high confidence. Moreover, they offer limited explainability, as there is no clear attribution of generated content to specific knowledge sources.

These shortcomings are especially critical in high-stakes domains such as legal reasoning [14, 15], scientific research [16, 17, 18], and clinical decision-making [19, 20, 21], where the verifiability and traceability of information are paramount. The pressing need to extend language models with the ability to reason over current, reliable, and externally maintained knowledge bases gave rise to the paradigm of retrieval-augmented generation (RAG).

1.2. Conceptual Foundations of RAG

Retrieval-Augmented Generation presents a hybrid framework [22, 23, 24]. It decomposes the task of question answering or knowledge-intensive generation into two interdependent stages, document retrieval and conditioned text generation. Unlike conventional LLMs that rely exclusively on internalized knowledge, RAG systems explicitly incorporate evidence from an external corpus during inference. In this architecture, a retriever first identifies relevant documents or passages based on a user query. These retrieved items are then supplied as auxiliary input to a generator, typically a sequence-to-sequence transformer, which produces a grounded response.

This approach reflects a cognitively aligned methodology, closely mimicking how humans synthesize information, by retrieving relevant content and reasoning over it to construct meaningful answers. The introduction of retrieval as an intermediate step not only enhances factual grounding but also enables continuous knowledge updates without the need for retraining or finetuning the generator component. RAG thereby offers a compelling solution to the inherent trade-off between static knowledge representation and generation flexibility.

1.3. Historical Trajectory and Milestones

The conceptual underpinnings of RAG can be traced to earlier advances in open-domain question answering (ODQA, [25, 26, 27]), particularly systems that employed retrieval-then-reading pipelines. Early models such as DrQA [28, 29, 30] utilized sparse term-based retrieval (e.g., TF-IDF [31, 32], [33, 34, 35]) followed by deep neural reading comprehension modules. However, these models were limited by their brittle retrieval quality and lack of integration with generation capabilities.

A pivotal advancement of RAG [22] formalized the integration of dense retrieval and generation within a unified probabilistic framework. In this model, retrieval was treated as a latent variable, and end-to-end training was performed via marginal likelihood estimation over retrieved contexts. Dense Passage Retrieval (DPR [36, 37, 38]) was employed to enable semantic similarity search in embedding space, improving both recall and relevance. The generator, implemented using the BART architecture [39, 40], conditioned on both the query and the retrieved passages, thus achieving superior performance across benchmarks such as Natural Questions [41] and WebQuestions [42].

Subsequent works expanded the RAG paradigm to include multi-document fusion (e.g., Fusion-in-Decoder [43, 44]), feedback-driven retriever refinement (e.g., RAG-Fusion [45, 46]), and multi-hop reasoning across evidence chains. Research also progressed toward modular frameworks such as Haystack [47] and LangChain [48], which enabled scalable, composable, and domain-adaptable RAG pipelines.

1.4. Methodological Advancements and Technical Merits

The architectural modularity of RAG introduces several methodological advantages. By decoupling knowledge access from generation, it enables dynamic context adaptation and fine-grained control over the provenance of information. Unlike monolithic models that are

computationally burdensome to retrain, RAG systems can update their knowledge base in real time through corpus expansion and reindexing, thereby maintaining temporal relevance without parameter modification.

Furthermore, RAG systems enhance explainability by allowing users to inspect retrieved documents that serve as the epistemic basis for generated outputs. This traceability is of particular significance in domains where transparency and accountability are essential. In terms of performance, empirical studies have demonstrated that RAG models outperform purely generative baselines on tasks requiring factual accuracy, domain specificity, and long-context reasoning.

1.5. Limitations and Challenges

Despite advantages, RAG systems face notable challenges. The reliance on a retrieval subsystem introduces latency and computational overhead [49, 50, 51], especially when querying large corpora. Meanwhile, retrieval quality is also critical. If irrelevant or noisy documents are retrieved, the generator may be misled, resulting in diluted or erroneous output [52, 53, 54]. The effective handling of conflicting or ambiguous evidence remains an open problem, as does the optimization of retrieval strategies under resource constraints.

Additionally, the fusion of multiple retrieved contexts into a fixed-length input remains constrained by the context window limitations of current language models [55, 50]. Techniques such as passage reranking [56, 57] and/or hierarchical encoding [58, 59, 60] can be used to address this issue, while each introduces its own complexity trade-offs.

RAG represents a broader shift in the design philosophy of NLP systems toward modular, interpretable, and dynamically extensible architectures. Its ability to separate retrieval, representation, and reasoning into loosely coupled yet jointly trainable components aligns with the goals of explainable and trustworthy AI. It signals a departure from fully parametric, end-to-end deep learning pipelines and paves the way for systems that can continuously learn, reason, and adapt in open-world settings.

1.6. Requirements for efficient AI-driven RAG systems

The need for efficient AI-driven RAG systems has become increasingly prominent due to the rapid development of LLM and AI applications. First, credibility and transparency are critical, especially in high-stakes fields such as healthcare, law, and science. RAG systems must not only retrieve relevant evidence, but also present the evidence in a way that ensures consistency with the source content and is interpretable.

Second, in a dynamic environment where information is constantly changing, static retrieval processes and fixed model weights are no longer sufficient. This makes adaptability and lifelong learning increasingly important. Future RAG systems must support continuous learning, real-time updates, and fine-tuning for specific domains and user contexts.

Third, as applications are promoted and businesses become flatter, scalability and efficiency remain critical for industrial and enterprise deployments. This includes minimizing delays in the retrieval process, adopting cost-effective generation methods, and ensuring seamless integration with existing data and infrastructure.

Finally, there is an urgent need to prioritize global inclusiveness. Currently, most RAG research revolves around high-resource languages and well-structured datasets. To democratize the benefits of AI, RAG systems must support multilingual reasoning, low-resource knowledge access, and cultural sensitivity during retrieval and generation.

1.7. Gaps in Current Research

Despite a growing body of literature, the current research landscape remains fragmented. Studies vary widely in their focus and methodology, with few adhering to a unified or standardized framework. Much of the literature is driven by specific applications or system prototypes, often lacking deeper engagement with the underlying architectures, algorithms, or theoretical foundations of RAG.

This fragmentation poses a challenge for new entrants to the field. The absence of coherent methodological structures makes it difficult to gain a comprehensive understanding or develop innovations at the system level. Moreover, while many review articles emphasize high-level applications, there is limited systematic exploration of the technical underpinnings of RAG systems, such as component integration, retrieval strategies, algorithmic mechanisms, or computational logic.

To address this gap, the present review adopts a bottom-up perspective—focusing on the core architecture, system logic, and foundational methodologies that form the basis of RAG systems. This approach aims to provide the community with a more technical, engineering-oriented understanding of how RAG systems are built, evaluated, and extended.

1.8. Purpose of the Review

This review aims to provide a comprehensive and systematic examination of the current state of research in retrieval-augmented generation (RAG) within the context of large language models (LLMs). As the complexity and diversity of information needs grow, there is an urgent need to move beyond purely parametric models and toward systems that can integrate dynamic retrieval with generation. RAG represents a promising direction to address these needs by combining the generative capabilities of LLMs with the precision and adaptability of information retrieval systems.

The primary objective of this review is to identify dominant research trajectories, consolidate key technical solutions, and outline future directions that address the pressing challenges of factual accuracy, scalability, adaptability, and domain relevance. By synthesizing research across architecture, retrieval methodologies, generation models, evaluation techniques, and real-world applications, this work serves as a foundational resource for researchers and practitioners seeking to advance the development and deployment of RAG-enhanced LLMs.

In an era where LLMs are becoming core infrastructure across industries, the demand for intelligent systems that can retrieve and reason over external knowledge is increasing rapidly. RAG systems aim to fulfill this need by bridging parametric model knowledge with contextual, real-time information access. However, to bring enduring value, such systems must meet several critical requirements.

1.9. PRISMA-Based Methodology for Literature Selection

To ensure methodological rigor and transparency in the selection of literature for this review on Retrieval-Augmented Generation (RAG) in the context of large language models (LLMs), we adopted the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) framework. The PRISMA guidelines provide a structured process for identifying, screening, and including research works, which is crucial given the rapidly growing and heterogeneous body of literature in this domain. As shwon in Figure1, we detail each phase of the PRISMA process, along with a critical discussion of its technical implications for building a robust and comprehensive RAG literature review.

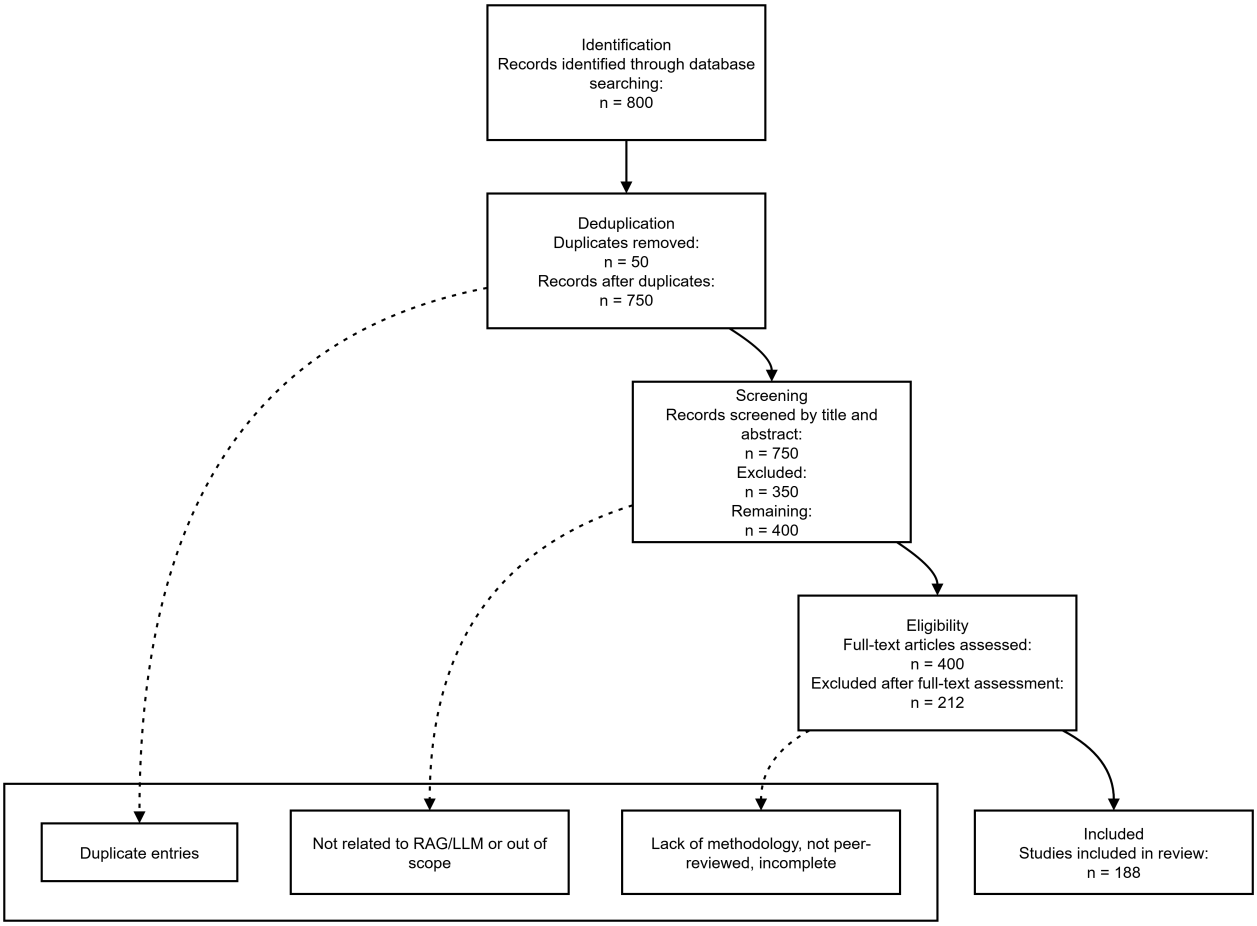

Figure 1: PRISMA Procedure

1.9.1. Identification Phase

The initial identification stage involved a comprehensive search across multiple academic databases, including ACM Digital Library, IEEE Xplore, SpringerLink, Elsevier ScienceDirect, and arXiv. We used combinations of search terms such as "retrieval-augmented generation", "RAG", "large language model", "dense retrieval", and "open-domain question answering" to maximize coverage. This resulted in a total of 800 potentially relevant articles published from 2019 to early 2025. These articles spanned a variety of publication types including conference proceedings, journal articles, preprints, and system descriptions.

At this stage, the challenge was not only volume but also heterogeneity. Many papers used different terminologies or focused on adjacent domains, which necessitated a broad but precise set of inclusion criteria. This initial pool captured a wide spectrum of studies ranging from foundational algorithms to downstream applications and extensions.

1.9.2. Deduplication and Screening

Following identification, a deduplication process was performed to remove redundant entries across databases. This step reduced the total corpus by 50 records, resulting in 750 unique articles. The screening phase then focused on titles and abstracts to filter out works that were tangentially related or entirely unrelated to the core topics of RAG and LLM integration. At this point, 350 studies were excluded for reasons such as focusing solely on language modeling without retrieval, or on IR systems without generation. This left 400 articles for full-text assessment.

This stage was technically significant, as many RAG-related studies are embedded within broader discussions of NLP systems or hybrid architectures. Thus, the ability to discern relevance based on abstract-level indicators required domain expertise in understanding whether the retrieval-generation integration was merely peripheral or central to the paper's contributions.

1.9.3. Eligibility Assessment

During the eligibility phase, the full texts of the remaining 400 articles were thoroughly reviewed. Each paper was evaluated against a set of inclusion criteria which emphasized the presence of explicit RAG architectures, detailed methodological exposition, empirical evaluation using standard benchmarks, or novel contributions to retrieval or generation mechanisms in the RAG pipeline.

A total of 212 papers were excluded at this stage. The most common reasons included lack of methodological depth, insufficient connection between retrieval and generation components, incomplete systems, or absence of peer review. This phase demanded not only technical discernment but also consistency in the application of evaluative criteria to ensure comparability across studies.

Notably, this stage also highlighted gaps in current literature. Many works were found to reuse existing LLMs without addressing the unique challenges of knowledge integration, dynamic retrieval, or evaluation groundedness. Such findings further informed our taxonomy and identified the boundaries of what constitutes core versus peripheral contributions in the RAG field.

1.9.4. Final Inclusion

Ultimately, 188 studies were included in this systematic review. These articles collectively represent the breadth and depth of contemporary RAG research, from retrieval strategies

(e.g., sparse, dense, hybrid) and generation models (e.g., BART, T5, GPT) to complex system extensions like feedback loops and multi-hop reasoning. Their inclusion was based on technical novelty, experimental rigor, architectural transparency, and relevance to real-world or high-value applications.

The final dataset offers not only a representative view of the current state of the field but also a robust foundation for developing the taxonomy, identifying research gaps, and proposing future directions. This rigorously selected body of literature forms the empirical and conceptual backbone of the subsequent sections of this review.

1.9.5. Technical Reflection and Value of PRISMA

Adopting the PRISMA methodology introduced multiple benefits beyond systematic filtering. First, it provided traceability and replicability to the review process—key requirements for any scientific synthesis. Second, it enabled a statistically grounded understanding of the field's maturity: for example, the high exclusion rate at the eligibility phase revealed how many published studies in the broader LLM landscape lack integration with retrieval mechanisms, indicating an over-reliance on parametric models.

Additionally, the PRISMA process revealed thematic patterns in research evolution. The majority of eligible studies were concentrated in the post-2021 period, correlating with the surge in transformer-based retrievers and the broader push toward grounded, fact-aware generation. It also exposed underexplored areas, such as multilingual RAG, real-time indexing, and evaluation of groundedness—thus offering data-driven support for future research prioritization.

By adhering to the PRISMA framework, this review ensured methodological rigor and conceptual clarity, while also yielding insights into the structure and development trajectory of RAG research. The process not only filtered literature but surfaced meaningful patterns, helped establish research boundaries, and framed the technical scaffolding for the taxonomy and synthesis that follows.

1.10. Contributions

1.10.1. A Comprehensive Taxonomy

A central contribution of this review is the development of a comprehensive taxonomy that categorizes the RAG literature into coherent and interpretable dimensions. This taxonomy spans retrieval methods, generation techniques, evaluation strategies, applications, and auxiliary topics such as interpretability and multilingual support.

Beyond classification, the taxonomy enables a data-driven mapping of the field. By combining content-year analysis with co-occurrence patterns across research themes, this study surfaces critical trends, reveals underexplored intersections, and highlights imbalances across domains. It not only helps readers understand what has been done, but also uncovers areas ripe for future exploration.

1.10.2. Focus on Core

This review places particular emphasis on the system-level architecture and core technical pillars of RAG, dissecting them across several key dimensions:

The RAG Architecture Overview outlines the interaction among the three foundational components—Retriever, Generator, and Knowledge Source—providing a structured view of input-output pathways and modular integration at the engineering level.

In the Retrieval Methodologies section, sparse and dense retrieval techniques are systematically categorized, from traditional approaches like TF-IDF and BM25 to advanced neural methods such as Dense Passage Retrieval (DPR) and ColBERT. This analysis highlights the trade-offs between precision, latency, and semantic relevance in various retrieval paradigms.

The Generator Methodologies section discusses the evolution and application of generative models like BART, GPT, and T5. These models form the generative core of RAG systems and are analyzed in terms of their capabilities to produce context-aware, coherent responses.

The RAG Extensions section examines the scalability and adaptability of modern RAG systems. Topics such as multi-hop reasoning, conversational RAG, and feedback-based refinement demonstrate the increasing complexity and intelligence expected from next-generation RAG applications.

Finally, the Evaluation Metrics section presents a multidimensional approach to system assessment. It incorporates traditional NLP metrics (Exact Match, BLEU, ROUGE), retrieval-specific indicators (Recall@k), and newer criteria such as groundedness, latency, and cost-efficiency—reinforcing the importance of reproducible and objective evaluation in system development.

1.10.3. Toward a More Contextual and Responsible RAG Ecosystem

This review does more than organize existing knowledge—it also seeks to define the functional and ethical expectations for the next generation of RAG systems. In an increasingly information-rich world, context-aware, adaptive, and inclusive RAG systems will be essential in domains ranging from enterprise search to scientific discovery.

By identifying structural gaps and proposing targeted directions for research, this review contributes to shaping a more robust, interpretable, and equitable AI ecosystem. It serves as both a reference point for current practitioners and a roadmap for researchers aiming to push the boundaries of RAG-augmented language modeling.

In the subsequent section, we review the architectural components of RAG in greater technical detail. We focus on the interplay between retriever, generator, and knowledge base, analyzing how different design choices affect the system's performance, generalizability, and scalability. Understanding these architectural underpinnings is essential to evaluating and improving RAG-based solutions for real-world, knowledge-intensive applications.

