Improving LLM Reasoning Through Ontology-driven Knowledge Graphs

Theyaneshwaran Jayaprakash
thjaya@iu.edu

May 8, 2025

Abstract

This work explores the development of an ontology-aware biomedical question answering (QA) system that combines large language models (LLMs) with structured retrieval from a knowledge graph (KG). Motivated by the limitations of conventional retrieval-augmented generation (RAG) pipelines such as hallucinations, shallow retrieval, and weak factual grounding, we propose a hybrid framework that leverages ontological reasoning to improve answer fidelity, semantic consistency, and interpretability.

Our system constructs a Neo4j-based biomedical knowledge graph from PubMed abstracts using a two-stage triple extraction strategy. We compare LLM-driven extraction with classical NLP and ontology-informed methods, emphasizing the role of biomedical ontologies for entity normalization, synonym resolution, and relationship generalization. These ontologies enable reasoning over semantically equivalent entities and support multi-hop inference across distributed graph fragments.

We curate a benchmark QA dataset across five medical domains: Alzheimer's Disease, COVID-19, Mental Health, Obesity, and Parkinson's Disease, where each question is linked to gold-standard answers and grounded triples. The pipeline performs symbolic subgraph retrieval followed by constrained LLM-based generation using the top-k relevant triples.

Ultimately, our goal is to establish a semantically aligned and ontology-driven QA framework that facilitates trustworthy biomedical information access. This work lays the foundation for scalable, explainable QA systems that integrate ontological reasoning for accurate and traceable knowledge retrieval.

1 Introduction

Biomedical question answering (QA) has the potential to transform clinical decision-making, scientific discovery, and health communication by enabling systems to provide structured, evidence-based answers to complex medical questions. However, delivering accurate and explainable answers in the biomedical domain presents unique challenges due to the density of technical language, rapidly evolving knowledge, and the need for high factual precision.

Traditional QA systems often rely on pre-trained language models (LMs) that lack explicit grounding in structured knowledge, making them prone to hallucinations and poor traceability. In contrast, knowledge graph (KG)-based approaches can offer transparent and semantically rich representations of biomedical knowledge. Despite their promise, existing biomedical KGQA systems suffer from several key limitations: (i) they rarely integrate biomedical ontologies to unify synonymous or related entities, (ii) they typically support only shallow, 1-hop retrievals, missing complex multi-hop associations, and (iii) they lack robust benchmarks that trace answers back to specific triples in the KG.

To address these gaps, we propose a benchmark pipeline for biomedical QA that combines graph-based subgraph retrieval, ontology-driven reasoning, and language model-based answer generation. Our system is grounded in a Neo4j-based biomedical knowledge graph constructed from scientific abstracts and enriched with ontological normalization. The QA benchmark spans five biomedical domains—Alzheimer's Disease, COVID-19, Mental Health, Obesity, and Parkinson's Disease—and includes gold-standard answers with fine-grained triple-level evidence annotations.

Our contributions are threefold:

- We release a benchmark dataset for biomedical QA with document- and chunk-level evidence grounding across five major disease areas.
- We integrate biomedical ontologies into both entity normalization and relationship generalization, improving semantic coherence and reasoning coverage.
- We demonstrate how combining ontology-enhanced retrieval with LLM-based generation enables scalable, explainable QA with strong recall and fidelity.

This work lays the foundation for future systems that support trustworthy biomedical QA through explicit reasoning, ontological alignment, and interpretable answer generation.

