

# Applying GraphRAG for improved LLM results



# Table of contents

Abbreviations	3
Introduction	4
Business challenges	5
Problem statement	5
Solution	6
GraphRAG lifecycle	7
GraphRAG workflow	7
Examples of GraphRAG improvements over vector-only RAG	8
Benefits	9
Conclusion	10
References	10
Author information	11

# Abbreviations

Short Form	Abbreviation
RAG	Retrieval Augmented Generation
GenAI	Generative AI
LLM	Large Language Model
GPU	Graphics Processing Unit

# Introduction

## Knowledge graphs in the Gartner Hype Cycle

The [Gartner 2024 Hype Cycle](#) for AI has mapped out the maturity and adoption phases of AI technologies. The report that highlights emerging trends, potential applications and the expected timeline for mainstream adoption of AI innovations has given a pivotal position to knowledge graphs in it.

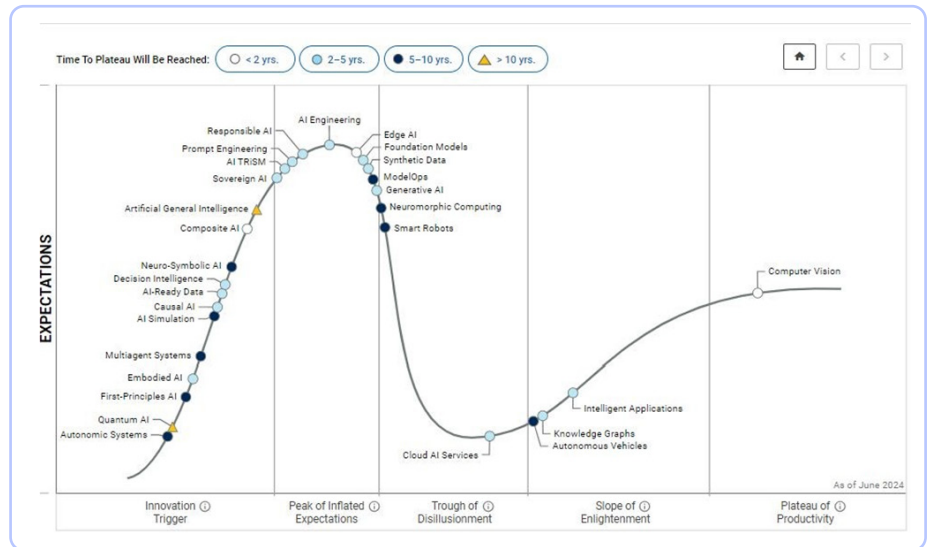


Figure 1 Gartner 2024 Hype Cycle for AI

Most AI technologies are expected to go through the “Trough of Disillusionment,” a phase where initial hype and inflated expectations give way to more measured and realistic assessments. For GenAI, this shift reflects an acknowledgment of the limitations and challenges around applications at scale, including ethical and privacy implications. But one standout in the Gartner 2024 Hype Cycle is the pivotal position of knowledge graphs. Positioned on the “Slope of Enlightenment,” knowledge graphs are increasingly recognized for their benefits to enterprises, leading to an increase in pilot projects. They are recognized as critical enablers for effectively applying GenAI in enterprise environments [4]

Knowledge graphs, integrated with GenAI, hold massive potential for driving business value. They provide efficient data management and enhanced decision-making capabilities, ensuring organizations are well-equipped to navigate the complexities of modern data landscapes [4]

## Introducing GraphRAG

Building upon the foundational benefits of knowledge graphs, GraphRAG emerges as a powerful technique that combines the strengths of knowledge graphs with vector-based Retrieval Augmented Generation (RAG). GraphRAG enhances the performance of Large Language Models (LLMs) by integrating knowledge graphs into the retrieval process, thereby addressing the limitations of vector-only RAG methods.

GraphRAG leverages the structured relationships and semantic richness of knowledge graphs to improve the accuracy and contextual relevance

of information retrieval. This approach not only enhances the retrieval process but also ensures that the generated answers are grounded in the underlying data, reducing the likelihood of model hallucination and improving overall reliability.

In this paper, we will look in detail at how the GraphRAG helps with improving the performance of LLMs.