1.11. Organization of the Paper

This review is structured to provide a comprehensive, technically grounded, and forward-looking examination of retrieval-augmented generation (RAG) systems within the context of large language models (LLMs). The paper is organized into eleven sections, each addressing a critical aspect of RAG research:

1.11.1. Section 2: Literature Overview and Insights

This section presents a high-level synthesis of the RAG literature from both a temporal and thematic perspective. It includes an overview of research trends by year, content-based insights, content-year correlation analyses, and co-occurrence analyses to highlight core themes, empirical trends, underexplored areas, and future research directions.

1.11.2. Section 3: RAG Architecture Overview

This section outlines the core components of RAG systems, including the retriever, generator, and knowledge source. It also describes the typical data flow and system workflow, offering

a structural foundation for understanding how RAG systems function at a systems-engineering level.

1.11.3. Section 4: Retrieval Methodologies

Here, we delve into the technical foundations of retrieval within RAG systems. This includes a detailed classification of sparse retrieval (e.g., TF-IDF, BM25) and dense retrieval techniques (e.g., DPR, BERT-based retrievers, ColBERT), along with comparative insights and architectural implications.

1.11.4. Section 5: Generator Methodologies

This section explores the generative backbone of RAG systems by analyzing various LLMs used in generation tasks, including BART, GPT, T5, and BERT. The discussion highlights their respective strengths, architectures, and integration strategies within RAG pipelines.

1.11.5. Section 6: RAG Extensions

This section investigates advanced use cases and system enhancements, such as open-domain QA, multi-hop reasoning, feedback loops, hybrid retrieval systems, and conversational RAG. These extensions demonstrate the adaptability and growing sophistication of RAG architectures.

1.11.6. Section 7: Evaluation Metrics

A comprehensive taxonomy of evaluation approaches is presented here, covering both generative and retrieval aspects. Metrics include Exact Match, F1, BLEU, ROUGE, Recall@k, groundedness, latency, and cost. Comparative analyses underscore the trade-offs and contextual relevance of each metric.

1.11.7. Section 8: Applications

This section showcases real-world applications of RAG systems across various domains, including enterprise search, academic research, legal and healthcare QA, customer support, and code documentation. These examples illustrate the practical value and versatility of RAG in different industries.

1.11.8. Section 9: Challenges, Limitations, and System Constraints

This section addresses the critical technical and operational challenges in current RAG research and deployment. Topics include latency, factual consistency, context selection, knowledge drift, scalability, and the limitations of parametric LLMs.

1.11.9. Section 10: Future Directions

Building upon previous sections, this part outlines promising research avenues, including personalized and multimodal RAG, structured–unstructured knowledge fusion, multilingual and low-resource adaptation, dynamic indexing, and lifelong learning mechanisms.

1.11.10. Section 11: Conclusion

The final section summarizes the key findings, reaffirms the contributions of this review, and reflects on the broader implications for the development of more transparent, adaptive, and contextually aware RAG systems.

2. Literature Overview and Insights

2.1. Overview and Insights by Year

2.1.1. Overview

We collect the publications from 1998 to 2025. The stats of the collections highlights the evolution of research in Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs), tracing a path from foundational principles in information retrieval to their deep integration in modern AI systems.

Figure 2 shows the trend of collections by year. It can be seen that from 1998 to 2016,

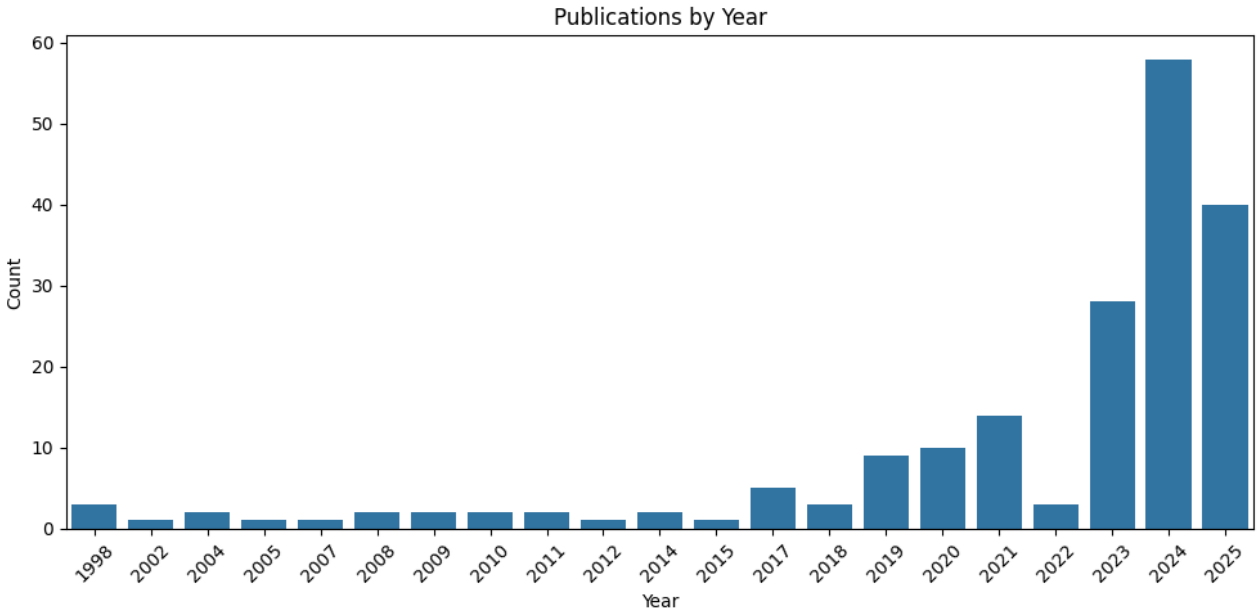

Figure 2: Publications by Year

publication activity was sparse, typically fewer than three papers annually, reflecting a phase where information retrieval (IR) and generation research developed in parallel but largely in isolation. This period focused on traditional IR models such as BM25 and early question answering frameworks, setting the groundwork for later developments.

Shifting around 2017, coinciding with the rise of transformer-based LLMs like BERT, GPT-2, and T5, research output began to climb, driven by growing recognition of the limitations of parametric-only models, particularly their struggles with factual consistency and knowledge scalability. From 2017 to 2020, publication numbers increased steadily, as researchers explored hybrid architectures that fused retrieval and generation, initiating the conceptual basis for modern RAG systems.

The acceleration continued between 2021 and 2023. This period marked the mainstream adoption of RAG as a research priority. Key innovations included the tight coupling of retrieval modules with LLM training and inference pipelines, the creation of retrieval-aware benchmarks, and the increased emphasis on trustworthiness, factual grounding, and explainability. These advances positioned RAG not just as an academic construct but as a practical necessity for deploying LLMs in high-stakes domains such as healthcare and legal reasoning.

The momentum peaked in 2024, in which the explosion of interest reflected both widespread industrial adoption and academic investment in fine-tuning, latency reduction, streaming retrieval, and domain-specific adaptation. Multi-modal RAG systems also began to emerge,

broadening the application landscape. The research focus during this phase shifted from conceptual design to large-scale optimization, real-world deployment, and system robustness.

In 2025, publications further mature stabilization of the field. The sustained output at this level suggests a mature and deeply embedded paradigm. Attention has turned toward refining retrieval strategies for low-resource settings, continual learning, and underrepresented languages, as well as standardizing evaluation methods and improving interpretability. RAG has transitioned from an experimental approach to a default architecture for grounded, scalable generation.

2.1.2. Insights

It can be seen that, the field of natural language processing (NLP) is shifting from purely parametric models, which encode all knowledge internally, to hybrid systems that dynamically retrieve external information based on context. The rise of Retrieval-Augmented Generation (RAG) reflects the maturation of information retrieval (IR), advances in model scalability, and the growing demand for reliable, cost-effective, and adaptable language generation.

The recent research shift enables models to combine external knowledge access with flexible generation, addressing limitations of parametric-only approaches. Despite ongoing challenges in optimizing retrieval relevance, latency, and alignment, RAGs have firmly established their value in both research and practical applications.

As retrieval quality and efficiency continue to improve, RAG architectures are evolving toward more interpretable and effective designs. No longer an emerging concept, RAG is now a foundational strategy for building fact-based, scalable, and adaptive AI systems. Looking forward, RAG-enhanced LLMs are poised to become the backbone of responsible, knowledge-intensive AI solutions.

2.2. Overview and Insights by Content

2.2.1. Overview

Figure 3 shows the distribution of publications on Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs), revealing a robust and retrieval-centric research landscape. Retrieval stands out as the dominant focus, with the majority of papers addressing retrieval strategies, indexing, and relevance optimization, highlighting retrieval as the foundational driver of the RAG paradigm. Closely following are works on model architecture and generation techniques, reflecting efforts to seamlessly integrate retrieval mechanisms into LLM frameworks.

A significant portion of RAG-related research overlaps with studies on language models, encompassing pretraining, fine-tuning, and adaptation for retrieval-augmented tasks. This intersection underscores the role of LLMs as the technical backbone enabling effective RAG implementations.

Application-focused research also plays an important role, demonstrating the growing deployment of RAG systems across domains such as healthcare, legal reasoning, and enterprise automation. In particular, studies centered on healthcare and medical use cases emphasize the importance of trust, factual grounding, and domain-specific retrieval, highlighting the practical potential and impact of RAG in high-stakes environments.

Complementing this, a substantial number of studies focus on datasets and analytical evaluations, indicating a balanced emphasis on empirical assessment and critical reflection on model behavior, limitations, and trade-offs. Survey papers contribute high-level syntheses and

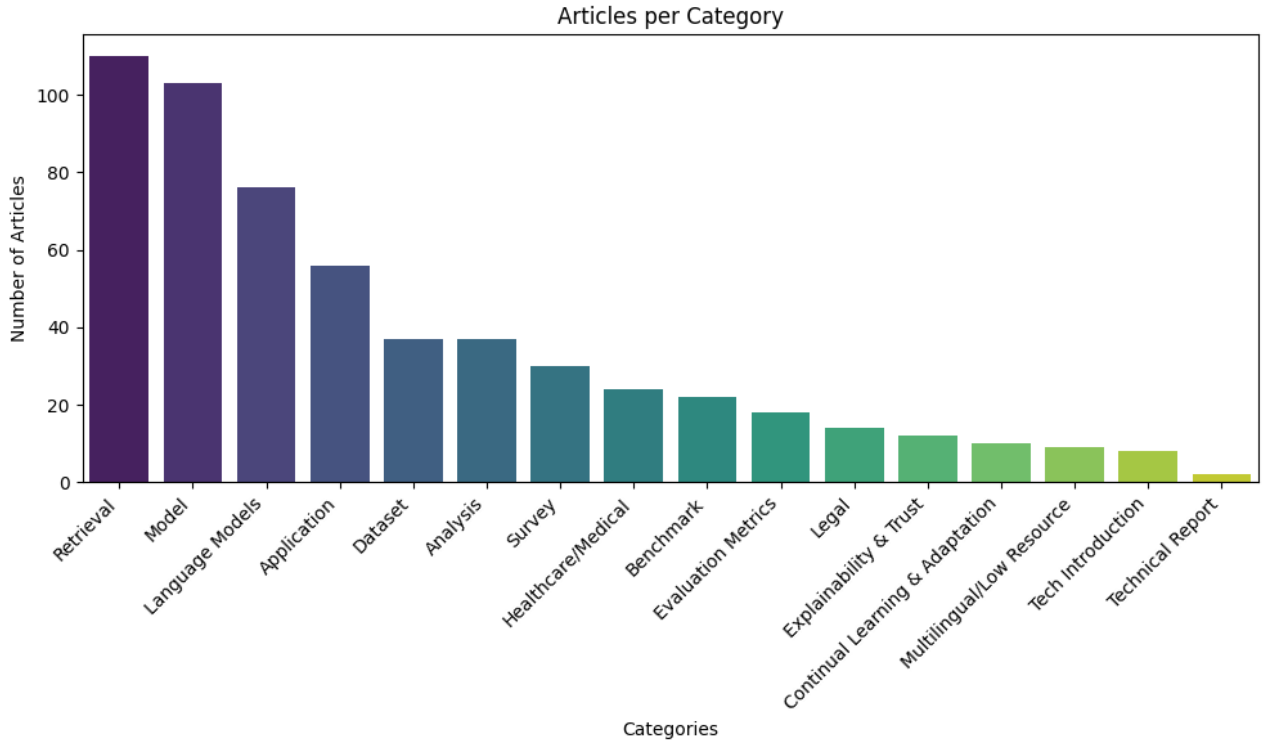

Figure 3: Article Category

trend mapping, aiding both newcomers to the field and domain-specific investigations. Meanwhile, benchmarks and evaluation metrics signal an ongoing commitment to standardizing performance measurement, an essential step for sustained progress.

2.2.2. Insights

Despite the field's maturity, key areas such as explainability and trust, continual learning, and multilingual or low-resource adaptation remain underexplored. Their limited representation suggests the need for deeper investigation, especially given their importance in advancing responsible and accessible AI. The small number of technical reports may also reflect the current publication preferences of peer-reviewed venues, which often prioritize theoretical contributions over non-peer-reviewed dissemination. However, as practical evaluation methodologies and datasets continue to evolve, these reports may become more prevalent.

This thematic distribution reveals a research ecosystem centered on the engineering of retrieval-enhanced generation. The strong emphasis on retrieval and modeling aligns with the technical demands of building scalable, fact-grounded, and context-aware systems. At the same time, increasing attention to application-driven studies reflects RAG's transition from theoretical exploration to operational deployment. Looking forward, research is expected to deepen in areas such as interpretability, trust, multilingual capabilities, and adaptation to low-resource settings, factors critical to the advancement of inclusive, transparent, and robust AI systems.

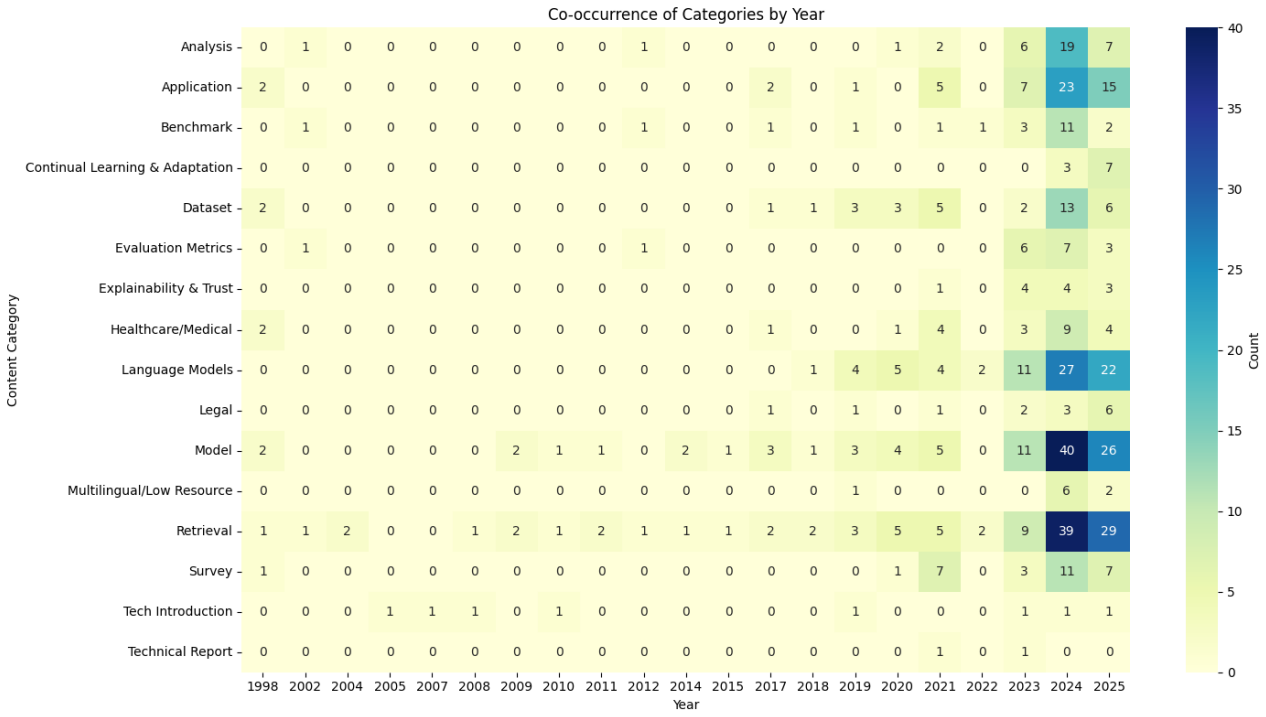

Figure 4: Content-Year Co-occurrence Analysis

2.3. Content-Year Analysis

2.3.1. Overview on Trends in RAG and LLM Research

The longitudinal analysis of research on Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) reveals a clear trajectory from foundational work toward a mature, application-driven field. As shown in Figure 4, early years, spanning from the late 1990s through the early 2010s, show limited and fragmented research activity, mostly focused on isolated aspects of retrieval or language modeling. During this period, foundational questions about the integration of retrieval with generative models were only beginning to be posed. The scarcity of empirical datasets, benchmarks, and applications during these formative years indicates the nascent stage of the field.

By the late 2010s, there was a noticeable increase in research coalescing around the core RAG concept—integrating retrieval mechanisms with large-scale generative models. This period witnessed growing convergence between retrieval strategies and model architecture research, suggesting the formation of a unified paradigm. However, while technical integration advanced, empirical evaluations and real-world applications remained limited. Early applications were often proof-of-concept rather than deployment-focused, and challenges such as latency, relevance, and model alignment were yet to be thoroughly addressed.

From 2021 onwards, the field has experienced rapid expansion in research volume and thematic diversity. The intertwining of retrieval, modeling, and application research reflects the shift from conceptual exploration to practical, domain-specific deployment. Healthcare and legal sectors have emerged as prominent use cases, driven by the need for factually accurate, trustworthy AI systems in critical domains. This reflects a growing awareness that RAG systems must meet rigorous domain-specific standards, especially where errors could have significant consequences.

2.3.2. Insights into Current Research Landscape

Despite the growth, several thematic areas reveal important gaps and ongoing challenges. Explainability and trust, though recognized as vital for adoption in sensitive domains, remain underrepresented in the literature. The limited overlap between explainability and core technical research suggests that transparency has yet to become an integral design consideration in RAG architectures. This represents both a challenge and an opportunity: as these systems are deployed more widely, providing interpretable outputs and understanding model decision pathways will be essential for user acceptance and regulatory compliance.

Multilingual and low-resource language contexts also remain marginal in current research. The scarcity of work in these areas indicates that the benefits of RAG and LLM advancements are not yet equitably distributed across global language communities. This gap points to a significant research frontier with high societal value, as addressing it would democratize access to advanced AI technologies and foster inclusivity in language AI.

Continual learning and adaptation show emerging interest but are still in their infancy within the RAG context. The dynamic nature of information environments necessitates models that can update knowledge on the fly without costly retraining. Progress in this area will be critical for maintaining model relevance, reducing obsolescence, and enabling personalized or domain-adaptive applications.