2 Previous Work

2.1 Biomedical Knowledge Graph Construction

Constructing structured biomedical knowledge graphs (KGs) from scientific literature is a foundational task in biomedical AI. **MedKGraph** Zhang et al. [2023] introduced a large-scale framework for biomedical KG construction, leveraging syntactic parsing and a scalable extraction pipeline applied to the entire PubMed corpus. While MedKGraph achieved expert-level accuracy and demonstrated broader concept coverage than curated databases, its reliance on shallow linguistic features limited semantic consistency across extracted entities. In contrast, **OntoMed** Liu et al. [2024] addressed entity disambiguation through ontology-based normalization, incorporating MeSH and UMLS hierarchies to reduce ambiguity by 37%. However, OntoMed did not extend these improvements to graph construction or downstream QA tasks. Our work unifies these lines by integrating ontology-driven normalization directly into the KG construction and retrieval pipeline, thereby supporting semantic interoperability and reasoning across heterogeneous biomedical sources.

2.2 Biomedical Question Answering over Knowledge Graphs

Despite the potential of KGs to support explainable biomedical question answering (QA), few QA benchmarks provide fine-grained graph-level evidence. **MedQA-KG** Roberts et al. [2023] introduced a benchmark linking QA tasks to structured knowledge, but its scope was limited to two clinical areas and lacked complex reasoning chains. More broadly, biomedical QA efforts like **MedCPT** Agrawal et al. [2022] explored the use of pre-trained language models (PLMs) for answering medical questions, yet these models operate without structured knowledge or retrievable evidence. Our benchmark advances the field by providing multi-domain coverage, triple-aligned graph evidence from a Neo4j KG, and evaluation metrics at the document, chunk, and semantic levels.

2.3 Knowledge-Augmented Retrieval and Graph Reasoning

Integrating structured knowledge with generative models has shown promise for improving factual grounding. **MedRAG** Pan et al. [2023] enhanced retrieval-augmented generation by combining

UMLS-based knowledge graphs with dense retrieval. While effective, MedRAG relied primarily on vector similarity and did not perform symbolic graph traversal or multi-hop subgraph reasoning. In contrast, our pipeline uses hybrid retrieval—combining symbolic Cypher queries with semantic ranking—and supports relaxed chunk-level evidence matching. This enables high-recall evidence selection while preserving the interpretability of explicit graph paths.

2.4 Our Contribution

Building on these foundations, our system introduces an ontology-integrated pipeline for KG-based biomedical QA that supports multi-hop reasoning and semantically coherent entity linking. We release a benchmark dataset that spans five disease domains and includes gold-standard answers, extracted triples, and chunk-level evidence alignment. Our evaluation framework enables both lexical and semantic fidelity analysis, facilitating the development of explainable, graph-grounded QA systems in biomedical research.

3 Experiments

3.1 Dataset

The dataset used in this study consists of 95 biomedical research abstracts spanning five health domains: Alzheimer's Disease, COVID-19, Mental Health, Obesity, and Parkinson's Disease. These abstracts were preprocessed and segmented into overlapping chunks by a collaborating member of the research team to preserve contextual integrity across sentences.

To support downstream evaluation tasks, each chunk was paired with five biomedical questions generated using a language model. These questions were designed to probe factual content and causal reasoning present within the abstracts.

Following question generation, I created ground truth answers for each chunk–question pair using a controlled prompt with Claude. The model was instructed to provide concise, text-only answers based solely on the information within the chunk. If the answer could not be found, the response was explicitly marked as *"Not mentioned in the text"*. This ensured that the answers remained grounded in the source material without introducing external knowledge.

This curated dataset—comprising chunked text, corresponding biomedical questions, and ground truth answers—formed the foundation for all subsequent components of the pipeline. It powered triple extraction for knowledge graph construction, enabled fine-tuning and validation of prompt templates, and supported evaluation of retrieval-based question answering through structured evidence matching.

3.2 Triple Construction: Two-Pass Extraction Strategy

To construct a structured representation of biomedical knowledge, we employed a two-pass extraction strategy using a large language model (LLM). Each text chunk was processed to identify named entities and infer subject–predicate–object triples that capture meaningful biomedical relationships. The method is designed to maximize coverage by combining direct extraction with a targeted verification step.

