Enhancing Biomedical Question Answering with Knowledge-Augmented Generation : an Empirical Evaluation of Med-KAG

Edouard Haddag[1,*], Lina F. Soualmia[1]

[1]Univ. Rouen Normandie, TIBS-LITIS UR4108, Saint Etienne du Rouvray, 76801, France

Abstract

Large language models (LLMs) remain underutilized in clinical practice due to their opacity and the constant risk of hallucinations. While the traditional retrieval-augmented generation (RAG) method mitigates these issues by embedding responses within sourced data, it is often prone to semantic loss. To address this limitation, this study investigates GraphRAG, a method that uses knowledge graphs to represent and preserve complex relationships between concepts. In this context, we developed Med-KAG that incoprporates a comprehensive knowledge graph specific to the biomedical domain into the RAG framework, in order to improve accuracy while reducing concrete errors. The knowledge graph was built from the Unified Medical Language System (UMLS) Metathesaurus, enriched with additional relationships that were computed by the BioSTransformers predictive model. Although its primary objective is to reduce hallucinations, preliminary tests with Med-KAG on the PubMedQA dataset showed that this model performed worse than the baseline Qwen3.5. A more in-depth diagnostic analysis revealed that the knowledge graph architecture is the main factor responsible for these shortcomings. These results highlight the importance of improving the quality and accuracy of knowledge to ensure the effective integration of GraphRAG systems in the biomedical field.

Keywords

Knowledge Graph, Retrieval-Augmented Generation, PubMedQA, Qwen3.5, Biomedical Domain.

CEUR-WS.org/Vol-4313/glow-paper10.pdf

1. Introduction

Artificial intelligence (AI), and more specifically large language models (LLMs), has fundamentally transformed medical research and practice [1]. Tasks ranging from the analysis of electronic health records (EHRs) [2] to clinical decision support [3] now leverage deep learning. Recently, the introduction of open-source models such as Qwen3.5 [4] and Gemma4 [5] has democratized access to powerful natural language processing (NLP) tools.

However, the reliability of these models in a clinical context is severely compromised by the phenomenon of hallucination, in which the model generates fluent but factually incorrect information [6]. In the medical field, such errors can have disastrous consequences for patient safety.

To mitigate these hallucinations, Research-Augmented Generation (RAG) was introduced by Lewis *et al.* [7]. This paradigm enriches the model's context by dynamically retrieving relevant information from an external knowledge base. Although standard RAG methods based on dense vector databases offer notable improvements [8], they often struggle to capture the complex, multi-hop relationships inherent in medical reasoning.

This is where the concept of GraphRAG comes into play [9, 10]. By structuring knowledge in the form of a knowledge graph [11], systems such as MED-COPILOT [12] or Think-on-Graph [13] aim to guide the model through a network of interconnected entities. The KAG (Knowledge Augmented Generation) framework [14] is part of this trend, facilitating the integration of specific professional domains.

Graph-enhanced LLMs for trustwOrthy Web data management (GLOW), in ISWC 2026 Workshops Joint Proceedings, October 25–26, 2026, Bari, Italy

*Corresponding author.

contact@c2fc2f.com (E. Haddag); lina.soualmia@litislab.fr (L. F. Soualmia)

https://sagbot.com/ (E. Haddag); https://pagesperso.litislab.fr/lsoualmia/ (L. F. Soualmia)

0009-0003-7646-7619 (E. Haddag); 0000-0001-7668-2819 (L. F. Soualmia)



© 2026 Copyright © 2026 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

published 2026-11-05

The Unified Medical Language System (UMLS) [15, 16] is one of the world's most extensive biomedical knowledge base, comprising millions of concepts, hundreds of semantic relationships, and lexical variants. Integrating UMLS into a GraphRAG architecture promises to endow the LLM with explicit symbolic reasoning. However, the density and complexity of UMLS triples pose major engineering challenges in terms of context integration.

The objective of this study is to empirically quantify the impact of the UMLS knowledge graph on the response capabilities of an open LLM (Qwen3.5-9b and Gemma4-12b) when faced with the PubMedQA benchmark [17], using two formatting heuristics (formal *vs.* textual) deployed on a Neo4j infrastructure [18].

2. Method

The proposed architecture introduces a robust system based on knowledge-augmented generation (KAG) [14], designed to improve the reliability of medical diagnoses. The system uses a knowledge graph (KG) from the Unified Medical Language System (UMLS) [16] instead of unstructured text search and deterministic extraction. We have enriched this graph by integrating new similarity relationships calculated by the BioSTransformers model [19, 20]: specifically, as soon as two distinct UMLS concepts achieve a semantic similarity score greater than 0.75, they are connected, and this score becomes the explicit weight of their relationship. This massive infrastructure thus comprises 50.6 million nodes and 72 million relationships with weights ranging from 0.75 (partial match) to 1 (exact match), ensuring comprehensive coverage of complex biomedical interconnections.