Another observation is the limited presence of technical introductions and practical technical reports, which may reflect publishing norms favoring peer-reviewed research over grey literature or industry white papers. The relative scarcity of such contributions restricts the dissemination of implementation insights and real-world system challenges, potentially slowing innovation and technology transfer from research to production.

2.3.3. Future Research Directions and Value Propositions

Addressing the identified gaps presents rich avenues for impactful research. Deepening the focus on explainability and trust will not only improve the usability and safety of RAG systems but will also foster regulatory compliance and ethical AI deployment. Research efforts should prioritize methods for transparent retrieval mechanisms, rationale generation, and uncertainty quantification to build user confidence.

Expanding multilingual and low-resource language research holds strategic importance for achieving global AI inclusivity. Developing efficient retrieval and generation methods that can function across diverse languages and dialects would broaden the applicability of RAG systems, ensuring they serve a truly global user base.

Continual learning represents another vital direction, enabling RAG models to stay current with rapidly evolving information landscapes. Research into scalable incremental updating, domain adaptation, and lifelong learning algorithms can enhance system longevity and reduce the operational costs of model maintenance.

Finally, encouraging more industry-academic collaboration and fostering avenues for technical reports and reproducibility studies will enrich the research ecosystem. Sharing practical experiences and challenges can accelerate the maturation of RAG technology and ensure robust deployment in complex, real-world environments.

The collected research on RAG and LLMs reveals a dynamic, evolving field with strong technical foundations in retrieval and modeling, increasingly grounded by real-world applications and evaluation efforts. However, the underrepresentation of critical areas such as explainability, multilingual support, and continual learning underscores both ongoing challenges

and rich opportunities. As RAG-augmented models become integral to knowledge-intensive AI applications, focusing on these gaps will be essential for developing responsible, scalable, and inclusive AI systems that meet societal and technological demands.

2.4. Content Co-occurrence Analysis

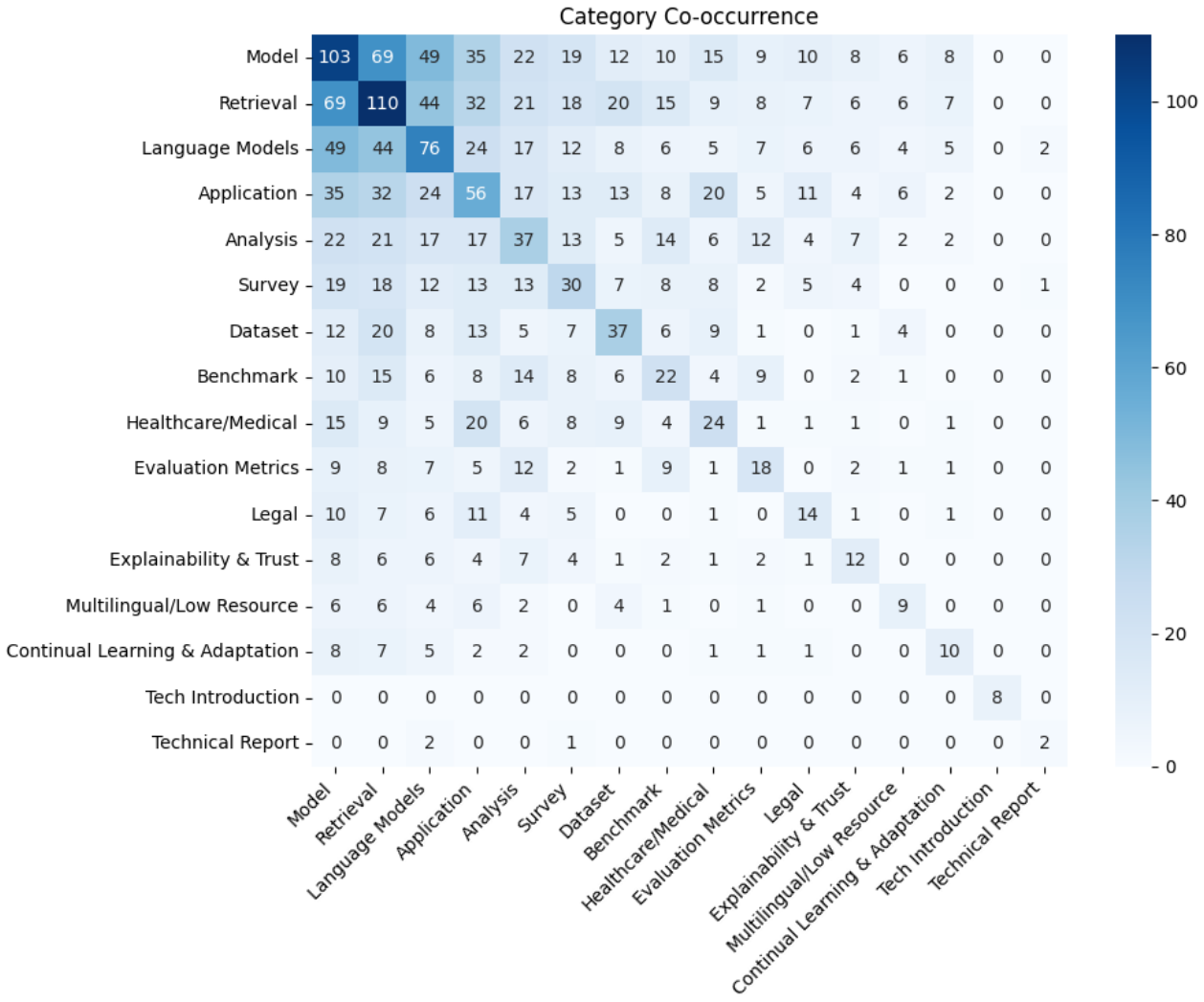

Figure 5: Publication Content Category Co-occurrence

We also perform the co-occurrence analysis of publication content categories in the domains of Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) offers a comprehensive view of the field's topical structure and interdisciplinary relationships. Figure **??** reveals not only the dominant research areas but also the interplay among them, providing valuable insights into the field's maturity, thematic diversity, and emerging trajectories.

2.4.1. Core Themes and Their Intersections

The categories of Retrieval and Model stand out as the most frequent and interconnected, co-occurring in 69 publications. This underscores their foundational role in the RAG paradigm, which centers on integrating efficient retrieval mechanisms with generative model architectures.

Language Models similarly serve as a bridge between retrieval and modeling, reinforcing their pivotal role in enabling retrieval-aware generation.

Application-focused research also features prominently, frequently intersecting with Model, Retrieval, and Language Models. This indicates that RAG is no longer confined to theoretical constructs but is being actively adopted in practical scenarios. Notably, its strong presence in Healthcare/Medical and Legal domains highlights its relevance in high-stakes settings where factual accuracy and reliability are essential.

2.4.2. Empirical and Evaluation-Driven Research

The categories of Analysis, Dataset, and Evaluation Metrics reflect the field's empirical sophistication. High co-occurrence between Analysis and both Language Models and Retrieval indicates a deep investigation into model behaviors, limitations, and trade-offs. The presence of Benchmark, often alongside Retrieval and Evaluation Metrics, points to ongoing efforts to standardize evaluation—an essential step toward reproducibility and meaningful comparison.

Survey papers play a significant role in synthesizing trends and knowledge across the field. Their frequent intersections with key categories like Retrieval, Model, and Dataset suggest an active community of researchers curating and contextualizing progress for both newcomers and experts alike.

2.4.3. Underexplored Areas and Future Opportunities

Despite this depth, some areas remain underrepresented. Topics such as Explainability and Trust, Continual Learning and Adaptation, and Multilingual/Low Resource exhibit limited overlap with other categories, indicating that these crucial research directions are still in their early stages. The Tech Introduction category appears largely isolated, and the scarcity of Technical Reports may reflect either the academic emphasis on peer-reviewed publications or the proprietary nature of industry research, which is often withheld from open dissemination.

2.4.4. Outlook and Research Implications

The current distribution of research suggests a maturing yet evolving ecosystem. While retrieval and modeling form the technical backbone of RAG systems, the increasing emphasis on datasets, evaluation, and application demonstrates a shift from conceptual development to real-world deployment. This convergence of foundational and applied work signals the field's progression from exploration to implementation.

To sustain this momentum, future research must expand into underexplored but critical areas such as explainability, lifelong learning, and multilingual capabilities. These directions are not only technologically essential but also vital for building responsible, inclusive, and globally scalable AI systems. As RAG-enhanced LLMs become the backbone of knowledge-intensive applications, ensuring trustworthiness, adaptability, and equity will define the next frontier of innovation.

3. RAG Architecture Overview

3.1. Core Components

3.1.1. Retriever

The retriever functions as the initial filtering mechanism, tasked with selecting the most relevant documents or passages from a large corpus based on a given input query. Traditionally, retrieval methods such as Sparse Retrieval System BM25 [34, 61, 62] have relied on term-based

matching, leveraging lexical overlap between query and documents. However, these classical approaches often struggle with synonymy and polysemy, motivating the adoption of dense retrieval techniques. Dense retrievers employ learned sentence embeddings, enabling semantic similarity matching beyond mere surface-level word overlap. Prominent examples include Dense Passage Retrieval (DPR [38, 37, 36]) and Contextualized Late Interaction over BERT (ColBERT [63, 64, 65]). Moreover, hybrid retrieval strategies combine both sparse and dense approaches, seeking to leverage the complementary strengths of lexical matching and semantic understanding. The choice and design of the retriever significantly influence the quality and relevance of the candidate documents fed to subsequent components.

3.1.2. Generator

Following retrieval, the generator module assumes the responsibility of producing coherent and contextually appropriate responses. This component is commonly instantiated as a Transformer-based sequence-to-sequence model, such as BART [39, 66, 67], T5 [8, 68, 69], or GPT [70, 71, 72]. The generator conditions on the retrieved documents, effectively integrating the external knowledge with the input query to produce informative and fluent outputs. The architecture of the generator, its pretraining objectives, and fine-tuning strategies critically impact the fidelity and creativity of the generated answers. Its ability to attend selectively to relevant retrieved context enables the generation of responses that are not only factually grounded but also contextually nuanced.

3.1.3. Knowledge Source

Underlying both the retriever and generator is the knowledge source, which constitutes the foundational repository of information. This corpus can vary widely, encompassing encyclopedic knowledge bases like Wikipedia, domain-specific documentation such as product manuals, or proprietary enterprise data. The size, quality, and topical coverage of this knowledge source directly affect the breadth and accuracy of the information accessible to the RAG system. Maintaining and updating this corpus is essential to ensure the system's responsiveness to evolving knowledge and information needs.

The RAG architecture synthesizes retrieval and generative modeling through a well-orchestrated interplay among the retriever, generator, and knowledge source. Each component's design choices and implementation intricacies bear significant influence on the system's ability to deliver relevant, accurate, and coherent information, positioning RAG as a powerful paradigm in modern natural language processing applications.

3.2. Workflow and Data Flow in Retrieval-Augmented Generation

The operational workflow of a Retrieval-Augmented Generation (RAG) system is characterized by a tightly coupled interaction between retrieval and generation stages, mediated through a centralized architecture that orchestrates the query-response cycle. Upon receiving an input query from the user, the RAG system delegates the responsibility of document selection to a retriever component. The retriever searches a designated knowledge source and returns the top-k documents deemed most relevant to the input query. These retrieved documents serve as external evidence or contextual support for the subsequent generative process.

The underlying data flow of this RAG logic pipeline is illustrated in Figure 6. The flow begins with the user query, which is submitted to the RAG system. The system forwards the query to the retriever, which interfaces with the knowledge source, a structured or unstructured

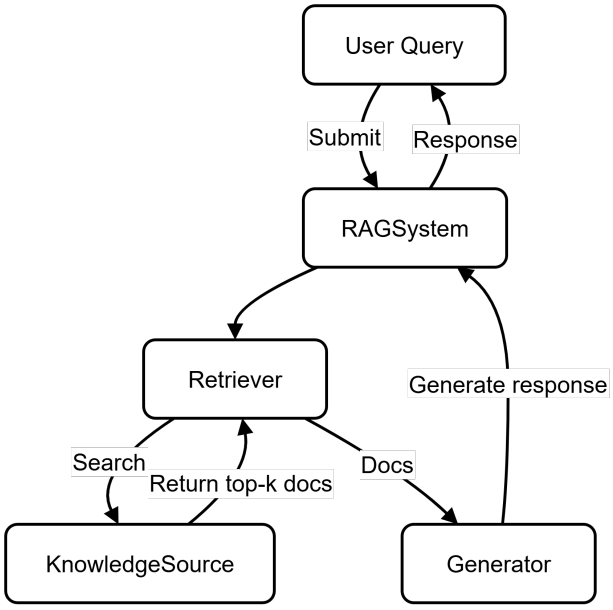

Figure 6: Dataflow of RAG logic pipeline

corpus of domain-specific content such as Wikipedia or enterprise documents. The retriever searches this corpus and returns the top-k most relevant documents. These documents are then passed to the generator, which integrates them with the original query to produce a response. This response is returned through the RAG system to the user, completing the end-to-end information retrieval and synthesis cycle.

Once the top-ranked documents have been retrieved, the generator component synthesizes the information from both the query and the documents to produce a coherent, contextually grounded response. This generator is typically instantiated as a large language model fine-tuned for conditional text generation, enabling it to attend over the retrieved contexts and dynamically integrate factual content with fluent language modeling. The final output is returned to the user in the form of a natural language response, ideally optimized for informativeness, relevance, and factual accuracy.

To further clarify the temporal and component-wise dynamics of the RAG system, Figure 7 presents a sequence diagram of the RAG workflow. This representation captures the runtime interactions between the system's core actors: the user, the RAG system controller, the retriever, the knowledge source, and the generator. The sequence begins when the user issues a query to the RAG system, which then activates the retriever. The retriever accesses the knowledge source to locate relevant documents and returns them to the system. These documents are subsequently passed to the generator, which conditions on both the query and the retrieved context to generate an appropriate response. Finally, the system delivers this response back to the user.

This tightly coupled interaction between retrieval and generation underpins the core value proposition of RAG systems. By explicitly grounding generative outputs in retrieved evidence, the architecture not only enhances factual reliability but also mitigates hallucination, a common shortcoming in standalone generative models. Moreover, the modular separation between retrieval and generation allows for flexible system tuning, hybridization of retrieval strategies, and targeted updates to the underlying knowledge source, thereby enabling contin-

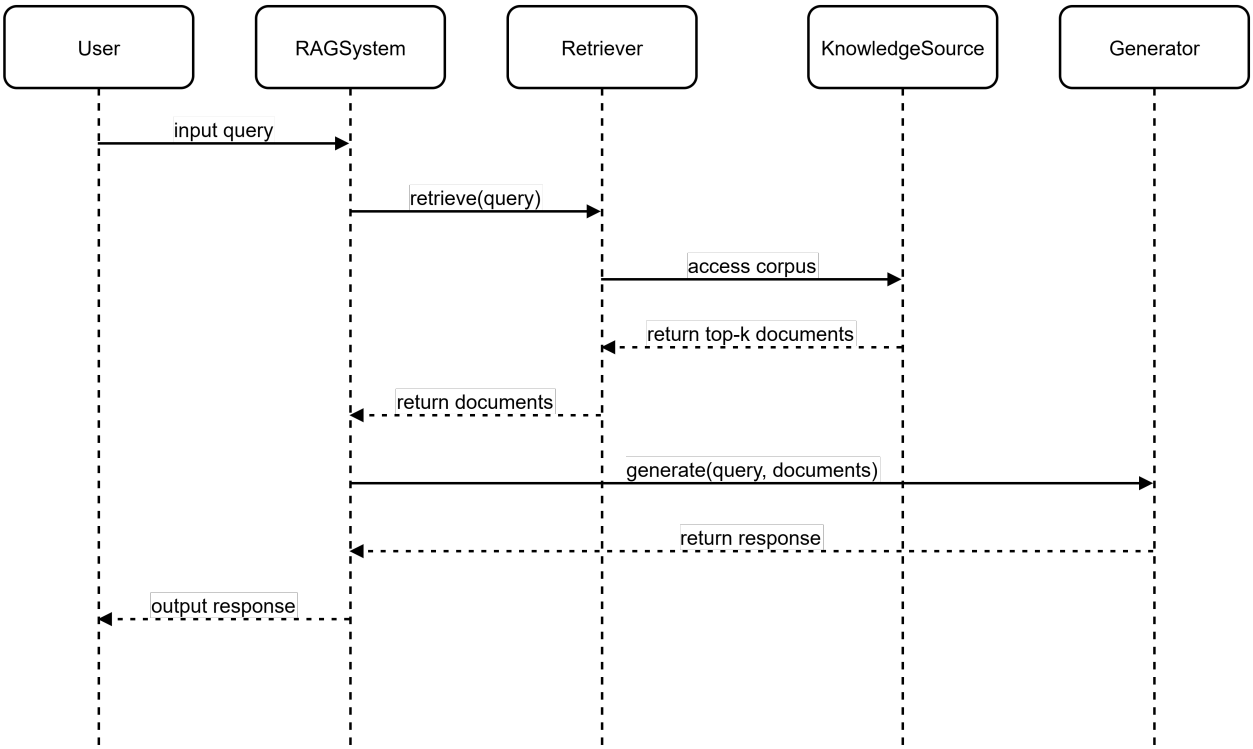

Figure 7: Sequence logic of RAG workflow

ued adaptability in dynamic or data-rich environments. Together, the data flow and workflow encapsulate the core operational logic of RAG and illustrate the pipeline's capacity to deliver high-quality, context-aware responses in real-world applications.

4. Retrieval Methodologies

4.1. Sparse Retrieval Techniques

4.1.1. Term-Based Retrieval Methods

Term-based retrieval techniques, which include simple models like Boolean retrieval [73, 74, 75] and the Vector Space Model (VSM [76, 77, 78]), form the foundation of many information retrieval systems. These methods focus on ranking documents based on the presence or frequency of query terms in the documents, without considering the semantic meaning or context of the terms involved.

The Boolean retrieval model is the simplest approach in this category. It retrieves documents based on the exact presence of query terms. In this model, a document is considered relevant if it contains the query terms, and irrelevant if it does not. The Boolean model, while computationally efficient, lacks the ability to rank documents by relevance or handle partial matches. This binary approach makes it unsuitable for more complex retrieval tasks that require nuanced judgment about the importance of terms within documents.

In contrast, the Vector Space Model (VSM) represents a more sophisticated approach, where both documents and queries are modeled as vectors in a high-dimensional space. Each dimension corresponds to a term in the corpus, and the weight assigned to each term typically reflects its importance, often measured by TF-IDF. The relevance of a document to a query is

determined by calculating the similarity between the query vector and the document vectors, with cosine similarity being the most commonly used metric. The formula for cosine similarity between a query vector q and a document vector d can be formulated as

$$\text{cosine\_similarity}(q, d) = \frac{q \cdot d}{\|q\|\|d\|} \tag{1}$$

where q and d are the vector representations of the query and document, and $\|\cdot\|$ represents the Euclidean norm of the vector. This similarity measure computes the cosine of the angle between the two vectors, providing a quantitative measure of the relevance of a document to the query. Cosine similarity allows for the ranking of documents by how closely they match the query in terms of term distributions, offering a more flexible approach compared to the Boolean model.