3.2.1 Primary Extraction

In the first pass, the LLM was prompted to extract a comprehensive set of named entities and all explicitly stated relationships among them. The prompt was tailored to cover a broad range

of biomedical concepts, including genes, proteins, diseases, treatments, chemical compounds, cell types, biological processes, statistical indicators, and study parameters.

To account for linguistic variation, the prompt included guidance on resolving syntactic structures such as appositions, coordinated subjects, and negations. For example, the sentence *"Protein A, a transcription factor, regulates gene expression"* was expected to yield both an identity triple (*Protein A* – `is` – *transcription factor*) and a functional triple (*Protein A* – `regulates` – *gene expression*). The output was formatted as a list of structured triples.

3.2.2 Verification Pass

In the second pass, the model was prompted to review the original text alongside the previously extracted entities and triples. This phase aimed to uncover additional relationships that may have been missed during the first pass. It focused specifically on implicit, long-range, or cross-sentential relationships, as well as those involving statistical associations or coreferent expressions.

Any new triples identified in this stage were added to the existing set, and duplicates were resolved through case-insensitive matching and structural normalization.

3.2.3 Post-Processing

The full set of extracted triples was cleaned to ensure consistency and readiness for graph construction. This involved standardizing entity casing, removing stopwords where appropriate, and merging duplicate triples. The resulting output served as the foundation for constructing a biomedical knowledge graph, enabling downstream tasks such as retrieval, reasoning, and evaluation.

3.3 Knowledge Graph Construction

Following triple extraction, we constructed a biomedical knowledge graph using the Neo4j graph database. Each subject and object in the extracted triples was instantiated as a node, while each predicate was represented as a directed edge between corresponding nodes. To reduce redundancy, entity names were normalized through lowercasing and lemmatization, and semantically equivalent nodes were merged. Duplicate relationships were also removed to ensure structural consistency.

Where possible, nodes were annotated with semantic type labels (e.g., *disease*, *compound*, *biological process*), either inferred from the surrounding context or derived from rule-based heuristics. These annotations enabled typed querying and improved subgraph traversal efficiency. To facilitate rapid retrieval, the graph was indexed using both full-text search and lexical similarity mechanisms within Neo4j.

After ingestion and normalization, the final graph consisted of 4,813 nodes and 5,696 directed relationships. These statistics reflect the combined output of all document chunks after deduplication, entity resolution, and relationship consolidation.

3.3.1 Graph Statistics

To understand the topological and semantic structure of the graph, we examined node connectivity and edge frequency distributions. The most highly connected node was *"study"*, which appeared in 143 direct relationships, reflecting its central role in biomedical abstracts. Other frequently connected nodes included *"participants"* (62), *"parkinson's disease"* (59), *"patients"* (57), and *"alzheimer's disease"* (50). These hubs suggest a concentration of extracted knowledge around clinical populations and neurodegenerative diseases.

In terms of relationship types, the most common predicates were `HAVE` (128), `HAS` (128), `IS` (116), `IS_ASSOCIATED_WITH` (108), and `INCLUDE` (101). These relations formed the semantic backbone of the graph, capturing key biomedical assertions such as ownership, identity, categorization, and statistical association.

This graph structure enabled downstream reasoning and retrieval tasks by providing an interpretable and densely connected knowledge space aligned with biomedical discourse.

3.4 Graph Retrieval and Fine-Tuning

Building on the constructed biomedical knowledge graph, we implemented a hybrid retrieval pipeline designed to support evidence-grounded question answering. This pipeline integrates symbolic graph traversal with dense vector similarity to identify relevant knowledge triples in response to user queries. The process comprises two stages: offline indexing and online retrieval.

3.4.1 Offline Indexing

In the offline phase, all unique entity names from the knowledge graph were embedded using the `all-roberta-large-v1` model from the `SentenceTransformers` library. These embeddings were stored in a FAISS index (`entity_index.faiss`) to support efficient approximate nearest neighbor (ANN) search. Associated metadata—including normalized entity strings—was serialized in a dictionary file (`entity_names.pkl`) for reverse lookup and annotation.

