Assessing the Robustness of Knowledge Graph RAG Architectures in the Presence of Incorrect Data

Zachary Grider[1], Ajay Abburi[1] and Alexander Nelson[1,*]

[1]University of Arkansas, Department of Electrical Engineering and Computer Science, 1 University of Arkansas, Fayetteville, AR 72704, USA

Abstract

Retrieval-Augmented Generation (RAG) using knowledge graphs effectively mitigates model hallucinations and improves reasoning over multi-hop queries. However, as RAG systems dynamically ingest data from unverified web sources, corrupt or contradictory information can severely degrade output reliability—a critical vulnerability in high-stakes domains such as healthcare. Using the HealthContradict dataset, we systematically evaluate knowledge-graph-based RAG architectures under noisy data conditions. Specifically, we measure how erroneous triples in both structured and unstructured contexts impact downstream performance, compare the robustness of general-purpose versus domain-adapted (fine-tuned) language models against false information, and analyze the downstream effects of varied graph traversal depth (single-hop, two-hop, and semantic cutoff strategy). Our findings highlight key trade-offs between context depth and noise resilience in medical fact verification systems.

Keywords

Knowledge Graphs, Language Models, Retrieval Augmented Generation, Knowledge Poisoning, QA Systems

1. Introduction

Humanity entered the "Zettabyte Era" in 2016 [1] and is projected to reach the Yottabyte Era by 2030 [2], representing a thousandfold increase in less than 2 decades. This growth has enabled big-data approaches to achieve advancements across a wide range of disciplines. However, integrating and reasoning from heterogeneous, highly-interconnected data remains a central challenge [3]. While relational models excel for tabular data, they struggle to capture complex, interconnected, multi-dimensional relationships. Knowledge graphs (KGs) have emerged as a promising technology for modeling this complexity [4]. KGs address this challenge by representing entities and their typed relationships, enriching both with properties and structuring them through ontologies. The adaptation and usage of knowledge graphs for storing and representing linked, semantic data has shown success and growth across various disciplines. For instance, in bioinformatics, KGs have been shown to model the complex, intricate interconnectedness of relationships between proteins, genes, diseases and symptoms[5] [6].

At the same time, large language models (LLMs) have demonstrated remarkable capabilities in question answering, document summarization, and scientific reasoning tasks. Despite these advancements, LLMs remain susceptible to hallucinations and generate responses that appear plausible, but are unsubstantiated or even factually incorrect. This limitation is especially concerning in biomedical domains where inaccurate information can influence medical decision making. To mitigate these hallucinations, retrieval augmented generation[7] (RAG) systems incorporate external knowledge sources into the generation process, grounding the responses in the retrieved evidence rather than solely on pre-trained model parameters.

RAG approaches largely are based on vector-based retrieval from unstructured text collections to reduce the context window. Unstructured retrieval can struggle with complex relational queries, especially queries that require explicit reasoning across multiple entities and relationships. Knowledge

RAGE-KG 2026: The Third International Workshop on Retrieval-Augmented Generation Enabled by Knowledge Graphs, co-located with ISWC 2026, October 25–26, 2026, Bari, Italy

*Corresponding author.

zg009@uark.edu (Z. Grider); aabburi@uark.edu (A. Abburi); ahnelson@uark.edu (A. Nelson)

https://ahnelson.uark.edu/ (A. Nelson)

0009-0009-6235-6487 (Z. Grider); 0009-0007-2804-2148 (A. Abburi); 0000-0003-3383-0193 (A. Nelson)



© 2026 Copyright 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 2026-11-04

CEUR-WS.org/Vol-4312/paper30.pdf

Graph Retrieval-Augmented Generation (KG-RAG) has emerged as a promising alternative that combines reasoning with a more structured presentation provided by the KG. By representing the information as entities connected through relationships, KGs provide interpretable retrieval pathways that support multi-hop reasoning and provide explicit provenance for retrieved facts.

However, the effectiveness of KG-RAG systems depends on the quality of the underlying knowledge graph. In real-world deployments, a default assumption of truth in the knowledge graph is broken as it cannot be assumed that graphs are constantly curated by domain experts. In this model, we assume that KG will be constructed and continually updated from large and continuously augmented document collections using named entity recognition, entity linking, and relation extraction. This is an attractive solution because it improves scalability by reducing the amount of manual curation, but does introduce opportunities for error, including extraction errors, missing relationships, and contradictory facts. Once incorporated into the graph, these errors may be assumed as supporting evidence and used in downstream reasoning tasks by LLMs.

Despite these risks, limited research has been devoted to understanding how errors propagate through KG-augmented pipelines including 1) How KG construction choices and how they influence downstream reasoning performance, 2) how retrieval depth affects the propagation of correct or incorrect evidence, or 3) how different generator models respond when presented with graph-derived information of varying quality. To address these limitations, we conduct a systematic evaluation of KG-RAG robustness using the HealthContradict dataset [8]. We investigate different KG construction models, retrieval strategies, how these perform when provided with only correct context, or in the presence of incorrect context. The goal of our study was to understand how errors introduced during graph construction and retrieval influence the reliability of downstream language model reasoning tasks.

Research Questions:

1. How does the choice of knowledge graph construction model impact biomedical fact-verification accuracy in KG-RAG systems?
2. How does retrieval depth influence the effectiveness of graph-based retrieval for fact verification?
3. How robust are different generator models when reasoning over graph-derived evidence?
4. How does the presence of correct versus incorrect biomedical knowledge affect KG-RAG performance?
5. What combinations of KG construction, retrieval strategy, and generator model produce the highest fact-verification accuracy on HealthContradict?

The primary contributions of this work are:

1. We conduct a systematic factorial evaluation ($N = 60$ configurations) across five knowledge graph extraction pipelines, three graph retrieval depths, two generator models, and two context correctness modes using the HealthContradict benchmark.
2. We demonstrate that knowledge graph construction quality is the single dominant factor governing downstream fact-verification accuracy ($F = 27.85$, $p < 0.001$), outperforming choices in retrieval depth and generator architecture.

Section II will cover related background work regarding knowledge graphs, entity extraction, retrieval augmented generation, and how to evaluate these systems. Section III will cover the methodology including dataset curation, knowledge graph generation, custom retrieval strategy, our RAG pipeline, an optional LLM-as-a-judge stage, and our statistical analysis framework. Section IV demonstrates the results from the methodological section, describing the results for each of our five research questions as well as a full-factor ANOVA analysis for the results. Section V is a discussion on the results from our methodology and how those impact KG-RAG approaches without an assumption of absolute correct context. The work concludes in Section VI.

2. Background

2.1. Knowledge Graphs in Biomedical Domains

A knowledge graph is a structure defined as $G = \{E, R\}$, where E is the set of all entities contained in the graph and R is the set of all relations in the graph. An RDF knowledge graph is a knowledge graph expressed using the Resource Description Framework (RDF)[9], with a relationship of entities in the knowledge graph being expressed as a "triple" in the form of h, r, t where $h \subset E$, $r \subset R$, and $t \subset E$. Each of these triples is considered to be a "fact" within the RDF knowledge graph which is assumed to be true for the given context. Knowledge graphs can be traversed by walking relationships which may point to other head entities from tail entities, with tail entities also consisting of URIs which may point to domains within the graph or to external resources. Knowledge graphs as a way to store and represent data are used in many applications including the biomedical domain[10], cybersecurity domain[11], and quality engineering domain[12].

