**Project: AccurateRAG: Leveraging Knowledge Graphs with LLMs**

**Team: Ali Gardezi, Ganesh Devaraj, Kishore Kolli**

**Code Repo: https://github.com/aligardezi/ai_healthcare_high_risk/tree/master**

**Presentation Deck: AccurateRAG**

**Presentation Video: https://utexas.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=a94b1b53-d1f5-4ba1-ac14-b24200346258**

# AccurateRAG: Leveraging Knowledge Graphs with LLMs

Ali Gardezi
College of Natural Sciences
University of Texas at Austin
Austin, TX USA
gardezi@utexas.edu

Ganesh Devaraj
College of Natural Sciences
University of Texas at Austin
Austin, TX USA
ganesh.devaraj.msai@utexas.edu

Kishore Kolli
College of Natural Sciences
University of Texas at Austin
Austin, TX USA
kkolli@utexas.edu

## ABSTRACT

This project tackles the challenges of improving prediction accuracy in medical question answering (QA) by leveraging a knowledge graph constructed from entities extracted from medical texts. By integrating knowledge graphs with large language models (LLMs), the project enhances the retrieval and contextualization of domain-specific information across extensive datasets. This approach aims to improve data transparency, reduce hallucination rates, and generate accurate, context-aware insights for medical QA. The paper outlines the project's methodology, reviews related work, and explores its potential applications in advancing medical information systems.

## KEYWORDS

RAG, LLM, MedRAG, MIRAGE, Elastic Search, UMLS, Neo4J

## 1 Introduction

The increasing volume of domain-specific textual data poses significant challenges in extracting, structuring, and leveraging meaningful insights. Traditional keyword-based and vector search systems struggle to address complex, context-dependent queries, often leading to incomplete or inaccurate results. In critical fields like medicine, where decisions hinge on precise and reliable information, these limitations can have serious consequences.

To address these challenges, this project leverages a knowledge graph constructed using entities extracted from medical texts via the Core NLP[4] model. The knowledge graph provides a structured and semantically rich representation of textual data, enabling a deeper understanding of relationships between entities. By integrating this graph with large language models (LLMs), the project creates a hybrid framework for improved information retrieval and contextual understanding.

The model offers several key benefits:

**Improved Accuracy and Transparency:** By grounding LLM responses in a structured knowledge graph, the system ensures traceable and verifiable insights, reducing hallucination rates.

**Resilience to Adversarial Inputs:** The structured data representation minimizes vulnerabilities to injection attacks and enhances the robustness of generated responses.

**Enhanced Query Understanding:** The combination of contextual processing and graph-based methodologies addresses the limitations of traditional search systems in handling complex medical queries.

This approach not only improves the precision of medical question answering (QA) but also establishes a transparent and scalable foundation for decision support systems. The project's methodology and findings are detailed in the subsequent sections, highlighting its potential to revolutionize information retrieval in critical domains such as medicine and law.

## 2 Related Work

This project builds on advancements in knowledge graph integration with large language models (LLMs), retrieval-augmented generation (RAG), and medical question answering (QA). Several previous works have influenced its methodology, datasets, and evaluation metrics.

### 2.1 Graph-Based Retrieval-Augmented Generation (RAG):

The integration of knowledge graphs with LLMs for retrieval-augmented generation has been explored in projects like Microsoft GraphRAG[3]. This work demonstrates how structured data representations can enhance contextual understanding and improve the accuracy of generated responses. Inspired by GraphRAG, this project adopts a similar approach to build knowledge graphs using entities extracted from domain-specific texts, enabling structured representation and advanced contextual query processing.