Concurrently, a full-text index was constructed over all entity nodes in Neo4j to enable lexical string matching. This dual indexing strategy—semantic and symbolic—was designed to maximize coverage and retrieval precision, with the FAISS index acting as a fallback when exact lexical matches were not found.

3.4.2 Online Retrieval

At inference time, user queries were processed in real-time to extract entity mentions using Claude, a language model optimized for biomedical entity recognition and question parsing. Extracted entities were first matched against the Neo4j index. If no exact matches were found, the system expanded the query using the FAISS index to retrieve semantically similar entities.

A 1-hop Cypher query was then issued in Neo4j to collect candidate triples linked to the retrieved entities. These candidate triples were ranked based on their semantic similarity to the input question, computed using `SentenceTransformer` embeddings of the question and each triple's subject, predicate, and object. The highest-ranked triples were presented as supporting evidence for the answer.

For example, in response to the query *"What causes AD?"*, the system identified *AD* as a mention of *Alzheimer's disease*, enabling the retrieval of triples such as *genetic predisposition* $\rightarrow$ *increases risk of* $\rightarrow$ *Alzheimer's disease* and *misfolded proteins* $\rightarrow$ *accumulate in* $\rightarrow$ *Alzheimer's disease*. These graph-grounded results demonstrate the system's ability to return concise, interpretable, and text-aligned answers.

Further analysis of retrieval accuracy and alignment with ground truth answers is presented in the Evaluation section.

4 Evaluation

4.1 Triple Extraction Effectiveness

To assess the effectiveness of our two-pass triple extraction strategy, we conducted a comparative evaluation against a baseline single-pass method. Both strategies were applied to the same set of 94 biomedical abstracts, which were segmented into approximately 470 overlapping chunks to preserve context across sentence boundaries.

The baseline approach employed a single prompt that instructed the language model to extract all subject–predicate–object triples from each chunk in one pass. In contrast, our two-pass strategy consisted of a primary extraction step followed by a verification pass, allowing the model to identify additional long-range, implicit, or cross-sentential relationships that may have been missed initially.

Metric	**Single-Pass Prompt**	**Two-Pass Prompt**
Total Triples	3,641	6,528
Unique Abstracts	94	94
Unique Chunks	470	473
Avg Triples per Abstract	38.73	69.45
Min / Max per Abstract	2 / 121	4 / 227
Avg Triples per Chunk	7.75	13.80
Min / Max per Chunk	1 / 32	2 / 46

Table 1: Comparison of extraction statistics between a single-pass prompt and the proposed two-pass strategy across 94 biomedical abstracts.

As shown in Table 1, the two-pass strategy significantly increased triple yield across all aggregation levels. The number of extracted triples nearly doubled, with the average triples per chunk rising from 7.75 to 13.80. The expanded coverage reflects the verification step's ability to capture implicit and complex relationships that may not be directly stated in the text.

This enriched triple set served as the foundation for downstream graph-based retrieval, enabling the system to surface more comprehensive evidence in response to biomedical questions.

4.2 Retrieval Effectiveness

We evaluated whether the top-k retrieved triples from Neo4j correctly captured evidence from the source document and chunk associated with each question, stratified by biomedical domain. For this analysis, we set $k = 10$, meaning each query was answered using the top-10 ranked triples based on semantic similarity. The following metrics were computed:

- **Doc-Level Recall@10:** Whether any of the top-10 triples originated from the same document as the question.
- **Chunk-Level Recall@10:** Whether any triple was retrieved from the exact chunk linked to the question.
- **Relaxed Chunk Recall@10:** Whether any triple came from a chunk within a ± 2 offset from the reference chunk.
- **Precision@10:** Fraction of the top-10 triples that originated from the correct document or chunk.

- **Mean Reciprocal Rank (MRR@10):** Inverse of the rank of the first correctly matched triple among the top-10.