The usage of knowledge graphs has also integrated them into various systems such as recommender systems[13, 14], and retrieval-augmented generation (RAG) systems[15]. One common approach of using knowledge graphs is to create knowledge graph embeddings (KGEs) using various approaches[16, 17, 18, 19, 20], from the entities and relationships to extract patterns which allow for question answering[21, 22, 23]. These KGEs can be improved by creating strong negative triples since KGs are assumed to be full of facts[24]. Knowledge graphs and embedding representations have also found plenty of use in the biomedical domain, especially for representing drug-target interactions[10, 25, 26, 27, 28]. Common tasks for knowledge graphs include knowledge graph creation which is creating knowledge graphs from an existing body of unstructured data[29, 30], and knowledge graph completion to infer relationships (link prediction)[31] or infer missing nodes (entity completion).

2.2. Named Entity Recognition and Relation Extraction

Named Entity Recognition is the process of extracting noun phrases from text.[32] Classical flat NER tasks used large, fine-tuned datasets with a preset number of entity spans which are manually annotated by humans in order to train models to identify entities aligning with human reasoning[33]. These classical models employed rule-based learning, machine learning, or a hybrid approach[34]. However, due to rigid labels of traditional NER systems, their performance in specific domains degrades due to the dissimilarity between the trained entity labels and domain-specific labels[35]. This motivated the creation of domain-specific datasets for NER[36, 37]. Advances in machine learning and neural networks led to new approaches using Large Language Models (LLMs)[38], which show promising performance across a variety of NLP tasks. For NER tasks, GPT-NER[39] shows comparable performance to the SOTA supervised-learning models on benchmark NER datasets, especially under the few-shot learning[40] and in low-resource settings. DEER[41] is an architecture for improving the retrieval of entity spans for in-context learning (ICL)[42] of LLMs for NER and found increased performance of LLMs for NER across various flat NER benchmark datasets, outperforming other ICL methods and approaching supervised SOTA NER models.[43] Evaluated LLMs using different input/output formats, ICL strategies, and encoders for NER against fine-tuned models and finding strong ICL strategies can outperform fine-tuned LLMs on some biomedical NER datasets at a lower price. They also evaluated a retrieval-augmented generation (RAG)[7] approach for zero-shot NER, which demonstrated SOTA performance in some cases.

Relation extraction (RE) is the process of determining a relationship type between two entities, which can be expressed as three-tuples <E1, R, E2>.[44] Directly related to NER, datasets and tasks were developed from human-annotated text for RE including NER.[45] RE methods have improved over the years from traditional machine learning models[46] to deep learning methods with attention mechanisms[47]. With the explosion of LLMs, they have been applied to the RE task with mixed results. SciER[48] is a dataset created over 100 manually annotated full-text scientific documents. The authors evaluated finetuned supervised SOTA models against LLMs, finding that even with few-shot prompting the LLMs performed significantly worse than the SOTA models for RE. BioGPT[49] is a GPT

model finetuned upon medical information and evaluated across NLP tasks, and demonstrated the best performance on the RE task, outperforming the SOTA models and base GPT2.

Common approaches for NER and RE tasks work either as a pipeline, where NER is performed and then RE is performed in a second-pass given the NER label spans, or as joint extraction, where NER and RE are performed in a single task. While LLMs have strong performance across many NLP tasks[50], research shows that supervised encoder models achieve comparable or better performance in many cases, including zero-shot NER[51], zero-shot RE[52], pipeline NERRE[53], and joint NERRE[54], while only needing a fraction of the time or space. However, these models may require NER or RE labels as seeds where LLMs may benefit from ICL but do not require it.

2.3. Retrieval Augmented Generation

Retrieval-augmented generation (RAG)[7] is a framework which is meant to allow LLMs access to knowledge bases in order to provide more relevant information and also avoid hallucinations[55]. RAG has become a popular harness for applying LLM capabilities in domain-specific scenarios, such as the biological and medical domains[56], the legal and policy domains[57], and even for enabling AI agents to improve their tool-retrieval capabilities[58]. A RAG system consists of a knowledge base, a retriever which finds documents relevant to a given input or query, and a generator which is usually an LLM to form more accurate or relevant responses. Standard RAG systems use a vector store, which have multiple types and availabilities[59]. The store acts as the knowledge base for the documents, which are usually split into separate parts - called ”chunks” - due to the limitations of the generator’s context space. The improvement for different chunking strategies is inconclusive[60], but chunking is required for performing embedding-based retrieval[61], as each chunk is converted to an embedding for searching similar chunks based upon the incoming query. Other retrieval methods include sparse retrieval[62], retrieval with ”late interaction”[63], and others.

Other approaches to RAG are current research topics. Graph-based RAG is the RAG architecture but instead a graph-based store is used instead of the vector store, and retrievers can extract entity-relations, perform semantic lookup, or offer a hybrid approach. Graph-based RAG has shown increased performance compared to vector store RAG for NLP tasks such as multi-hop question-answering[64] and whole corpus focused queries[65]. Since graph RAG is based upon graph structures, which are usually knowledge graphs, they have found many applications in the biomedical domain. HyKGE[66] uses hypothesis optimization to first generate hypotheses based upon user medical queries, then use the retrieved hypotheses to locate related entities rather than solely based on semantic similarity of entities related to the user query. The authors report HyKGE has the best performance on medical QA benchmarks regardless of generator choice. MedGraphRAG[67] generates a layered graph based upon factual medical data, and uses a summary and tag to tie a query to the layer of the graph which best answers their question. The authors report state of the art performance across multiple medical benchmarks for both fact-checking and QA benchmarks. In [68] the authors explore the performance of open-weight LLMs for biomedical accuracy using the HealthContradict dataset[8]. They observe the next-token prediction statistics based upon incorrect and correct biomedical documents and control for parametric knowledge, context positions, and context correctness with their results showing all models have reduced accuracy and change their answers. They also experiment with a scoring mechanism which uses model confidence and conflicting evidence detection which demonstrated improvements in model accuracy compared to model confidence only.

2.4. Evaluating RAG systems

RAG systems have demonstrated some advances compared to base LLM-only systems, but increase the system complexity. While evaluation metrics for text generation tasks can suffice depending on the query, such as ”yes or no” for answer accuracy, more complex queries and responses such as summarization or logical reasoning require advanced metrics, such as text overlap[69] or using other LLMs to gauge response quality, referred to as ”LLM-as-a-judge”[70], RAG systems incorporate extra

components which need to be measured to determine the output of the system, such as retrieval quality of chunks and the performance of the retriever itself. RAGAS[71] is a system developed which uses LLM-as-a-judge to report metrics such as answer relevance, faithfulness, and context relevance.

3. Methodology

3.1. Dataset

The HealthContradict dataset consists of 920 biomedical claim-verification instances. Each instance contains a query, a supporting document containing evidence consistent with the claim, and a contradicting document containing evidence that opposes the claim. The dataset was designed to evaluate the ability of models to distinguish factual and contradictory biomedical information. In this work, the supporting and contradicting documents were used independently to construct knowledge graphs for retrieval and downstream fact verification.

3.2. Experimental Design

The objective of this study is to evaluate how knowledge graph construction methods, retrieval strategies, and generator model selection influence downstream biomedical fact-verification performance in KG-RAG systems. To achieve this objective, we employed a factorial experimental design in which each component of the KG-RAG pipeline was systematically varied while the remaining components were held constant.

Four experimental factors were considered. The first factor was the knowledge graph construction model, consisting of five extraction pipelines: GLiNER-Relex[54], GLiNER-Bio[51] with GLiREL[52], GLiNER2[53], Llama3-OpenBio-8B[72], and GPT-OSS-20B[73]. The second factor was the retrieval strategy, which included 1-hop retrieval, 1–2 hop retrieval, and semantic 1–2 hop retrieval. The third factor was the generator language model used for fact verification, consisting of Llama3-OpenBio-8B and Gemma2-2B-IT[74]. The final factor was the document context type, where knowledge graphs were generated from either supporting (correct) documents or contradicting (incorrect) documents within the HealthContradict dataset.