## Business challenges

GenAI adoption is increasing in the industry at rapid rate for its associated benefits like productivity improvement, quality improvement, helping up-skill resources, etc. Today, almost every organization irrespective of the domain/field are aggressively training their resources on GenAI technologies to stay relevant in the evolving industry scenario and customer requirements. Although GenAI holds great potential, certain challenges must be addressed to harness its full benefits effectively. Some of these are given below:

- Need for accurate retrieval, efficient querying, scalable AI performance
- Building the solutions that are cheaper and scalable
- Avoid model hallucination and keep the results grounded

## Problem statement

LLMs are getting more and more acceptance in the industry across domains for their proven capabilities of aiding the users and developers in resolving real life problems and improving the productivity of human. Even though these LLMs are trained on gigantic amounts of data, they are generic in nature as their capability is limited to the data they are trained on. Users can apply the power of information retrieval of LLMs on their specific data through techniques like model fine-tuning, vector-based RAG. Though these are quite effective but still they have some limitations. Though these techniques increase the probability of a correct answer but do not ensure the certainty of a correct answer. Further, fine-tuning of LLMs is an expensive process and requires large setup with the need for number of GPUs directly proportional to the model parameter size.

A research blog published by Microsoft titled "GraphRAG: Unlocking LLM discovery on narrative private data" <sup>[6]</sup> in Feb 2024 has highlighted the following shortcoming of vector-only RAG:

- Baseline RAG struggles to connect the dots. This happens when answering a question requires traversing disparate pieces of information through their shared attributes in order to provide new synthesized insights.
- Baseline RAG performs poorly when being asked to holistically understand summarized semantic concepts over large data collections or even singular large documents.

# Solution

There are two types of knowledge representation prevalent in the technology world – vectors and graphs.

Vectors represent any given text in form of an array of numbers and are found to be very effective in capturing the essence of the text.

Knowledge graphs are symbolic representations of the world around the domain on which one deals with.

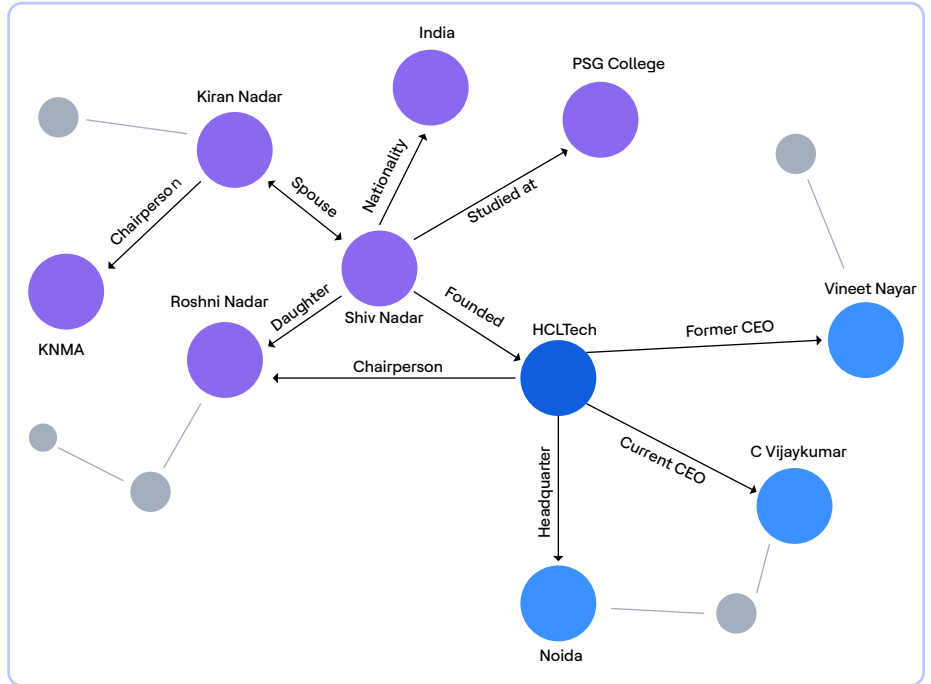


Figure 2 Knowledge graph depiction

In the above knowledge graph diagram, the nodes represent the entities and the edges connecting nodes present the relationship between them. The knowledge graph efficiently covers the word semantics that makes the processing of graph information by the systems more effective. For example, the entity "Shiv Nadar" can be linked to the entity "HCLTech" as Shiv Nadar has clear relation with HCLTech being the founder of the company.

RAG technique is based on dividing the text data into chunks, converting these chunks into embedding (vector representation of text in numbers) and storing it in database. When the query is applied, RAG retrieves the conceptually similar text through the similarity search technique (using algorithm like cosine similarity of vectors) applied over the chunks stored in the embedding database. Though vectors are very good at capturing essence of text, it fails in making accurate assessment of what is inside of vector, what is present around it and how it connects with the other things part of overall text.