4.1.2. TF-IDF: Term Frequency-Inverse Document Frequency

The Term Frequency-Inverse Document Frequency (TF-IDF) model is a cornerstone of information retrieval and text representation. It is commonly employed to evaluate the significance of a term within a document relative to a corpus. TF-IDF consists of two components, including Term Frequency (TF [79, 80, 81]) and Inverse Document Frequency (IDF [82, 83, 84]). The former measures how often a term appears within a document, while the latter gauges the importance of the term across the entire corpus. The product of these two values gives a score that reflects the relevance of the term in the document, considering its rarity in the larger corpus. The mathematical expression for Term Frequency (TF) is defined as

$$tf(t, d) = \frac{f(t, d)}{\sum_{t' \in d} f(t', d)} \tag{2}$$

where $f(t, d)$ denotes the frequency of term t in document d, and the denominator sums the frequencies of all terms in d. This normalization prevents longer documents from having an unfair advantage, ensuring that term frequency is balanced across varying document lengths. The Inverse Document Frequency (IDF), which quantifies the informativeness of a term across the entire corpus, is calculated as

$$idf(t, D) = \log\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right) \tag{3}$$

where $|D|$ is the total number of documents in the corpus, and $|\{d \in D : t \in d\}|$ counts the number of documents that contain the term t. IDF reduces the weight of common terms that appear in many documents, which are likely to be less informative, and increases the weight of rare terms that appear in fewer documents, highlighting their uniqueness.

The TF-IDF score for a term t in document d within a corpus D is the product of the TF and IDF, which can be formulated as

$$tf - idf(t, d, D) = tf(t, d) \times idf(t, D) \tag{4}$$

This formulation ensures that terms which are frequent in a given document but rare across the entire corpus are emphasized, making TF-IDF particularly useful for document retrieval and classification tasks. It is computationally efficient, scalable, and interpretable, which explains its continued popularity in information retrieval systems.

However, despite its strengths, TF-IDF comes with notable limitations. The method assumes that terms are independent of each other, neglecting any syntactic or semantic relationships between them. This assumption limits the model's ability to capture more complex linguistic structures, such as polysemy (words with multiple meanings) and synonymy (words with similar meanings). Additionally, TF-IDF does not account for word order, which can result in a loss of semantic meaning that is context-dependent. As a result, more sophisticated techniques, such as word embeddings and transformer-based models, have been developed to better capture the deeper relationships between words and their contexts.

4.1.3. Best Match 25 (BM25)

The BM25 model, an extension of the probabilistic information retrieval framework, is one of the most effective term-based retrieval techniques in practice today. It builds upon the Vector Space Model by incorporating a probabilistic approach that adjusts for term frequency saturation and document length normalization. The BM25 formula for ranking a document d given a query q is

$$\text{score}(d, q) = \sum_{t \in q} idf(t) \times \frac{tf(t, d) \times (k_1 + 1)}{tf(t, d) + k_1 \times (1 - b + b \times \frac{|d|}{avg\_len})} \tag{5}$$

where$q$ refers to the query, d the document, $tf(t, d)$ the term frequency of term t in document d, $idf(t)$ the inverse document frequency of term t, $|d|$ the length of document d, avg_len the average document length in the corpus, k_1 and b parameters that control the sensitivity of the model to term frequency and document length.

The BM25 model incorporates two key features, i.e. term frequency saturation and document length normalization. The term frequency saturation ensures that the relevance score does not increase indefinitely as the term frequency grows, which reflects the diminishing returns of adding more occurrences of a term in a document. The length normalization term adjusts for the fact that longer documents are likely to contain more terms, and it ensures that shorter documents are not unfairly penalized.

BM25 has become the actual standard in information retrieval systems, such as search engines, due to its ability to effectively balance term importance and document characteristics. It is widely used because it provides a robust and interpretable ranking method that adapts well to different corpus sizes and query types.

4.1.4. Remarks

Term-based methods continue to offer value in certain contexts, highly effective for tasks where efficiency, simplicity, and interpretability are critical. Furthermore, for smaller corpora or applications where exact term matching is sufficient, models like TF-IDF and BM25 remain highly relevant. The simplicity of these models allows for rapid implementation and easy tuning, making them invaluable tools in a wide range of information retrieval applications.

While term-based retrieval methods such as TF-IDF and BM25 have served as the foundation for many information retrieval systems, recent advancements in the field have shifted towards models that incorporate deeper semantic understanding. Dense retrieval methods, which leverage word embeddings and neural networks, represent a significant departure from these traditional approaches. These methods embed both terms and documents into continuous vector spaces, allowing for more flexible and accurate retrieval by measuring semantic similarity rather than relying on exact term matching. Models like Dense Passage Retrieval

(DPR [38, 37, 36]), which are based on pre-trained neural networks, have demonstrated significant improvements in retrieval performance, especially for complex queries involving synonyms, polysemy, or other semantic nuances.

4.2. Dense Retrieval Techniques

Dense retrieval models, which leverage dense vector representations of both queries and documents, have gained significant prominence in the field of information retrieval. These models are typically powered by deep learning-based methods, which produce continuous embeddings for text, rather than relying on sparse term-based representations such as TF-IDF. The transition from sparse retrieval methods to dense retrieval represents a shift towards capturing the semantic meaning of text, rather than just keyword matching. This move is particularly beneficial for handling complex queries that involve synonyms, polysemy, and other linguistic variations. Dense retrieval models, which generally incorporate pre-trained language models, such as BERT, are designed to provide more flexible, accurate, and context-aware retrieval.

4.2.1. Dense Passage Retrieval (DPR)

Dense Passage Retrieval (DPR) is a state-of-the-art dense retrieval method that leverages dense representations of both the query and the document passage. DPR utilizes two separate neural encoders, one for encoding the query and one for encoding the document passages, both of which are typically based on transformer architectures. These encoders are trained jointly to produce vector representations for both queries and passages in a shared embedding space, allowing for efficient retrieval based on vector similarity.

The core idea behind DPR is to map both queries and passages into the same dense vector space such that the vector of a query is close to the vectors of the relevant passages. The model is trained using a contrastive loss function, typically the InfoNCE loss [85, 86, 86], which encourages the distance between the query vector and the correct passage vector to be smaller than the distance between the query vector and any irrelevant passage vector. The loss function can be expressed as

$$\mathcal{L}_{\text{DPR}} = -\log \frac{\exp(\text{sim}(q, d_{\text{positive}}))}{\sum_{d \in D} \exp(\text{sim}(q, d))} \tag{6}$$

where $\text{sim}(q, d)$ is the cosine similarity between the query vector q and a document passage vector d, and D is the set of all passages. The model maximizes the similarity between the query and the correct passage while minimizing the similarity between the query and other passages.

DPR has demonstrated remarkable success in various information retrieval tasks, particularly in scenarios that involve large-scale document corpora and complex queries. One of the primary advantages of DPR is its ability to capture rich semantic relationships between queries and documents, making it more effective than traditional sparse methods in capturing the intent behind the query.

However, DPR models require substantial computational resources for training, as they involve training large neural networks on massive amounts of text data. Moreover, retrieval during inference can be slow, as it requires calculating the similarity between the query and all document passages in the corpus, although this can be mitigated by employing approximate nearest neighbor search techniques.

4.2.2. BERT-based Dense Retrievers

BERT-based dense retrievers represent a powerful class of models that extend the pre-trained BERT (Bidirectional Encoder Representations from Transformers) architecture for information retrieval tasks. Unlike traditional retrieval models, which rely on keyword matching or shallow term frequencies, BERT-based retrievers aim to represent both the query and the document passages as dense vectors in a continuous vector space. These vectors capture more sophisticated semantic relationships and contextual information, enabling the retrieval system to perform better on tasks that involve nuances such as synonyms, polysemy, and paraphrasing.

A key feature of BERT-based dense retrievers is the fine-tuning of a pre-trained BERT model on the retrieval task. During this fine-tuning process, the BERT model is trained to generate embeddings for both queries and documents in such a way that the similarity between the embeddings of a query and a relevant document is maximized. This fine-tuning process typically employs a contrastive loss function similar to that used in DPR. The general formulation for BERT-based retrievers is as follows

$$\mathcal{L}_{\text{BERT}} = -\log \frac{\exp(\text{sim}(q, d_{\text{positive}}))}{\sum_{d \in D} \exp(\text{sim}(q, d))} \tag{7}$$

where q and d are the query and document embeddings generated by the BERT model, respectively, and D represents the set of candidate documents. The loss function ensures that the query and its relevant document are closer in the embedding space compared to other non-relevant documents.

While BERT-based retrievers have proven effective, they often face challenges in terms of computational efficiency and scalability. One solution to mitigate the computational overhead of BERT is the use of interpolation with BM25, which combines the strengths of dense retrieval models and traditional sparse retrieval models. BM25, with its efficient and interpretable scoring mechanism, is used to narrow down the set of candidate documents, which are then ranked using the dense representations provided by the BERT model. This hybrid approach improves retrieval speed while maintaining high accuracy in document ranking.

Furthermore, BERT-based retrievers require substantial hardware resources, especially when dealing with large-scale datasets. In practice, these models often need specialized hardware such as GPUs to enable efficient training and inference.

4.2.3. ColBERT (Contextualized Late Interaction over BERT)

ColBERT, which stands for Contextualized Late Interaction over BERT, represents an innovative approach that seeks to combine the efficiency of traditional sparse retrieval methods with the power of dense, context-aware embeddings. The ColBERT approach can be formulated as

$$\text{score}(q, d) = \max_{t \in d} \text{sim}(q, t) \tag{8}$$

where t represents a token in the document passage d, and $\text{sim}(q, t)$ measures the cosine similarity between the query q and the token t. The max-pooling operation selects the token that is most semantically similar to the query, which reduces the number of comparisons that need to be made, significantly improving retrieval efficiency.

Unlike traditional BERT-based retrieval systems, which perform full interactions between the query and every document, ColBERT introduces a novel two-phase mechanism, including

a late interaction between the query and document representations. This approach balances the need for fine-grained semantic understanding with computational efficiency.

In the first phase, ColBERT generates dense, fixed-length representations for both the query and document passages using the BERT model. In the second phase, instead of performing expensive, full interactions between the query and all documents, ColBERT applies a fast interaction technique where the query is compared to a compressed version of the document passage, typically by using max-pooling over the token embeddings of the document. This allows the model to reduce the computational cost significantly while still benefiting from the rich semantic information contained in the dense embeddings.

ColBERT has shown to be particularly effective in large-scale retrieval tasks, where traditional full-interaction models like BERT can be prohibitively slow. By applying late interaction techniques, ColBERT achieves a good trade-off between the high-quality semantic retrieval of BERT-based models and the computational efficiency of traditional sparse methods. Additionally, ColBERT can be easily integrated with existing sparse retrieval systems, making it highly adaptable for real-world applications.

While ColBERT improves efficiency, it still requires substantial training time and resources, especially when dealing with large-scale document corpora. Moreover, like other dense retrieval methods, it benefits from the use of approximate nearest neighbor search techniques for efficient retrieval during inference.

4.3. Remarks

The evolution from sparse retrieval methods to dense retrieval techniques such as DPR, BERT-based retrievers, and ColBERT marks a significant leap forward in the field of information retrieval. Dense retrieval models provide superior semantic understanding by leveraging powerful transformer-based models like BERT, enabling more accurate retrieval of relevant documents. However, they come with trade-offs in terms of computational complexity and scalability. As retrieval systems continue to evolve, hybrid approaches that combine the efficiency of sparse methods with the rich semantic representations of dense models are likely to become the standard. Models like ColBERT, which offer a balance between accuracy and efficiency, represent an exciting direction for the future of large-scale retrieval tasks.

5. Generator Methodologies

In retrieval-augmented generation (RAG) systems, the generator plays a critical role by synthesizing final outputs from retrieved evidence. These generators are typically implemented using large language models (LLMs) capable of producing coherent, contextually appropriate, and semantically rich responses. This section examines four foundational models, BART, BERT as a generator, GPT, and T5, analyzing their architectures, generation capabilities, and theoretical underpinnings within the context of generative NLP.

5.1. BART: Bidirectional and Auto-Regressive Transformer

BART (Bidirectional and Auto-Regressive Transformer) represents a hybrid model architecture that combines the strengths of both bidirectional and autoregressive training objectives, making it particularly suitable for generation tasks in RAG pipelines. The BART model follows an encoder-decoder architecture, where the encoder operates in a denoising autoencoder setup and the decoder functions autoregressively.

The training objective of BART centers on reconstructing corrupted inputs. Let x denote an original sequence and $\tilde{x}$ be its noised version. The model is trained to minimize the negative log-likelihood of reconstructing x from $\tilde{x}$ using the encoder-decoder pair (E, D), where:

$$\mathcal{L}_{\text{BART}} = -\sum_{t=1}^{|x|} \log P(x_t | x_{<t}, E(\tilde{x})) \tag{9}$$

This formulation allows BART to generalize over a wide range of sequence generation tasks, including abstractive summarization, translation, and open-domain question answering.

BART's encoder benefits from deep bidirectional context modeling, similar to BERT, while its decoder employs autoregressive generation akin to GPT. This dual configuration grants BART strong performance across both comprehension and generation tasks. In retrieval-augmented systems, BART excels by conditioning its generation on retrieved documents, using the encoder to integrate contextual passages and the decoder to generate fluent, informative answers.

5.2. BERT (Bidirectional Encoder Representations from Transformers)

While BERT was originally conceived as an encoder-only model for representation learning, various adaptations have enabled its use in generation tasks. BERT's bidirectional masked language modeling (MLM) pretraining objective does not inherently support left-to-right generation, as is standard in autoregressive models. However, it can be repurposed for generation through techniques such as mask-filling or by serving as an encoder within an encoder-decoder framework. The pretraining objective of BERT can be formulated as

$$\mathcal{L}_{\text{MLM}} = -\sum_{i \in \mathcal{M}} \log P(x_i | x_{\setminus \mathcal{M}}) \tag{10}$$

where $\mathcal{M}$ is the set of masked positions in the input sequence x. This formulation trains the model to recover masked tokens using bidirectional context, but does not directly model sequential dependencies needed for generative output.

To adapt BERT for generation, one common approach involves coupling it with a transformer decoder or applying strategies like iterative refinement, where BERT successively refines partially completed sequences. Another approach is the use of infilling generation, where parts of the sequence are iteratively predicted, drawing on BERT's ability to condition on both left and right context. Despite these efforts, BERT remains more effective as a retriever or encoder than a standalone generator, especially compared to models natively trained for left-to-right generation.

5.3. GPT (Generative Pretrained Transformer)

The Generative Pretrained Transformer (GPT) is among the most influential autoregressive language models used for generation in RAG systems. GPT utilizes a unidirectional transformer decoder architecture, pretrained on large corpora using a causal language modeling (CLM [87, 88, 89]) objective. This architecture enables the model to predict the next token in a sequence, making it naturally aligned with text generation tasks.

Formally, the GPT training objective minimizes the negative log-likelihood of the next token given the preceding context:

$$\mathcal{L}_{\text{GPT}} = -\sum_{t=1}^{|x|} \log P(x_t | x_{<t}) \tag{11}$$

This autoregressive nature allows GPT to model long-range dependencies and produce coherent, sequentially logical text outputs. Unlike bidirectional models, GPT does not condition on future context during generation, which makes it particularly suitable for real-time, incremental generation in applications such as dialogue systems and question answering.

GPT's ability to generalize across diverse generative tasks stems from its unsupervised pretraining over a vast range of internet-scale data. In RAG frameworks, GPT serves as a generator that conditions its output on retrieved documents, often concatenated with the user query. This capacity is further enhanced by fine-tuning or prompt engineering techniques that steer GPT's responses based on the structure and intent of the task.

The latest versions of GPT, such as GPT-3 and GPT-4, are equipped with few-shot and zero-shot capabilities, enabling them to adapt to new generation tasks without explicit training. Their autoregressive architecture and massive parameter count allow them to serve as universal generators across various domains and downstream applications.

5.4. T5 (Text-to-Text Transfer Transformer)

The T5 (Text-to-Text Transfer Transformer) model reframes all natural language processing tasks into a unified text-to-text format, including both inputs and outputs as text strings. This architectural design allows T5 to handle classification, summarization, translation, and generation uniformly under a single framework. T5 uses a standard encoder-decoder transformer architecture and is pretrained on a denoising objective similar to that of BART.

The training loss for T5 is based on a span corruption task. Given an input text sequence x, spans of tokens are randomly masked and replaced with sentinel tokens. The model then learns to reconstruct the missing spans, which is formulated as

$$\mathcal{L}_{\text{T5}} = -\sum_{t=1}^{|y|} \log P(y_t | y_{<t}, E(\tilde{x})) \tag{12}$$

in which $E(\tilde{x})$ denotes the encoding of the corrupted input and y is the target sequence that reconstructs the masked spans. This objective trains T5 to produce semantically meaningful and contextually appropriate sequences conditioned on partial inputs.

T5's text-to-text formulation simplifies the development and deployment of NLP pipelines by converting disparate tasks into a common format. In generative retrieval systems, T5 acts as a powerful generator that can ingest retrieved evidence and user prompts and synthesize detailed responses that are coherent and task-specific.

Variants such as Sentence-T5 have been developed to further improve performance on sentence-level generation and ranking tasks. These variants modify the pretraining or fine-tuning regimes to produce better representations for sentence matching and generation.

T5's flexibility, combined with its competitive performance across benchmarks, makes it one of the versatile generators in retrieval-augmented generation. Its encoder-decoder architecture enables deep conditioning on retrieved evidence, and its pretraining objective equips it with a robust understanding of syntax and semantics across a wide range of contexts.

6. RAG Extensions

6.1. Open-Domain Question Answering

Open-domain question answering (QA) represents a foundational application area for retrieval-augmented generation, where models are tasked with answering arbitrary questions

by accessing a large unstructured corpus rather than relying solely on internal parametric knowledge. Traditional language models often struggle with factual consistency in this setting due to their inability to update knowledge dynamically. RAG-based approaches mitigate this by grounding generated answers in retrieved documents, thereby combining parametric fluency with non-parametric factual access.

Formally, let q denote a user query. A retriever $R(q)$ returns a set of candidate passages $\{p_i\}_{i=1}^{k}$, and a generator G synthesizes the final answer a based on q and $\{p_i\}$, optimizing:

$$a = \arg\max_{a'} P(a'|q, \{p_i\}_{i=1}^{k}) \tag{13}$$

This design has demonstrated significant improvements on established benchmarks such as Natural Questions [41] and WebQuestions [42], where the factual grounding provided by the retrieval component enables superior precision and recall. Early contributions to this paradigm, including work by [25, 26], helped solidify the structure of retrieval-based QA pipelines, emphasizing the separation of retrieval and generation as distinct yet complementary stages.