For each HealthContradict query, a knowledge graph was first constructed from the associated document collection using one of the five extraction approaches. The custom retrieval framework was then used to generate a retrieval context according to the selected graph traversal strategy. The retrieved triples were supplied as evidence to a generator model, which produced an initial classification of *Yes*, *No*, or *Unknown*. Responses and associated reasoning were subsequently passed to the LLM-as-Judge framework to generate a final prediction. This prediction was compared against the HealthContradict ground-truth label to determine classification correctness.

The resulting experimental design evaluates the independent and combined effects of graph construction quality, retrieval depth, and language model selection on downstream fact-verification accuracy. The complete experimental matrix is summarized in Table 1.

Table 1
Experimental factors and evaluated configurations.

Factor	Configurations
KG Construction	GLiNER-Relex, GLiNER-Bio+GLiREL, GLiNER2, Llama3-OpenBio-8B, GPT-OSS-20B
Retrieval Strategy	1-hop, 1–2 hop, Semantic 1–2 hop
Generator Model	Gemma2-2B-IT, Llama3-OpenBio-8B
Context Type	Correct, Incorrect

The full factorial design produced a total of $5 \times 3 \times 2 \times 2 = 60$ unique KG-RAG configurations. Each configuration was evaluated across all 920 HealthContradict queries, resulting in a comprehensive

assessment of how retrieval and reasoning performance change as knowledge graph construction methods, retrieval depth, and generator models are varied.

This design enables the investigation of the research questions outlined in Section 1, including the impact of knowledge graph construction quality on fact verification, the role of multi-hop retrieval, the robustness of biomedical versus general-purpose language models, and the influence of correct versus incorrect biomedical knowledge on KG-RAG system performance.

3.3. Knowledge Graph Creation

To evaluate the impact of knowledge graph construction on downstream fact verification performance, we generated knowledge graphs from the HealthContradict dataset using both the correct and incorrect document types using five open-source extraction models GLiNER-Relex, GLiNER-bio, GLiNER2, GPT-OSS, and LLAMA3-OPENBIO.

Prior to knowledge graph extraction, documents were segmented using semantic chunking with `semchunk`[75]. Chunk sizes were selected based on the computational requirements of the extraction model. For the structured extraction pipelines, documents were segmented using a chunk size and overlap ratio of 256:32 tokens. Larger chunk sizes of 1024:128 tokens were used for dense language-model-based extraction approaches to reduce memory consumption while preserving contextual information.

The evaluated extraction approaches consist of both structured and unstructured knowledge graph construction methods. Three structured pipelines utilize zero-shot Named Entity Recognition (NER) frameworks: GLiNER-Relex, GLiNER-Bio , and GLiNER2. These models require predefined entity and relation labels during extraction. Standard biomedical entity categories were provided to the models[76]. However, relation labels required manual definition due to the absence of benchmark relation schemas within HealthContradict. GLiNER-Bio performs entity extraction only and does not directly generate relationships. Consequently, entities identified by GLiNER-Bio were passed to GLiREL, which utilized the predefined relation labels to extract relationship triples.

In contrast, the unstructured extraction approaches do not require predefined entity or relation schemas. Instead, Llama3-OpenBio-8B and GPT-OSS-20B were prompted to directly generate biomedical subject-predicate-object triples from document text. This approach provides greater flexibility in entity and relation discovery, allowing extraction of relationships that may not fit a predefined label. Llama3-OpenBio-8B is additionally specialized through biomedical fine-tuning on healthcare-related corpora, including doctor-patient interactions, making it well suited for biomedical knowledge extraction tasks.

The extracted triples from each model were subsequently ingested into a Neo4j graph database to create property knowledge graphs. Each graph node and relationship was enriched with associated metadata, including document chunks, entity labels, instance identifiers, document type indicators (correct or incorrect), document identifiers, and entity names. This process enabled the construction of a structured biomedical knowledge repository for retrieval and downstream reasoning tasks.

Because structured extraction pipelines rely on predefined entity and relation schemas, they generally produce graphs with a more constrained set of node and relationship types. Conversely, the unstructured extraction approaches generate a broader and more diverse set of entities and relations, increasing graph expressiveness at the cost of reduced standardization. All generated knowledge graphs were deployed to Neo4j AuraDB and served as the retrieval backend for the subsequent KG-RAG fact-verification experiments.

3.4. Custom Retrieval Strategy

To ensure that differences in downstream performance could be attributed to knowledge graph construction quality rather than the retrieval capabilities of a separate language model, we developed a custom graph retrieval framework. A strictly-LLM-based retriever can also introduce variability due to their non-deterministic nature. Consequently, retrieval was performed directly from the graph structure using deterministic graph traversal methods.

Retrieval begins by extracting entities from the input query using the same Named Entity Recognition (NER) model that was used to construct the corresponding knowledge graph. The extracted entities are then used to query the graph database, retrieving all triples in which the identified entities appear as either the source or target node. To ensure that retrieved knowledge remains associated with the appropriate HealthContradict query instance, all graph queries utilize the `instanceID_type` property, allowing retrieval to be constrained to the subgraph corresponding to a specific query-document pair.

The baseline retrieval strategy consists of a 1-hop neighborhood search, in which all directly connected triples associated with the query entities are returned. To support more complex reasoning, this approach was extended to include two additional retrieval strategies: a 1–2 hop retriever and a semantic 1–2 hop retriever. The 1–2 hop retriever expands the retrieved subgraph by traversing one additional edge from the initial set of retrieved triples, allowing access to indirectly related entities and relationships. Because unrestricted graph expansion can rapidly increase the number of retrieved triples and exceed available GPU memory during generation, retrieval depth was constrained through a series of heuristics. Specifically, only the first 50 triples retrieved during the initial 1-hop search were considered for expansion, and the final retrieval context supplied to the generator model was capped at 150 triples.

For the semantic 1–2 hop retrieval strategy, an additional filtering stage was introduced prior to graph expansion. Similarity scores between the query and the retrieved 1-hop triples were computed using the Qwen3-4B[77] embedding model. Only triples with a semantic similarity score greater than 0.55 were retained as candidates for expansion. To further control context size, a maximum of 50 semantically relevant triples were selected and expanded into their corresponding 2-hop neighborhoods. This process prioritizes graph regions that are more closely aligned with the query while reducing the inclusion of unrelated or noisy relationships.

The resulting retrieval strategies, namely 1-hop, 1–2 hop, and semantic 1–2 hop retrieval, provide progressively larger and more context-rich subgraphs. These retrieved triples serve as the evidence context supplied to downstream generator models for biomedical fact-verification and retrieval-augmented generation experiments.

3.5. Retrieval-Augmented Generation

The triples retrieved using the custom retrieval framework serve as the context provided to the downstream generation models for fact verification. Given a query and its associated retrieved triples, the objective of the generator is to classify the claim as *Yes*, *No*, or *Unknown* based solely on the evidence present in the retrieved subgraph.

To evaluate the influence of generator model selection, two language models were used: Llama3-OpenBio-8B and Gemma2-2B-IT. Llama3-OpenBio-8B is a biomedical domain-adapted model designed to incorporate medical and healthcare knowledge, whereas Gemma2-2B-IT is a general-purpose instruction-tuned language model. Both models were provided with identical query and evidence contexts and prompted using a common fact-verification framework. Minor modifications were made to the prompt templates to accommodate model-specific response formatting and output parsing requirements.