GraphRAG harnesses the power of vector-only RAG and blends it with strength of knowledge graphs. GraphRAG enhances the performance of LLMs by integrating knowledge graphs into the retrieval process, thereby addressing the limitations of vector-only RAG methods. GraphRAG leverages the structured relationships and semantic

richness of knowledge graphs to improve the accuracy and contextual relevance of information retrieval. This approach not only enhances the retrieval process but also ensures that the generated answers are grounded in the underlying data, reducing the likelihood of model hallucination and improving overall reliability.

A GenAI application that uses GraphRAG follows the same workflow as vector-only RAG application with an additional step of creating a knowledge graph at the beginning.

## GraphRAG lifecycle

GraphRAG lifecycle can be broken down into the following steps:

1. Graph creation: Before diving into the retrieval process, a knowledge graph is created. This graph can be a domain graph representing the world model relevant to the application or a lexical graph representing the document structure.
  - Domain graph: Represents the world model relevant to the application. Domain graph creation depends on the source data whether it is structured or unstructured text.
  - Lexical graph: Represents the document structure, including relationships between chunks, document objects, chapters, sections and more. Lexical graph creation is typically done through simple parsing and chunking strategies. Knowledge graphs can be created by LLMs. Also, there are commercial tools available for creating Knowledge graphs from unstructured data. For example, Neo4j Knowledge Graph Builder takes PDF documents, web pages, YouTube clips, or Wikipedia articles and automatically creates a knowledge graph from them.
2. Vector embedding: Text data is divided into chunks and converted into vector embeddings.
3. Graph-enhanced retrieval: Queries are applied over both the vector data and the knowledge graph, leveraging the graph structure to retrieve more contextually relevant information.
4. Answer generation: The retrieved information is used to generate accurate and contextually rich answers.



Figure 3 GraphRAG application building steps

## GraphRAG workflow

In case of vector-based RAG, first the source data is processed through the embedding model that results in the word vectors also called word embeddings. These word embeddings are then stored in the database that is known by the term embedding database. When the user sends the query to the GenAI application, it retrieves the information from the embedding

database based on the similarity of embeddings of query words that is measured by the distance or similarity to other embedding vectors in database. Application then constructs the prompt with the user query and the information retrieved from the vector database.

In case of GraphRAG, the knowledge graph constructed from the source data is additional step compared to vector-only RAG. Knowledge graph is stored in database along with the word embeddings. Knowledge graph and word embeddings can be stored in same database or two distinct databases. During the information retrieval step, retrieval path refers knowledge graph also in addition to the embedding database. Other than this, GraphRAG workflow is like that of vector-based RAG. Typical GraphRAG workflow is depicted in below diagram.

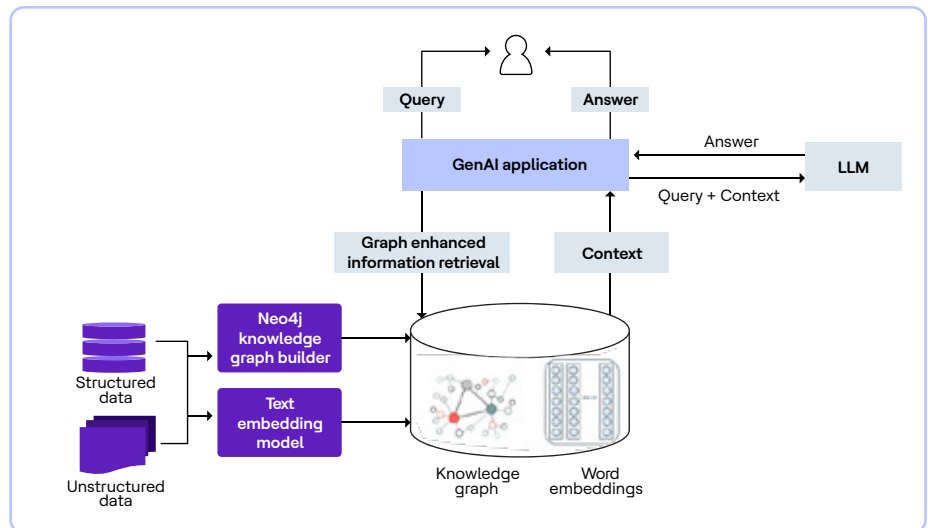


Figure 4 GraphRAG workflow  
(Ref: The GraphRAG Manifesto: Adding Knowledge to GenAI (1))

The inclusion of knowledge graphs in addition to the word embeddings in GraphRAG enhances the power of RAG as here the information retrieval path also includes knowledge graph. Knowledge graph contains the references to all the entities and relationships within the source data providing better reach to the disparate pieces of information through their shared attributes and gives newer insights.

