Available Online: 29 January 2026 DOI: 10.54254/2977-3903/2026.31567

A comprehensive survey of knowledge Graph-Augmented Generation (Graph-RAG) for trustworthy large language models

Jianwei Ren

Department of Statistics, University of Wisconsin Madison, Madison, USA

jren76@wisc.edu

Abstract. Large Language Models (LLMs), which are capable of generating diverse expressions or reasoning paths, are fundamentally limited by their reliance on Parametric Memory and hence suffer from the 'hallucination' problem. That is, LLMs are capable of generating false yet plausible statements that are not grounded in reality, which is problematic for scientific and clinical use. A popular solution for alleviating this problem is the Retrieval-Augmented Generation (RAG) technique. However, the retrieved sources are still large passages of semi-unstructured documents, and rigorous, verifiable, and logically coherent reasoning processes are still lacking. In this review, we summarise the latest progress in the direction of Knowledge Graph-Augmented Generation (Graph-RAG) as an attempt to achieve rigorous, verifiable, and logical reasoning processes using structured retrieval sources. We design a computation framework to provide a comprehensive taxonomy, classifying Graph-RAG methods into three areas: Graph Indexing, Graph Guided Retrieval, and Graph Enhanced Generation. We also discuss in depth the potential future directions for Graph-RAG, with an emphasis on moving beyond simple retrieval towards causal graph reasoning and actionable graphs. Finally, we present the main challenges in scaling Graph-RAG and its evaluation protocols, and conclude that Graph-RAG is a critical step toward trustworthy AI.

Keywords: Graph-RAG, AI, LLMs

1. Introduction

The advent of GPT series [1], Llama [2], and Claude [3] marks the rise of Large Language Models (LLMs), ushering in a new AI era. Equipped with vast web-scale text corpora, these models demonstrate remarkable, albeit unplanned, power across a wide array of tasks, including reasoning, fluent human language generation, and few-shot learning [4]. Moreover, these impressive capabilities allow them to serve as the foundation for a new breed of applications, including intelligent search engines [5], robotic chatbots, and even scientific discovery tools [6].

However, despite their strengths, they are hindered by a significant flaw that prevents their reliable use in sensitive environments such as health care, finance, and engineering. This flaw is hallucination [7, 8], where models have a tendency to generate text that is seemingly reasonable and plausible but is actually false, unverifiable, or even completely detached from the source context [9]. In essence, hallucination substantially impedes the trustworthiness of an LLM's outputs [10]. This "black box" nature, where all information is concentrated in billions of parameters, leads to several important challenges. First, the models are prone to factual errors, "hallucinating" statistically plausible but incorrect facts, references, or data points [11]. Second, they are subject to a knowledge cut-off, as model knowledge is frozen and outdated at the end of training, rendering them incapable of perceiving up-to-date or recent information [12]. Third, they offer no verifiability; it is typically impossible to trace the provenance of a specific factual claim made by a large language model, rendering its outputs unfit for high-stakes decisions [13].

To overcome these limitations, Retrieval-Augmented Generation (RAG) has emerged as the standard solution [14]. Its core concept is straightforward: shift knowledge from the opaque "parametric" memory inside the model's weights to an explicit "non-parametric" external knowledge source. In a typical RAG process, a user's question will first trigger a retriever (for example, the Dense Passage Retriever (DPR) [15]) to search through a large amount of information (usually a vector database). The system extracts the most relevant text segments and combines them with the original question to form an expanded query [16]. The large language model is then instructed to generate an answer using only the provided context. This grounding mechanism, anchoring responses in verifiable external sources, has become the standard approach for building knowledge-intensive applications.

However, standard RAG is no silver bullet. Its effectiveness is limited by the type of knowledge sources it uses: unstructured text [17, 18]. Raw text fragments are retrieved, which introduces issues such as noisy retrieval, where relevant facts may be buried within long, unrelated passages [19]. Additionally, unstructured text lacks the structural expressiveness required for handling complex queries that require multi-hop reasoning. For example, a query such as "Which drug inhibits protein A known to activate protein B?" cannot be answered by a single text fragment, as it requires finding and combining two separate but related pieces of information. Thus, large language models suffer a considerable drop in performance when accurate facts are located deep within extensive retrieval contexts – a phenomenon referred to as "getting lost in the middle" [20].

Table 1 provides a full comparison between traditional Text-RAG and the emerging Graph-RAG paradigm, illustrating how the underlying information-processing and reasoning mechanisms differ.

Table 1. Comprehensive comparison between Text-RAG and Graph-RAG

Feature	Text-RAG (Standard)	Graph-RAG (Proposed)
Data Source	Unstructured text chunks (documents)	Structured knowledge graphs (entities & relations)
Retrieval Unit	Vector-based text segments	Nodes, edges, subgraphs, and paths
Reasoning Capability	Implicit, co-occurrence based	Explicit, multi-hop, logical traversal
Explainability	Low (black box retrieval)	High (traceable reasoning paths)
Noise Sensitivity	High (irrelevant info in chunks)	Low (precise extraction of facts)
Context Efficiency	Prone to "Lost in the Middle"	Compact, structured schema representation

The central thesis of this study is that the next frontier for trustworthy AI lies in overcoming these limitations by using structured symbolic Knowledge Graphs (KGs) [21] as a non-parametric knowledge source. We term this paradigm Graph-RAG, which is key to achieving precise, verifiable, and complex reasoning. Knowledge graphs store information as explicit, discrete facts (e.g., Protein A, Inhibits, Drug X). Compared to text-based RAG, enhancing large language models with knowledge graphs offers three major advantages: precision, because the system retrieves single, distinct pieces of information or small subgraphs instead of messy text fragments; multi-hop reasoning, because graphs naturally support complex queries (such as "A→B→C") through structured traversal; and verifiability, because retrieval paths or subgraphs form a formal, machine-readable "chain of evidence" that clearly demonstrates how the generated answers are derived.