### 2.2 Domain-Specific Datasets for Medical QA:

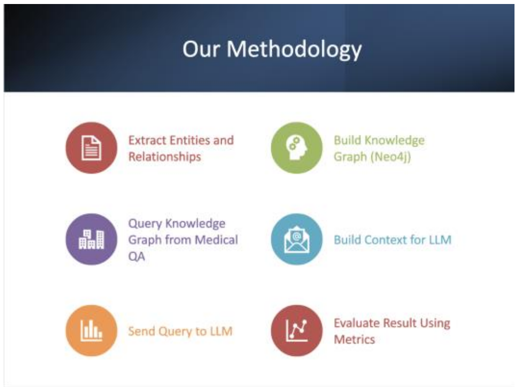
The MedRAG[2] project has highlighted the challenges and opportunities of using diverse medical datasets for retrieval-augmented generation. Building on their insights, this project focuses on constructing a comprehensive knowledge graph from medical texts processed through Core NLP models. By leveraging domain-specific entities and relationships, the system addresses the unique challenges of medical QA, such as handling ambiguous queries and ensuring response reliability.

### 2.3 Datasets and Evaluation Metrics

This project draws inspiration from the datasets and evaluation settings of MIRAGE to design a robust framework for assessing the performance of medical question answering (QA) systems. While not adopting MIRAGE directly, the project integrates key principles to ensure realistic and meaningful evaluation.

## 3. Methodology

Our methodology begins with **extracting entities** and **relationships** from domain-specific medical text using Stanford CoreNLP model, which enables a structured representation of the data. These extracted entities and relationships are then used to construct a knowledge graph in Neo4j[1], facilitating efficient storage and retrieval of interconnected medical information. For a given medical question, relevant entities are identified and used to query the Neo4j graph to retrieve associated information, which forms the basis for constructing a contextual document. This document is provided to a large language model (LLM), enriching its ability to generate informed and contextually accurate answers. Finally, the generated responses are evaluated using metrics inspired by MIRAGE, focusing on aspects such as accuracy, resilience to adversarial inputs, and alignment with real-world medical QA scenarios.



Our Methodology

Extract Entities and Relationships

Build Knowledge Graph (Neo4j)

Query Knowledge Graph from Medical QA

Build Context for LLM

Send Query to LLM

Evaluate Result Using Metrics

**Figure 1: Our Methodology**

## 4. Results

This section presents a comparative analysis of two different approaches: our implementation using the Stanford CoreNLP model and Microsoft's GraphRAG. **The comparison focuses on key metrics such as execution time, cost, the number of entities and relationships extracted, and accuracy in answering medical QA questions.** To assess the effectiveness of each method, a sample anatomy question was posed to both systems, followed by an extensive evaluation involving 135 medical QA questions. These questions were processed using Microsoft's GraphRAG, which was set up and executed locally, providing insights into the system's capabilities and performance against our graph-based retrieval methodology.

Here is a tabular representation of the metrics from two approaches, GraphRAG and Stanford CoreNLP + Neo4J:

| Metric | GraphRAG | Stanford CoreNLP+Neo4J |
|---|---|---|
| Execution Time (Anatomy Book) | Approx. 2 hr. 30 min | Approx. 8 hrs. |
| Cost to Run | Approx.$55 | $0 |
| Number of Entities | 35,652 | 78,075 |
| Number of Relationships | 85,968 | 195,756 |
| Number of Communities | 12,950 | N/A |

Table 1: Microsoft GraphRAG vs AccurateRAG Metrics

### 4.1 Answer to Anatomy Question

The following are the responses from two different approaches—GraphRAG and AccurateRAG—for the given Anatomy question:

**Question**: lesion causing compression of the facial nerve at the stylomastoid foramen will cause ipsilateral:
A. Paralysis of the facial muscles.
B. Paralysis of the facial muscles and loss of taste.
C. Paralysis of the facial muscles, loss of taste, and lacrimation.
D. Paralysis of the facial muscles, loss of taste, lacrimation, and decreased salivation.

Both GraphRAG and our AccurateRAG provided the correct **answer**

A. Paralysis of the facial muscles.

### 4.2 Results for 135 Medical QA Questions

In the evaluation involving 135 medical QA questions, GraphRAG successfully provided the correct answer for 114 out of 135 questions. The remaining discrepancies were attributed to a few limitations:
(1) GraphRAG generated an answer but did not provide the corresponding answer choice (A, B, C, D);
(2) it was unable to confidently select one correct answer, or
(3) it lacked sufficient information to provide an answer.