Our methodology, which draws inspiration from the traditional research-augmented generation (RAG) paradigm, employs a hybrid pipeline that combines vector similarity search with localized graph traversal. In this section, we explain in detail the search process, the dynamic translation process for serializing the context, and the explicit prompt engineering strategies used to guide the large language model (in this case, Qwen3.5-9b and Gemma4-12b).

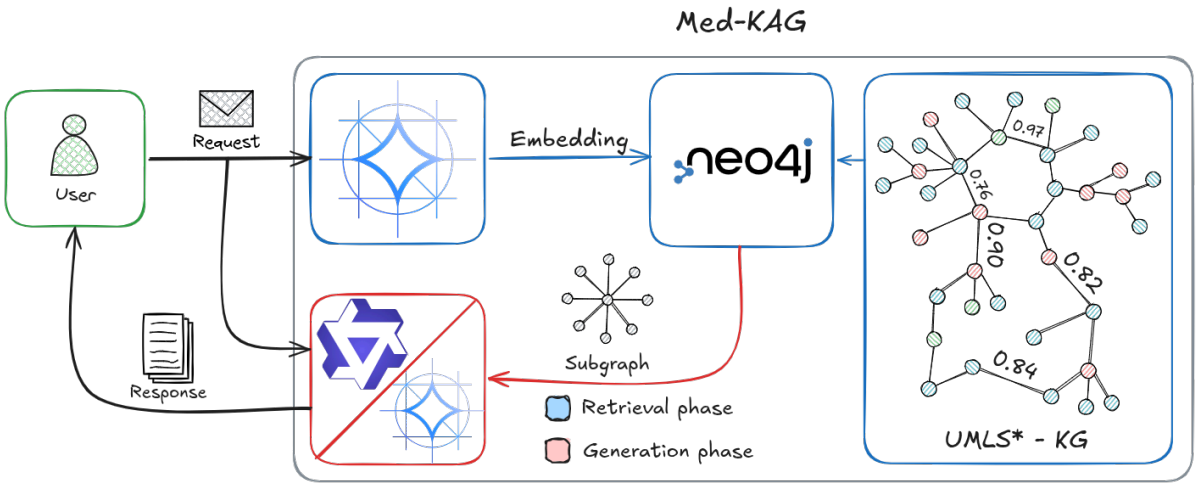


Figure 1: Diagram of the Med-KAG architecture [21], where UMLS* is the extended UMLS with weighted similarity relations

2.1. Knowledge Graph and Vector Representation

To bridge the semantic gap between natural language clinical queries and standardized medical ontologies, the system treats concepts and queries as atomic semantic units without arbitrary text chunking. By utilizing high-dimensional embeddings (specifically embeddinggemma:latest via Ollama), both user intents and node descriptions are projected into a shared continuous vector space. The similarity between the embedded user query (u) and the concept vectors (v) is measured using cosine similarity :

$$\text{similarity}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|} \tag{1}$$

2.2. Hybrid Semantic and Structural Retrieval

The retrieval module executes a hybrid search against the Neo4j graph database. Neo4j was selected because it natively integrates vector search algorithms within a graph-oriented architecture, allowing us to store complex graph structures and perform semantic similarity searches in a single, unified database. Furthermore, its available free tiers lower the barrier to entry, simplifying its deployment and use in medical applications. By blending vector index querying with local neighborhood expansion, the retrieval dynamically adapts to a parameterized topological depth (`neighborhood = 1` in our experiments). Unlike static graph extractions, this approach dictates how many relational hops to traverse outward from the initial seed nodes (top-k = 4).

The retrieval process is executed via an optimized Cypher query leveraging the `UMLS_INDEX`. The query first identifies the top-k most semantically relevant nodes using the vector index. It then performs a variable-length expansion to gather the local neighborhood context (*e.g.*, hierarchical is-a relationships, like parent/child) up to the specified depth. Finally, it extracts the discrete subgraphs, consisting of source nodes, predicates, and target nodes, while filtering out duplicates to construct a dense, highly relevant context network.

The Cypher query utilized by the pipeline is defined as follows:

```
MATCH (node)
  SEARCH node IN (
    VECTOR INDEX $index
    FOR $embedding
    LIMIT $top_k
  )
MATCH (node)-[*0..$neighborhood]-(reachable)
WITH collect(DISTINCT reachable) AS nodes
UNWIND nodes AS n
MATCH (n)-[rel]->(m)
WHERE m IN nodes
WITH DISTINCT rel
RETURN
  startNode(rel) AS source,
  rel AS predicate,
  endNode(rel) AS target
```

Figure 2: Hybrid GraphRAG Cypher Query: Retrieves a localized subgraph by expanding a defined neighborhood around nodes identified via vector similarity search.

This approach ensures that the LLM is provided with a verifiable, multi-hop semantic context, rather than disjointed paragraphs of text. By explicitly capturing `reachable` entities and matching internal relationships (`WHERE m IN nodes`), the query guarantees that the resulting subgraph remains topologically cohesive.