As shown in Figure 1, retrieval performance was consistently strong across all five biomedical domains—Alzheimer's Disease, COVID-19, Mental Health, Obesity, and Parkinson's Disease. Doc-level recall@10 exceeded 0.93 in every domain, peaking at 0.969 for Parkinson's. Relaxed chunk-level recall@10 mirrored this trend, remaining above 0.91 throughout.

In contrast, chunk-level precision@10 exhibited greater variability, ranging from 0.362 (Obesity) to 0.374 (COVID-19), reflecting the challenge of retrieving triples from the exact source chunk. However, relaxed chunk-level precision@10 was substantially higher, surpassing 0.69 in all domains and reaching 0.737 for Parkinson's.

MRR@10 scores further underscored this pattern: chunk-level MRR ranged from 0.599 to 0.652, while relaxed chunk-level MRR consistently exceeded 0.83. Parkinson's again achieved the highest value (0.877), indicating that relevant triples were often retrieved at top ranks.

These results demonstrate the effectiveness of the graph-based retrieval system in capturing semantically relevant evidence, even when exact chunk alignment is difficult. The strong relaxed metrics validate the system's robustness for downstream biomedical QA and reasoning tasks.

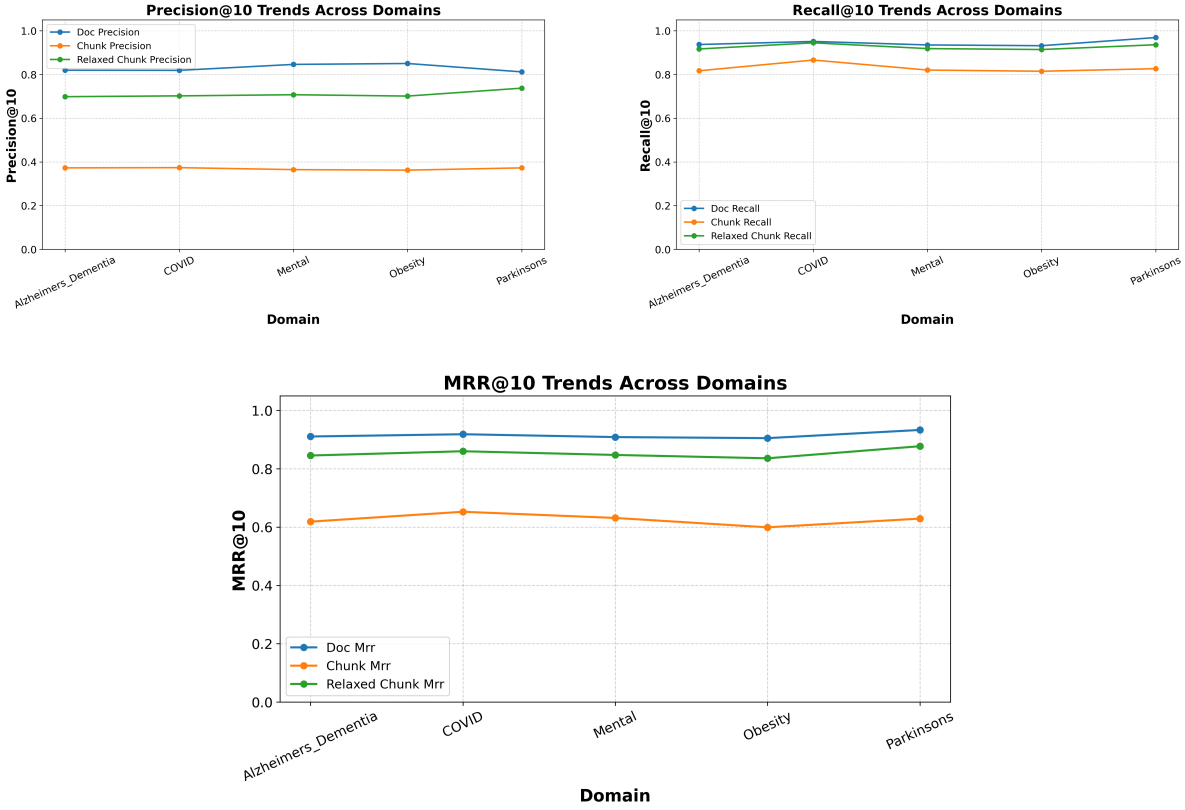

Figure 1: Domain-level evaluation of graph retrieval using top-10 triples, showing Precision@10, Recall@10, and Mean Reciprocal Rank (MRR@10) across strict and relaxed levels.

4.3 Answer Fidelity

We further evaluated whether the answers generated by Claude, using only the top-10 retrieved triples—were faithful to the original ground truth answers. To assess this, we computed both

lexical and semantic similarity using the following metrics:

- **BLEU** and **ROUGE-L**: Measure lexical overlap between generated and reference answers.
- **Cosine Similarity**: Captures semantic closeness using `all-roberta-large-v1` sentence embeddings.
- **Exact Match**: Measures cases where the generated answer exactly matched the ground truth.
- **"Not mentioned in the text" Rate**: Reflects appropriate abstention when the answer is not derivable from the retrieved triples.

Across the full evaluation set of 2,527 questions, the average BLEU score was 0.267, ROUGE-L was 0.438, and cosine similarity averaged 0.609. While exact string matches were rare—highlighting the model's tendency to paraphrase—the relatively high cosine scores indicate strong semantic alignment in many responses.

Figure 2 presents two complementary visualizations. Subfigure (a) shows domain-specific trends in BLEU, ROUGE-L, and cosine similarity. While BLEU and ROUGE scores remained modest across domains (reflecting limited lexical overlap), cosine similarity was consistently higher—ranging from 0.590 to 0.617—indicating that Claude generated semantically appropriate answers despite surface-level variation. Subfigure (b) presents the global cosine similarity distribution across all questions, revealing that:

- 794 answers (31%) achieved a cosine similarity ≥ 0.9, indicating high semantic fidelity;
- 893 answers (35%) fell in the medium range (0.5–0.9);
- 840 answers (33%) scored below 0.5, often due to incomplete evidence or vague phrasing.

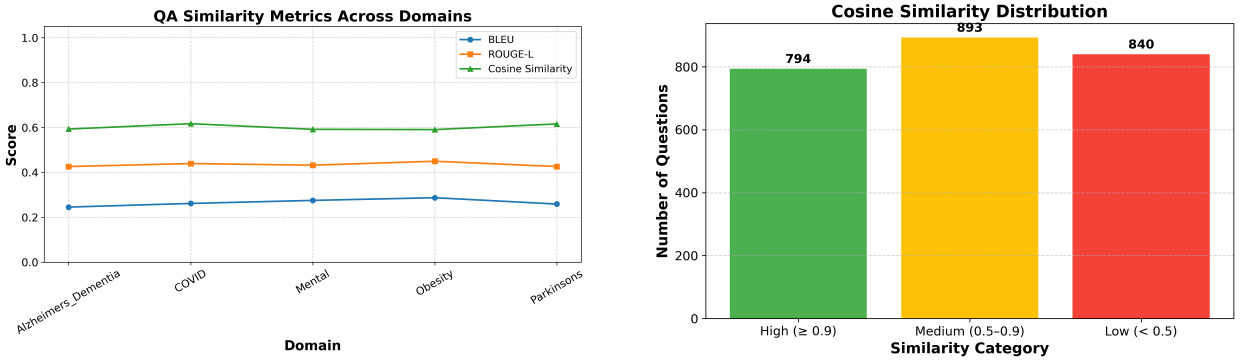

Figure 2: Answer similarity evaluation using lexical (BLEU, ROUGE-L) and semantic (cosine) metrics. Left: Domain-wise average scores. Right: Global cosine similarity distribution.

