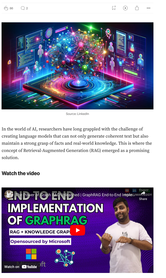
End-to-End Implementation of GraphRAG(Knowledge Graphs + Retrieval Augmented Generation): Unlocking LLM Discovery on Narrative Private Data

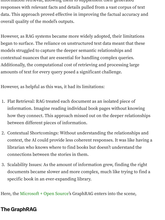
Vinod Kumar S.R. · Follow
9 min read · Jul 24, 2024



Source Lexica

In the world of AI, researchers have long grappled with the challenge of creating language models that can not only generate coherent text but also maintain a strong grasp of facts and real-world knowledge. This is where the concept of Retrieval Augmented Generation (RAG) emerged as a promising solution.

Watch the video



Traditional RAG models combined the power of large language models with information retrieval, allowing them to supplement their generated responses with relevant facts and details pulled from a vast corpus of text data. This approach proved effective in improving the factual accuracy and overall quality of the model's outputs.

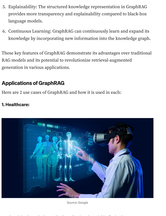
However, as RAG systems became more widely adopted, their limitations began to surface. The reliance on unstructured text data meant that these models struggled to capture the deeper semantic relationships and contextual nuances that are essential for handling complex queries. Additionally, the computational cost of retrieving and processing large amounts of text for every query proved a significant challenge.

However, as helpful as this was, it had its limitations:

1. Pat Retrieval: RAG treated each document as an isolated piece of information. Imagine reading individual book pages without knowing how they connect. This approach missed out on the deeper relationships between different pieces of information.
2. Contextual Shortcomings: Without understanding the relationships and context, the AI could provide less coherent responses. It was like having a librarian who knows where to find books but doesn't understand the connections between the stories in them.
3. Scalability Issues: As the amount of information grew, finding the right documents became slower and more complex, much like trying to find a specific book in an ever-expanding library.

Here, the Microsoft + Open Source's GraphRAG enters the scene,

The GraphRAG



Source Google

GraphRAG is a revolutionary advancement in the field of retrieval-augmented generation. Instead of working with unstructured text, GraphRAG leverages the power of knowledge graphs — structured representations of entities, their attributes, and the relationships between them.

By representing information in this structured format, GraphRAG can overcome the limitations of traditional RAG approaches. The knowledge graph allows for more precise and comprehensive retrieval of relevant information, enabling the model to generate responses that are not only factually accurate but also contextually relevant and cover all the necessary aspects of a query.

GraphRAG has been shown to significantly improve the accuracy and relevance of generated text, making it an ideal solution for applications where accurate, sensible real-time answers are critical. It has also been praised for its ability to handle complex queries by following chains of relationships within the graph, providing a richer understanding of the information needed to answer questions.

Key Features of GraphRAG

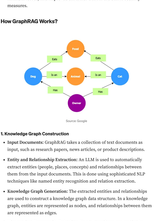
1. Structured Knowledge Representation: GraphRAG uses knowledge graphs to represent information, capturing entities, relationships, and hierarchical structure.
2. Contextual Retrieval: The knowledge graph enables GraphRAG to understand semantic context and relationships, allowing for more comprehensive retrieval.
3. Efficient Processing: Pre-processing data into a knowledge graph reduces computational cost and enables faster retrieval compared to traditional RAG approaches.
4. Multi-Faceted Query Handling: GraphRAG can handle complex, multi-faceted queries by synthesizing relevant information from multiple parts of the knowledge graph.
5. Explainability: The structured knowledge representation in GraphRAG provides more transparency and explainability compared to black-box language models.
6. Continuous Learning: GraphRAG can continuously learn and expand its knowledge by incorporating new information into the knowledge graph.

These key features of GraphRAG demonstrate its advantages over traditional RAG models and its potential to revolutionize retrieval-augmented generation in various applications.