2.3. Graph Translation and Context Serialization Pipeline

Raw graph outputs (streams of nodes and edges) are incompatible with standard LLM context windows. To resolve this, we implemented a highly concurrent, asynchronous translation pipeline that processes the incoming stream of database rows into structured, token-efficient textual representations.

This serialization engine operates on a translation strategy, designed to strictly format the extracted subgraphs. We evaluate two distinct strategies:

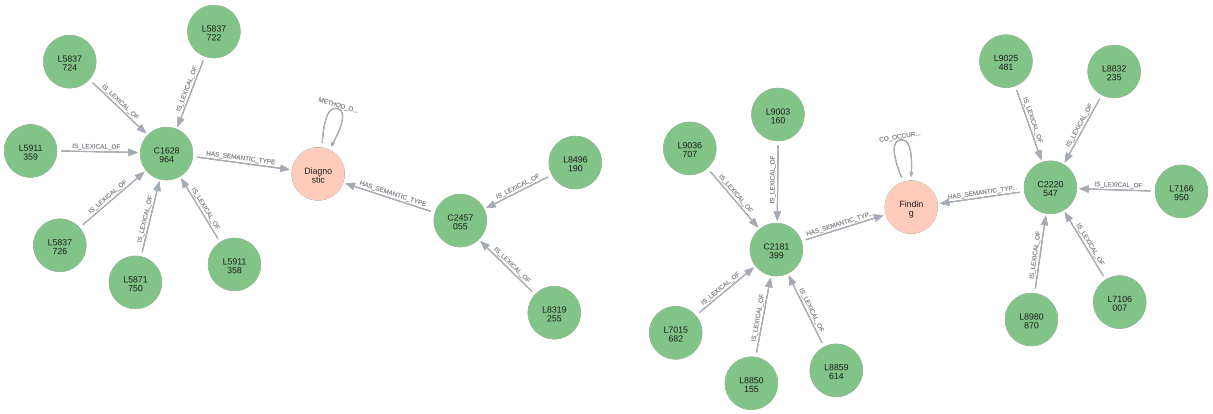


Figure 3: A sample subgraph returned by the retriever. Nodes prefixed with "C" represent specific biomedical concepts (e.g., C1628964: CT and aspiration of abdomen; C2457055: Computerized Tomography (CT Scan) of Abdomen; C2181399: computed tomography of abdomen: foreign body of stomach; C2220547: computed tomography of abdomen: foreign body of alimentary tract), "L" represents their associated lexical terms or exact string variants, and beige nodes denote the Semantic Types categorizing these concepts (e.g., T033: Finding; T060: Diagnostic Procedure). This specific graph was selected for its small size to ensure readability.

- **FormalTriplet Strategy:** This strategy generates a rigorous, Cypher-like syntax representing the structural connections. As the asynchronous stream of rows is consumed, the engine dynamically extracts properties based on predefined label sets and relationship filters. The serialization produces formal edge definitions (*e.g.*, `(source_id)-[:REL_TYPE {props}]->(target_id)`) and individual node property listings. For the UMLS dataset, we strictly filter properties, retaining only the `ui` for `UMLSConcept` nodes to minimize noise. To optimize the LLM's context window, the pipeline utilizes algorithmic deduplication (via internal hash sets) to ensure that nodes and their associated properties are only appended once, even if they participate in multiple retrieved relationships.
- **TextualTriplet Strategy:** This strategy attempts to naturalize the graph structure into fluent English sentences. Node formats are explicitly defined (*e.g.*, a `UMLSConcept` becomes "the UMLS concept {ui}"). Furthermore, complex relational properties and edge definitions are serialized into descriptive sentences. For instance, the `HAS_DEFINITION` relationship is translated as "{FROM} is described by {TO}", and `IS_ATOM_OF` as "{FROM} is an atom of {TO} (preferred: {isPreferred})". This approach evaluates whether LLMs perform better with human-readable, grammatically cohesive context compared to structural notation.