This paper presents a structured review of the rapidly evolving research area of schema-based RAG, which structures the topic by connecting basic concepts, important technologies, and key issues. In more detail, the review is organised as follows: Section 2 establishes the background for KGs [21, 22], formalising their structure and providing key public examples [23-26]. Section 3 examines the core technologies for computational use of knowledge graphs: knowledge acquisition [27-29], knowledge completion [30-32], and graph representation using Graph Neural Networks (GNNs) [33-35]. Section 4 discusses the phenomenon of hallucination in LLMs and how they are classified [7, 8] and evaluated [36, 37]. Section 5 reviews classical [13] and advanced text-based RAG frameworks [17, 18, 38]. Section 6, the main chapter, introduces a classification system for the latest graph RAG, categorising techniques according to different approaches to indexing, retrieving, and generating [39-46] (including hybrid approaches [46]). Section 7 explores future directions, such as causal reasoning [47-51] and the development of dynamic, evolving graphs [52-55]. Lastly, Section 8 discusses current problems [56-58] and outlines promising avenues for future work.

2. Foundations: Knowledge Graphs (KGs)

Before we can appreciate how Graph-RAG addresses LLM hallucinations, we need to understand the tool at the heart of the solution: the Knowledge Graph (KG). Whereas unstructured text represents information as loosely organised sequences of words with relationships only implicitly embedded, a Knowledge Graph provides a precise, structured map of facts. It links concepts and their connections explicitly, eliminating ambiguity. This section explains what a KG formally represents and traces how these graphs evolved from early academic constructs into foundational components of modern intelligent systems.

2.1. Formal definitions and structure

At its simplest, a Knowledge Graph is a network of facts. We can think of it formally as a graph $G = (E, R, T)$ [21], but its essential components can be described as follows:

• Entities (E): These constitute the "nodes" or dots of the graph and correspond to well-defined real-world or conceptual objects, such as "Aspirin", "New York City", or "Inflation".

• Relations (R): These are the lines that connect the dots. They tell us exactly how two things are connected (for example, "treats", "located at", "causes").

• Triples (T): These are the most basic units of knowledge. A triple is simply a sentence: (Subject, Predicate, Object).

Consider the biological statement, "Rapamycin inhibits mTOR". In unstructured text, this information may appear in various forms of language. However, in a KG, it is represented as a single, unambiguous triple: (Rapamycin, inhibits, mTOR). This format is powerful because it enables the computer to take action beyond word matching; instead, it can follow a path and resolve questions based on logical graph traversal and inference [22]. It transforms variable natural language into a precise, machine-interpretable structure.

2.2. Key characteristics

Why construct these graphs when one could simply provide raw text to an LLM? The literature identifies three clear advantages. The first is semantic explicitness. Human language is inherently ambiguous, with many terms carrying multiple meanings dependent on context. In a KG, a relation such as "treats" has a fixed, domain-specific definition. This explicitness acts as a guardrail, preventing the LLM from hallucinating based on the most probable definition. The second advantage is multi-hop connectivity. Complex questions often require linking information across several intermediate concepts. For example, if a doctor asks, "What disease does a drug targeting protein X treat?", one must traverse Drug->Protein->Pathway->Disease. KGs are explicitly designed to support such multi-step reasoning, mirroring experts' analytical processes. Lastly, KGs provide interoperability. In practice, the same concept may be referred to by many different names; for example, Tylenol, Acetaminophen, and APAP all refer to the same drug. A KG consolidates these synonyms under a single unique identifier, ensuring that the system does not overlook relevant information simply because of terminological variation.

2.3. Major public Knowledge Graphs

The large Knowledge Graphs used today did not emerge fully formed. They developed through a long process of development, starting with rigid, handmade regulations and evolving into the large-scale, community-generated graphs that we have today.

• This period also includes the beginning of the "Gene Ontology" (GO) project [23]. It was a pivotal moment for biology: the field recognised that a shared, structured vocabulary could unify biological knowledge across species and systems.

– The Big Picture: It demonstrated that domain-specific graphs could be deployed successfully and laid the foundations for the scientifically rigorous graphs now used in Graph-RAG.

• After GO came Freebase [24] and the work of Bollacker and colleagues, who proposed that the community should construct a "graph of everything". Freebase was intentionally broad and flexible, a significant departure from earlier, narrowly scoped academic ontologies.

– Significance: Its success led to Google acquiring it in 2010, and it became the basis of the Google Knowledge Graph launched in 2012. This demonstrated that graph technology can scale to encompass the breadth of the web.

• The period beginning in 2012 marks the Open Data Era (Wikidata and DBpedia). The launch of Wikidata [25] in 2012 and the continued evolution of DBpedia [26] made structured knowledge widely accessible at an unprecedented scale.

– Wikidata is now ubiquitous and serves as the backbone for many sites, including Wikipedia, supported by a global community of editors.

– DBpedia extracts structured "infobox" data from Wikipedia articles.

– Together, these projects provide the breadth and diversity needed to train modern AI systems and our Graph-RAG models with the extensive grounding required to answer general knowledge questions with scientific defensibility.