The models were instructed to evaluate the retrieved evidence and return one of three possible stances: *Yes*, *No*, or *Unknown*. In addition to the classification label, both models were prompted to provide a justification explaining the reasoning process used to arrive at the predicted stance. While Gemma2-2B-IT generally produced structured responses that explicitly separated the predicted stance from supporting reasoning, Llama3-OpenBio-8B often generated less standardized outputs due to its conversational fine-tuning objectives. Consequently, normalization procedures were applied to extract the final prediction labels from model responses.

During preliminary experimentation, we observed that both insufficient retrieval context and excessively large retrieval contexts frequently resulted in an *Unknown* prediction. In some cases, relevant evidence was absent from the retrieved subgraph, while in others, the volume of retrieved triples introduced ambiguity and reduced the model's confidence in selecting a definitive stance. Although the *Unknown* category is appropriate when supporting evidence is genuinely inconclusive, excessive use of this response can obscure meaningful differences in fact-verification capability across retrieval and

graph construction configurations.

3.6. LLM-as-Judge Evaluation

The proportion of inconclusive predictions varied substantially across the evaluated KG-RAG configurations. In particular, Llama3-OpenBio-8B demonstrated a strong tendency to predict *Unknown* when presented with either limited evidence or large retrieval contexts containing potentially conflicting information. Although abstaining from a definitive prediction is appropriate when insufficient evidence exists, excessive use of the *Unknown* category makes direct comparison between retrieval strategies and generator models difficult and may obscure meaningful differences in fact-verification performance.

To enable a more consistent evaluation across all experimental configurations, an LLM-as-Judge framework was introduced as a secondary decision stage. The objective of the judge model was not to generate new evidence, but rather to evaluate the reasoning and supporting context already produced by the KG-RAG pipeline and determine whether a conclusive stance could be inferred from the available information. For this purpose, Qwen3-14B[78] was selected as the judge model. The judge was provided with the original query, the retrieved knowledge graph triples, and the response generated by the primary KG-RAG model, including any supporting reasoning. Using this information, the judge model was instructed to produce a final verdict of *Yes*, *No*, or *Unknown* based on the evidence available in the retrieval context.

The use of an independent judge model serves two purposes. First, it reduces variability introduced by differences in response style and confidence across generator models. Second, it provides a mechanism for reevaluating inconclusive predictions using the reasoning already generated by the primary model, allowing potentially valid evidence to be considered before assigning a final classification. This approach promotes a more equitable comparison between generator models and retrieval strategies while maintaining reliance on the same underlying knowledge graph evidence.

The final verdict produced by the judge model was used as the prediction for downstream evaluation. These predictions were subsequently compared against the HealthContradict ground-truth labels to calculate classification accuracy, error rate, and other performance metrics across all knowledge graph construction methods, retrieval configurations, and generator models considered in this study.

3.7. Statistical Analysis Framework

To rigorously evaluate the individual and combined effects of knowledge graph construction, retrieval strategy, generator model choice, and context mode on biomedical fact-verification performance, we employed a multi-tiered statistical evaluation framework. All statistical tests were evaluated at a significance threshold of $\alpha = 0.05$.

Factorial Analysis of Variance (ANOVA): To evaluate main effects and interaction terms across our 5×3×2×2 experimental matrix ($N = 60$ unique configurations), we computed a 4-Factor Type-II Analysis of Variance (ANOVA [79]) on classification accuracy ($\text{Accuracy}_{\%}$). Type-II ANOVA was selected to account for main effects while controlling for higher-order interactions without assuming complete orthogonality among factors. The test evaluated four experimental factors: (1) **KG Construction Model** (5 levels: *GLiNER-Relex*, *GLiNER-Bio+GLiREL*, *GLiNER2*, *Llama3-OpenBio-8B*, *GPT-OSS-20B*); (2) **Retrieval Strategy** (3 levels: *1-hop*, *1–2 hop*, *Semantic 1–2 hop*); (3) **Generator Model** (2 levels: *Gemma2-2B-IT*, *Llama3-OpenBio-8B*); and (4) **Context Mode** (2 levels: *Correct*, *Incorrect*).

Tukey's Honestly Significant Difference (HSD) Post-Hoc Tests: Where ANOVA indicated statistically significant main or interaction effects ($p < 0.05$) among multi-level categorical factors (KG construction models and retrieval strategies), we conducted Tukey's HSD post-hoc pairwise comparisons [80]. Tukey's HSD controls the Family-Wise Error Rate (FWER) across multiple pairwise comparisons, allowing us to identify specific pairwise differences between extraction frameworks and retrieval depths without inflating Type I error rates.

Paired Hypothesis Testing: Because each unique architectural pipeline (KG Model × Retrieval Strategy × Generator Model) was evaluated under both *Correct* and *Incorrect* context conditions ($N = 30$ paired configurations), we performed paired statistical comparisons using both a **Paired Student's *t*-test** (to measure systematic mean accuracy shifts assuming normality) and a **Wilcoxon Signed-Rank Test** [81] (a non-parametric alternative ranking paired differences without assuming normality). This provided robust verification of performance degradation when systems reason over contradicting biomedical evidence.

3.8. Compute Infrastructure & Reproducibility

Experiments were conducted on Pittsburgh Supercomputing Cluster Briges-2 as described in the Acknowledgments. Depending on model sizing and extraction memory demands, the resources were selected from NVIDIA V100 (32GB), L40S (48GB), and H100 (80GB) GPUs. The underlying software environment is built on Python 3.10 with PyTorch (v2.x) and CUDA 12.6, utilizing FlashAttention-2 for accelerated transformer inference. Graph operations were backed by a Neo4j instance (v5.28) and orchestrated using the LlamaIndex framework.

To ensure exact reproducibility, the complete environment configuration (‘requirements.txt‘) containing precise package versions is made publicly available at https://github.com/AESIRLab/graphRAG-reasoning.

4. Experiments and Results

4.1. Experiment 1: KG-Construction Models

For Experiment 1, we investigated the performance of the five Knowledge Graph constructions described in Table 1. For each construction method, we calculated the per-family mean accuracy across the 920 total queries ($N = 12$ pipeline configurations per family). These results are reported in Table 2.

Table 2
Mean accuracy (%) for each KG Construction Model family across all pipeline configurations.

KG Configuration	Mean (%)	Std	Min	Max	N
Llama3-OpenBio-8B	74.10	0.98	72.61	75.65	12
GLiNER-Bio+GLiREL	65.45	8.12	54.35	75.98	12
GLiNER-Relex	63.02	3.56	56.30	68.37	12
GPT-OSS-20B	60.79	11.26	45.76	72.93	12
GLiNER2	51.93	3.58	46.09	56.96	12

The primary factorial ANOVA revealed a significant main effect of KG construction model on fact verification accuracy, $F(4, 51) = 27.85$, $p < 0.001$. This result indicates that choice of KG construction method is a major determinant of downstream fact verification accuracy. Post-hoc Tukey HSD comparisons showed that `Llama3-OpenBio-8B` consistently outperformed every other model family ($p < 0.02$ for all pairwise comparisons), achieving the highest mean accuracy (74.10%) with minimal variance (SD = 0.98). Conversely, `GLiNER2` demonstrated the lowest performance (51.93%), lagging significantly behind `Llama3-OpenBio-8B` ($\Delta = -22.17\%$, $p < 0.001$). `GLiNER-Bio+GLiREL` (65.45%), `GLiNER-Relex` (63.02%), and `GPT-OSS-20B` (60.79%) formed an overlapping middle-tier cluster with no statistically significant differences between one another ($p > 0.40$).

4.2. Experiment 2: Retrieval Depth

To evaluate research question 2 on the effect of retrieval depth, we evaluated fact-verification accuracy across the retrieval depths described in Section 3.4, namely: 1-hop, 1-2 hop, and Semantic 1-2 hop ($N = 20$ configurations per strategy). Descriptive statistics are summarized in Table 3.