Applications of GraphRAG

Here are 2 use cases of GraphRAG and how it is used in each:

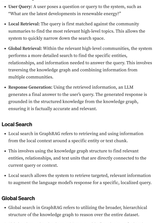
1. Healthcare:



Source Google

- GraphRAG can help medical professionals quickly find relevant information from large volumes of medical literature to answer complex questions about patient symptoms, treatments, and outcomes.
- By representing medical knowledge as a structured knowledge graph, GraphRAG enables multi-hop reasoning to connect disparate pieces of information and provide comprehensive answers.
- This can lead to faster and more accurate diagnoses, better-informed treatment decisions, and improved patient outcomes.

2. Banking:



Source Google

- GraphRAG can help banks detect and prevent fraud by representing customer transactions, account information, and behavioral patterns in a knowledge graph.
- By analyzing the relationships and anomalies in the graph, GraphRAG can identify suspicious activities and alert the bank's fraud detection system.
- This can lead to faster and more accurate fraud detection, reduced financial losses, and improved customer trust in the bank's security measures.

How GraphRAG Works?

Source Google

1. Knowledge Graph Construction

- **Input Documents:** GraphRAG takes a collection of text documents as input, such as research papers, news articles, or product descriptions.
- **Entity and Relationship Extraction:** An LLM is used to automatically extract entities (people, places, concepts) and relationships between them from the input documents. This is done using sophisticated NLP techniques like named entity recognition and relation extraction.
- **Knowledge Graph Generation:** The extracted entities and relationships are used to construct a knowledge graph data structure. In a knowledge graph, entities are represented as nodes, and relationships between them are represented as edges.
- **Hierarchical Community Detection:** Graph algorithms are applied to detect communities of densely connected nodes in the knowledge graph. These communities represent topics or themes that span multiple documents. The communities are organized hierarchically, with high-level communities containing lower-level sub communities.

2. Knowledge Graph Summarization

- **Community Summarization:** For each detected community in the knowledge graph, an LLM is used to generate a natural language summary. These summaries describe the key entities, relationships, and themes within each community.
- **Hierarchical Summaries:** The hierarchical structure of the communities is preserved in the summaries. High-level summaries describe broad themes, while lower-level summaries provide more granular details.

3. Retrieval-Augmented Generation

- **User Query:** A user poses a question or query to the system, such as "What are the latest developments in renewable energy?"
- **Local Retrieval:** The query is first matched against the community summaries to find the most relevant high-level topics. This allows the system to quickly narrow down the search space.
- **Global Retrieval:** Within the relevant high-level communities, the system performs a more detailed search to find the specific entities, relationships, and information needed to answer the query. This involves traversing the knowledge graph and combining information from multiple communities.
- **Response Generation:** Using the retrieved information, an LLM generates a final answer to the user's query. The generated response is grounded in the structured knowledge from the knowledge graph, ensuring it is factually accurate and relevant.

Local Search

- Local search in GraphRAG refers to retrieving and using information from the local context around a specific entity or text chunk.
- This involves using the knowledge graph structure to find relevant entities, relationships, and text units that are directly connected to the current query or context.
- Local search allows the system to retrieve targeted, relevant information to augment the language model's response for a specific, localized query.

Global Search

- Global search in GraphRAG refers to utilizing the broader, hierarchical structure of the knowledge graph to reason over the entire dataset.
- This involves using the detected "communities" or clusters of related entities and text units in the graph to generate summaries and overviews of the key themes and topics in the full dataset.
- Global search enables the system to answer queries that require aggregating information across multiple documents or entities, such as "What are the top 5 themes in the data?"
- By leveraging the pre-computed community summaries, global search can provide comprehensive responses without needing to search the entire dataset from scratch for each query.

How GraphRAG Represents the Information?

Let's understand how GraphRAG will store the information. Let's say I have this below sentence:

"Type 2 diabetes is a chronic condition characterized by high blood sugar levels, which can lead to complications such as nerve damage, kidney disease, and cardiovascular problems."