This trajectory illustrates the evolution from small, rigid graphs to large, flexible, and richly interconnected graphs. AI researchers no longer need to build such resources from scratch; mature, high-quality graphs are readily available. These are the very resources that enable us to embed structured representations of reality into large language models.

Figure 1 illustrates the evolutionary timeline of Knowledge Graph technologies, culminating in the current Graph-RAG paradigm.

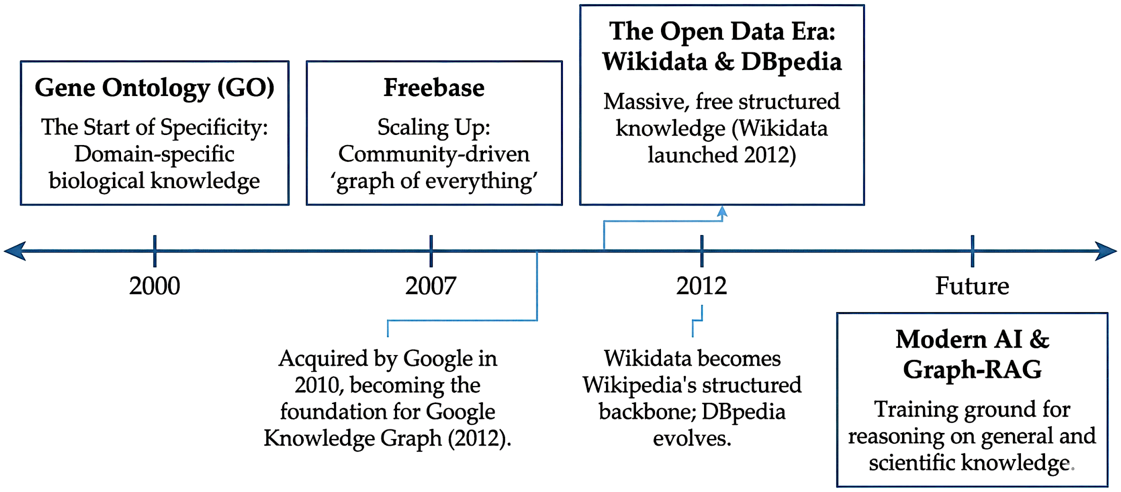
Gene Ontology (GO)
The Start of Specificity: Domain-specific biological knowledge
Freebase
Scaling Up: Community-driven 'graph of everything'
The Open Data Era: Wikidata & DBpedia
Massive, free structured knowledge (Wikidata launched 2012)
2000
2007
2012
Future
Acquired by Google in 2010, becoming the foundation for Google Knowledge Graph (2012).
Wikidata becomes Wikipedia's structured backbone; DBpedia evolves.
Modern AI & Graph-RAG
Training ground for reasoning on general and scientific knowledge.

Figure 1. Evolution of Graph-RAG technologies

3. Core technologies: KG construction and representation

Having a map is of limited value without the ability to construct or interpret it. This section outlines the core technological foundations underlying Graph-RAG: the extraction of structured facts from unstructured text (Construction), the inference of missing information (Completion), and the transformation of symbolic graph structures into numerical representations suitable for machine learning (Representation).

3.1. Knowledge Extraction (KE): from pipelines to LLMs

The construction of Knowledge Graphs (KGs) begins with Knowledge Extraction (KE), which identifies entities and their semantic interrelations by mining unstructured text. Between 2007 and 2020, this was typically referred to as the Traditional Pipeline. This process begins with Named Entity Recognition (NER) and concludes with Relation Extraction (RE), which predicts connections between entities. Relation Extraction (RE) was achieved in 2007 with Open Information Extraction (OpenIE) [27] when Banko et al. removed the dependence on hand-labelled datasets and enabled systems to extract relational facts directly from web-scale corpora. However, their systems produced large amounts of noisy data, which had to undergo profiling to remove noise.

The emergence of LLMs (2023 - Present) has fundamentally transformed this landscape. Instead of training separate specialised models for NER and for RE, LLMs can function as "general-purpose annotators", capable of performing one-shot or zero-shot extraction. Recent reviews [28, 29] demonstrate that LLMs can reliably extract structured biomedical relations such as drug-target pairs, substantially reducing the complexity of constructing high-quality KGs.

3.2. Knowledge Graph Completion (KGC): filling the gaps

There is no such thing as a perfect map. Real-world knowledge graphs are inherently sparse: they may know that "Drug A treats Disease B", while omitting that "Drug A also blocks Protein C". Knowledge Graph Completion (KGC), a form of link prediction, uses machine learning to infer such missing facts.

- 2013: Geometry of Knowledge (TransE). Bordes et al. introduced TransE [30], a seminal model based on a simple geometric intuition: if Head + Relation = Tail, then relations can be modelled as translations in a vector space.

– Impact: This enabled logical reasoning to be approximated through vector arithmetic. However, the model struggled with complex relational patterns, such as one-to-many mappings (e.g., a single director associated with multiple movies).

• 2016-2019: Handling Complexity (ComplEx & RotatE). Researchers extended TransE to address its limitations.

– ComplEx [31] uses complex numbers to deal with asymmetrical relationships (e.g., "is the father of").

– RotatE [32] does not treat relations as translations, but rather as rotations in a complex vector space, enabling the model to naturally capture symmetry and inversion.

3.3. Graph Representation Learning (GRL): teaching networks to "see" graphs

Once a graph has been constructed, it must be encoded in a form suitable for neural computation. Graph Neural Networks (GNNs) address this need by learning vector representations for nodes that capture both their intrinsic properties and their structural context within the graph.