To illustrate these approaches, the following examples demonstrate the serialized output of a sample subgraph using the FormalTriplet (top) and TextualTriplet (bottom) strategies.

```
(2066291)-[:BROADER_THAN]->(1075911)
(2066291)-[ui]->(C0346627)
(1075911)-[ui]->(C1522206)
(407814)-[:MENAD_SIMILARITY { weight: 0.78192264 }]->(2066291)
(407814)-[ui]->(C0022374)
(350842)-[:MENAD_SIMILARITY { weight: 0.8034035 }]->(407814)
(350842)-[ui]->(C0009375)
(2066291)-[:HAS_DEFINITION]->(445136)
(445136)-[ui]->(AT198116685)
(445136)-[source]->(NCI)
(445136)-[value]->(A primary or metastatic malignant neoplasm involving the
  small intestine, large intestine, or both. Representative examples are
  carcinomas, lymphomas, and sarcomas.)
(48922262)-[:IS_LEXICAL_OF { isPreferred: true }]->(1075911)
(48922262)-[ui]->(L5370840)
(34952460)-[:IS_STRING_OF { isPreferred: true }]->(48922262)
```

```
(34952460)-[ui]->(S6132613)
(14092158)-[:IS_ATOM_OF { isPreferred: true }]->(34952460)
(14092158)-[sourceIdentifier]->(C22151)
(14092158)-[ui]->(A7653159)
(14092158)-[source]->(NCI)
(14092158)-[value]->(Malignant Neoplasms of the Mouse Intestinal Tract)
(14092158)-[sourceAssertedConceptIdentifier]->(C22151)
(2066291)-[:HAS_SEMANTIC_TYPE]->(21388719)
(21388719)-[usageNote]->(All neoplasms are assigned to this type. Do not also
  assign a type from the 'Anatomical Abnormality' hierarchy.)
(21388719)-[ui]->(T191)
(21388719)-[name]->(Neoplastic Process)
(21388719)-[definition]->(A new and abnormal growth of tissue in which the
  growth is uncontrolled and progressive. The growths may be malignant or
  benign.)
(21388719)-[abbreviation]->(neop)
(21388719)-[treeNumber]->(B2.2.1.2.1.2)

the UMLS concept C0346627 is broader than the UMLS concept C1522206 through
  the relationship ''.
the UMLS concept C0022374 is semantically similar to the UMLS concept
  C0346627 with a MENAD similarity of 0.78192264.
the UMLS concept C0009375 is semantically similar to the UMLS concept
  C0022374 with a MENAD similarity of 0.8034035.
the UMLS concept C0346627 is described by the definition 'A primary or
  metastatic malignant neoplasm involving the small intestine, large intestine,
  or both. Representative examples are carcinomas, lymphomas, and sarcomas.'.
the lexical variant group L5370840 is a lexical variant of the UMLS concept
  C1522206 (preferred: true).
the string S6132613 is a string of the lexical variant group L5370840
  (preferred: true).
the atom 'Malignant Neoplasms of the Mouse Intestinal Tract' from the source
  vocabulary NCI is an atom of the string S6132613 (preferred: true).
the UMLS concept C0346627 is classified as the semantic type 'Neoplastic
  Process'.

the definition 'A primary or metastatic malignant neoplasm involving the small
  intestine, large intestine, or both. Representative examples are carcinomas,
  lymphomas, and sarcomas.' comes from the source vocabulary NCI and has the
  attribute unique identifier AT198116685.
the definition '...' has the source-asserted attribute identifier null.

the atom 'Malignant Neoplasms of the Mouse Intestinal Tract' from the source
  vocabulary NCI has the atom unique identifier A7653159 and the source
  identifier C22151.
the atom '...' has the source-asserted atom identifier null.
the atom '...' has the source-asserted concept identifier C22151.
the atom '...' has the source-asserted descriptor identifier null.

the semantic type 'Neoplastic Process' is defined as: A new and abnormal growth of
  tissue in which the growth is uncontrolled and progressive. The growths may be
  malignant or benign.
the semantic type 'Neoplastic Process' has the abbreviation neop, the unique
  identifier T191, and the tree number B2.2.1.2.1.2 in the semantic network.
the semantic type 'Neoplastic Process' carries the usage note: All neoplasms are
  assigned to this type. Do not also assign a type from the 'Anatomical
  Abnormality' hierarchy.
```

2.4. Prompt Engineering and Configurations

To empirically isolate the impact of our KAG architecture, we evaluated the system using distinct prompt configurations. The prompts were engineered to enforce strict adherence to the provided context and format constraints, mitigating generative verbosity.

For the baseline evaluation (relying solely on the model's parametric knowledge), we utilized a zero-shot prompt defined in `prompt_mcq.md`:

```
Based on your internal medical knowledge, answer the following question:

{{ INPUT }}

Choices:
{{ CHOICE }}

Follow these strict rules:
1. Your final output must be EXACTLY AND EXCLUSIVELY the single key
   corresponding to the correct choice (e.g., 'A', 'B', 'C'). Do not
   include any explanations, reasoning, punctuation, or additional
   text.
2. If you do not have the information needed to answer the question,
   reply exactly with: 'I don't know.'
```