Table 3
Mean accuracy (%) grouped by retrieval strategy.

Retrieval Strategy	Mean (%)	Std	Min	Max	N
Semantic 1-2 hop	64.56	8.92	50.76	75.98	20
1-2 hop	63.53	9.86	45.76	75.98	20
1-hop	61.09	10.28	46.09	75.65	20

Although multi-hop retrieval strategies yielded higher descriptive mean accuracies (Semantic 1-2 hop reaching 64.56% compared to 61.09% for 1-hop), the overarching factorial ANOVA indicates that the main effect of retrieval depth alone is not statistically significant, $F(2, 51) = 2.30, p = 0.111$. A comprehensive decomposition of retrieval interactions across KG models is provided in Section 4.6.

4.3. Experiment 3: Generator Models and Abstention Behavior

To evaluate generator robustness, we evaluated both overall fact-verification accuracy and initial abstention rates (the proportion of *Unknown* responses returned prior to judging) between `Llama3-OpenBio-8B` and `Gemma2-2B-IT` ($N = 30$ per generator).

When controlling for variance across KG construction, retrieval, and context modes in our primary factorial model, generator choice exhibited a statistically significant main effect on final accuracy, $F(1, 51) = 9.03, p = 0.0041$, with `Llama3-OpenBio-8B` (65.10%) outperforming `Gemma2-2B-IT` (61.02%).

Table 4
Initial abstention rate (proportion of *Unknown* responses prior to LLM-as-Judge) by generator model.

Generator Model	Initial Unknown Rate (%)	Std	Min	Max	N
Llama3-OpenBio-8B	75.82	0.1451	50.22	97.61	30
Gemma2-2B-IT	32.95	0.1690	16.30	65.11	30

Prior to the judge stage, the two generators exhibited starkly different initial decision-making behaviors, $F(1, 58) = 111.19, p < 0.001$ (Table 4). `Llama3-OpenBio-8B` adopted a highly conservative strategy, abstaining on an average of 697.57 queries per run (75.82% initial unknown rate), whereas `Gemma2-2B-IT` abstained on 303.10 queries (32.95%).

The LLM-as-Judge stage virtually eliminated abstentions entirely across both models. The judge stage successfully resolved over 99.9% of all initial *Unknown* responses, driving the final unclassified query rate down to near-zero (0.036% for `Llama3-OpenBio-8B` and 0.011% for `Gemma2-2B-IT`). This demonstrates that while the generator choice heavily dictates initial prediction confidence, the secondary judge phase acts as an effective complete resolution layer for ambiguous or uncommitted outputs.

4.4. Experiment 4: Presence of Incorrect Context Data

In Experiment 4, we assessed system resilience when presented with deliberately corrupted or incorrect context data compared to solely correct context ($N = 30$ matched configurations per mode).

Table 5
Fact-verification accuracy (%) by context correctness mode.

Context Mode	Mean (%)	Std	Min	Max	N
correct	66.28	8.30	48.37	75.98	30
incorrect	59.83	9.94	45.76	75.98	30

When pipelines are provided with correct context, they achieved a mean accuracy of 66.28%, compared to 59.83% when provided incorrect context as shown in Table 5. A paired t-test was used to measure the effect of this condition because each experiment was conducted for both correct and incorrect context modes. The resulting t-statistic confirms that context corruption causes a statistically significant

degradation in performance, $t(29) = 4.54$, $p < 0.001$. To verify this finding without an assumption of normally distributed differences, we conducted a non-parametric Wilcoxon signed-rank test, which confirmed the paired t-test result ($W = 6.50$, $p < 0.001$).

4.5. Experiment 5: Top Pipeline Configurations

To identify optimal end-to-end architectures, we ranked all 60 pipeline configurations by fact-verification accuracy. The top 10 pipelines are presented in Table 6.

Table 6
Top 10 highest-performing experimental pipeline configurations.

KG Model	Retrieval	Generator	Context Mode	Accuracy (%)
GLiNER-Bio+GLiREL	1–2 Hop	Llama3-OpenBio-8B	Incorrect	75.98
GLiNER-Bio+GLiREL	Semantic 1–2 Hop	Llama3-OpenBio-8B	Correct	75.98
Llama3-OpenBio-8B	1-Hop	Llama3-OpenBio-8B	Correct	75.65
GLiNER-Bio+GLiREL	1–2 Hop	Llama3-OpenBio-8B	Correct	75.33
Llama3-OpenBio-8B	1–2 Hop	Llama3-OpenBio-8B	Correct	75.11
Llama3-OpenBio-8B	Semantic 1–2 Hop	Llama3-OpenBio-8B	Correct	74.78
Llama3-OpenBio-8B	Semantic 1–2 Hop	Gemma2-2B-IT	Correct	74.78
Llama3-OpenBio-8B	Semantic 1–2 Hop	Llama3-OpenBio-8B	Incorrect	74.67
Llama3-OpenBio-8B	1-Hop	Gemma2-2B-IT	Correct	74.35
GLiNER-Bio+GLiREL	Semantic 1–2 Hop	Llama3-OpenBio-8B	Incorrect	74.35

The highest individual accuracy reached 75.98%, achieved by `gliner-bio-glirel` coupled with `OpenBio`. However, `llama3-openbio` demonstrated superior stability, accounting for 6 of the top 10 overall configurations across multiple retrieval depths and both generator models.

4.6. Unified Factorial ANOVA Analysis

A 4-factor Type-II ANOVA evaluated the main and interaction effects of our architectural choices across all 60 experimental runs ($R^2_{\text{adj}} = 0.730$). The complete ANOVA results are detailed in Table 7.

Table 7
Unified 4-Factor ANOVA table for fact-verification accuracy.

Factor	Sum of Squares	df	F-Value	p-value
C(kg_model)	3080.01	4	27.85	< 0.001
C(retrieval)	127.16	2	2.30	0.1106
C(generation_model)	249.70	1	9.03	0.0041
C(context_mode)	624.17	1	22.58	< 0.001
Residual	1409.92	51	–	–

The factorial analysis establishes that Knowledge Graph construction model selection is the predominant factor governing system accuracy ($F = 27.85$, $p < 0.001$), followed by context integrity ($F = 22.58$, $p < 0.001$) and generator selection ($F = 9.03$, $p = 0.0041$). Retrieval depth did not achieve main-effect significance when accounting for model choice ($p = 0.1106$), indicating that graph quality outweighs retrieval depth in downstream verification performance.

5. Discussion

The superior performance of biomedical-specialized extraction frameworks, particularly GLiNER-Bio, highlights the critical necessity of domain alignment during knowledge graph construction. We hypothesize that GLiNER-Bio's pre-training on domain-specific corpora allowed it to extract more nuanced biomedical entities and relations, yielding denser, high-fidelity subgraphs that provided richer

context for downstream verification. However, architectural constraints imposed operational trade-offs during graph generation. Specifically, structured encoders like GLiNER-Bio are constrained to smaller chunk sizes ($\leq$ 300 tokens), whereas dense generative models (e.g., GPT-OSS-20B) accommodate larger windows (1024 tokens). While processing dense models at smaller chunk sizes would have incurred prohibitive computational costs ($\approx$ 14 days per graph variant), using larger chunks introduced a granularity trade-off, diluting localized relation density.