• 2017: GCN revolution. Kipf and Welling introduced the Graph Convolutional Network (GCN) [33], which formalised a simple yet powerful principle: a node's representation should be derived from an aggregation of information from its neighbours.

– Limitation: Standard GCNs treat all neighbouring nodes uniformly. In a knowledge graph, however, different relations carry different semantic significance; a "friend" relation should not be considered the same as an "enemy" relation.

• 2018: Adding Attention and Relational Awareness (GAT & R-GCN). The field rapidly evolved to address the particular requirements of KGs.

– Graph Attention Networks (GAT) [34] introduced an "attention mechanism" designed by Veličković et al. This allows nodes to focus on important neighbours and ignore irrelevant ones, which is essential when filtering noise in Graph-RAG.

– Relational-GCN (R-GCN) [35] incorporated explicit modelling of relation types into the GCN architecture. Given the importance of ensuring the model is aware of the significance of different types of connections, this method has become the leading approach for processing knowledge graphs.

4. The problem: LLM hallucination

Once we have a solid understanding of graph theory, we turn to an immediate concern: the phenomenon of hallucination. Although large language models demonstrate impressive capabilities, they continue to pose significant risks. This section examines the underlying causes of false or unfounded model outputs and outlines methods for their identification.

4.1. Defining the problem: intrinsic vs. extrinsic

The literature distinguishes between two types of hallucination-related errors [7]. An intrinsic error arises when the model contradicts information explicitly present in the source. For example, if a document states, "Apollo 11 landed on the moon in 1969", but the model summarises this as "Apollo 11 landed on the moon in 1970", this discrepancy constitutes an intrinsic error. An extrinsic error occurs when the model introduces information unsupported by the source, for instance, inventing details about Armstrong's hobbies when the document contains none. Extrinsic errors therefore involve statements that cannot be supported by the available context. In high-stakes domains such as scientific research, the most concerning category is Factual Hallucination [10], where the model produces content that is objectively false.

4.2. Root causes: why do models hallucinate

Recent surveys [8] identify three primary sources of hallucination in the LLM lifecycle. First are data issues: web-scale training corpora contain misinformation, bias, and fictional material, which the model learns to imitate. Second, are training dynamics: LLMs are designed to maximise the probability of the next word rather than factual accuracy. If there's an error that has some statistical likelihood (like a common misconception), then the model will produce that content. Lastly, inference stochasticity: the decoding procedure (sampling) introduces randomness. While beneficial for creativity, it can degrade factual reliability [13].

4.3. Measuring the truth

Reliable mitigation requires reliable measurement. Standard metrics such as BLEU or ROUGE assess surface-level lexical overlap rather than factual correctness.

- TruthfulQA: Lin et al. [36] introduced a benchmark designed to expose models' susceptibility to human fallacies and misconceptions, demonstrating that larger models can be less truthful due to broader memorisation.
- HaluEval: Li et al. [37] released a large-scale dataset for evaluating a model's ability to avoid hallucinations.

5. The initial solution: Retrieval-Augmented Generation (RAG)

To address the recurring problem of hallucination, the AI community adopted a strategy similar to human problem-solving: consulting external references. This approach, known as Retrieval-Augmented Generation (RAG) [13], reduces reliance on a model's imperfect internal memory by grounding responses in verifiable external sources.

5.1. The anatomy of standard RAG

The RAG pipeline, introduced by Lewis et al. [13] and improved by dense retrievers such as DPR [15], operates under the "Retrieve-then-Read" model. In indexing, trusted source documents (e.g., medical textbooks and company wikis) are segmented and embedded into a vector space for storage. During retrieval, a user query is converted into a numerical vector representation and compared against the database to identify the top-k most relevant segments. Finally, in generation, the extracted documents are pasted into the prompt context of the LLM. The response generated by the model is constrained by the available information, making it accurate.

5.2. Advanced RAG techniques

As researchers sought to overcome the limitations of the basic pipeline, a range of "advanced RAG" methods emerged [17]. Pre-Retrieval Optimisation addresses users' propensity for asking unclear questions. Query rewriting techniques [38] leverage an LLM to clarify or expand users' questions before they reach the vector database, yielding better search results. Post-Retrieval Refinement ensures that only the most relevant material is passed to the LLM. Reranking models [18] act as a quality control step, re-ranking the retrieved passages so that the most relevant content is provided to the LLM, thereby reducing noise.

5.3. Why text RAG fails at reasoning

Despite engineering improvements, text-based RAG remains fundamentally limited in its ability to support complex reasoning [20]. The core reason is that it treats knowledge as isolated textual fragments rather than as a structured system.

Consider a multi-hop question: "What drug targets the protein that is controlled by Gene X?" Suppose two documents exist:

- Document A says: "Gene X regulates Protein Y."
- Document B says: "Drug A targets Protein Y."

Text RAG may retrieve both documents, but it relies on the LLM to infer the connection. If the two facts are buried in long-form passages, or if the "bridge entity" (protein Y in this case) is replaced by various synonyms, the model often fails to infer the relationships. It may find the puzzle pieces, but it has no scaffolding to arrange them. This is precisely the type of structural void that Graph-RAG is designed to address.

6. The frontier: Graph-RAG (knowledge Graph-augmented generation)

This section marks the transition to current research frontiers. Whereas standard RAG resembles keyword lookup in a textbook index, Graph-RAG is similar to consulting domain experts who follow a coherent chain of reasoning. Instead of using (or adding to) a list of plain text documents, the system fetches reasoning paths – explicit chains of evidence – rather than individual pieces of text.