Conversely, for the knowledge-augmented configurations, the prompt dynamically injects the serialized graph payload based on the selected retriever (`neo4j-embedding-triplet-formal` or `neo4j-embedding-triplet-textual`). The model is explicitly instructed to trace specific relationships, such as *TRIBUTARY_OF*, *TREATS*, or *MEASUREMENT_OF*, and is strictly forbidden from relying on external knowledge. This forces the generator to act purely as a reasoning engine over the extracted graph logic, as shown in the augmented prompts (`umls_formal_prompt_mcq.md`:

```
Based strictly on the provided UMLS knowledge graph context, answer the following
question:

{{INPUT}}

Graph Context:
{{ RETRIEVAL | default('<empty>') }}

Choices:
{{ CHOICE }}

Follow these strict rules:
1. Rely ONLY on the provided graph context. Do not use external medical knowledge.
2. Internally trace the relationships (e.g., 'TRIBUTARY_OF', 'TREATS',
  'EXHIBITS', 'MEASUREMENT_OF') between the medical nodes to determine the
  correct answer.
3. Your final output must be EXACTLY AND EXCLUSIVELY the single key corresponding
  the correct choice (e.g., 'A', 'B', 'C'). Do not include any explanations,
  reasoning, punctuation, or additional text.
4. If the provided graph context does not contain the information needed to answer
  the question, reply exactly with: 'I don't know.'
```

and `umls_textual_prompt_mcq.md`:

```
Based strictly on the provided UMLS knowledge graph context, answer the following
question:

{{INPUT}}
```

```
Graph Context:
{{ RETRIEVAL | default('<empty>') }}

Choices:
{{ CHOICE }}

Follow these strict rules:
1. Rely ONLY on the provided graph context. Do not use external medical knowledge.
2. The context is given as plain sentences. Some sentences describe an entity (its
   term, source vocabulary, definition, semantic type, identifiers); others state
   how two entities are linked ("X treats Y", "X is a tributary of Y", "X exhibits
   Y", "X is a measurement of Y", "X is a kind of Y", "X is a synonym of Y").
   Internally chain these statements together, and use both what each sentence
   says about an entity and the links the sentences express between entities, to
   determine the answer.
3. Your final output must be EXACTLY AND EXCLUSIVELY the single key corresponding
   to the correct choice (e.g., 'A', 'B', 'C'). Do not include any explanations,
   reasoning, punctuation, or additional text.
4. If the provided graph context does not contain the information needed to answer
   the question, reply exactly with: 'I don't know.'
```

By standardizing the output formats to a single choice key across all configurations, we established a robust framework for benchmarking the precision of the graph-augmented context against the model's native capabilities.

3. Results

This section presents the quantitative performance metrics evaluated on subsets of the PubMedQA dataset consisting of up to 500 total scorable questions (See an example question in Figure 4). For the graph-augmented evaluations, the retrieval module was configured to fetch the top four most relevant seed nodes ($top_k = 4$) and expand the local graph context by a topological depth of one hop ($neighborhood = 1$). The experimental evaluation benchmarks the baseline parametric performance of the foundation models against the proposed knowledge-augmented configurations. Specifically, we compare the unaugmented `qwen3.5-9b-native` and `gemma4-12b-native` setups against their hybrid graph-retrieval pipeline counterparts utilizing FormalTriplet and Textual serialization strategies.

Question : *Does desflurane alter left ventricular function when used to control surgical stimulation during aortic surgery?*
Options : (A) yes, (B) no, (C) maybe
Correct answer : B

Figure 4: Example of a multiple-choice question taken from the PubMedQA dataset.

For the evaluation of the generated answers, a strict output matching criterion is applied. A prediction is classified as OK if the model outputs exactly the correct option letter. If the model explicitly responds with « I don't know. », the prediction is categorized as IDK. Any other generated text, whether it is an incorrect option letter or any other unexpected string, is considered WRONG.

The primary metrics monitored during execution are accuracy (ACC%), as well as runtime profiling metrics such as the mean generation time per question (GEN(s)) and the mean retrieval time per question (RET(s)). For the graph-augmented pipeline, architectural and structural payload complexities are quantified using the average number of retrieved vertices, relations, and properties per query execution. The comprehensive results are summarized in Table 1.

Table 1
Generation performance and classification metrics on the PubMedQA dataset.

Setup	ACC%	OK	WRONG	IDK	TOT	GEN(s)
Qwen3.5-9B Native Baseline	53.80	269	226	5	500	14.920
Gemma4-12B Native Baseline	50.40	252	246	2	500	905.843
Qwen3.5-9B Med-KAG (Formal)	5.40	27	63	410	500	116.303
Qwen3.5-9B Med-KAG (Textual)	4.20	21	66	413	500	112.762
Gemma4-12B Med-KAG (Formal)	4.60	23	67	410	500	853.265
Gemma4-12B Med-KAG (Textual)	2.80	14	21	465	500	537.365

Table 2
Retrieval latency and graph structural metrics for the Med-KAG Framework.