A central trade-off observed between structured and unstructured graph construction is the balance between schema expressiveness and extraction consistency. Open-ended LLMs generated a significantly more diverse array of entities and edge types. However, this high expressiveness introduced severe entity-linking inconsistencies during retrieval. When processing query-time entities, generative models like Llama3-OpenBio-8B frequently extracted surface forms that diverged from those generated during document ingestion. This representation drift inhibited Cypher-based subgraph matching, directly driving the high rate of initial *Unknown* abstentions across non-judge configurations. These alignment challenges were further compounded by deployment hurdles. Specifically, base models lacking standardized chat templates on HuggingFace[82] required extensive prompt engineering and few-shot formatting to ensure reliable output parsing when acting as primary generators.

Our evaluation also highlights key design considerations regarding retrieval architectures and generator selection. The deterministic Cypher retrieval strategies employed in this work prioritized strict experimental control over adaptive search. Implementing dynamic, LLM-guided graph traversal algorithms could mitigate entity-mismatch issues but have an increased computational cost. Furthermore, while computational constraints prevented evaluating additional domain-adapted open-weights architectures (e.g. Meditron[83]) or conducting exhaustive ablation studies across isolated error modes, our unified framework effectively isolates model robustness in the presence of noisy and contradictory context. Future work will explore instruction-tuned biomedical generators and adaptive graph traversal frameworks to further enhance KG-RAG resilience in high-stakes reasoning domains.

6. Conclusion

In this study, we evaluated the robustness of KG-RAG architectures under correct and corrupted context conditions across 60 unique pipeline configurations. Our findings reveal that KG construction quality had the strongest influence on downstream fact-verification performance, more than retrieval depth and generator choice. Additionally, we demonstrated that an LLM-as-a-Judge resolution layer eliminates nearly all generator abstentions, recovering actionable predictions from non-instruction-tuned models. Ultimately, ensuring high-fidelity graph construction remains the most critical factor in building resilient KG-RAG systems capable of reasoning through misinformation and context conflicts. For reproducibility, our results and prompts are provided at the following repository https://github.com/AESIRLab/graphRAG-reasoning.

Declaration on Generative AI

During the preparation of this work, the authors used Gemini 3.6 Flash in order to: paraphrase and reword, grammar and spelling check. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

Acknowledgments

This work used Bridges-2 at Pittsburgh Supercomputer Cluster through allocation CIS251294 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program[84], which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. This material is based upon work supported by the National Science Foundation under Grant No.# 2237945 and #2113903.

References

[1] T. J. Barnett, The zettabyte era officially begins (how much is that?), https://blogs.cisco.com/sp/the-zettabyte-era-officially-begins-how-much-is-that, 2016. Cisco Blogs, accessed 2025-12-11.

[2] Huawei Global Industry Vision (GIV), Data storage 2030: White paper, 2024. URL: https://www-file.huawei.com/-/media/corp2020/pdf/giv/2024/data_storage_whitepaper_2030_en.pdf, accessed: 2025-12-11.

[3] I. M. Putrama, P. Martinek, Heterogeneous data integration: Challenges and opportunities, Data in Brief 56 (2024) 110853.

[4] X. Wang, D. Bo, C. Shi, S. Fan, Y. Ye, P. S. Yu, A survey on heterogeneous graph embedding: methods, techniques, applications and sources, IEEE transactions on big data 9 (2022) 415–436.

[5] J.-M. Huan, X.-J. Wang, Y. Li, S.-J. Zhang, Y.-L. Hu, Y.-L. Li, The biomedical knowledge graph of symptom phenotype in coronary artery plaque: machine learning-based analysis of real-world clinical data, BioData Mining 17 (2024) 13. URL: https://doi.org/10.1186/s13040-024-00365-1. doi:10.1186/s13040-024-00365-1.

[6] A. A. K. Annadurai, S. Bhandary, S. G. Hegde, J. Chatterjee, Assembling a multiscale biomedical knowledge graph for explainable drug repurposing in rare diseases, Network Modeling Analysis in Health Informatics and Bioinformatics 14 (2025) 37. URL: https://doi.org/10.1007/s13721-025-00532-2. doi:10.1007/s13721-025-00532-2.

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021. URL: https://arxiv.org/abs/2005.11401. arXiv:2005.11401.

[8] B. Zhang, A. Bornet, R. Yang, N. Liu, D. Teodoro, Healthcontradict: Evaluating biomedical knowledge conflicts in language models, npj Digital Medicine 9 (2026) 152. URL: https://doi.org/10.1038/s41746-025-02336-0. doi:10.1038/s41746-025-02336-0.

[9] K. S. Candan, H. Liu, R. Suvarna, Resource description framework: metadata and its applications, SIGKDD Explor. Newsl. 3 (2001) 6–19. URL: https://doi.org/10.1145/507533.507536. doi:10.1145/507533.507536.

[10] P. Gao, F. Gao, J. Ni, Y. Wang, F. Wang, Q. Zhang, Medical knowledge graph question answering for drug-drug interaction prediction based on multi-hop machine reading comprehension, CAAI Transactions on Intelligence Technology 9 (2024) 1217–1228. URL: https://doi.org/10.1049/cit2.12332. doi:10.1049/cit2.12332.

[11] P. Wang, J. Liu, D. Hou, S. Zhou, A Cybersecurity Knowledge Graph Completion Method Based on Ensemble Learning and Adversarial Training, Applied Sciences 12 (2022). URL: https://www.mdpi.com/2076-3417/12/24/12947. doi:10.3390/app122412947.

[12] X. Wu, Y. She, X. Wang, H. Lu, Q. Gao, Domain-Specific Knowledge Graph for Quality Engineering of Continuous Casting: Joint Extraction-Based Construction and Adversarial Training Enhanced Alignment, Applied Sciences 15 (2025) 5674. doi:10.3390/app15105674.

[13] N. L. Le, M.-H. Abel, P. Gouspillou, A Personalized Recommender System Based-on Knowledge Graph Embeddings, in: The 3rd International Conference on Artificial Intelligence and Computer Vision (AICV2023), March 5–7, 2023, Springer Nature Switzerland, 2023, pp. 368–378. URL: http://dx.doi.org/10.1007/978-3-031-27762-7_35. doi:10.1007/978-3-031-27762-7_35.

[14] N. L. Le, M.-H. Abel, P. Gouspillou, A Constraint-based Recommender System via RDF Knowledge Graphs, in: 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), IEEE, 2023, pp. 849–854. URL: http://dx.doi.org/10.1109/CSCWD57460.2023.10152701. doi:10.1109/cscwd57460.2023.10152701.

[15] X. Zhao, S. Liu, S.-Y. Yang, C. Miao, MedRAG: Enhancing Retrieval-augmented Generation with Knowledge Graph-Elicited Reasoning for Healthcare Copilot, 2025. URL: https://arxiv.org/abs/2502.04413, _eprint: 2502.04413.

[16] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier, G. Bouchard, Complex Embeddings for Simple Link Prediction, in: M. F. Balcan, K. Q. Weinberger (Eds.), Proceedings of The 33rd International Conference on Machine Learning, volume 48 of *Proceedings of Machine Learning Research*, PMLR,

New York, New York, USA, 2016, pp. 2071–2080. URL: https://proceedings.mlr.press/v48/trouillon16.html.

[17] D. Q. Nguyen, T. D. Nguyen, D. Q. Nguyen, D. Q. Phung, A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network, in: North American Chapter of the Association for Computational Linguistics, 2017. URL: https://api.semanticscholar.org/CorpusID:3882054.

[18] Z. Sun, Z. Deng, J.-Y. Nie, J. Tang, RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space, ArXiv abs/1902.10197 (2018). URL: https://api.semanticscholar.org/CorpusID:67855617.

[19] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI'14, AAAI Press, Québec City, Québec, Canada, 2014, pp. 1112–1119.

[20] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13, Curran Associates Inc., Red Hook, NY, USA, 2013, pp. 2787–2795. Event-place: Lake Tahoe, Nevada.