Figure 2 visualises the unified architecture of the Graph-RAG framework, demonstrating the flow from user query to final answer through graph-guided processes.

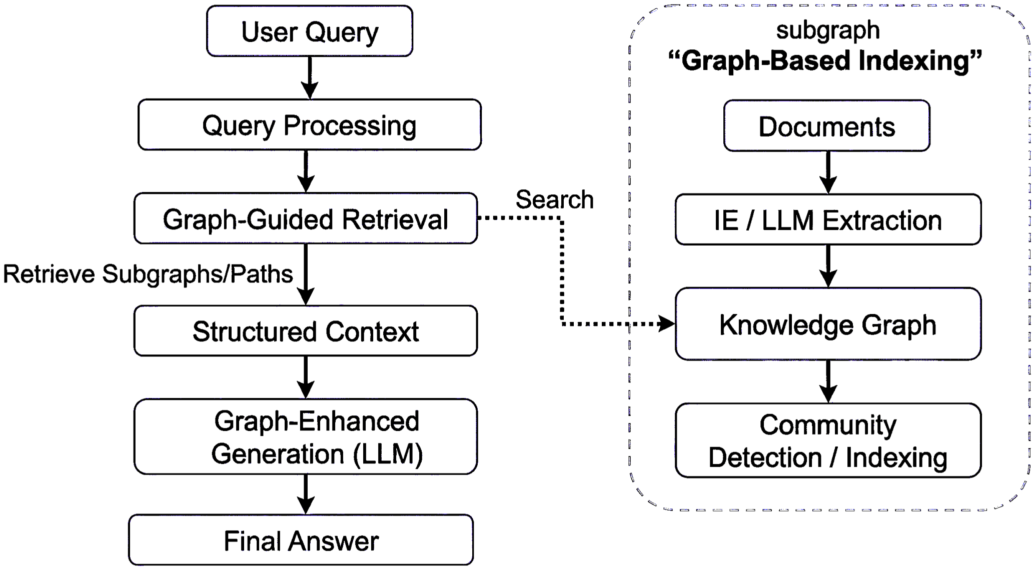

Figure 2. Unified architecture of Graph-RAG

6.1. Why graphs change the game

Graph-RAG is not merely a shift in storage format; it represents a fundamental change in what is retrieved and how reasoning is supported. First, it offers precision: instead of retrieving lengthy, noisy paragraphs, the system can provide discrete factual units such as Drug A, treats, Disease B. Secondly, it gives connectivity: the graph structure connects the gaps that may be dispersed across multiple documents, enabling the model to traverse from one fact to another and access the relevant information. Lastly, it guarantees explainability: the retrieved subgraphs act as a roadmap for the model's "thought process", making it clear which facts ultimately resulted in the conclusion.

6.2. A taxonomy of Graph-RAG architectures

Recent comprehensive surveys [39-41] categorise the various Graph-RAG landscapes into three different stages of intervention.

Stage 1: Graph-based indexing (sourcing). This stage concerns how information is organised prior to retrieval. The Microsoft Approach [42] constructs a graph from documents by extracting entities using LLMs and organising them into different levels of a community hierarchy. This creates a multi-ordered index and enables the model to respond to "global" inquiries, for example, identifying the key themes of a given dataset. This is achieved by making generalisations across entire graph communities, which is an improvement over the capabilities of traditional vector search.

Stage 2: Graph-Guided Retrieval (Understanding the Reasoning). Here, the system utilises the structure of the graph in its search. Think-on-Graph [43] introduced an agentic paradigm, in which LLMs do not perform isolated lookups but instead "walk" through the graph as if they were independent agents. From a given node, they look at adjacent nodes, determine if a pathway exists, and then advance a step. This form of iteration is analogous to human reasoning, which enables the model to tackle complex problems that require advanced structuring. Similarly, in addressing complexity, Atomic Decomposition [44] decomposes complex queries into smaller, more manageable "atomic" constituents. This is analogous to resolving complex equations by performing simpler, constituent arithmetic operations. Figure 3 describes an example reasoning path, showing how a model may infer a drug-disease association through intermediary proteins and genes.

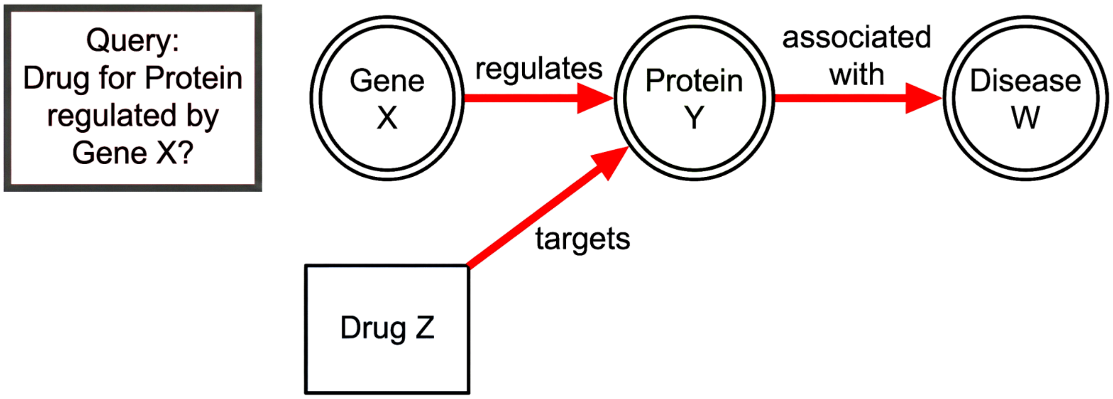

Figure 3. Example of a reasoning path in Graph-RAG

Stage 3: Graph-enhanced generation (refining the output). Once relevant paths or subgraphs have been retrieved, the challenge becomes how to present them to the LLM. Path Selection (KELP) [45] addresses the issue that providing an entire subgraph can overwhelm the model. Xu et al. proposed KELP for smartly scoring and choosing only the most informative routes. It ensures the LLM focuses on the signal rather than the noise, pruning the reasoning tree before generation.