6.2. DrQA

DrQA, one of the earliest modern implementations of retrieval-augmented QA, established a clear modular separation between document retrieval and machine reading comprehension. As a Retrieval-Then-Read Framework, it retrieves relevant documents using a TF-IDF based retriever and subsequently feeds them into a neural reader trained to extract answer spans. The architecture is formally decomposed as follows: given a question q, retrieve documents $D = R(q)$, and then extract an answer span $a \in D$ by maximizing $P(a|D, q)$.

The reader in DrQA is typically a neural model such as a bi-directional LSTM or a transformer trained using supervised span extraction objectives on datasets like SQuAD [90, 91, 92]. This pipeline significantly improved performance on datasets requiring large knowledge bases, such as CoQA [28, 93] and the original SQuAD. While the retrieval component remains relatively naive compared to dense retrievers, the pipeline's simplicity and modularity set the foundation for subsequent RAG innovations.

6.3. RAG with Feedback Loops

Recent enhancements to the RAG framework incorporate feedback mechanisms that allow information to flow from the generator back to the retriever. This interaction is motivated by the observation that initial retrievals may not be optimal for answer generation, and the generator's contextual needs can guide more informed second-stage retrieval.

One such extension is the Fusion-in-Decoder (FiD [43, 44, 94]) architecture, which processes multiple retrieved passages jointly within a single decoder. In contrast to early-fusion methods that concatenate passages before encoding, FiD encodes each passage independently using a shared encoder and performs late fusion at the decoding stage. This architectural choice preserves the distinct semantic contributions of each passage while enabling the decoder to attend globally across all encoded inputs.

Formally, let q denote the input query and $\{p_1, p_2, \ldots, p_k\}$ be the top-k retrieved passages. Each passage p_i is encoded independently to produce a contextual representation $E(p_i)$. The decoder is then conditioned jointly on the query and the encoded passages to generate the final answer. Let a' denote a candidate answer sequence and a be the optimal output. The model maximizes the conditional probability

$$a = \arg\max_{a'} P(a' \mid q, E(p_1), \ldots, E(p_k)) \tag{14}$$

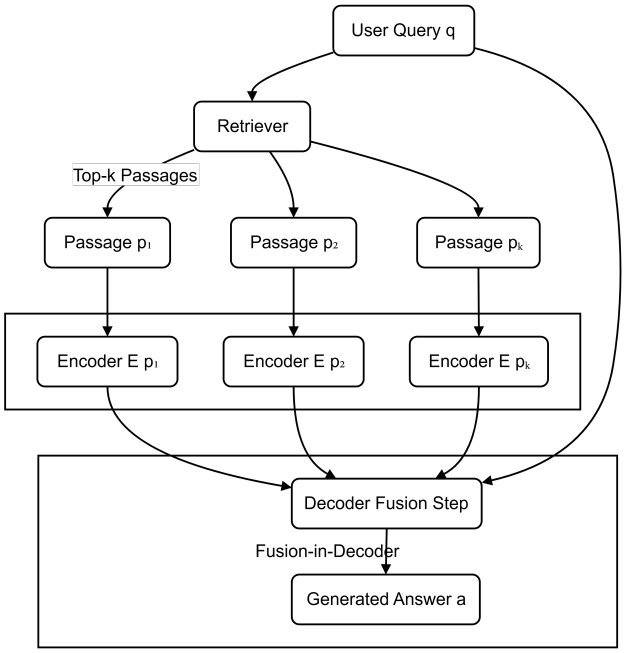

Figure 8: Fusion in Decoder Logic Flow

in which a represents the generated answer that maximizes the likelihood under the model given the query and the encoded representations of the retrieved passages. The formulation reflects the autoregressive decoding objective typical of sequence-to-sequence models, while the fusion mechanism in the decoder ensures that evidence from multiple passages is jointly leveraged without interference at the encoder stage.

Figure 8 presents the architecture of the Fusion-in-Decoder (FiD) model, a representative late-interaction framework in retrieval-augmented generation. The process begins with a retriever module that selects the top-k relevant passages $\{p_1, p_2, \ldots, p_k\}$ corresponding to the input query q. Each passage p_i is then independently encoded by a shared encoder, yielding a contextual embedding $E(p_i)$. This independent encoding strategy preserves the semantic fidelity of each passage and prevents early-stage interference among retrieved documents, in contrast to early-fusion approaches that concatenate passages prior to encoding.

Following the encoding stage, the decoder is jointly conditioned on the query and the set of encoded passages. The fusion of information from multiple sources is deferred to the decoding phase, allowing the decoder to attend over the entire collection of encoded passages simultaneously. This late fusion mechanism enhances the model's capacity to aggregate evidence across disjoint or partially overlapping information sources. Formally, the decoder generates the most probable answer a by solving the optimization objective (14) where a' ranges over all candidate output sequences.

This formulation follows the autoregressive sequence generation paradigm, where the decoder incrementally generates the answer token-by-token based on the conditional probability distribution. The FiD architecture is particularly advantageous for multi-hop reasoning and open-domain question answering, as it allows for fine-grained, token-level integration of retrieved content during generation, thus effectively capturing long-range dependencies across passages.

In parallel, RAG-Fusion introduces a bidirectional interaction between retrieval and gener-

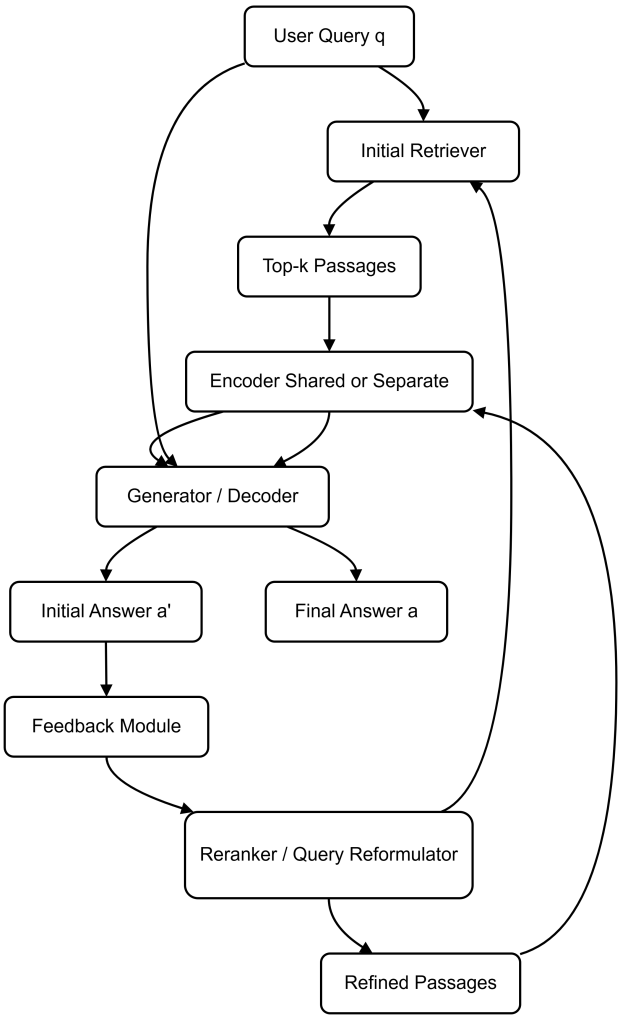

Figure 9: RAG Fusion Logic Flow

ation stages [45, 46]. The initial generation result informs retrieval reranking or reformulation, creating a feedback loop that refines the relevance of retrieved content. This feedback-enhanced retrieval operates under the assumption that generation-aware signals (e.g., confidence scores or attention weights) can indicate retrieval deficiencies. These techniques have shown promise in refining retrieval granularity, particularly when answer evidence is diffuse or implicitly expressed.

Figure 9 illustrates the component-level architecture of RAG-Fusion, an extension of the standard RAG framework that introduces bidirectional interaction between the retriever and generator. Its feedback-enhanced retrieval shows that the initial generation informs a secondary retrieval phase that is more aligned with generation requirements. The process begins with a user query q, which is passed to an initial retriever that fetches a set of top-k passages. These passages are then encoded and fed into the generator, which produces an initial response a'.

Unlike traditional RAG pipelines, RAG-Fusion includes a feedback module that evaluates the quality of the initial generation. This module extracts generator-informed signals, such as attention weights, confidence scores, or token-level entropy, which are indicative of generation uncertainty or retrieval insufficiency. These signals are used to drive a secondary retrieval

process through reranking or query reformulation, thereby producing a refined set of passages that better align with the generative intent.

The revised passages are re-encoded and passed back into the generator, which then produces the final answer a. This iterative feedback loop allows the system to condition retrieval on partial decoding outputs, effectively closing the retrieval-generation loop and improving the relevance and coherence of the final response. This architecture is particularly advantageous when the answer spans multiple documents or when the information required is implicit, distributed, or requires higher-order reasoning.

The feedback loop also introduces new optimization challenges. For example, feedback signals must be carefully designed to be informative and stable, as noisy gradients or poorly calibrated confidence scores can degrade reranking. Nevertheless, empirical studies [45, 46] suggest that such feedback mechanisms substantially enhance retrieval granularity and downstream generation quality in complex, open-domain QA settings.

6.4. Hybrid and Feedback-Enhanced Retrieval Mechanisms

Beyond linear retrieval-generation pipelines, recent work has explored hybrid systems that integrate sparse and dense retrieval or introduce reranking modules for relevance optimization. These approaches seek to leverage the complementary strengths of different retrieval signals while enhancing the precision of evidence selection.

For instance, a hybrid retriever may initially combine BM25 with a dense retriever (e.g., DPR) using score interpolation or weighted fusion. Once candidate passages are retrieved, a reranker model, often based on a cross-encoder, is applied to refine the top-k selection by re-evaluating the semantic fit between the query and each passage:

$$s_i = f_{\text{reranker}}(q, p_i) \tag{15}$$

where s_i is a relevance score and f is a transformer-based scoring function. This reranking step has demonstrated substantial improvements in both precision@1 and answer exact match metrics, as shown in [56, 57].

Further complexity is introduced by hierarchical retrieval models, which first identify coarse-level clusters or topics before performing fine-grained retrieval within selected partitions. This two-stage hierarchical architecture reduces search space while maintaining high recall, and it is mathematically modeled as optimizing:

$$\text{cluster}(q) \rightarrow \text{top-}k(p_i \in \mathcal{C}_q) \tag{16}$$

where $\mathcal{C}_q$ is the relevant cluster for query q, and passages p_i are selected from within $\mathcal{C}_q$. The use of hierarchical encodings further improves the model's efficiency and scalability on large corpora, as explored in [58, 59, 60].

6.5. Multi-Hop and Conversational RAG

Traditional retrieval-augmented models often assume that a single passage contains sufficient information to answer a query. However, real-world queries may require reasoning across multiple documents or adapting to multi-turn conversational history. Multi-hop RAG architectures extend the standard paradigm by enabling sequential retrieval steps, where each step conditions on previously retrieved content. Let $R^{(1)}(q)$ be the initial retrieval and $R^{(2)}(q, p_1)$ be

a second-hop retrieval conditioned on the first passage p_1. The final answer is then generated by:

$$a = G(q, R^{(1)}(q), R^{(2)}(q, R^{(1)}(q))) \tag{17}$$

Such architectures allow the model to aggregate and reason over distributed evidence, essential for answering questions that require logical inference or synthesis from multiple sources.

Conversational RAG further generalizes the architecture to handle dialogue systems where the query q evolves over time as a function of conversation history h_t. The model retrieves based on $h_t \cup q_t$, adapting the retrieval strategy dynamically to follow the evolving discourse context. This framework is particularly effective for customer support, task-oriented dialogue, and information-seeking interactions that span multiple conversational turns.

7. Evaluation Metrics

Evaluating Retrieval-Augmented Generation (RAG) systems demands a multifaceted approach that incorporates both traditional natural language generation metrics and retrieval-specific diagnostics. A comprehensive evaluation must assess the factual correctness, linguistic quality, and retrieval alignment of generated responses. This section provides an in-depth analysis of the principal evaluation metrics used in the literature.

7.1. Exact Match

The Exact Match (EM) metric evaluates whether the generated answer is textually identical to a ground-truth reference answer. Formally, let a_{pred} be the predicted answer and a_{gold} be the gold standard. The EM score is defined as:

$$\text{EM}(a_{\text{pred}}, a_{\text{gold}}) = \begin{cases} 1 & \text{if } a_{\text{pred}} = a_{\text{gold}} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

The overall EM score is then the mean across all examples in the evaluation set. EM is particularly suitable for factoid-style question answering where responses must be precise. However, it tends to be overly strict in the presence of minor linguistic variations or paraphrasing, thereby underestimating the model's true semantic accuracy.

7.2. F1 Score (QA Quality)

To capture partial correctness and semantic overlap, the F1 score is frequently employed in RAG-based QA systems. It computes the harmonic mean of precision and recall at the token level. Let T_{pred} and T_{gold} denote the sets of tokens in the predicted and reference answers, respectively. Then:

$$\text{Precision} = \frac{|T_{\text{pred}} \cap T_{\text{gold}}|}{|T_{\text{pred}}|}, \quad \text{Recall} = \frac{|T_{\text{pred}} \cap T_{\text{gold}}|}{|T_{\text{gold}}|} \tag{19}$$

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{20}$$

F1 provides a more forgiving assessment than EM and is thus better aligned with human judgments, particularly in open-ended generation settings where multiple correct answers may exist.

7.3. BLEU and ROUGE (Generation Fluency)

For evaluating the fluency and stylistic similarity of generated content relative to reference texts, BLEU [95, 96, 97] and ROUGE [98, 99, 100] scores are widely adopted. The BLEU score measures n-gram precision between the generated and reference texts, typically up to four-grams, and penalizes overly short outputs via a brevity penalty:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{4} w_n \log p_n\right) \tag{21}$$

where p_n is the modified precision for n-grams and w_n are uniform weights.

The *Brevity Penalty* (BP) in the BLEU metric is introduced to penalize translations that are shorter than the reference, which could otherwise achieve artificially high precision scores. The penalty is defined by the following equation:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ \exp\left(1 - \frac{r}{c}\right) & \text{if } c \leq r \end{cases} \tag{22}$$

in which c denotes the total length of the candidate translation, measured in tokens. The variable r represents the effective reference length, which is typically chosen as the length of the reference sentence that is closest in length to the candidate. When the candidate is longer than the reference, the brevity penalty is set to 1, meaning no penalty is applied. However, if the candidate is shorter, the penalty decreases the overall BLEU score exponentially, with greater penalties for greater length disparities. This formulation ensures that generated translations are not only precise in terms of local n-gram matches, but also adequate in length, encouraging complete and fluent outputs.

In contrast, ROUGE, particularly ROUGE-L, is based on recall of longest common subsequences (LCS), which better captures phrasal alignment and is more robust to free-form paraphrasing. ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation – Longest Common Subsequence) is a widely used evaluation metric that captures phrasal and sequential overlap between a generated text and a reference text. Unlike precision-based n-gram metrics such as BLEU, ROUGE-L relies on the Longest Common Subsequence (LCS), making it more tolerant to free-form paraphrasing and syntactic variation. It is especially suitable for tasks such as summarization and response generation in retrieval-augmented generation (RAG) frameworks, where lexical variety is common.

Let $X = x_1, x_2, \ldots, x_m$ be the sequence of tokens in the reference text and $Y = y_1, y_2, \ldots, y_n$ be the sequence of tokens in the candidate (generated) text. The Longest Common Subsequence (LCS) between X and Y is the longest sequence of tokens that appears in both X and Y in the same order, but not necessarily contiguously.

Define $LCS(X, Y)$ as the length of the longest common subsequence. The recall-oriented ROUGE-L metric is computed using the following formula

$$\text{ROUGE-L}_{\text{recall}} = \frac{LCS(X, Y)}{m} \tag{23}$$

Similarly, a precision-oriented version can be expressed as

$$\text{ROUGE-L}_{\text{precision}} = \frac{LCS(X, Y)}{n} \tag{24}$$

To obtain a balanced score that accounts for both recall and precision, the F-measure (or F1 score) of ROUGE-L is used

$$\text{ROUGE-L}_{F1} = \frac{(1+\beta^2) \cdot \text{ROUGE-L}_{\text{precision}} \cdot \text{ROUGE-L}_{\text{recall}}}{\text{ROUGE-L}_{\text{precision}} + \beta^2 \cdot \text{ROUGE-L}_{\text{recall}}} \tag{25}$$

where β is a parameter that determines the relative importance of recall versus precision. In most implementations, $\beta = 1$ is used, giving equal weight to both.

The use of LCS in ROUGE-L allows it to align phrasal structures across reference and candidate texts even when they are not exact matches in contiguous spans. This feature makes ROUGE-L particularly well-suited for evaluating the fluency and coverage of RAG-generated responses, which often involve varied and paraphrased expressions that cannot be effectively captured by exact matching methods alone.

While these metrics were originally developed for machine translation and summarization, they remain useful proxies for output fluency and lexical overlap in RAG-generated responses.

7.4. Recall@k (Retrieval Accuracy)

Since retrieval constitutes a critical sub-component of RAG, retrieval accuracy must be independently evaluated. Recall@k measures whether the relevant document appears among the top k retrieved results for a given query. If D_{gold} is the ground-truth evidence and $\{D_1, \ldots, D_k\}$ are the top-k retrieved passages, then:

$$\text{Recall@}k = \begin{cases} 1 & \text{if } D_{\text{gold}} \in \{D_1, \ldots, D_k\} \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

Averaging this indicator across a dataset yields the overall recall. High Recall@k is a prerequisite for downstream answer correctness in RAG pipelines and provides insight into the retriever's capacity to localize supporting evidence.

7.5. Groundedness

Groundedness refers to the degree to which generated content is supported by retrieved documents. This metric is crucial in preventing hallucination, where the model generates fluent but fabricated content. Formally, for a generated response a and retrieved set $\{p_i\}$, groundedness can be defined as a binary or probabilistic measure of whether all factual claims in a are entailed by $\{p_i\}$. Evaluating groundedness typically involves human annotation or automatic methods such as natural language inference (NLI [101, 102, 103, 104]) or entailment scoring [105, 106, 107]. Groundedness metrics are essential for high-stakes applications like medical or legal QA, where factual alignment is paramount.

7.6. Latency and Cost

In addition to quality metrics, RAG systems are often evaluated on efficiency metrics such as inference latency and computational cost. Let T_{ret} and T_{gen} denote the time spent in retrieval and generation respectively. The total latency T_{total} is given by:

$$T_{\text{total}} = T_{\text{ret}} + T_{\text{gen}} \tag{27}$$

In scenarios requiring real-time interaction, such as dialogue systems or search interfaces, reducing T_{total} is critical. Moreover, computational cost, often measured in FLOPs or GPU-hours, becomes a limiting factor in large-scale deployment. Trade-offs between retrieval depth, generation complexity, and system throughput must be carefully balanced to meet practical constraints without compromising output quality.