Our methodology got only 53 out of 135 questions correctly. The comparison table below summarizes the performance of **GraphRAG** against our implementation using the **Stanford CoreNLP model**.

| Metric | GraphRAG | Stanford CoreNLP + Neo4J |
|---|---|---|
| Total Questions Asked | 135 | 135 |
| Correct Answers | 114 | 53 |
| No Answer Choice Provided | 9 | - |
| Could Not Pick One Correct Answer | 6 | - |
| Not Enough Information | 6 | 82 |

Table 2: Microsoft GraphRAG vs AccurateRAG Results

For further details, including code and complete results, please visit the repository: GitHub Repository for AI Healthcare High Risk Project

### 4.3 Future Directions

Our Model did not have enough information to answer 82 questions correctly. We believe adding textual context and community relationships would improve the results when compared to the Microsoft GraphRAG model.

### 4.4 UMLS Data

We attempted to use UMLS[5] data for mapping entities with relationships to enrich the graph-based representation. However, due to the substantial size of the UMLS dataset, it became challenging to complete this task within the project timeline. Moving forward, we plan to improve the efficiency of data retrieval by loading UMLS data into an RDBMS, which will allow for optimized querying and storage.

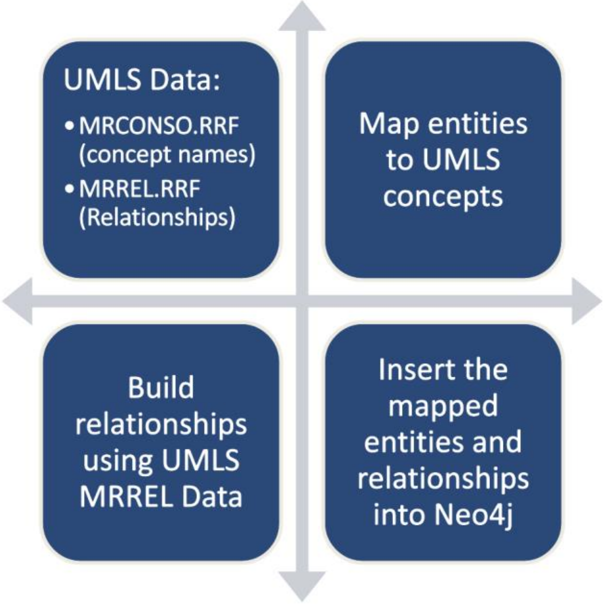

UMLS Data:
• MRCONSO.RRF (concept names)
• MRREL.RRF (Relationships)
Map entities to UMLS concepts
Build relationships using UMLS MRREL Data
Insert the mapped entities and relationships into Neo4j

**Figure** 2: UMLS Workflow

### 4.5 Cosine Similarity between embeddings

Another method which showed promise involves generating relationships between entities using embeddings and cosine similarity through the Neo4j Graph Data Science (GDS) library. We faced similar performance challenges due to the dataset's scale.

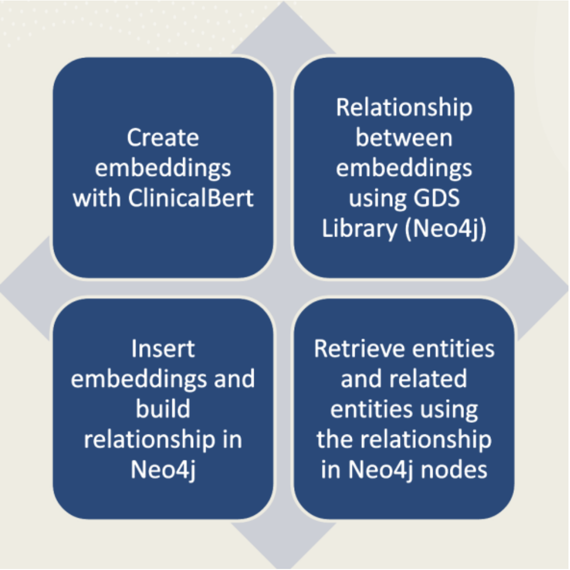

Create embeddings with ClinicalBert
Relationship between embeddings using GDS Library (Neo4j)
Insert embeddings and build relationship in Neo4j
Retrieve entities and related entities using the relationship in Neo4j nodes

**Figure 3: Workflow using embedding relationships to build context**

### 4.6 Textual Context with ElasticSearch

We attempted to build textual context by loading medical text data into ElasticSearch and using entities derived from medical questions to identify relevant documents. This approach demonstrated potential for improving the context-building process, but scalability remains an area of focus.