6.3. Hybrid RAG: the best of both worlds

The field is increasingly converging on Hybrid RAG [50]. These systems recognise a simple truth: graphs excel at representing structure and logical dependencies, whereas text excels at conveying nuance and context. Hybrid models therefore combine both modalities - using graphs to identify the essential backbone of an answer and text to supply detailed, contextually grounded explanations. In effect, the graph provides the "skeleton" of logic, while the text provides the "flesh" that completes the final response.

Table 2 provides a taxonomy of Graph-RAG methods, categorising them by their primary stage of intervention.

Table 2. Taxonomy of Graph-RAG methods

Category	Sub-category	Description	Representative Works
Indexing	Text-to-Graph	Builds KG from docs using LLMs	Microsoft GraphRAG [42]
	Graph Indexing	Hierarchical clustering/indexing	Edge et al. [42]
Retrieval	Agentic Traversal	LLM walks the graph step-by-step	Think-On-Graph[43]
	Atomic Decomposition	Breaks query into sub-graph queries	Li et al. [44]
Generation	Path Pruning	Select most relevant paths	KELP [45]
	Hybrid Integration	Combines text chunks + graph paths	HybridRAG [46]

7. Advanced topics & future directions

If Graph-RAG represents the current state of the art, what lies ahead? The field is rapidly progressing beyond systems that merely retrieve existing knowledge towards systems capable of explaining why things occur (causality) and imagining what might take place (generation). This section examines these emerging frontiers.

7.1. Causal Graph-RAG: understanding "why"

Most contemporary AI models operate as correlation machines. They recognise that "taking aspirin" and "headache relief" frequently co-occur, but they do not inherently understand the causal mechanism linking the two. In scientific contexts, correlation is insufficient; causal reasoning is essential. Standard GNNs aggregate information based on structural proximity rather than causal delineation. This limitation can lead to false reasoning, where the model may state that a medicine is efficient merely due to its frequent association with a sickness in writing, even if it has no therapeutic effect [47].

The solution is Causally-Aware GNNs. Researchers are now incorporating the mathematically strict Causal Inference [48, 49] into graph neural networks. Approaches such as Causal-GNN [50] and the framework proposed by Luo et al. in Causal Graphs Meet Thoughts [51] introduce a new type of "causally-aware GNN". These models do more than propagate features: they learn to separate spurious associations from genuine cause-effect relations. This enables Graph-RAG systems to respond to counterfactual questions ("What would have happened to the patient if we hadn't given them that drug?"), which is necessary for making safe clinical decisions.

7.2. Generative graph retrieval: simulating the unknown

Standard RAG has a fundamental limitation: it can only retrieve information that already exists in the database. Scientific discovery, however, requires the ability to explore what is not yet known. How can we identify a molecule that has never been observed?

• The Innovation: Retrieval to Generation. The field is shifting from "retrieving a subgraph" to "generating a subgraph", driven by advances in Generative Graph Models [52].

• Graph Diffusion Models: Inspired by the diffusion models underlying image generators such as Stable Diffusion, researchers have modified these methods for use on graphs. Surveys by Zhang et al. [53] and models such as DiGress [54] demonstrate how to train models to create valid, new molecular structures or biological pathways by reversing a noise process.

• The Vision: Consider the query, "Design a protein that binds to Target X". Rather than looking up a protein in a database that may not exist, a future Graph-RAG system would use a conditional diffusion model [55] to create a completely new, hypothetical protein structure and how it interacts. Here, the "retrieved" content is simulated, not recorded.

8. Conclusion

This review has mapped the evolution of trustworthy AI from the statistical fragility of LLMs to the structured grounding of Graph-RAG. The argument is clear: while standard RAG offers a necessary "open book" memory, it is an insufficiently "logical" reader of that book. Graph-RAG fills this structural gap. Retrieval is elevated from text chunk matching to path-based reasoning, yielding three undeniable advantages for high-stakes applications: precision, through the retrieval of discrete, unambiguous facts; reasoning, through the integration of multi-hop dependencies; and verifiability, through explicit evidence trails that render the model's reasoning transparent.

Graph-RAG, however, is far from a solved problem. Several challenges remain. The Latency Bottleneck persists: graph traversal is expensive, especially when running iterative agentic models like Think-on-Graph. Doing so in real time for web-scale graphs remains a challenge of scalability [56]. The "Garbage In" Risk is equally pressing: a Graph-RAG system is only as reliable as the underlying graph. If we rely on hallucinating LLMs for KG construction, the system risks entering a self-reinforcing loop of misinformation. Ensuring data quality in automated KG construction [57] may become a critical discipline. Finally, the Evaluation Gap remains unresolved. How does one score the quality of a "reasoning path"? Traditional metrics such as BLEU or recall do not capture logical validity. No standardised benchmarks currently distinguish between a correct answer derived through flawed reasoning and one derived through sound reasoning [58].