7.7. Comparative Analysis of Evaluation Metrics

The performance of Retrieval-Augmented Generation (RAG) systems is inherently multifaceted, requiring evaluation across both the retrieval and generation stages. Each metric employed in this context reflects a different axis of system behavior, and understanding their complementary roles is essential for robust benchmarking and system development.

The *Exact Match (EM)* metric provides a binary assessment of correctness, marking a prediction as valid only if it exactly matches the ground-truth answer. While this strictness ensures high precision for factoid-style queries, it often penalizes semantically correct paraphrases or alternative surface forms, especially in natural language generation. In contrast, the *F1 score* softens this rigidity by accounting for the degree of token overlap between the generated and reference answers. This makes it more suit able for evaluating responses with partial correctness, particularly in QA settings with longer or multi-token answers.

Metrics such as *BLEU* and *ROUGE* offer insights into linguistic fluency and stylistic fidelity. BLEU emphasizes precision by measuring n-gram overlap, often favoring shorter answers, whereas ROUGE, especially ROUGE-L, captures sequence-level recall, better aligning with longer summarizations. However, both metrics fall short in assessing factual consistency or grounding, making them less informative in scenarios where truthfulness is critical.

Retrieval quality is assessed via *Recall@k*, which verifies whether the relevant passage appears within the top-k retrieved documents. While essential for diagnosing retriever effectiveness, this metric does not reflect how the generator utilizes the retrieved content. Moreover, high recall does not guarantee the generation of accurate answers, particularly in noisy retrieval scenarios.

The notion of *groundedness* has emerged as a vital metric for RAG systems, especially in mitigating hallucination and ensuring factual reliability. Groundedness quantifies the alignment between generated content and retrieved evidence, typically through human judgment or entailment models. However, this metric is inherently subjective and often lacks robust, fully automated implementations.

From a systems perspective, *latency* and *computational cost* are indispensable for evaluating deployment feasibility. These metrics do not assess correctness but are essential in balancing retrieval depth, passage reranking, and decoder complexity for scalable inference.

Table 1 summarizes these metrics across several key dimensions, including what they measure, their strengths, known limitations, and typical application contexts.

As evident in Table 1, no single metric provides a complete picture of system performance. Therefore, comprehensive RAG evaluation typically involves a combination of these metrics to balance correctness, fluency, retrieval relevance, and operational feasibility. For instance, high Recall@k must be complemented by groundedness and F1 to ensure that the retrieved documents are not only relevant but are also used effectively in generating accurate and faithful responses. Similarly, latency metrics must be considered in production systems where response time is a critical factor alongside model quality.

8. Applications

The deployment of Retrieval-Augmented Generation (RAG) architectures has expanded rapidly across diverse domains, each presenting unique challenges in reliability, factual grounding, and domain-specific reasoning. These applications share a core dependence on accurate retrieval of contextually appropriate information, yet the nature of this retrieval, as well as the standards of adequacy, varies considerably depending on the target domain.

Table 1: Comparison on Evaluation Metrics in RAG Systems

Metric	**Aspect Measured**	**Strengths**	**Limitations**	**Typical Use Case**
Exact Match (EM)	Answer correctness (binary)	High precision, interpretable	Penalizes paraphrases, no partial credit	Short-answer QA, factoid datasets
F1 Score	Token-level overlap	Partial credit, better recall than EM	Sensitive to word order, lacks semantics	Passage-level QA, extractive answers
BLEU / ROUGE	Fluency and lexical similarity	Measures surface fluency, widely adopted	Ignores factual correctness, shallow overlap	Summarization, paraphrase generation
Recall@k	Retrieval accuracy	Captures retriever effectiveness, easy to compute	No insight into generator behavior	Retriever diagnostics, pipeline tuning
Groundedness	Factual support from retrieved evidence	Evaluates hallucination, fact alignment	Often needs human or entailment models	Trustworthy generation, knowledge-intensive tasks
Latency / Cost	Runtime and inference cost	Reflects system efficiency	Not task-specific, ignores semantic quality	Production deployment, efficiency benchmarking

8.1. Enterprise Search Assistants

In enterprise environments, large-scale systems with Retrieval have redefined the paradigm of internal knowledge access [108, 109, 110]. These systems serve as search assistants capable of surfacing proprietary content, technical documentation, and operational data in response to natural language queries. The primary concern in this setting revolves around the faithful grounding of generated content in structured knowledge bases and internal documentation [111, 112, 113]. Enterprises typically demand high reliability and low hallucination rates, as generated outputs can influence decision-making processes at scale. The sensitivity of enterprise data also introduces constraints on retrieval scope, emphasizing the need for secure indexing, fine-grained access control, and robust query reformulation techniques that are aware of organizational context.

8.2. Academic Research Assistants

RAG systems have found increasing utility in the domain of academic research [114, 115, 116]. These systems aim to assist researchers by summarizing articles, extracting claims, identifying citations, and proposing relevant literature. The scientific setting presents particular challenges due to the domain-specific terminology, evolving conceptual frameworks, and the requirement for epistemic humility in generated answers. Studies have highlighted risks of misrepresentation, oversimplification, or hallucinated citations when domain-adapted retrieval fails to retrieve sufficiently precise evidence [16, 17, 18]. Moreover, the dynamic and iterative nature of research inquiry demands flexible retrieval strategies that can evolve over the course of exploratory search and hypothesis refinement.

8.3. Legal QA

Legal question answering is a domain where precision and precedent are paramount [117, 118, 119]. RAG systems in this context are tasked with retrieving relevant statutes, case law, or legal commentary and generating reasoned summaries or argument structures [120, 121, 122]. Unlike general-domain QA, legal systems must conform to jurisdiction-specific constraints and preserve legal logic in output generation [123, 124, 125]. Recent work has identified the challenges of maintaining logical consistency, accurate referencing, and non-biased interpretation of ambiguous legal text [14, 15, 120, 121, 124, 125]. Additionally, RAG models in this space must handle documents with complex structure and interpretive ambiguity, raising the bar for both retrieval granularity and generative reasoning fidelity.

8.4. Healthcare QA

In clinical and healthcare contexts, RAG systems have been explored for use in supporting medical decision-making, patient education, and summarization of electronic health records [126, 127, 128, 129]. However, the potential consequences of misinformation are profound [130, 131, 132]. Generation must be not only factually grounded but also calibrated to reflect clinical uncertainty, guideline variability, and patient-specific nuances [19, 20, 21]. Furthermore, the retrieval component must effectively navigate structured sources such as SNOMED [133, 134, 135], UMLS [136, 137, 138], or clinical trial databases [139, 140, 141], while also incorporating unstructured clinical narratives. The intersection of domain expertise, ethical safety constraints, and the need for transparent traceability makes healthcare QA one of the most demanding applications of RAG in terms of both precision and accountability.

8.5. Customer Support Chatbots

In customer service, RAG-based chatbots are used to provide automated support across a wide range of industries [142, 143, 144]. These systems typically retrieve FAQ articles, support documentation, or account-specific data to answer user queries. The primary technical challenge lies in optimizing retrieval for high coverage while preserving domain alignment, such that generated responses remain fluent and helpful without introducing false commitments. Customer-facing systems are often evaluated not only by technical correctness but also by measures of user satisfaction and perceived responsiveness, which can be strongly influenced by generation fluency and latency. This application benefits greatly from retriever fine-tuning on historical ticket data and multi-turn conversational grounding.

8.6. Code Generation and Documentation Systems

RAG systems are increasingly integrated into programming assistants and development tools, where they support tasks such as code completion, bug explanation, and documentation synthesis [145, 146, 147, 148]. In such systems, retrieval typically sources from large corpora of open-source code or API documentation. The success of generation depends critically on accurately identifying relevant functions, usage patterns, and edge-case behaviors. A major concern in this context is the generation of insecure or deprecated code due to poor retrieval or model hallucination. Effective systems must therefore integrate retrieval with static or dynamic code analysis pipelines, ensuring syntactic correctness and semantic robustness. This domain exemplifies the synergy between symbolic and neural reasoning, with retrieval acting as a bridge between learned representations and formal code semantics.

Across these applications, RAG continues to evolve toward more adaptive and domain-aware paradigms. The increasing use of hybrid retrievers, iterative feedback mechanisms,

and retriever-generator co-training reflects a broader shift toward retrieval systems that are sensitive not only to input queries but also to generation objectives and contextual demands. The integration of such techniques is essential to meet the varying standards of truthfulness, utility, and interpretability demanded by high-stakes real-world deployments.

9. Challenges, Limitations, and System Constraints

Despite the substantial advances introduced by Retrieval-Augmented Generation (RAG) frameworks, their deployment in real-world settings is accompanied by a host of architectural and epistemic challenges. These arise both from the hybrid dual-stage nature of RAG, comprising a retriever and a generator, and from the underlying constraints inherited from large language models (LLMs). In what follows, we examine key limitations associated with latency, context management, factual consistency, temporal knowledge stability, and system scalability.

9.1. Latency

One of the most fundamental challenges in RAG is latency [149, 50]. Unlike end-to-end parametric models that generate responses directly from internal representations, RAG systems require an initial retrieval stage to surface relevant textual evidence before generation can proceed. This dual-stage pipeline introduces non-trivial delays, particularly when retrieval must search across large document indices or invoke re-ranking heuristics. Moreover, latency compounds under high-concurrency settings, affecting user experience and system throughput. Recent work has sought to quantify and mitigate this overhead by proposing compressed retrieval pipelines, caching mechanisms, and retriever acceleration strategies [49, 50, 51]. Nevertheless, in latency-sensitive domains such as real-time dialogue systems or customer service chatbots, achieving optimal trade-offs between retrieval quality and system responsiveness remains an unresolved tension.

9.2. Context Selection

Another critical bottleneck in RAG performance lies in the selection and ingestion of contextual evidence [150, 151, 152]. Since LLMs have limited input lengths, not all retrieved documents can be passed into the generator, which necessitates aggressive filtering or ranking of candidate passages. However, if irrelevant or marginally related documents are included, they can distract the generator or anchor it to spurious content, thereby degrading response quality. Conversely, excluding key passages risks omitting necessary evidence. This challenge is exacerbated in tasks involving long documents, overlapping evidence, or ambiguous queries [55, 50]. Furthermore, due to fixed token constraints, truncation or compression strategies may distort meaning, undermining the model's ability to generate faithful and coherent responses. Thus, context selection is not merely a retrieval task but a delicate optimization problem involving informativeness, relevance, and budget-aware encoding.

9.3. Factual Consistency

While RAG is designed to mitigate hallucination by grounding generation in external documents, factual inconsistency remains a persistent issue [153, 154, 155]. Even when relevant evidence is retrieved, the generator may ignore it, misinterpret it, or hallucinate unsupported content, particularly when retrieval is noisy or sparse. This disconnect arises in part because

the generator is still a large autoregressive model trained to maximize likelihood, not necessarily factual accuracy. Generator-level hallucination has been linked to model overconfidence, exposure bias, and poor calibration [52, 53, 54]. Furthermore, when irrelevant or contradictory evidence is retrieved, it can actively degrade generation quality, leading to confident but incorrect responses. Ensuring factual consistency thus requires not only effective retrieval but also alignment between retrieved content and generation focus. Approaches such as answer-aware re-ranking and confidence-controlled decoding are being explored to mitigate these issues.

9.4. Knowledge Drift

A further limitation arises from the temporal stasis of RAG's retrieval corpus [156, 157, 158]. If the document index is not regularly updated, the retrieved evidence may become outdated, introducing knowledge drift into generated responses. This is particularly problematic for domains that evolve rapidly, such as law, medicine, or technology. Even if the generator is capable of producing fluent and coherent answers, the absence of temporally relevant grounding can result in outdated, misleading, or unsafe outputs. Static corpora, by design, cannot reflect emergent facts or corrections, and continual index maintenance becomes both a technical and operational burden. This constraint has renewed interest in hybrid systems that combine fixed retrieval indices with online, dynamic sources or crowd-sourced updates.

9.5. Limitations of Parametric LLMs

RAG architectures were originally motivated in part by the limitations of parametric language models, particularly their inability to access knowledge not seen during pretraining [159, 160]. Closed-book models are known to hallucinate plausible-sounding but factually incorrect content, especially when queried on rare or specialized topics [11, 12, 13]. By externalizing factual knowledge through retrieval, RAG attempts to bridge the gap between fluency and factuality. Nevertheless, these parametric limitations persist in the generation module, which remains susceptible to hallucination even when retrieval is correct. Therefore, the retrieval component serves not as a silver bullet but as a necessary corrective mechanism to augment and constrain the generation process.

9.6. Scalability

Scalability is another major concern for RAG deployment, particularly when document indices grow to billions of passages [161, 162]. The computational cost of indexing, updating, and querying such large-scale corpora imposes significant burdens on memory, storage, and retrieval latency. Techniques such as approximate nearest neighbor search (ANNS [163, 164, 165]) and inverted indexing [166, 167, 168] have been employed to alleviate this burden, but these often introduce trade-offs between recall and speed. In addition, large-scale RAG systems must balance retrieval performance with user privacy, index freshness, and compute efficiency. As the demand for personalized or domain-specific retrieval grows, scalable and adaptive retriever architectures remain an area of active research.

9.7. Remarks

These challenges highlight that RAG systems, while powerful, inherit both the epistemic limitations of language models and the infrastructural complexities of large-scale retrieval systems. Addressing these issues requires a holistic understanding of interaction dynamics between retrieval and generation, along with practical engineering constraints that govern latency, scalability, and robustness in real-world applications.

10. Future Directions

As Retrieval-Augmented Generation (RAG) systems continue to evolve, several promising research trajectories are emerging that aim to enhance the adaptability, expressiveness, and contextual richness of RAG-based architectures. These directions reflect both the growing demand for domain-specific intelligence and the limitations of current systems in handling dynamic, multimodal, and low-resource environments.

10.1. Personalized RAG

One prominent direction involves the development of personalized RAG systems, which tailor retrieval and generation to individual user profiles [169, 170, 171]. In contrast to generic retrieval, personalized RAG dynamically conditions the retriever on user history, preferences, or contextual metadata to surface documents that are not just relevant to the query but also aligned with the user's intent and prior knowledge. This necessitates modeling users as probabilistic entities and incorporating long-term user representations into the retrieval stage. Challenges remain in preserving privacy and avoiding bias propagation, yet the potential benefits are substantial, particularly for recommendation, education, and personal digital assistants.

10.2. Multimodal RAG

Another major advancement lies in extending RAG beyond text to accommodate multimodal inputs, including images, tables, and videos. Multimodal RAG aims to enable systems that can retrieve from heterogeneous content repositories and generate coherent outputs that blend textual and non-textual information [172, 173, 174]. This introduces architectural complexities in both encoding and fusion stages, as each modality requires distinct embedding strategies and cross-attention mechanisms. For instance, integrating vision-language models with textual retrievers demands alignment in representation space and careful co-training protocols. Nevertheless, this direction opens critical applications in document understanding, scientific question answering, and medical diagnostics where visual information plays a central role.

10.3. Structured–Unstructured Fusion

A third frontier is the integration of structured data sources, such as relational databases, SQL queries, and knowledge graphs, into the retrieval process, alongside traditional unstructured corpora. Structured–unstructured fusion enables systems to combine precise, symbolic reasoning with rich contextual grounding [175, 176, 177]. One challenge is designing retrieval mechanisms that can navigate both structured triples and raw textual passages under a unified query representation. Moreover, the fusion of retrieved outputs demands careful calibration between factual correctness from structured data and contextual elaboration from unstructured sources. This hybrid setup is particularly compelling for complex decision-support systems in domains like finance, law, and enterprise knowledge management.

10.4. Low-Resource and Multilingual RAG

RAG models are typically data-hungry and tuned on English-centric corpora, limiting their effectiveness in low-resource [178, 179, 180] and/or multilingual settings [181, 182, 183]. Expanding RAG to underrepresented languages requires innovations in cross-lingual retrieval, translation-augmented pretraining, and transfer learning. Dual-encoder architectures with

shared multilingual embeddings or alignment-based retrieval offer promising avenues. Furthermore, low-resource domains may lack large-scale corpora for both retrieval and generation, necessitating data augmentation techniques or synthetic corpus creation. The challenge is to maintain retrieval relevance and generation fluency across linguistically diverse and underdocumented contexts.

10.5. Dynamic Indexing and Continual Learning

Traditional RAG systems rely on static document indices, which struggle to capture emerging knowledge or adapt to evolving corpora. Future RAG models must incorporate dynamic indexing and continual learning mechanisms that allow for real-time document ingestion, entity updates, and retrieval adaptation [184, 185, 186]. This may include streaming retrieval models [187, 188] that can incrementally update embeddings and data structures, as well as retrieval-aware lifelong learning strategies that prevent catastrophic forgetting. Such capabilities are essential for applications requiring up-to-date factual grounding, such as news summarization, crisis monitoring, and scientific discovery.

10.6. Remarks

The future of RAG lies in its ability to become more adaptive, inclusive, and semantically expressive. Advancements in personalization, multimodality, hybrid data integration, linguistic inclusivity, and dynamic adaptability are poised to shape the next generation of retrieval-augmented systems. These directions not only address existing limitations but also open new research landscapes for intelligent and context-aware generation.

11. Conclusion

This review has systematically examined the landscape of Retrieval-Augmented Generation (RAG), covering foundational methodologies, architectural variants, evaluation metrics, application domains, system challenges, and promising future directions. Beginning with dense retrieval techniques such as Dense Passage Retrieval (DPR), BERT-based dense retrievers, and the contextual late interaction framework of ColBERT, we analyzed the mathematical formulations and practical implications that underpin effective retrieval in RAG systems. The survey further explored generator architectures including BART, BERT, GPT, and T5 models, emphasizing their unique contributions to language generation conditioned on retrieved evidence.

We discussed critical advancements in hybrid and feedback-enhanced retrieval, highlighting how architectures like Fusion-in-Decoder and RAG-Fusion enable iterative refinement of retrieved passages informed by generation outputs. The review also addressed evaluation metrics ranging from exact match and F1 scores for answer accuracy to fluency metrics such as BLEU and ROUGE, alongside retrieval-specific measures like Recall@k and groundedness, providing a nuanced critique of their relative strengths, limitations, and applicability to RAG performance assessment.

Application-wise, RAG's impact spans multiple specialized domains including enterprise search, academic research, legal and healthcare question answering, customer support, and code generation systems. Each area demands tailored approaches to ensure reliability, interpretability, and domain-specific robustness. Challenges related to latency, scalability, factual consistency, and knowledge drift were examined with an emphasis on current mitigation strategies and the need for continued innovation.

Looking forward, the review outlined future research trajectories such as personalized and multimodal RAG, structured-unstructured data fusion, multilingual and low-resource adaptations, and dynamic indexing with continual learning. These directions reflect the growing complexity and user-centric demands of real-world AI deployments.

RAG embodies a principled convergence of retrieval and generation that enhances the factual grounding, contextual relevance, and flexibility of language models. The surveyed literature collectively demonstrates significant progress toward scalable, interpretable, and domain-adaptive systems. As foundational models and retrieval techniques continue to evolve, RAG is poised to remain a cornerstone architecture, driving the next generation of trustworthy, knowledge-aware artificial intelligence applications.