[21] J. Li, K. Qu, J. Yan, L. Zhou, L. Cheng, TEBC-Net: An Effective Relation Extraction Approach for Simple Question Answering over Knowledge Graphs, in: H. Qiu, C. Zhang, Z. Fei, M. Qiu, S.-Y. Kung (Eds.), Knowledge Science, Engineering and Management, Springer International Publishing, Cham, 2021, pp. 154–165.

[22] X. Huang, J. Zhang, D. Li, P. Li, Knowledge Graph Embedding Based Question Answering, Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (2019). URL: https://api.semanticscholar.org/CorpusID:59528287.

[23] M. Samir, N. Fathy, W. Gad, Knowledge-based question answering using graph neural networks and contextual language representations., Scientific reports 16 (2026) 2640. doi:10.1038/s41598-025-33854-2.

[24] L. Cai, W. Y. Wang, KBGAN: Adversarial learning for knowledge graph embeddings, in: M. Walker, H. Ji, A. Stent (Eds.), Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1470–1480. URL: https://aclanthology.org/N18-1133/. doi:10.18653/v1/N18-1133.

[25] C. Zhang, T. Zang, T. Zhao, KGE-UNIT: toward the unification of molecular interactions prediction based on knowledge graph and multi-task learning on drug discovery, Briefings in Bioinformatics 25 (2024). URL: https://api.semanticscholar.org/CorpusID:267634271.

[26] Y. Li, Z.-H. You, Y. Yuan, C.-G. Mi, Y.-a. Huang, H. Yi, L.-X. Hou, Integrated Knowledge Graph and Drug Molecular Graph Fusion via Adversarial Networks for Drug-Drug Interaction Prediction, Journal of chemical information and modeling (2024). URL: https://api.semanticscholar.org/CorpusID:273702373.

[27] Y. Qiu, S. Wang, X. Yang, X. Qiu, C. Wu, Y. Cui, C. Yang, DrugProtKGE: Weakly Supervised Knowledge Graph Embedding for Highly-Effective Drug-Protein Interaction Representation, in: 2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2023, pp. 1386–1393. doi:10.1109/BIBM58861.2023.10385638.

[28] A. K. Mohan, T. John, G. Anoop, A Topological Deep Knowledge Graph Approach for Protein Interaction: Sheaf Neural Networks in Drug Discovery, in: 2025 2nd International Conference on Trends in Engineering Systems and Technologies (ICTEST), volume 1, 2025, pp. 1–6. doi:10.1109/ICTEST64710.2025.11042524.

[29] C. Han, Y. Li, X. Tang, DocPolicyKG: A Lightweight LLM-Based Framework for Knowledge Graph Construction from Chinese Policy Documents, in: Proceedings of the 34th ACM International Conference on Information and Knowledge Management, CIKM '25, Association for Computing Machinery, New York, NY, USA, 2025, pp. 4753–4757. URL: https://doi.org/10.1145/3746252.3760904. doi:10.1145/3746252.3760904.

[30] N. Mihindukulasooriya, S. Tiwari, C. F. Enguix, K. Lata, Text2KGBench: A Benchmark

for Ontology-Driven Knowledge Graph Generation from Text, in: The Semantic Web – ISWC 2023: 22nd International Semantic Web Conference, Athens, Greece, November 6–10, 2023, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2023, pp. 247–265. URL: https://doi.org/10.1007/978-3-031-47243-5_14. doi:10.1007/978-3-031-47243-5_14.

[31] Y. Cao, X. Ji, X. Lv, J. Li, Y. Wen, H. Zhang, Are Missing Links Predictable? An Inferential Benchmark for Knowledge Graph Completion, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 6855–6865. URL: https://aclanthology.org/2021.acl-long.534/. doi:10.18653/v1/2021.acl-long.534.

[32] H. L. Chieu, H. T. Ng, Named entity recognition with a maximum entropy approach, in: Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003, 2003, pp. 160–163.

[33] E. F. Tjong Kim Sang, Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition, in: COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002), 2002. URL: https://aclanthology.org/W02-2024/.

[34] A. Mansouri, L. S. Affendey, A. Mamat, Named entity recognition approaches, International Journal of Computer Science and Network Security 8 (2008) 339–344.

[35] H. Cho, H. Lee, Biomedical named entity recognition using deep neural networks with contextual information, BMC Bioinformatics 20 (2019) 735.

[36] Ö. Uzuner, B. R. South, S. Shen, S. L. DuVall, 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text, J Am Med Inform Assoc 18 (2011) 552–556.

[37] R. I. Doğan, R. Leaman, Z. Lu, NCBI disease corpus: a resource for disease name recognition and concept normalization, J Biomed Inform 47 (2014) 1–10.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, volume 30, 2017.

[39] S. Wang, X. Sun, X. Li, R. Ouyang, F. Wu, T. Zhang, J. Li, G. Wang, Gpt-ner: Named entity recognition via large language models, 2023. URL: https://arxiv.org/abs/2304.10428. arXiv:2304.10428.

[40] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, 2020. URL: https://arxiv.org/abs/2005.14165. arXiv:2005.14165.

[41] F. Bai, H. Hassanzadeh, A. Saeedi, M. Dredze, LLMs are better than you think: Label-guided in-context learning for named entity recognition, in: C. Christodoulopoulos, T. Chakraborty, C. Rose, V. Peng (Eds.), Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Suzhou, China, 2025, pp. 28372–28392. URL: https://aclanthology.org/2025.emnlp-main.1441/. doi:10.18653/v1/2025.emnlp-main.1441.

[42] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, B. Chang, X. Sun, L. Li, Z. Sui, A survey on in-context learning, 2024. URL: https://arxiv.org/abs/2301.00234. arXiv:2301.00234.

[43] M. Monajatipoor, J. Yang, J. Stremmel, M. Emami, F. Mohaghegh, M. Rouhsedaghat, K.-W. Chang, Llms in biomedicine: A study on clinical named entity recognition, 2024. URL: https://arxiv.org/abs/2404.07376. arXiv:2404.07376.

[44] Q. Zhang, M. Chen, L. Liu, A review on entity relation extraction, in: 2017 Second International Conference on Mechanical, Control and Computer Engineering (ICMCCE), 2017, pp. 178–183. doi:10.1109/ICMCCE.2017.14.

[45] Automatic content extraction 2008 evaluation plan (ace 08) assessment of detection and recognition of entities and relations within and across documents, 2008. URL: https://api.semanticscholar.org/CorpusID:15253237.

[46] G. Zhou, J. Su, J. Zhang, M. Zhang, Exploring various knowledge in relation extraction, in: Annual Meeting of the Association for Computational Linguistics, 2005. URL: https://api.semanticscholar.org/CorpusID:3160937.

[47] Y. Lin, S. Shen, Z. Liu, H. Luan, M. Sun, Neural relation extraction with selective attention over

instances, in: K. Erk, N. A. Smith (Eds.), Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 2124–2133. URL: https://aclanthology.org/P16-1200/. doi:10.18653/v1/P16-1200.

[48] Q. Zhang, Z. Chen, H. Pan, C. Caragea, L. J. Latecki, E. Dragut, Scier: An entity and relation extraction dataset for datasets, methods, and tasks in scientific documents, 2024. URL: https://arxiv.org/abs/2410.21155. arXiv:2410.21155.

[49] R. Luo, L. Sun, Y. Xia, T. Qin, S. Zhang, H. Poon, T.-Y. Liu, Biogpt: generative pre-trained transformer for biomedical text generation and mining, Briefings in Bioinformatics 23 (2022). URL: http://dx.doi.org/10.1093/bib/bbac409. doi:10.1093/bib/bbac409.