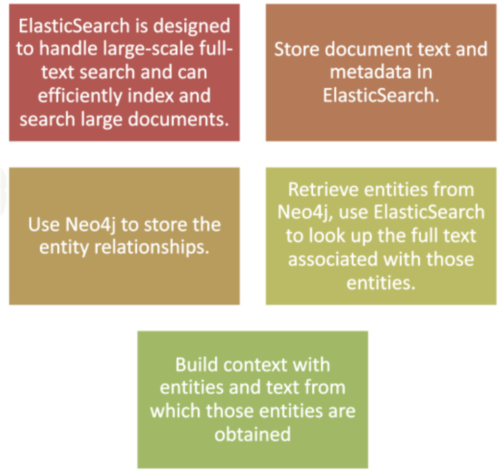
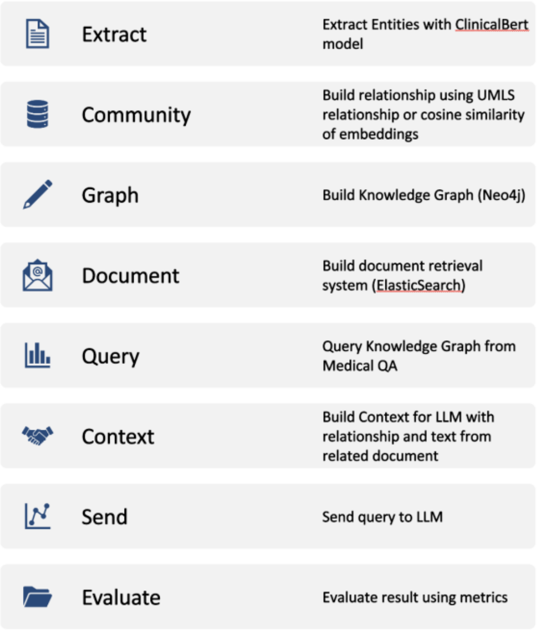

ElasticSearch is designed to handle large-scale full-text search and can efficiently index and search large documents.
Store document text and metadata in ElasticSearch.
Use Neo4j to store the entity relationships.
Retrieve entities from Neo4j, use ElasticSearch to look up the full text associated with those entities.
Build context with entities and text from which those entities are obtained

**Figure 4: Workflow to build textual context**



| | |
|---|---|
| Extract | Extract Entities with ClinicalBert model |
| Community | Build relationship using UMLS relationship or cosine similarity of embeddings |
| Graph | Build Knowledge Graph (Neo4j) |
| Document | Build document retrieval system (ElasticSearch) |
| Query | Query Knowledge Graph from Medical QA |
| Context | Build Context for LLM with relationship and text from related document |
| Send | Send query to LLM |
| Evaluate | Evaluate result using metrics |

**Figure 5: Target architecture to enrich context**

## 5 Conclusion

Leveraging Knowledge Graphs with LLMs demonstrates the feasibility and impact of combining large language models with graph-based knowledge representations. The project showcases significant potential in domains such as healthcare and legal research by providing accurate and transparent responses. We believe that further enhancing the system by extracting entities using Clinical BERT, building relationships with UMLS data or using embedding similarity, and adding context through documents retrieved via Elasticsearch will lead to even better results. These enhancements are expected to improve the depth and reliability of the information retrieval process, ultimately advancing the accuracy of medical question answering systems.

## REFERENCES

[1] Neo4j Documentation. "Graph Data Science Library." Neo4j
[2] Guangzhi Xiong, Qiao Jin, Zhiyong Lu, Aidong Zhang . Benchmarking Retrieval-Augmented Generation for Medicine (MedRAG & MIRAGE)
[3] Project GraphRAG (Microsoft)
[4] Stanford CoreNLP
[5] UMLS