References

[1] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *AI Open*, 5:208–215, 2024.

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

[3] Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30:681–694, 2020.

[4] Manuel Trajtenberg. Artificial intelligence as the next gpt. *The economics of artificial intelligence: An agenda*, 175, 2019.

[5] Benyamin Ghojogh and Ali Ghodsi. Attention mechanism, transformers, bert, and gpt: tutorial and survey. 2020.

[6] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[7] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[8] Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith B Hall, Daniel Cer, and Yinfei Yang. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*, 2021.

[9] Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. Rankt5: Fine-tuning t5 for text ranking with ranking losses. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2308–2313, 2023.

[10] Michael Fu, Chakkrit Tantithamthavorn, Trung Le, Van Nguyen, and Dinh Phung. Vulrepair: a t5-based automated software vulnerability repair. In *Proceedings of the 30th ACM joint european software engineering conference and symposium on the foundations of software engineering*, pages 935–947, 2022.

[11] Annalisa Szymanski, Noah Ziems, Heather A Eicher-Miller, Toby Jia-Jun Li, Meng Jiang, and Ronald A Metoyer. Limitations of the llm-as-a-judge approach for evaluating llm outputs in expert knowledge tasks. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pages 952–966, 2025.

[12] Qian Wang, Jiaying Wu, Zhenheng Tang, Bingqiao Luo, Nuo Chen, Wei Chen, and Bingsheng He. What limits llm-based human simulation: Llms or our design? *arXiv preprint arXiv:2501.08579*, 2025.

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[14] Inyoung Cheong, King Xia, KJ Kevin Feng, Quan Ze Chen, and Amy X Zhang. (a) i am not a lawyer, but...: engaging legal experts towards responsible llm policies for legal advice. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 2454–2469, 2024.

[15] Venkatesh Mishra, Bimsara Pathiraja, Mihir Parmar, Sat Chidananda, Jayanth Srinivasa, Gaowen Liu, Ali Payani, and Chitta Baral. Investigating the shortcomings of llms in step-by-step legal reasoning. *arXiv preprint arXiv:2502.05675*, 2025.

[16] Luca Rossi, Katherine Harrison, and Irina Shklovski. The problems of llm-generated data in social science research. *Sociologica*, 18(2):145–168, 2024.

[17] Ibrahim Al Azher, Venkata Devesh Reddy Seethi, Akhil Pandey Akella, and Hamed Alhoori. Limtopic: Llm-based topic modeling and text summarization for analyzing scientific articles limitations. In *Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries*, pages 1–12, 2024.

[18] Hao-Wen Cheng. Challenges and limitations of chatgpt and artificial intelligence for scientific research: a perspective from organic materials. *Ai*, 4(2):401–405, 2023.

[19] Paul Hager, Friederike Jungmann, Robbie Holland, Kunal Bhagat, Inga Hubrecht, Manuel Knauer, Jakob Vielhauer, Marcus Makowski, Rickmer Braren, Georgios Kaissis, et al. Evaluation and mitigation of the limitations of large language models in clinical decision-making. *Nature medicine*, 30(9):2613–2622, 2024.

[20] Tobias Roeschl, Marie Hoffmann, Djawid Hashemi, Felix Rarreck, Nils Hinrichs, Tobias D Trippel, Matthias I Gröschel, Axel Unbehaun, Christoph Klein, Jörg Kempfert, et al. Assessing the limitations of large language models in clinical practice guideline-concordant treatment decision-making on real-world data. *medRxiv*, pages 2024–11, 2024.

[21] Justin T Reese, Daniel Danis, J Harry Caufield, Tudor Groza, Elena Casiraghi, Giorgio Valentini, Christopher J Mungall, and Peter N Robinson. On the limitations of large language models in clinical diagnosis. *medRxiv*, pages 2023–07, 2024.

[22] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[23] Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992, 2023.

[24] Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. Corrective retrieval augmented generation. 2024.

[25] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*, 2021.

[26] Danqi Chen and Wen-tau Yih. Open-domain question answering. In *Proceedings of the 58th annual meeting of the association for computational linguistics: tutorial abstracts*, pages 34–37, 2020.

[27] John Prager et al. Open-domain question–answering. *Foundations and Trends® in Information Retrieval*, 1(2):91–231, 2007.

[28] Siva Reddy, Danqi Chen, and Christopher D Manning. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

[29] Linlong Xiao, Nanzhi Wang, and Guocai Yang. A reading comprehension style question answering model based on attention mechanism. In *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pages 1–4. IEEE, 2018.

[30] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*, 2017.

[31] Shahzad Qaiser and Ramsha Ali. Text mining: use of tf-idf to examine the relevance of words to documents. *International journal of computer applications*, 181(1):25–29, 2018.

[32] Ho Chung Wu, Robert Wing Pong Luk, Kam Fai Wong, and Kui Lam Kwok. Interpreting tf-idf term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, 26(3):1–37, 2008.

[33] Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple bm25 extension to multiple weighted fields. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49, 2004.

[34] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009.

[35] Andrew Trotman, Antti Puurula, and Blake Burgess. Improvements to bm25 and language models examined. In *Proceedings of the 19th Australasian Document Computing Symposium*, pages 58–65, 2014.

[36] Benjamin Reichman and Larry Heck. Dense passage retrieval: Is it retrieving? *arXiv preprint arXiv:2402.11035*, 2024.

[37] Thilina C Rajapakse. Dense passage retrieval: architectures and augmentation methods. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3494–3494, 2023.

[38] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781, 2020.

[39] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[40] Timo Johner, Abhik Jana, and Chris Biemann. Error analysis of using bart for multi-document summarization: A study for english and german language. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 391–397, 2021.

[41] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

[42] Maria Wang, Srinivas Sunkara, Gilles Baechler, Jason Lin, Yun Zhu, Fedir Zubach, Lei Shu, and Jindong Chen. Webquest: A benchmark for multimodal qa on web page sequences. *arXiv preprint arXiv:2409.13711*, 2024.

[43] Qinyuan Ye, Iz Beltagy, Matthew E Peters, Xiang Ren, and Hannaneh Hajishirzi. Fid-icl: A fusion-in-decoder approach for efficient in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8158–8185, 2023.

[44] Kosuke Akimoto, Kunihiro Takeoka, and Masafumi Oyamada. Context quality matters in training fusion-in-decoder for extractive open-domain question answering. *arXiv preprint arXiv:2403.14197*, 2024.

[45] Zackary Rackauckas. Rag-fusion: a new take on retrieval-augmented generation. *arXiv preprint arXiv:2402.03367*, 2024.

[46] Yuki Momii, Tetsuya Takiguchi, and Yasuo Ariki. Rag-fusion based information retrieval for fact-checking. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 47–54, 2024.

[47] Tom Taulli and Gaurav Deshmukh. Haystack. In *Building Generative AI Agents: Using LangGraph, AutoGen, and CrewAI*, pages 237–249. Springer, 2025.

[48] Vasilios Mavroudis. Langchain. 2024.

[49] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Gui, Ziran Jiang, Ziyu Jiang, et al. Crag-comprehensive rag benchmark. *Advances in Neural Information Processing Systems*, 37:10470–10490, 2024.

[50] Quinn Leng, Jacob Portes, Sam Havens, Matei Zaharia, and Michael Carbin. Long context rag performance of large language models. *arXiv preprint arXiv:2411.03538*, 2024.

[51] Giulio Corallo, Orion Weller, Fabio Petroni, and Paolo Papotti. Beyond rag: Task-aware kv cache compression for comprehensive knowledge reasoning. *arXiv preprint arXiv:2503.04973*, 2025.

[52] Shengming Zhao, Yuheng Huang, Jiayang Song, Zhijie Wang, Chengcheng Wan, and Lei Ma. Towards understanding retrieval accuracy and prompt quality in rag systems. *arXiv preprint arXiv:2411.19463*, 2024.

[53] Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–729, 2024.

[54] Jinyang Wu, Shuai Zhang, Feihu Che, Mingkuan Feng, Chuyuan Zhang, Pengpeng Shao, and Jianhua Tao. Pandora's box or aladdin's lamp: A comprehensive analysis revealing the role of rag noise in large language models. *arXiv preprint arXiv:2408.13533*, 2024.

[55] Kush Juvekar and Anupam Purwar. Introducing a new hyper-parameter for rag: Context window utilization. *arXiv preprint arXiv:2407.19794*, 2024.

[56] Nicholas Ampazis. Improving rag quality for large language models with topic-enhanced reranking. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 74–87. Springer, 2024.

[57] Pradumn Mishra, Aditya Mahakali, and Prasanna Shrinivas Venkataraman. Searchd-advanced retrieval with text generation using large language models and cross encoding re-ranking. In *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, pages 975–980. IEEE, 2024.

[58] Minchae Song. Enhancing rag performance by representing hierarchical nodes in headers for tabular data. *IEEE Access*, 2025.

[59] Pei Liu, Xin Liu, Ruoyu Yao, Junming Liu, Siyuan Meng, Ding Wang, and Jun Ma. Hm-rag: Hierarchical multi-agent multimodal retrieval augmented generation. *arXiv preprint arXiv:2504.12330*, 2025.

[60] Jiajie Jin, Xiaoxi Li, Guanting Dong, Yuyao Zhang, Yutao Zhu, Yongkang Wu, Zhonghua Li, Qi Ye, and Zhicheng Dou. Hierarchical document refinement for long-context retrieval-augmented generation. *arXiv preprint arXiv:2505.10413*, 2025.

[61] Guilherme Moraes Rosa, Ruan Chaves Rodrigues, Roberto Lotufo, and Rodrigo Nogueira. Yes, bm25 is a strong baseline for legal case retrieval. *arXiv preprint arXiv:2105.05686*, 2021.

[62] Rahul Seetharaman, Kaustubh D Dhole, and Aman Bansal. Insertrank: Llms can reason over bm25 scores to improve listwise reranking. *arXiv preprint arXiv:2506.14086*, 2025.

[63] Omar Khattab and Matei 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*, pages 39–48, 2020.

[64] Minjin Choi, Sunkyung Lee, Eunseong Choi, Heesoo Park, Junhyuk Lee, Dongwon Lee, and Jongwuk Lee. Melbert: Metaphor detection via contextualized late interaction using metaphorical identification theories. *arXiv preprint arXiv:2104.13615*, 2021.

[65] Sebastian Hofstätter, Omar Khattab, Sophia Althammer, Mete Sertkan, and Allan Hanbury. Introducing neural bag of whole-words with colberter: Contextualized late interactions using enhanced reduction. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 737–747, 2022.

[66] Hugh A Chipman, Edward I George, and Robert E McCulloch. Bart: Bayesian additive regression trees. 2010.

[67] Veronika Ročková and Enakshi Saha. On theory for bart. In *The 22nd international conference on artificial intelligence and statistics*, pages 2839–2848. PMLR, 2019.

[68] Bhimasen Moharana, Vinay Kumar Singh, Tiyas Sarkar, Dhanpratap Singh, Manik Rakhra, and Vikas Kumar Pandey. Automated questions answering generation system adopting nlp and t5. In *2024 International Conference on Cybernation and Computation (CYBERCOM)*, pages 363–369. IEEE, 2024.

[69] Altaj Virani, Rakesh Yadav, Prachi Sonawane, and Smita Jawale. Automatic question answer generation using t5 and nlp. In *2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, pages 1667–1673. IEEE, 2023.

[70] Min Zhang and Juntao Li. A commentary of gpt-3 in mit technology review 2021. *Fundamental Research*, 1(6):831–833, 2021.

[71] Katharine Sanderson. Gpt-4 is here: what scientists think. *Nature*, 615(7954):773, 2023.

[72] Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. In *NeurIPS*, 2023.

[73] Budi Yulianto, Widodo Budiharto, and Iman Herwidiana Kartowisastro. The performance of boolean retrieval and vector space model in textual information retrieval. *CommIT (Communication and Information Technology) Journal*, 11(1):33–39, 2017.

[74] Stefan Pohl, Alistair Moffat, and Justin Zobel. Efficient extended boolean retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 24(6):1014–1024, 2011.

[75] Arash Habibi Lashkari, Fereshteh Mahdavi, and Vahid Ghomi. A boolean model in information retrieval for search engines. In *2009 International Conference on Information Management and Engineering*, pages 385–389. IEEE, 2009.

[76] Massimo Melucci. Vector-space model. In *Encyclopedia of database systems*, pages 1–6. Springer, 2017.

[77] Vaibhav Kant Singh and Vinay Kumar Singh. Vector space model: an information retrieval system. *Int. J. Adv. Engg. Res. Studies/IV/II/Jan.-March*, 141(143), 2015.

[78] Liping Jing, Michael K Ng, and Joshua Z Huang. Knowledge-based vector space model for text clustering. *Knowledge and information systems*, 25(1):35–55, 2010.

[79] Nouman Azam and JingTao Yao. Comparison of term frequency and document frequency based feature selection metrics in text categorization. *Expert Systems with Applications*, 39(5):4760–4768, 2012.

[80] Elie J Baghdady, Richard N Lincoln, and Bert D Nelin. Short-term frequency stability: Characterization, theory, and measurement. *Proceedings of the IEEE*, 53(7):704–722, 2005.

[81] Deqing Wang, Hui Zhang, Rui Liu, Weifeng Lv, and Datao Wang. t-test feature selection approach based on term frequency for text categorization. *Pattern Recognition Letters*, 45:1–10, 2014.

[82] Donald Metzler. Generalized inverse document frequency. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 399–408, 2008.

[83] Stephen Robertson. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520, 2004.

[84] Jaekeol Choi and Sang-Woong Lee. Improving fasttext with inverse document frequency of subwords. *Pattern Recognition Letters*, 133:165–172, 2020.

[85] Advait Parulekar, Liam Collins, Karthikeyan Shanmugam, Aryan Mokhtari, and Sanjay Shakkottai. Infonce loss provably learns cluster-preserving representations. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 1914–1961. PMLR, 2023.

[86] Yujian Long, Dian Gu, Xinrui Li, Peiqing Lu, and Jing Cao. Enhancing educational content matching using transformer models and infonce loss. In *2024 IEEE 7th International Conference on Information Systems and Computer Aided Education (ICISCAE)*, pages 11–15. IEEE, 2024.

[87] Zhaorui Zhu, Hongyi Yu, Caiyao Shen, Jianping Du, Zhixiang Shen, and Zhenyu Wang. Causal language model aided sequential decoding with natural redundancy. *IEEE Transactions on Communications*, 71(5):2685–2697, 2023.

[88] Ning Shi, Bradley Hauer, and Grzegorz Kondrak. Lexical substitution as causal language modeling. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (* SEM 2024)*, pages 120–132, 2024.

[89] Michał Perełkiewicz, Sławomir Dadas, and Rafał Poświata. Smclm: Semantically meaningful causal language modeling for autoregressive paraphrase generation. *IEEE Access*, 2025.

[90] Tilahun Abedissa Taffa, Debayan Banerjee, Yaregal Assabie, and Ricardo Usbeck. Hybrid-squad: Hybrid scholarly question answering dataset. *arXiv preprint arXiv:2412.02788*, 2024.

[91] Simeon Monov, Detelinka Trifonova, Nikolay Pavlov, and Andrey Nikolov. Automatic translation of squad and race question answering datasets in bulgarian language. *International Journal of Differential Equations and Applications*, 23(1):83–95, 2024.

[92] Elior Sulem, Jamaal Hay, and Dan Roth. Do we know what we don't know? studying unanswerable questions beyond squad 2.0. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4543–4548, 2021.

[93] Abbas Saliimi Lokman, Mohamed Ariff Ameedeen, and Ngahzaifa Ab Ghani. Question classification of coqa (qcoc) dataset. In *2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM)*, pages 644–648. IEEE, 2021.

[94] Michiel de Jong, Yury Zemlyanskiy, Joshua Ainslie, Nicholas FitzGerald, Sumit Sanghai, Fei Sha, and William Cohen. Fido: Fusion-in-decoder optimized for stronger performance and faster inference. *arXiv preprint arXiv:2212.08153*, 2022.

[95] Mozhgan Ghassemiazghandi. An evaluation of chatgpt's translation accuracy using bleu score. *Theory and Practice in Language Studies*, 14(4):985–994, 2024.

[96] Yayan Heryanto and Agung Triayudi. Evaluating text quality of gpt engine davinci-003 and gpt engine davinci generation using bleu score. *SAGA: Journal of Technology and Information System*, 1(4):121–129, 2023.

[97] Shweta Chauhan, Philemon Daniel, Archita Mishra, and Abhay Kumar. Adableu: A modified bleu score for morphologically rich languages. *IETE Journal of Research*, 69(8):5112–5123, 2023.

[98] Max Grusky. Rogue scores. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1914–1934, 2023.

[99] Nattapong Sanchan. Comparative study on automated reference summary generation using bert models and rouge score assessment. *Journal of Current Science and Technology*, 14(2):26–26, 2024.

[100] Alessia Auriemma Citarella, Marcello Barbella, Madalina G Ciobanu, Fabiola De Marco, Luigi Di Biasi, and Genoveffa Tortora. Assessing the effectiveness of rouge as unbiased metric in extractive vs. abstractive summarization techniques. *Journal of Computational Science*, 87:102571, 2025.

[101] Zongbao Yang, Yinxin Xu, Jinlong Hu, and Shoubin Dong. Generating knowledge aware explanation for natural language inference. *Information Processing & Management*, 60(2):103245, 2023.

[102] Petros Eleftheriadis, Isidoros Perikos, and Ioannis Hatzilygeroudis. Evaluating deep learning techniques for natural language inference. *Applied Sciences*, 13(4):2577, 2023.

[103] Reto Gubelmann, Aikaterini-Lida Kalouli, Christina Niklaus, and Siegfried Handschuh. When truth matters-addressing pragmatic categories in natural language inference (nli) by large language models (llms). In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (* SEM 2023)*, pages 24–39, 2023.

[104] Yanran Chen and Steffen Eger. Menli: Robust evaluation metrics from natural language inference. *Transactions of the Association for Computational Linguistics*, 11:804–825, 2023.

[105] Swagat Shubham Bhuyan, Saranga Kingkor Mahanta, Partha Pakray, and Benoit Favre. Textual entailment as an evaluation metric for abstractive text summarization. *Natural Language Processing Journal*, 4:100028, 2023.

[106] Hongyin Luo and James Glass. Logic against bias: Textual entailment mitigates stereotypical sentence reasoning. *arXiv preprint arXiv:2303.05670*, 2023.

[107] Diego de Vargas Feijo and Viviane P Moreira. Improving abstractive summarization of legal rulings through textual entailment. *Artificial intelligence and law*, 31(1):91–113, 2023.

[108] Fabio Rizzi. *Developing an Enterprise Chatbot using Machine Learning Models: A RAG and NLP based approach*. PhD thesis, Politecnico di Torino, 2024.

[109] Rui Yang, Michael Fu, Chakkrit Tantithamthavorn, Chetan Arora, Lisa Vandenhurk, and Joey Chua. Ragva: Engineering retrieval augmented generation-based virtual assistants in practice. *arXiv preprint arXiv:2502.14930*, 2025.