Overall, these findings validate the effectiveness of our graph-based retrieval pipeline in supporting accurate and semantically faithful answer generation. While the model often rephrased content rather than copying verbatim, it largely preserved meaning—demonstrating suitability for biomedical question answering where paraphrasing is common and surface overlap is not the sole indicator of correctness.

5 Discussion

Despite achieving a high overall success rate of **95.5%** across 2,647 biomedical QA runs, several failure cases revealed important limitations in the current system. Most errors fell into three categories:

1. **Entity Extraction Failures:** Claude failed to extract valid named entities from vague or underspecified questions.
2. **No Triples Found:** Neo4j returned no results for extracted or expanded entities, indicating sparsity in KG coverage.
3. **Unexpected Format Issues:** Some triples were malformed or failed to unpack properly during parsing.

These issues were particularly prominent for questions such as "*What is the main focus of the study?*" that lacked clear grounding in specific entities or chunk-local content.

Additionally, our current graph retrieval setup only supports **1-hop neighborhood** queries. This restricts the system's ability to reason over complex biomedical mechanisms that require multi-hop inference. Early tests using deeper retrieval queries (e.g., 2-hop subgraphs) reveal that richer pathways, such as *inflammation* $\rightarrow$ *AD pathology* $\rightarrow$ *cognitive decline*, are feasible and interpretable.

Another observation is that while strict chunk-level matching may miss some targets, relaxed chunk-level metrics (within ± 2 offset) consistently capture semantically adjacent information. This affirms the effectiveness of the retrieval mechanism, even when exact chunk alignment fails.

Finally, current entity representations are based solely on surface-level text. The lack of formal semantic alignment through biomedical ontologies (e.g., MeSH, UMLS, SNOMED CT) leads to redundancy and inconsistency in entity linking. This hampers deeper reasoning and standardization across documents.

5.1 Conclusion

This work introduces a benchmark pipeline for biomedical question answering built upon graph-based retrieval and language model-based answer generation. The evaluation demonstrates strong document-level recall and high semantic fidelity across five biomedical domains: Alzheimer's Disease, COVID-19, Mental Health, Obesity, and Parkinson's Disease.

However, the study highlights several limitations that must be addressed to achieve scalable and explainable QA:

- **Ontological Alignment:** Incorporating biomedical ontologies can unify equivalent entities, enable relationship generalization, and support knowledge inference.
- **Multi-Hop Reasoning:** Retrieval should move beyond 1-hop to capture causal or associative chains spread across multiple graph segments.
- **Improved Entity Linking:** Robust linking techniques are needed to handle vague or abstract queries.

Ultimately, by integrating ontology-driven reasoning into the graph retrieval and QA pipeline, we aim to build a semantically rich system capable of high-precision, high-recall biomedical knowledge access. This work lays a foundation for future efforts in benchmarking, expansion, and real-world deployment of ontology-aware biomedical question answering systems.

References

M. Agrawal, B. V. Srinivasan, and M. M. Khapra. Medcpt: Medical concepts, procedures, and treatments in pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 4907–4918. Association for Computational Linguistics, 2022.

Z. Liu, Y. Chen, J. Wang, and K. Chang. Ontomed: Ontology-based normalization techniques for biomedical entity recognition. *Journal of Biomedical Informatics*, 151:104503, 2024.

L. Pan, Y. Song, Z. Zhang, and Z. Liu. Medrag: Integrating medical knowledge graphs with retrieval augmented generation. *Journal of Biomedical Informatics*, 143:104417, 2023.

A. Roberts, R. Johnson, and M. Lewis. Medqa-kg: A benchmark dataset for medical question answering with knowledge graphs. In *Proceedings of the 2023 ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 278–287. ACM, 2023.

H. Zhang, Y. Wang, and Z. Chen. Medkgraph: A framework for constructing biomedical knowledge graphs from literature. *Biomedical Informatics Insights*, 15:11779322231151294, 2023.