Setup	RET(s)	VERT	REL	PROP
Qwen3.5-9B Med-KAG (Formal)	0.212	33.56	45.68	73.04
Qwen3.5-9B Med-KAG (Textual)	0.245	33.56	45.68	73.04
Gemma4-12B Med-KAG (Formal)	0.472	33.56	45.68	73.04
Gemma4-12B Med-KAG (Textual)	0.502	33.56	45.68	73.04

3.1. Task Accuracy and Generation Performance

An analysis of the benchmarking classification performance highlights a substantial quantitative divergence between the native and augmented evaluation regimes, as well as between the model architectures. The baseline configurations, utilizing exclusively parametric knowledge, demonstrated superior classification effectiveness. The `qwen3.5-9b-native` setup achieved an accuracy of 53.80% over 500 questions (269 OK, 226 WRONG, 5 IDK), while the `gemma4-12b-native` setup achieved an accuracy of 50.40% on 500 questions (252 OK, 246 WRONG, 2 IDK).

In contrast, the knowledge-augmented configurations utilizing the Neo4j vector graph retrieval index exhibited a severe decrease in classification effectiveness across both models and serialization strategies. The `qwen3.5-9b` model accuracy degraded to 5.40% using the FormalTriplet strategy and 4.20% using the Textual strategy. Similarly, the `gemma4-12b` model performance dropped to 4.60% (FormalTriplet) and 2.80% (Textual).

3.2. Computational Overhead and Execution Latency

The introduction of the structured retrieval pipeline and serialized text formatting context significantly altered the execution profiling logs of both models. For the native baseline configurations, operating entirely as closed-book zero-shot prompts without external documentation, retrieval latency (RET(s)) was identically zero. The mean generation time was clocked at 14.920 seconds for Qwen-9B and 905.843 seconds for Gemma-12B.

For the Med-KAG pipelines, generation latency expanded notably due to the enriched context windows. The Qwen-9B augmented configurations registered runtime metrics of 116.303 seconds (Formal) and 112.762 seconds (Textual). The Gemma-12B augmented setups experienced even higher overheads, peaking at 853.265 seconds (Formal) and 537.365 seconds (Textual). Additionally, the external retrieval component introduced an extra layer of computational overhead, yielding average graph query execution latencies ranging from 0.212 to 0.502 seconds for Qwen, and 0.472 to 0.502 seconds for Gemma against the Neo4j database instance. It should be noted that these benchmarking tests were executed across multiple separate batches and sessions; consequently, the aggregated runtime metrics may encapsulate intermittent system overheads and are not strictly accurate reflections of a continuous, uninterrupted execution environment.

3.3. Graph Payload and Structural Complexity Metrics

The localized neighborhood expansion query deployed via Cypher extracted targeted topological contexts for each natural language question. For the Qwen-9B/Gemma-12B evaluations, the hybrid vector and structural retrieval module extracted an average of 33.56 vertices (nodes), 45.68 relationships (edges), and 73.04 distinct property pairs per question from the underlying knowledge graph database.

These elements were fully processed by the asynchronous translation pipeline, converted into their respective rigid text context configurations, and subsequently injected directly into the foundation models' prompt windows.

4. Discussion

The empirical findings gathered from the PubMedQA evaluation reveal critical insights into the systemic trade-offs and architectural vulnerabilities associated with integrating massive structured biomedical knowledge bases into generative pipelines. While the Med-KAG framework was intended to strengthen factual grounding, the sharp drop in accuracy from 53.80% in the native baseline to 5.40% in the graph-augmented setup highlights basic issues with existing GraphRAG approaches.

4.1. Architectural Paradigm and Component Justification

The selection of the core computational models within the Med-KAG framework was driven by the strict privacy, security, and operational demands of modern clinical deployment environments. To this end, our architecture compares two distinct state-of-the-art open-source Large Language Models: Qwen3.5 and Gemma4. Both models offer the exact same operational advantages, enabling the diagnostic pipeline to be deployed entirely on-premises within a secure, closed medical infrastructure. This design choice guarantees patient confidentiality and completely eliminates the data exfiltration risks associated with proprietary cloud-based APIs. The primary distinction between the two options is chronological, with Gemma4 simply serving as the more recent model. Evaluating both possibilities provides a robust baseline for assessing the effectiveness of the augmented retrieval pipeline across different iterations of open-source architectures.

4.2. Informational Scarcity and the Context Inflation Paradox

The initial cause of the observed performance decline is a structural paradox that arises from the knowledge graph's localized informational capacity and its overall topological density. The KAG framework was initially predicated on the idea that obtaining a validated ontological pathway would automatically yield the necessary information to address medical questions. Nevertheless, our assessment suggests that the discrete subgraphs extracted at conservative retrieval depths (e.g., $top_k = 4$ and $neighborhood = 1$) frequently lack the comprehensive informational depth necessary to address complex, multi-step biomedical questions. It's just that the knowledge graph doesn't hold enough information on its own in these small, localized clusters (as evidenced by the high number of IDK responses).