Here's how GraphRAG would represent this information in a knowledge graph:

```
# Entities
Type 2 Diabetes (CONDITION)
High Blood Sugar Levels (CONDITION)
Nerve Damage (CONDITION)
Kidney Disease (CONDITION)
Cardiovascular Problems (CONDITION)

# Relationships
Type 2 Diabetes -> has_symptom -> High Blood Sugar Levels
Type 2 Diabetes -> can_lead_to -> Nerve Damage
Type 2 Diabetes -> can_lead_to -> Kidney Disease
Type 2 Diabetes -> can_lead_to -> Cardiovascular Problems
```

In this knowledge graph representation:

- The entities (nodes) represent the key concepts mentioned in the sentence, such as the medical condition, symptoms, and complications.
- The relationships (edges) capture the connections between these entities, such as the fact that type 2 diabetes has the symptoms of high blood sugar levels and that it can lead to various complications.

This structured representation allows GraphRAG to understand the semantic relationships and context within the sentence, rather than just treating it as a bag of words.

When a user asks a question related to type 2 diabetes, such as *"What are the complications of type 2 diabetes?"*, GraphRAG can quickly traverse the knowledge graph, starting from the *"Type 2 Diabetes"* entity and following the "can_lead_to" relationships to retrieve the relevant information (*nerve damage, kidney disease, and cardiovascular problems*).

Advantages of GraphRAG

1. Structured Knowledge Representation: GraphRAG uses knowledge graphs to capture entities, relationships, and hierarchical structure, enabling more comprehensive and contextual retrieval of information.
2. Contextual Retrieval: The knowledge graph allows GraphRAG to understand the semantic context and relationships between different concepts, leading to more relevant and accurate responses.
3. Efficient Processing: By pre-processing data into a knowledge graph, GraphRAG can reduce the computational cost and enable faster retrieval compared to traditional RAG approaches.
4. Multi-Faceted Query Handling: GraphRAG can handle complex, multi-faceted queries by synthesizing relevant information from multiple parts of the knowledge graph.
5. Explainability: The structured knowledge representation in GraphRAG provides more transparency and explainability compared to black-box language models, which can foster trust and facilitate further discussions.

Potential Disadvantages of GraphRAG

1. Knowledge Graph Construction Complexity: Building a comprehensive and accurate knowledge graph can be a complex and time-consuming process, requiring sophisticated entity extraction and relationship modeling techniques.
2. Dependency on Underlying Data: The quality and coverage of the knowledge graph are heavily dependent on the input data sources. If the underlying data is incomplete or biased, it can limit the effectiveness of GraphRAG.
3. Scalability Challenges: As the knowledge graph grows in size and complexity, the computational resources required for efficient retrieval and reasoning may become a challenge, especially for real-time applications.

End-to-End Implementation

1. Knowledge Graph Construction: Use a combination of NLP techniques and LLMs to extract entities and relationships from the input documents. Store the extracted information in a graph database like Neo4j or a graph data structure in memory.
2. Hierarchical Community Detection: Apply graph algorithms like Leiden or Louvain to detect communities of densely connected nodes in the knowledge graph. Organize the communities hierarchically, with high-level communities containing lower-level sub communities.
3. Community Summarization: Generate natural language summaries for each community using an LLM. Provide the community's entities, relationships, and text units as context to the LLM.
4. Query Processing: Implement the local and global search mechanisms as described earlier. Use embeddings and similarity search to find relevant entities and communities for local search. Traverse the hierarchical community structure for global search.
5. Response Generation: Use the retrieved information and LLM to generate the final answer to the user's query. Ensure the response is grounded in the structured knowledge from the graph.

GitHub - Taksha-ai/End-to-End-Graphrag-implementation
Contribute to Taksha-ai/End-to-End-Graphrag-implementation development by creating an account on GitHub.
github.com

Here is the GitHub link, you can clone and run it locally. Follow the instructions mentioned in the Readme.md file.