Looking ahead, the future of AI is symbiosis: the seamless, creative power of connectionist models (e.g., Large Language Models, LLMs), colliding headlong with the rigid, logical scaffolding of symbolic systems (e.g., Knowledge Graphs, KGs). The most promising frontier is not static retrieval but causal generation. As causal inference and graph diffusion models are incorporated into the RAG pipeline, we move closer to AI systems that do not merely regurgitate information but develop a structured understanding of how the world works.

References

[1] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901.

[2] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J.,

Fu, W., … Scialom, T. (2023). *Llama 2: Open foundation and fine-tuned chat models*. arXiv. https: //arxiv.org/abs/2307.09288

[3] Anthropic. (2024). *The Claude 3 model family: Opus, Sonnet, Haiku* [Technical report]. https: //www.anthropic.com/research/claude-3-family

[4] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., & Fedus, W. (2022). Emergent abilities of large language models. *Transactions on Machine Learning Research*. https://arxiv.org/abs/2206.07682

[5] OpenAI. (2024). *GPT-4o system card*. OpenAI. https: //openai.com/index/gpt-4o-system-card/

[6] Jiang, A. Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bressand, F., Lengyel, G., Bour, G., Lavaud, L. R., Gervet, T., Bamford, C., Chaplot, D. S., de las Casas, D., Ebner, M., Bhotika, F., Hanna, E. B., Biken, F., … Lample, G. (2024). *Mixtral of experts.* arXiv. https: //arxiv.org/abs/2401.04088

[7] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys, 55*(12), Article 248. https: //doi.org/10.1145/3571730

[8] Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., & Liu, T. (2023). *A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions.* arXiv. https: //arxiv.org/abs/2311.05232

[9] Huang, J., & Chang, K. C.-C. (2022). *Towards reasoning in large language models: A survey.* arXiv. https: //arxiv.org/abs/2212.10403

[10] Tonmoy, S. M. T. I., Zaman, S. M. M., Jain, V., Rani, A., Rawte, V., Chadha, A., & Das, A. (2024). *A comprehensive survey of hallucination mitigation techniques in large language models.* arXiv. https: //arxiv.org/abs/2401.01313

[11] Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidl, T., … Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences, 103*, 102274. https: //doi.org/10.1016/j.lindif.2023.102274

[12] Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., … Wen, J.-R. (2023). *A survey of large language models*. arXiv. https: //arxiv.org/abs/2303.18223

[13] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, *33*, 9459–9474.

[14] Zhao, P., Zhang, H., Yu, Q., Wang, Z., Geng, Y., Fu, F., Yang, L., Zhang, W., Jiang, J., & Cui, B. (2024). *Retrieval-augmented generation for AI-generated content: A survey*. arXiv. https: //arxiv.org/abs/2402.19473

[15] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-T. (2020). Dense passage retrieval for open-domain question answering. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6769–6781). Association for Computational Linguistics.

[16] Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., van den Driessche, G. B., Lespiau, J.-B., Damoc, B., Clark, A., de Las Casas, D., Guy, A., Menick, J., Ring, R., Hennigan, T., Huang, S., Maggiore, L., Jones, C., Cassirer, A., … Sifre, L. (2022). Improving language models by retrieving from trillions of tokens. *Proceedings of the 39th International Conference on Machine Learning* (pp. 2206–2240). PMLR.

[17] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023). *Retrieval-augmented generation for large language models: A survey*. arXiv. https: //arxiv.org/abs/2312.10997

[18] Wang, X., Wang, Z., Gao, X., Zhang, F., Wu, Y., Xu, Z., Shi, T., Wang, Z., Li, S., Qian, Q., Yin, R., Lv, C., Zheng, X., & Huang, X. (2024). *Searching for best practices in retrieval-augmented generation*. arXiv. https:

//arxiv.org/abs/2407.01219

[19] Ovadia, O., Brief, M., Mishaeli, M., & Elisha, O. (2023). *Fine-tuning or retrieval? Comparing knowledge injection in LLMs.* arXiv. https: //arxiv.org/abs/2312.05934

[20] Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, *12*, 157–173.

[21] Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., & Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*, *54*(4), 1–37.

[22] Ji, S., Pan, S., Cambria, E., Marttinen, P., & Philip, S. Y. (2021). A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(2), 494–514.

[23] Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., & Sherlock, G. (2000). Gene ontology: Tool for the unification of biology. *Nature Genetics*, *25*(1), 25–29.

[24] Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: A collaboratively created graph database to structure human knowledge. *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, 1247–1250.

[25] Vrandečić, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM, 57*(10), 78–85.

[26] Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., & Bizer, C. (2015). DBpedia – A large-scale, multilingual knowledge graph extracted from Wikipedia. *Semantic Web, 6*(2), 167–195.

[27] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the web. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, 2670–2676.

[28] Bian, H. (2024). *LLM-empowered knowledge graph construction: A survey.* arXiv. https: //arxiv.org/abs/2510.20345

[29] Trajanoska, M., Stojanov, R., & Trajanov, D. (2023). *Enhancing knowledge graph construction using large language models.* arXiv. https: //arxiv.org/abs/2305.04676

[30] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems, 26*, 2787–2795.

[31] Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., & Bouchard, G. (2016). Complex embeddings for simple link prediction. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, 2071–2080.