Logically, resolving this informational scarcity requires expanding the search parameters to encompass a wider structural neighborhood. Yet, because the underlying biomedical knowledge graph is so massively integrated and densely connected, attempting to increase the traversal depth triggers an immediate and exponential context explosion. Fetching just one additional neighborhood hop results in an overwhelming influx of adjacent entities. The system is consequently trapped in a dichotomy: constrained neighborhoods fail to retrieve sufficient semantic knowledge, while expanded neighborhoods trigger severe serialization bottlenecks that instantly saturate and exceed the rigid context window limits of the foundational language models.

4.3. Graph Refinement and Future Trajectories

These dynamics illustrate that the raw integration of structured data into a RAG pipeline is heavily hindered by the underlying graph density. To navigate the trade-off between localized informational scarcity and broader context inflation, it is imperative to refine the knowledge graph prior to serialization. The graph structure itself must be optimized to condense relevant clinical insights into tighter, more expressive ontological pathways.

In order to dynamically rerank relationships and filter extended networks, modern solutions usually use secondary Large Language Models [13, 10]. However, these approaches come with prohibitive latency and computational overhead, making real-time clinical application unviable. Because the number of edges grows very quickly and exponentially when a neighborhood grows, future versions of the program will need to focus on more advanced graph refinement methods, like localized graph condensation and structural pruning algorithms. By systematically refining the graph, the architecture can facilitate deeper informational traversal without triggering an unmanageable explosion of context.

5. Conclusion

The present study introduced Med-KAG, a Knowledge-Augmented Generation framework that is intended to reduce factual hallucinations in clinical Large Language Models by incorporating a vast, highly structured biomedical knowledge base that is represented in graphs. The theoretical foundation of GraphRAG posits that the precision of diagnostics can be improved by anchoring generative models in verified ontological pathways. However, our empirical evaluation of the PubMedQA dataset revealed a counterintuitive reality. Both the Qwen3.5 and the more recent Gemma4 native, unaugmented foundation models performed noticeably better than the Med-KAG architecture, exposing flaws in the state-of-the-art graph-retrieval paradigms.

Our investigation indicates that the structural paradox between localized informational scarcity and exponential context inflation is the primary constraint on scaling GraphRAG for the biomedical domain in a question-answering task. With severely limited retrieval parameters, the extracted subgraphs often don't contain enough information to answer all of the complicated medical questions. On the other hand, the attempt to address this deficiency by increasing the search depth across the dense knowledge graph results in an excessive number of vertices and relationships. When serialized, this substantial structural payload induces severe context inflation, which exceeds the model's processing capacity. Additionally, the computational cost of utilizing secondary language models to filter this expanded data is prohibitive, making it unsuitable for rapid clinical decision support.

In the end, these results highlight the necessity of a fundamental change in retrieval mechanics for the effective incorporation of comprehensive knowledge graphs into clinical AI. In order to fully unleash the potential of GraphRAG in high-stakes biomedical environments, future research must transcend brute-force subgraph extraction. It is imperative to create pruning algorithms that are knowledge-aware, lightweight, and efficient in order to reduce enormous knowledge networks to sparse, highly relevant semantic pathways. This ensures that foundation models are empowered rather than overwhelmed by structured biomedical knowledge. Additionally, since the current study only examined the Question-Answering (QA) use-case, subsequent research will broaden this assessment to include a wider range of natural language processing tasks with expalainable resuts, such as coding Electronic Health Records, document retrieval and classification. In addition, we want to look into adding different structural bases, like the Precision Medicine Knowledge Graph (PrimeKG)[11], to evaluate how different graph topologies would affect the performance of both retrieval and generation in specific biomedical settings.

Declaration on Generative AI

During the preparation of this work, the author(s) used Gemini, LanguageTool, DeepL in order to: Grammar and spelling check, Paraphrase and reword, Text Translation. After using this tool/service,

the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] Y. A. Fahim, I. W. Hasani, S. Kabba, W. M. Ragab, Artificial intelligence in healthcare and medicine: clinical applications, therapeutic advances, and future perspectives, European Journal of Medical Research 30 (2025) 848. doi:10.1186/s40001-025-03196-w.

[2] H. Alghamdi, A. Mostafa, Advancing EHR analysis: Predictive medication modeling using LLMs, Information Systems 131 (2025) 102528. URL: https://www.sciencedirect.com/science/article/pii/S0306437925000134. doi:https://doi.org/10.1016/j.is.2025.102528.

[3] Y. Hu, W. Xuan, Q. Zhou, Z. Li, Y. Li, et al., A self-correcting Agentic Graph RAG for clinical decision support in hepatology, Frontiers in Medicine 12 (2025) 1716327. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC12748213/. doi:10.3389/fmed.2025.1716327.

[4] Q. Team, Qwen3.5-Omni Technical Report, 2026. URL: https://arxiv.org/abs/2604.15804. arXiv:2604.15804.

[5] G. Team, S. E. Abd, V. Aggarwal, R. Algayres, A. Andreev, et al., Gemma 4 Technical Report, 2026. URL: https://arxiv.org/abs/2607.02770. arXiv:2607.02770.