Conclusion

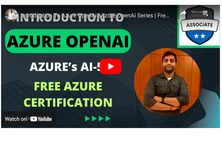
- In conclusion, GraphRAG represents a significant advancement in the field of retrieval-augmented generation. By leveraging the power of knowledge graphs, GraphRAG overcomes the limitations of traditional RAG models, providing more accurate, relevant, and comprehensive responses to complex queries.
- The structured knowledge representation in GraphRAG enables the system to understand the semantic context and relationships between different pieces of information, allowing it to handle complex, multi-faceted queries with ease.
- Additionally, the efficient processing capabilities of GraphRAG make it a practical solution for real-world applications, where speed and accuracy are crucial.
- GraphRAG has been shown to be effective in various domains, including healthcare and banking, where it can provide valuable insights and support decision-making processes.
- Its ability to continuously learn and expand its knowledge makes it a versatile tool that can adapt to new information and evolving domains.

. . .

If you have any questions, mention them in the comments or send a mail to below mentioned EmailD. Check out our YouTube channel for more Data Science related content.

. . .

Azure OpenAI Course and Assignment with Solutions

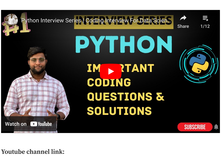


Machine Learning Playlist:

Deep Learning RoadMap 2024

Complete MongoDB Playlist

Python Interview Playlist



Youtube channel link:

SAI KUMAR REDDY
Hello all, I am going to upload videos related to Python, Data Science, Machine Learning, and other interesting AI...
www.youtube.com

My LinkedIn Account:

Vinod Kumar G R

And if you have any queries reach me out :

Email

Microsoft Graphrag | Graphrag | Retrieval Augmented Gen | Knowledge Graph | Large Language Models

98 2

Written by Vinod Kumar G R
93 Followers · 1 Following
YouTube Link -> https://youtube.com/@SAIKUMARREDDY76 Data scientist, I write articles related to Data Science.

Follow

More from Vinod Kumar G R

Vinod Kumar G R
12] Handling Missing Values in Machine Learning: Different...
You know what? when we get data, it's not always squeaky clean. We can't just pass it o...
Mar 31 · 55

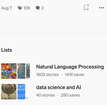
Vinod Kumar G R
07] Standardization and Normalization Techniques in...
Data is rarely perfect, and it often comes in various shapes and forms, with values that...
Jan 6 · 10 · 2

Vinod Kumar G R
09] Data Transformations in ML: Different transformations in...
As I mentioned earlier we cannot feed the data as it is, we need to convert or transform...
Jan 20 · 54

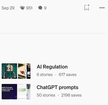
Vinod Kumar G R
Exploring Med-Gemini: A Breakthrough in Medical Imaging...
The rise of big language models (like GPT-4) and big multimodal models (like Flamingo) i...
May 13 · 2

See all from Vinod Kumar G R

Recommended from Medium



Zilliz
GraphRAG Explained: Enhancing RAG with Knowledge Graphs
Introduction to RAG and Its Challenges
Aug 7 · 120 · 2



In Towards AI by Barhoumi Mosbah
Anthropic's New RAG Approach
If you want a more detailed article about RAG from scratch or if you are new to RAG, feel...
Sep 29 · 950 · 9

Lists

Natural Language Processing
1902 stories · 1416 saves
AI Regulation
6 stories · 677 saves
data science and AI
40 stories · 280 saves
ChatGPT prompts
50 stories · 2168 saves

Zilliz
Graph Retrieval-Augmented Generation (Graph RAG) with...
In the rapidly evolving field of artificial intelligence, Retrieval-Augmented...
Sep 28 · 51

In Towards Data Science by Tomaz Bratanic
Building Knowledge Graphs with LLM Graph Transformer
A deep dive into LangChain's implementation of graph construction with LLMs
6d ago · 660 · 6

In WhyHow.AI by Chia Jeng Yang
Understanding the Knowledge Graph & RAG Opportunity
Case Studies, how Public companies are using KGs today, and where to apply KGs in...
May 27 · 355 · 6

In AI Advances by Anand Subramanian
Creating a Knowledge Graph for ICD Codes using LLMs
Representing ICD Codes as a Knowledge Graph
Nov 4 · 219 · 6

See more recommendations