[110] Maciej Pondel, Iwona Chomiak-Orsa, Małgorzata Sobińska, Wojciech Grzelak, Artur Kotwica, Andrzej Małowiecki, Kamila Łuczak, Andrzej Greńczuk, Peter Busch, David Chudán, et al. Ai tools for knowledge management–knowledge base creation via llm and rag for ai assistant. In *European Conference on Artificial Intelligence*, pages 3–15. Springer, 2024.

[111] Xiaofeng Zhu and Jaya Krishna Mandivarapu. Trustful llms: Customizing and grounding text generation with knowledge bases and dual decoders. *arXiv preprint arXiv:2411.07870*, 2024.

[112] Domenico Bulfamante. *Generative enterprise search with extensible knowledge base using ai*. PhD thesis, Politecnico di Torino, 2023.

[113] Yujia Zhou, Yan Liu, Xiaoxi Li, Jiajie Jin, Hongjin Qian, Zheng Liu, Chaozhuo Li, Zhicheng Dou, Tsung-Yi Ho, and Philip S Yu. Trustworthiness in retrieval-augmented generation systems: A survey. *arXiv preprint arXiv:2409.10102*, 2024.

[114] Hamid Vakilzadeh and David A Wood. The development of a rag-based artificial intelligence research assistant (aira). *Journal of Information Systems*, pages 1–23, 2025.

[115] Ioana Frîncu. In search of the perfect prompt. *Aalto University*, 2023.

[116] Yash Vaykar and Madhavi Chaudahri. Intelligent academic assistant: An agentic rag framework for qa from notes and syllabi.

[117] Xiaoxian Yang, Zhifeng Wang, Qi Wang, Ke Wei, Kaiqi Zhang, and Jiangang Shi. Large language models for automated q&a involving legal documents: a survey on algorithms, frameworks and applications. *International Journal of Web Information Systems*, 20(4):413–435, 2024.

[118] Aayushi Verma, Jorge Morato, Arti Jain, and Anuja Arora. Relevant subsection retrieval for law domain question answer system. In *Data Visualization and Knowledge Engineering: Spotting Data Points with Artificial Intelligence*, pages 299–319. Springer, 2019.

[119] Johny Moreira, Altigran da Silva, Edleno de Moura, and Leandro Marinho. A study on unsupervised question and answer generation for legal information retrieval and precedents understanding. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2865–2869, 2024.

[120] Justin Ho, Alexandra Colby, and William Fisher. Incorporating legal structure in retrieval-augmented generation: A case study on copyright fair use. *arXiv preprint arXiv:2505.02164*, 2025.

[121] S Ajay Mukund and KS Easwarakumar. Optimizing legal text summarization through dynamic retrieval-augmented generation and domain-specific adaptation. *Symmetry*, 17(5):633, 2025.

[122] Mahd Hindi, Linda Mohammed, Ommama Maaz, and Abdulmalik Alwarafy. Enhancing the precision and interpretability of retrieval-augmented generation (rag) in legal technology: A survey. *IEEE Access*, 2025.

[123] Gianmaria Ajani, Guido Boella, Luigi Di Caro, Livio Robaldo, Llio Humphreys, Sabrina Praduroux, Piercarlo Rossi, and Andrea Violato. The european legal taxonomy syllabus: a multi-lingual, multi-level ontology framework to untangle the web of european legal terminology. *Applied Ontology*, 11(4):325–375, 2017.

[124] Peizhang Shao, Linrui Xu, Jinxi Wang, Wei Zhou, and Xingyu Wu. When large language models meet law: Dual-lens taxonomy, technical advances, and ethical governance. *arXiv preprint arXiv:2507.07748*, 2025.

[125] Debarati Das, Khanh Chi Le, Ritik Sachin Parkar, Karin De Langis, Brendan Madson, Chad M Berryman, Robin M Willis, Daniel H Moses, Brett McDonnell, Daniel Schwarcz,

et al. Lawflow: Collecting and simulating lawyers' thought processes. *arXiv preprint arXiv:2504.18942*, 2025.

[126] Ali J Ramadhan, Sahar Yousif Mohammed, Mohammed Aljanabi, Maad M Mijwil, Mostafa Abotaleb, Hussein Alkattan, and Pushan Kumar Dutta. Enhancing ehr analysis: Leveraging rag-enabled generative ai for clinical data summarization. *Library of Progress-Library Science, Information Technology & Computer*, 44, 2024.

[127] Mohammad Alkhalaf, Ping Yu, Mengyang Yin, and Chao Deng. Applying generative ai with retrieval augmented generation to summarize and extract key clinical information from electronic health records. *Journal of biomedical informatics*, 156:104662, 2024.

[128] Josip Vrdoljak, Zvonimir Boban, Marino Vilović, Marko Kumrić, and Joško Božić. A review of large language models in medical education, clinical decision support, and healthcare administration. In *Healthcare*, volume 13, page 603. MDPI, 2025.

[129] Omid Kohandel Gargari and Gholamreza Habibi. Enhancing medical ai with retrieval-augmented generation: A mini narrative review. *Digital health*, 11:20552076251337177, 2025.

[130] John PA Ioannidis, Michael E Stuart, Shannon Brownlee, and Sheri A Strite. How to survive the medical misinformation mess. *European journal of clinical investigation*, 47(11):795–802, 2017.

[131] Victor Suarez-Lledo and Javier Alvarez-Galvez. Prevalence of health misinformation on social media: systematic review. *Journal of medical Internet research*, 23(1):e17187, 2021.

[132] Micheline Andel Goldwire, Steven T Johnson, Maha Abdalla, Ashish Advani, Allison Bernknopf, Angela Colella, Heather A Kehr, Karen Kier, Dianne May, J Russell May, et al. Medical misinformation: a primer and recommendations for pharmacists. *Journal of the American College of Clinical Pharmacy*, 6(5):497–511, 2023.

[133] Riikka Vuokko, Anne Vakkuri, and Sari Palojoki. Systematized nomenclature of medicine–clinical terminology (snomed ct) clinical use cases in the context of electronic health record systems: systematic literature review. *JMIR medical informatics*, 11:e43750, 2023.

[134] Christophe Gaudet-Blavignac, Vasiliki Foufi, Mina Bjelogrlic, Christian Lovis, et al. Use of the systematized nomenclature of medicine clinical terms (snomed ct) for processing free text in health care: systematic scoping review. *Journal of medical Internet research*, 23(1):e24594, 2021.

[135] Eunsuk Chang and Javed Mostafa. The use of snomed ct, 2013-2020: a literature review. *Journal of the American Medical Informatics Association*, 28(9):2017–2026, 2021.

[136] Leonardo Campillos-Llanos, Ana Valverde-Mateos, Adrián Capllonch-Carrión, and Antonio Moreno-Sandoval. A clinical trials corpus annotated with umls entities to enhance the access to evidence-based medicine. *BMC medical informatics and decision making*, 21(1):69, 2021.

[137] Andrew P Reimer and Alex Milinovich. Using umls for electronic health data standardization and database design. *Journal of the American Medical Informatics Association*, 27(10):1520–1528, 2020.

[138] Michel Joubert, Marius Fieschi, Jean-Jacques Robert, Françoise Volot, and Dominique Fieschi. Umls-based conceptual queries to biomedical information databases: an overview of the project ariane. *Journal of the American Medical Informatics Association*, 5(1):52–61, 1998.

[139] Shirley V Wang, Sebastian Schneeweiss, Jessica M Franklin, Rishi J Desai, William Feldman, Elizabeth M Garry, Robert J Glynn, Kueiyu Joshua Lin, Julie Paik, Elisabetta Patorno, et al. Emulation of randomized clinical trials with nonrandomized database analyses: results of 32 clinical trials. *Jama*, 329(16):1376–1385, 2023.

[140] Rajesh R Tampi, Brent P Forester, and Marc Agronin. Aducanumab: evidence from clinical trial data and controversies. *Drugs in context*, 10, 2021.

[141] Prakash M Nadkarni, Cynthia Brandt, Sandra Frawley, Frederick G Sayward, Robin Einbinder, Daniel Zelterman, Lee Schacter, and Perry L Miller. Managing attribute–value clinical trials data using the act/db client–server database system. *Journal of the American Medical Informatics Association*, 5(2):139–151, 1998.

[142] Data Analytics and Dishant Sukhwal. Retrieval augmented generation: An evaluation of rag-based chatbot for customer support. *Retrieval Augmented Generation: An Evaluation of RAG-based Chatbot for Customer Support*, 2024.

[143] Muhammed Rizwan, Lars Carlsson, and Mohammad Loni. Personabot: Bringing customer personas to life with llms and rag. *arXiv preprint arXiv:2505.17156*, 2025.

[144] J Benita, Kosireddy Vivek Charan Tej, E Vinay Kumar, G Venkata Subbarao, and CH Venkatesh. Implementation of retrieval-augmented generation (rag) in chatbot systems for enhanced real-time customer support in e-commerce. In *2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, pages 1381–1388. IEEE, 2024.

[145] Junyan Li, Sam-Zaak Wong, Gwok-Waa Wan, Xi Wang, and Jun Yang. Eda-debugger: An llm-based framework for automated eda runtime issue resolution. In *2025 26th International Symposium on Quality Electronic Design (ISQED)*, pages 1–7. IEEE, 2025.

[146] Quentin Romero Lauro, Shreya Shankar, Sepanta Zeighami, and Aditya Parameswaran. Rag without the lag: Interactive debugging for retrieval-augmented generation pipelines. *arXiv preprint arXiv:2504.13587*, 2025.

[147] Wilson Soto. Tool-based retrieval-augmented generative as an automated assistant in object-oriented programming course. In *2025 IEEE Engineering Education World Conference (EDUNINE)*, pages 1–6. IEEE, 2025.

[148] Carlos Alario-Hoyos, Rebiha Kemcha, Carlos Delgado Kloos, Patricia Callejo, Iria Estévez-Ayres, David Santín-Cristóbal, Francisco Cruz-Argudo, and José Luis López-Sánchez. Tailoring your code companion: Leveraging llms and rag to develop a chatbot

to support students in a programming course. In *2024 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*, pages 1–8. IEEE, 2024.

[149] Junkyum Kim and Divya Mahajan. An adaptive vector index partitioning scheme for low-latency rag pipeline. *arXiv preprint arXiv:2504.08930*, 2025.

[150] Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. Long-context llms meet rag: Overcoming challenges for long inputs in rag. *arXiv preprint arXiv:2410.05983*, 2024.

[151] Jérémie Farret, Jerin Jude, and Nitish Kumar Pilla. Enhancing real-time decision-making with scalable, safe, and private llmops and context-aware rag workflows. *Advances in Signal Processing and Artificial Intelligence*, page 119, 2025.

[152] Satya Narayana Cheetirala, Ganesh Raut, Dhavalkumar Patel, Fabio Sanatana, Robert Freeman, Matthew A Levin, Girish N Nadkarni, Omar Dawkins, Reba Miller, Randolph M Steinhagen, et al. Less context, same performance: A rag framework for resource-efficient llm-based clinical nlp. *arXiv preprint arXiv:2505.20320*, 2025.

[153] Xinxi Chen, Li Wang, Wei Wu, Qi Tang, and Yiyao Liu. Honest ai: Fine-tuning" small" language models to say" i don't know", and reducing hallucination in rag. *arXiv preprint arXiv:2410.09699*, 2024.

[154] Shreya Saxena, Siva Prasad, MV Prakash, Advaith Shankar, Vishal Vaddina, Saisubramaniam Gopalakrishnan, et al. Minimizing factual inconsistency and hallucination in large language models. *arXiv preprint arXiv:2311.13878*, 2023.

[155] Qiucheng Chen and Bo Wang. Valuable hallucinations: Realizable non-realistic propositions. *arXiv preprint arXiv:2502.11113*, 2025.

[156] Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, Avishek Anand, and Adam Jatowt. It's high time: A survey of temporal information retrieval and question answering. *arXiv preprint arXiv:2505.20243*, 2025.

[157] Jacky He, Guiran Liu, Binrong Zhu, Hanlu Zhang, Hongye Zheng, and Xiaokai Wang. Context-guided dynamic retrieval for improving generation quality in rag models. *arXiv preprint arXiv:2504.19436*, 2025.

[158] Alicia Russell-Gilbert. *RAAD-LLM: adaptive anomaly detection using LLMs and RAG integration*. PhD thesis, Mississippi State University, 2025.

[159] Zhengbao Jiang. *Towards More Factual Large Language Models: Parametric and Non-parametric Approaches*. PhD thesis, Carnegie Mellon University, 2024.

[160] Hao Zhang, Yuyang Zhang, Xiaoguang Li, Wenxuan Shi, Haonan Xu, Huanshuo Liu, Yasheng Wang, Lifeng Shang, Qun Liu, Yong Liu, et al. Evaluating the external and parametric knowledge fusion of large language models. *arXiv preprint arXiv:2405.19010*, 2024.

[161] Jiho Shin, Reem Aleithan, Hadi Hemmati, and Song Wang. Retrieval-augmented test generation: How far are we? *arXiv preprint arXiv:2409.12682*, 2024.

[162] Anupam Purwar et al. Evaluating the efficacy of open-source llms in enterprise-specific rag systems: A comparative study of performance and scalability. *arXiv preprint arXiv:2406.11424*, 2024.

[163] Wen Li, Ying Zhang, Yifang Sun, Wei Wang, Mingjie Li, Wenjie Zhang, and Xuemin Lin. Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1475–1488, 2019.

[164] Alexandr Andoni, Piotr Indyk, and Ilya Razenshteyn. Approximate nearest neighbor search in high dimensions. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3287–3318. World Scientific, 2018.

[165] Sunil Arya, David M Mount, Nathan S Netanyahu, Ruth Silverman, and Angela Y Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM (JACM)*, 45(6):891–923, 1998.

[166] Falk Scholer, Hugh E Williams, John Yiannis, and Justin Zobel. Compression of inverted indexes for fast query evaluation. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229, 2002.

[167] Giulio Ermanno Pibiri and Rossano Venturini. Techniques for inverted index compression. *ACM Computing Surveys (CSUR)*, 53(6):1–36, 2020.

[168] Manish Patil, Sharma V Thankachan, Rahul Shah, Wing-Kai Hon, Jeffrey Scott Vitter, and Sabrina Chandrasekaran. Inverted indexes for phrases and strings. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 555–564, 2011.

[169] Michael J Ryan, Danmei Xu, Chris Nivera, and Daniel Campos. Enronqa: Towards personalized rag over private documents. *arXiv preprint arXiv:2505.00263*, 2025.

[170] Xiaopeng Li, Pengyue Jia, Derong Xu, Yi Wen, Yingyi Zhang, Wenlin Zhang, Wanyu Wang, Yichao Wang, Zhaocheng Du, Xiangyang Li, et al. A survey of personalization: From rag to agent. *arXiv preprint arXiv:2504.10147*, 2025.

[171] Yahe Yang, Chao Xu, Jing Guo, Tianbao Feng, and Cailian Ruan. Improving the rag-based personalized discharge care system by introducing the memory mechanism. In *2025 IEEE 17th International Conference on Computer Research and Development (ICCRD)*, pages 316–322. IEEE, 2025.

[172] Peng Xia, Kangyu Zhu, Haoran Li, Tianze Wang, Weijia Shi, Sheng Wang, Linjun Zhang, James Zou, and Huaxiu Yao. Mmed-rag: Versatile multimodal rag system for medical vision language models. *arXiv preprint arXiv:2410.13085*, 2024.

[173] Peng Xia, Kangyu Zhu, Haoran Li, Hongtu Zhu, Yun Li, Gang Li, Linjun Zhang, and Huaxiu Yao. Rule: Reliable multimodal rag for factuality in medical vision language models. *arXiv preprint arXiv:2407.05131*, 2024.

[174] Yinghao Zhu, Changyu Ren, Shiyun Xie, Shukai Liu, Hangyuan Ji, Zixiang Wang, Tao Sun, Long He, Zhoujun Li, Xi Zhu, et al. Realm: Rag-driven enhancement of multimodal electronic health records analysis via large language models. *arXiv preprint arXiv:2402.07016*, 2024.

[175] Yumeng Sun, Renhan Zhang, Renzi Meng, Lian Lian, Heyi Wang, and Xuehui Quan. Fusion-based retrieval-augmented generation for complex question answering with llms. 2025.

[176] Darío Garigliotti. Explainable llm-powered rag to tackle tasks in the unstructured-structured data spectrum. 2023.

[177] Jiaan Wang, Fandong Meng, Yingxue Zhang, and Jie Zhou. Retrieval-augmented machine translation with unstructured knowledge. *arXiv preprint arXiv:2412.04342*, 2024.

[178] Berhanu Bogale, Tesfa Tegegne, Solomon Teferra, and Gebeyehu Belay. Rag based qa for low resource languages. 2024.

[179] Mitha Alshammary, Md Nahiyan Uddin, and Latifur Khan. Rfpg: Question-answering from low-resource language (arabic) texts using factually aware rag. In *2024 IEEE 10th International Conference on Collaboration and Internet Computing (CIC)*, pages 107–116. IEEE, 2024.

[180] Chen-Chi Chang, Chong-Fu Li, Chu-Hsuan Lee, and Hung-Shin Lee. Enhancing low-resource minority language translation with llms and retrieval-augmented generation for cultural nuances. *arXiv preprint arXiv:2505.10829*, 2025.

[181] Nadezhda Chirkova, David Rau, Hervé Déjean, Thibault Formal, Stéphane Clinchant, and Vassilina Nikoulina. Retrieval-augmented generation in multilingual settings. *arXiv preprint arXiv:2407.01463*, 2024.

[182] Syed Rameel Ahmad. Enhancing multilingual information retrieval in mixed human resources environments: A rag model implementation for multicultural enterprise. *arXiv preprint arXiv:2401.01511*, 2024.

[183] Jeonghyun Park and Hwanhee Lee. Investigating language preference of multilingual rag systems. *arXiv preprint arXiv:2502.11175*, 2025.

[184] Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From rag to memory: Non-parametric continual learning for large language models. *arXiv preprint arXiv:2502.14802*, 2025.

[185] Yingjie Xia, Yuhan Chen, Yunxiao Zhao, Li Kuang, Xuejiao Liu, Ji Hu, and Zhiquan Liu. Fcllm-dt: Enpowering federated continual learning with large language models for digital twin-based industrial iot. *IEEE Internet of Things Journal*, 2024.

[186] Lucian Gruia and Bogdan Ionescu. Exploring conversational agents and continual learning in artificial intelligence. 2024.

[187] Murugan Sankaradas, Ravi K Rajendran, and Srimat T Chakradhar. Streamingrag: Real-time contextual retrieval and generation framework. *arXiv preprint arXiv:2501.14101*, 2025.

[188] Yeonwoo Jeong, Kyuli Park, and Sungyong Park. Streamrag: a lock-aware and traffic-aware query coordinator in stream-based rag systems. In *2025 IEEE 25th International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, pages 01–10. IEEE, 2025.