[6] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, et al., A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions, ACM Trans. Inf. Syst. 43 (2025). URL: https://doi.org/10.1145/3703155. doi:10.1145/3703155.

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, et al., Retrieval-augmented generation for knowledge-intensive nlp tasks, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020, p. 16.

[8] J. Khlaut, C. Dancette, E. Ferreres, A. Bennani, P. Hérent, et al., Efficient Medical Question Answering with Knowledge-Augmented Question Generation, in: T. Naumann, A. Ben Abacha, S. Bethard, K. Roberts, D. Bitterman (Eds.), Proceedings of the 6th Clinical Natural Language Processing Workshop, Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 10–20. URL: https://aclanthology.org/2024.clinicalnlp-1.2/. doi:10.18653/v1/2024.clinicalnlp-1.2.

[9] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, et al., From Local to Global: A Graph RAG Approach to Query-Focused Summarization, 2025. URL: https://arxiv.org/abs/2404.16130. arXiv:2404.16130.

[10] J. Wu, J. Zhu, Y. Qi, J. Chen, M. Xu, et al., Medical Graph RAG: Evidence-based Medical Large Language Model via Graph Retrieval-Augmented Generation, in: W. Che, J. Nabende, E. Shutova, M. T. Pilehvar (Eds.), Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vienna, Austria, 2025, pp. 28443–28467. URL: https://aclanthology.org/2025.acl-long.1381/. doi:10.18653/v1/2025.acl-long.1381.

[11] P. Chandak, K. Huang, M. Zitnik, Building a knowledge graph to enable precision medicine, Scientific Data 10 (2023) 67. URL: https://doi.org/10.1038/s41597-023-01960-3.

[12] S. Chen, N. Patil, H. Pan, A. H.-C. Hwang, Y. Du, et al., MED-COPILOT: A Medical Assistant Powered by GraphRAG and Similar Patient Case Retrieval, 2026. URL: https://arxiv.org/abs/2603.00460. arXiv:2603.00460.

[13] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, et al., Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph, 2024. URL: https://arxiv.org/abs/2307.07697. arXiv:2307.07697.

[14] L. Liang, M. Sun, Z. Gui, Z. Zhu, Z. Jiang, et al., KAG: Boosting LLMs in Professional Domains via Knowledge Augmented Generation, 2024. URL: https://arxiv.org/abs/2409.13731. arXiv:2409.13731.

[15] O. Bodenreider, The Unified Medical Language System (UMLS): integrating biomedical terminology,

Nucleic Acids Research 32 (2004) D267–D270. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC308795/. doi:10.1093/nar/gkh061.

[16] National Library of Medicine (US), UMLS Knowledge Sources [dataset on the Internet], 2026. URL: http://www.nlm.nih.gov/research/umls/licensedcontent/umlsknowledgesources.html, release 2026AA.

[17] Q. Jin, B. Dhingra, Z. Liu, W. W. Cohen, X. Lu, PubMedQA: A Dataset for Biomedical Research Question Answering, 2019. URL: https://arxiv.org/abs/1909.06146. arXiv:1909.06146.

[18] Neo4j, Inc., Cypher Manual, v2026.06, Neo4j, Inc., 2026. URL: https://neo4j.com/docs/cypher-manual/current/.

[19] S. Menad, S. Abdeddaïm, L. F. Soualmia, Biomedical Ontologies Matching and Enrichment with Language Models, in: I. Lynce, N. Murano, M. Vallati, S. Villata, F. Chesani, M. Milano, A. Omicini, M. Dastani (Eds.), ECAI 2025 - 28th European Conference on Artificial Intelligence, 25-30 October 2025, Bologna, Italy - Including 14th Conference on Prestigious Applications of Intelligent Systems (PAIS 2025), volume 413 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2025, pp. 4460–4467. URL: https://doi.org/10.3233/FAIA251345. doi:10.3233/FAIA251345.

[20] S. Menad, G. H. A. Medeiros, L. F. Soualmia, Enhancing the Description-Detection Framework with Semantic Clustering Using Biostransformers, in: A. R. González, R. Sicilia, L. P. Santamaría, G. A. Papadopoulos, V. Guarrasi, M. T. Cazzolato, B. Kane (Eds.), 38th IEEE International Symposium on Computer-Based Medical Systems, CBMS 2025, Madrid, Spain, June 18-20, 2025, IEEE, 2025, pp. 654–659. URL: https://doi.org/10.1109/CBMS65348.2025.00135. doi:10.1109/CBMS65348.2025.00135.

[21] E. Haddag, G. H. Medeiros, L. F. Soualmia, Med-kag: Preliminary results of a medical knowledge-augmented generation approach, Studies in Health Technology and Informatics 336 (2026) 700–704. doi:10.3233/SHTI260261.

A. Online Resources

The sources for the prompts/configuration and the various programs are available on GitHub under the MIT License.

- Med-KAG.rs,
- kag,
- UMLS-to-KG.