[32] Sun, Z., Deng, Z.-H., Nie, J.-Y., & Tang, J. (2019). RotatE: Knowledge graph embedding by relational rotation in complex space. *7th International Conference on Learning Representations (ICLR 2019)*. https: //arxiv.org/abs/1902.10197

[33] Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *5th International Conference on Learning Representations (ICLR 2017)*. https: //arxiv.org/abs/1609.02907

[34] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). *Graph attention networks. 6th International Conference on Learning Representations (ICLR 2018)*. https: //arxiv.org/abs/1710.10903

[35] Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., & Welling, M. (2018). Modeling relational data with graph convolutional networks. *The Semantic Web (ESWC 2018), 10843*, 593–607.

[36] Lin, S., Hilton, J., & Evans, O. (2022). TruthfulQA: Measuring how models mimic human falsehoods. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers), 3214–3252.

[37] Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., & Wen, J.-R. (2023). HaluEval: A large-scale hallucination evaluation benchmark for large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6449–6464.

[38] Ma, Y., Cao, Y., Hong, Y., & Sun, A. (2023). *Large language model is not a good few-shot information extractor, but a good reranker for hard samples!* arXiv. https: //arxiv.org/abs/2303.08559

[39] Peng, B., Zhu, Y., Liu, Y., Bo, X., Shi, H., Hong, C., Zhang, Y., & Tang, S. (2024). *Graph retrieval-augmented generation: A survey.* arXiv. https: //arxiv.org/abs/2408.08921

[40] Zhang, Q., Chen, S., Bei, Y., Yuan, Z., Zhou, H., Hong, Z., Dong, J., Chen, H., Chang, Y., & Huang, X. (2025). *A survey of graph retrieval-augmented generation for customized large language models.* arXiv. https: //arxiv.org/abs/2501.13958

[41] Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., & Wu, X. (2024). *Unifying large language models and knowledge graphs: A comprehensive survey. IEEE Transactions on Knowledge and Data Engineering.* Advance online publication. https: //doi.org/10.1109/TKDE.2024.3402368

[42] Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., & Larson, J. (2024). *From local to global: A graph RAG approach to query-focused summarization*. arXiv. https: //arxiv.org/abs/2404.16130

[43] Sun, J., Xu, C., Tang, L., Wang, S., Lin, C., Gong, Y., Ni, L. M., Shum, H.-Y., & Guo, J. (2024). Think-on-Graph: Deep and responsible reasoning of large language model on knowledge graph. *The Twelfth International Conference on Learning Representations (ICLR 2024)*. https: //openreview.net/forum?id=1o8q7c7F9x

[44] Li, Y., Song, D., Zhou, C., Tian, Y., Wang, H., Yang, Z., & Zhang, S. (2024). A framework of knowledge graph-enhanced large language model based on question decomposition and atomic retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2024* (pp. 11472–11485). Association for Computational Linguistics.

[45] Liu, H., Wang, S., Zhu, Y., Dong, Y., & Li, J. (2024). Knowledge graph-enhanced large language models via path selection. In *Findings of the Association for Computational Linguistics: ACL 2024* (pp. 6311–6321). Association for Computational Linguistics.

[46] Sarmah, B., Benara, V., Awasthi, A., & Talukdar, P. (2024). *HybridRAG: Integrating knowledge graphs and vector retrieval for retrieval-augmented generation*. arXiv. https: //arxiv.org/abs/2408.04948

[47] Kıcıman, E., Ness, R., Sharma, A., & Tan, C. (2023). *Causal reasoning and large language models: Opening a new frontier for causality*. arXiv. https: //arxiv.org/abs/2305.00050

[48] Pearl, J. (2009). *Causality: Models, reasoning, and inference* (2nd ed.). Cambridge University Press.

[49] Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect.* Basic Books.

[50] Mesinovic, M., Buhlan, M., & Zhu, T. (2025). *Causal graph neural networks for healthcare*. arXiv. https: //arxiv.org/abs/2511.02531

[51] Luo, H., Zhang, J., & Li, C. (2025). *Causal graphs meet thoughts: Enhancing complex reasoning in graph-augmented LLMs*. arXiv. https: //arxiv.org/abs/2501.14892

[52] Guo, X., & Zhao, L. (2018). A systematic survey of deep generative models for graph generation. *IEEE Transactions on Knowledge and Data Engineering, 30*(5), 1036–1053.

[53] Zhang, M., Qamar, M., Kang, T., Jung, Y., Zhang, C., Bae, S., & Zhang, C. (2023). *A survey on graph diffusion models: Generative AI in science for molecule, protein and material.* arXiv. https: //arxiv.org/abs/2304.01565

[54] Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., & Frossard, P. (2023). DiGress: Discrete denoising diffusion for graph generation. *The Eleventh International Conference on Learning Representations (ICLR 2023)*. https: //openreview.net/forum?id=UaAD-Nu86WX

[55] Jo, J., Lee, S., & Hwang, S. J. (2022). Score-based generative modeling of graphs via the system of stochastic differential equations. *Proceedings of the International Conference on Machine Learning*, 10362–10383.

[56] Liang, K., Meng, L., Liu, M., Liu, Y., Tu, W., Wang, S., Zhou, S., Liu, X., & Sun, F. (2024). A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal. *IEEE Transactions on Pattern*

Analysis and Machine Intelligence, *46*(12), 9456–9478.

[57] Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., & Auer, S. (2016). Quality assessment for linked data: A survey. *Semantic Web*, *7*(1), 63–93.

[58] Chen, J., Lin, H., Han, X., & Sun, L. (2024). Benchmarking large language models in retrieval-augmented generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *38*(16), 17754–17762.