[50] Z. Wan, F. Cheng, Z. Mao, Q. Liu, H. Song, J. Li, S. Kurohashi, GPT-RE: In-context learning for relation extraction using large language models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 3534–3547. URL: https://aclanthology.org/2023.emnlp-main.214/. doi:10.18653/v1/2023.emnlp-main.214.

[51] U. Zaratiana, N. Tomeh, P. Holat, T. Charnois, Gliner: Generalist model for named entity recognition using bidirectional transformer, 2023. URL: https://arxiv.org/abs/2311.08526. arXiv:2311.08526.

[52] J. Boylan, C. Hokamp, D. G. Ghalandari, Glirel – generalist model for zero-shot relation extraction, 2025. URL: https://arxiv.org/abs/2501.03172. arXiv:2501.03172.

[53] U. Zaratiana, G. Pasternak, O. Boyd, G. Hurn-Maloney, A. Lewis, Gliner2: An efficient multi-task information extraction system with schema-driven interface, 2025. URL: https://arxiv.org/abs/2507.18546. arXiv:2507.18546.

[54] I. Stepanov, O. Lukashov, M. Shtopko, V. Kalyanarangan, Gliner-relex: A unified framework for joint named entity recognition and relation extraction, 2026. URL: https://arxiv.org/abs/2605.10108. arXiv:2605.10108.

[55] A. T. Kalai, O. Nachum, S. S. Vempala, E. Zhang, Why language models hallucinate, 2025. URL: https://arxiv.org/abs/2509.04664. arXiv:2509.04664.

[56] L. Y. Panchumarthi, S. Saleti, S. P. Gudari, A. Negi, P. R. Budime, H. Upadhya, RAG-BioQA: A Retrieval-Augmented Generation Framework for Long-Form Biomedical Question Answering, 2026. URL: https://arxiv.org/abs/2510.01612, _eprint: 2510.01612.

[57] M. R. Kabir, R. M. Sultan, F. Rahman, M. R. Amin, S. Momen, N. Mohammed, S. Rahman, LegalRAG: A Hybrid RAG System for Multilingual Legal Information Retrieval, 2025 International Joint Conference on Neural Networks (IJCNN) (2025) 1–8. URL: https://api.semanticscholar.org/CorpusID:278000677.

[58] T. Gan, Q. Sun, RAG-MCP: Mitigating Prompt Bloat in LLM Tool Selection via Retrieval-Augmented Generation, 2025. URL: https://arxiv.org/abs/2505.03275, _eprint: 2505.03275.

[59] R. Shan, A deep dive into vector stores: Classifying the backbone of retrieval-augmented generation, in: 2024 IEEE International Conference on Big Data (BigData), 2024, pp. 8831–8833. doi:10.1109/BigData62323.2024.10825992.

[60] R. Qu, R. Tu, F. Bao, Is semantic chunking worth the computational cost?, 2024. URL: https://arxiv.org/abs/2410.13070. arXiv:2410.13070.

[61] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, H. Jégou, The faiss library, 2025. URL: https://arxiv.org/abs/2401.08281. arXiv:2401.08281.

[62] S. Robertson, H. Zaragoza, The probabilistic relevance framework: Bm25 and beyond, Found. Trends Inf. Retr. 3 (2009) 333–389. URL: https://doi.org/10.1561/1500000019. doi:10.1561/1500000019.

[63] K. Santhanam, O. Khattab, J. Saad-Falcon, C. Potts, M. Zaharia, Colbertv2: Effective and efficient retrieval via lightweight late interaction, 2022. URL: https://arxiv.org/abs/2112.01488. arXiv:2112.01488.

[64] X. Zhu, Y. Xie, Y. Liu, Y. Li, W. Hu, Knowledge graph-guided retrieval augmented generation, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language

Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 8912–8924. URL: https://aclanthology.org/2025.naacl-long.449/. doi:10.18653/v1/2025.naacl-long.449.

[65] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, J. Larson, From local to global: A graph rag approach to query-focused summarization, 2025. URL: https://arxiv.org/abs/2404.16130. arXiv:2404.16130.

[66] X. Jiang, R. Zhang, Y. Xu, R. Qiu, Y. Fang, Z. Wang, J. Tang, H. Ding, X. Chu, J. Zhao, Y. Wang, HyKGE: A hypothesis knowledge graph enhanced RAG framework for accurate and reliable medical LLMs responses, 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. 11836–11856. URL: https://aclanthology.org/2025.acl-long.580/. doi:10.18653/v1/2025.acl-long.580.

[67] J. Wu, J. Zhu, Y. Qi, J. Chen, M. Xu, F. Menolascina, V. Grau, Medical graph rag: Towards safe medical large language model via graph retrieval-augmented generation, 2024. URL: https://arxiv.org/abs/2408.04187. arXiv:2408.04187.

[68] Y. Han, M. Lan, H. Kilicoglu, When evidence conflicts: Uncertainty and order effects in retrieval-augmented biomedical question answering, 2026. URL: https://arxiv.org/abs/2605.14115. arXiv:2605.14115.

[69] A. Lavie, A. Agarwal, Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments, in: Proceedings of the Second Workshop on Statistical Machine Translation, StatMT '07, Association for Computational Linguistics, USA, 2007, p. 228–231.

[70] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu, S. Wang, K. Zhang, Y. Wang, W. Gao, L. Ni, J. Guo, A survey on llm-as-a-judge, 2025. URL: https://arxiv.org/abs/2411.15594. arXiv:2411.15594.

[71] S. Es, J. James, L. Espinosa-Anke, S. Schockaert, Ragas: Automated evaluation of retrieval augmented generation, 2025. URL: https://arxiv.org/abs/2309.15217. arXiv:2309.15217.

[72] M. S. Ankit Pal, Openbiollms: Advancing open-source large language models for healthcare and life sciences, https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B, 2024.

[73] OpenAI, gpt-oss-120b & gpt-oss-20b model card, 2025. URL: https://arxiv.org/abs/2508.10925. arXiv:2508.10925.

[74] G. Team, Gemma (2024). URL: https://www.kaggle.com/m/3301. doi:10.34740/KAGGLE/M/3301.

[75] Issacus, semchunk: Fast and accurate-length text chunking, https://github.com, 2024. Version 4.1.1.

[76] Knowledgator Engineering, biomed_ner dataset, https://huggingface.co/datasets/knowledgator/biomed_NER, 2024. Accessed: 2026-06-11.

[77] Y. Zhang, M. Li, D. Long, X. Zhang, H. Lin, B. Yang, P. Xie, A. Yang, D. Liu, J. Lin, F. Huang, J. Zhou, Qwen3 embedding: Advancing text embedding and reranking through foundation models, arXiv preprint arXiv:2506.05176 (2025).

[78] Q. Team, Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

[79] D. C. Montgomery, Design and analysis of experiments, John wiley & sons, 2017.

[80] J. W. Tukey, Comparing individual means in the analysis of variance, Biometrics (1949) 99–114.

[81] W. J. Conover, Practical nonparametric statistics, john wiley & sons, 1999.

[82] Hugging Face, The Hugging Face platform: Open source machine learning community and hub, https://huggingface.co/, 2026. Accessed: July 2026.

[83] X. Theimer-Lienhard, M. El-Amin, F. Elhassan, S. Vaidya, V. Cartier-Negadi, D. Sasu, L. Klein, M.-A. Hartley, Fully open meditron: An auditable pipeline for clinical llms, 2026. URL: https://arxiv.org/abs/2605.16215. arXiv:2605.16215.

[84] T. J. Boerner, S. Deems, T. R. Furlani, S. L. Knuth, J. Towns, Access: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support, in: Practice and experience in advanced research computing 2023: Computing for the common good, 2023, pp. 173–176.