Further, the knowledge graph is used to organize the data hierarchically into semantic clusters. These cluster-based partitioning allows for the pre-summarization of semantic concepts. This pattern allows for a more holistic understanding of the data, enabling the retrieval of contextually relevant information that vector-only RAG methods might miss. Hence, GraphRAG has been found to give overall better results in cases the vector-based RAG fails to give expected results.

## Examples of GraphRAG improvements over vector-only RAG

In this sub-section, we will explore few examples that demonstrate the improvements achieved by GraphRAG over traditional vector-only RAG methods. These examples highlight the enhanced accuracy, contextual relevance and efficiency of GraphRAG in different applications.

## Example 1: Financial article analysis

One notable example comes from Lettria, an AWS Partner, which demonstrated that integrating graph-based structures into RAG workflows improves answer precision by up to 35% compared to vector-only retrieval methods<sup>[9]</sup>.

In this case, GraphRAG was applied to a text-to-graph workflow that ingested 10,000 financial articles into a knowledge graph. The quality of the answers improved markedly with GraphRAG and the process required one-third fewer tokens helping lower the cost also.

## Example 2: Narrative private data analysis

Microsoft Research has highlighted the effectiveness of GraphRAG in their blog post titled "GraphRAG: Unlocking LLM discovery on narrative private data." The blog underlines the observation that vector-only RAG struggles to handle the queries that require aggregation of information across the dataset to compose an answer. Vector-only RAG performs terribly for query such as "What are the top 5 themes in the data". This is because RAG relies on vector search of semantically similar content within the dataset, but the given query has nothing that can direct it to correct information through RAG <sup>[6]</sup>.

Whereas GraphRAG can produce effective results for such questions because knowledge graphs organize the data hierarchically into semantic clusters. These cluster-based partitioning allows for the pre-summarization of semantic concepts and a more holistic understanding of the data, enabling the retrieval of contextually relevant information.

# Benefits

GraphRAG provides following benefits over the vector-only RAG:

- Microsoft [6] has proved that the knowledge graph assisted information retrieval by GraphRAG provides vastly improved results in comparison to the vector-only RAG. The context window provided by GraphRAG has higher relevant content that results in more accurate answers.
- In case of data where relationships between data points are important, knowledge graph provides improved data understanding as it is good at navigating deep hierarchies, finding hidden connections between items and discovering relationships between items [2]
- Microsoft [6] has also discovered that GraphRAG required between 26% and 97% fewer tokens than alternative approaches that makes it cheaper option.
- GraphRAG reduces the model hallucination by keeping the results grounded.

# Conclusion

The word-based computations and language skills inherent in LLMs and vector-only RAG are being widely used today but fall short in certain scenarios. GraphRAG adds another step of the knowledge graph creation to the vector-only graph that helps identify the entities and relationship between the entities across varied datasets. GraphRAG complements the vector-only RAG well and enhances the effectiveness of LLM output that is more accurate and aligned with the input query, more complete and cheaper.

# References

1. [The GraphRAG Manifesto: Adding Knowledge to GenAI](#)
2. [RAG with a Graph database | OpenAI Cookbook](#)
3. [Re-evaluation of Knowledge Graph Completion Methods | by Farahnaz Akrami | Medium](#)
4. [Gartner Hype Cycle for AI: Why Knowledge Graphs Are Essential for Enterprises? | LinkedIn](#)
5. [Graph Assets - Best practises for your RAG based project.](#)
6. [GraphRAG: Unlocking LLM discovery on narrative private data - Microsoft Research](#)
7. [Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering](#)
8. [Knowledge Graphs: Opportunities and Challenges | Artificial Intelligence Review](#)
9. [Improving Retrieval Augmented Generation accuracy with GraphRAG | AWS Machine Learning Blog](#)

# Author information



## **Abhishek Tomar**

Abhishek Tomar is part of GenAI practice testing team in HCLTech with focus on bringing productivity and quality improvement in the testing projects. He has 25+ years of industry experience in leading high-impact engineering, automation and innovation initiatives in software development and testing for embedded systems. An AI/ML enthusiast with keen interest in applying GenAI, Deep Learning, ML concepts towards engineering innovation that bring measurable outcomes.

# HCLTech | Supercharging Progress™

HCLTech is a global technology company, home to more than 223,000 people across 60 countries, delivering industry-leading capabilities centered around digital, engineering, cloud and AI, powered by a broad portfolio of technology services and products. We work with clients across all major verticals, providing industry solutions for Financial Services, Manufacturing, Life Sciences and Healthcare, Technology and Services, Telecom and Media, Retail and CPG and Public Services. Consolidated revenues as of 12 months ending March 2025 totaled \$13.8 billion. To learn how we can supercharge progress for you, visit [hcltech.com](https://hcltech.com).

[hcltech.com](https://hcltech.com)

