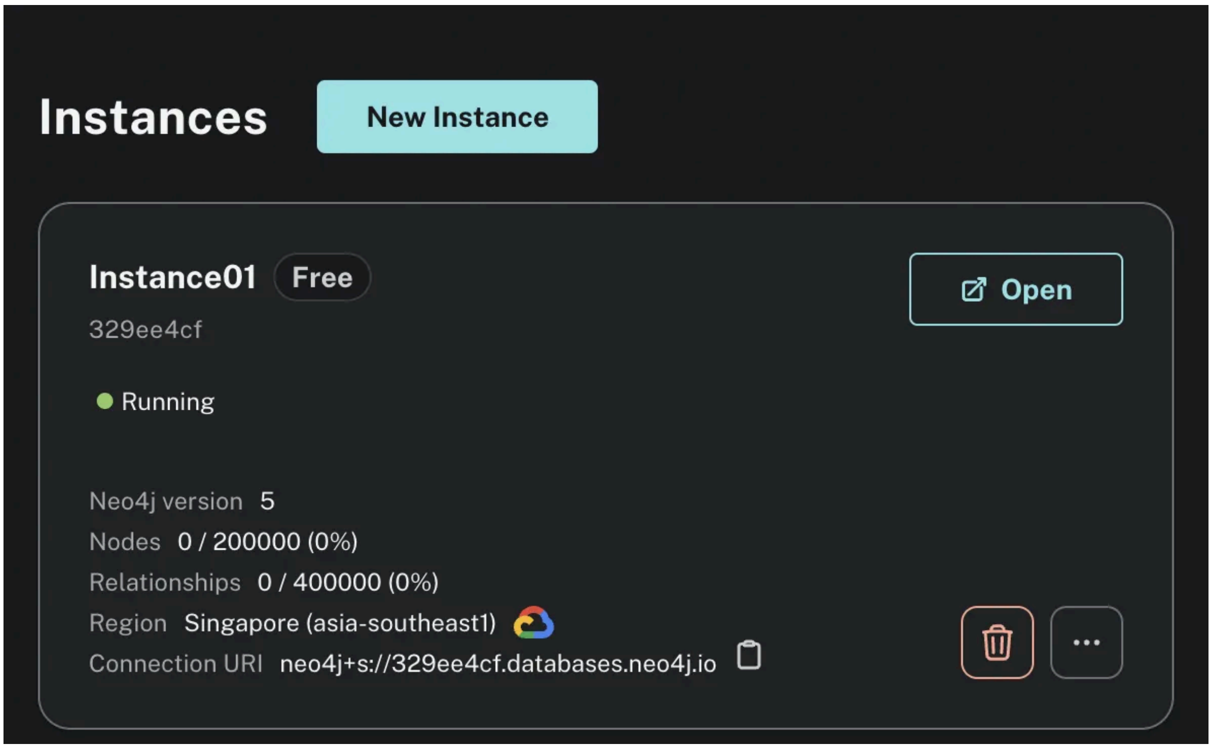
Graph-RAG

1. Document Setup and Graph Generation



Instances

New Instance

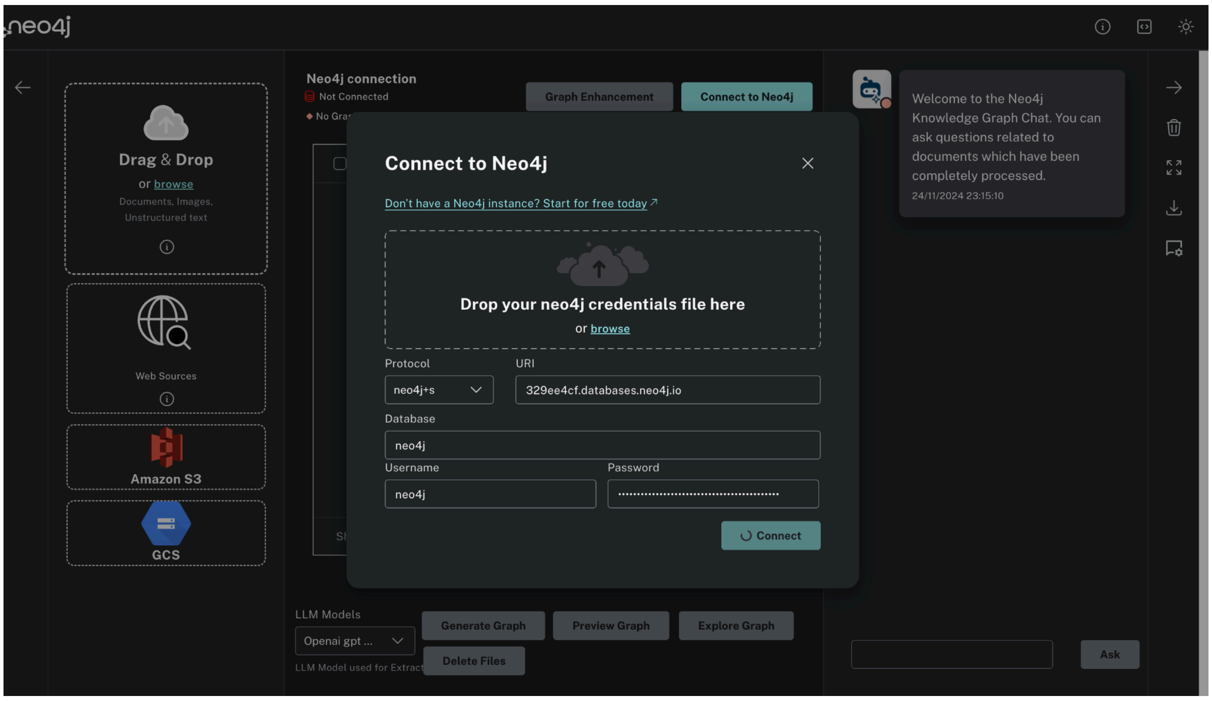
Instance01 Free

329ee4cf

● Running

Open

Neo4j version	5
Nodes	0 / 200000 (0%)
Relationships	0 / 400000 (0%)
Region	Singapore (asia-southeast1)
Connection URI	neo4j+s://329ee4cf.databases.neo4j.io



Connect to Neo4j

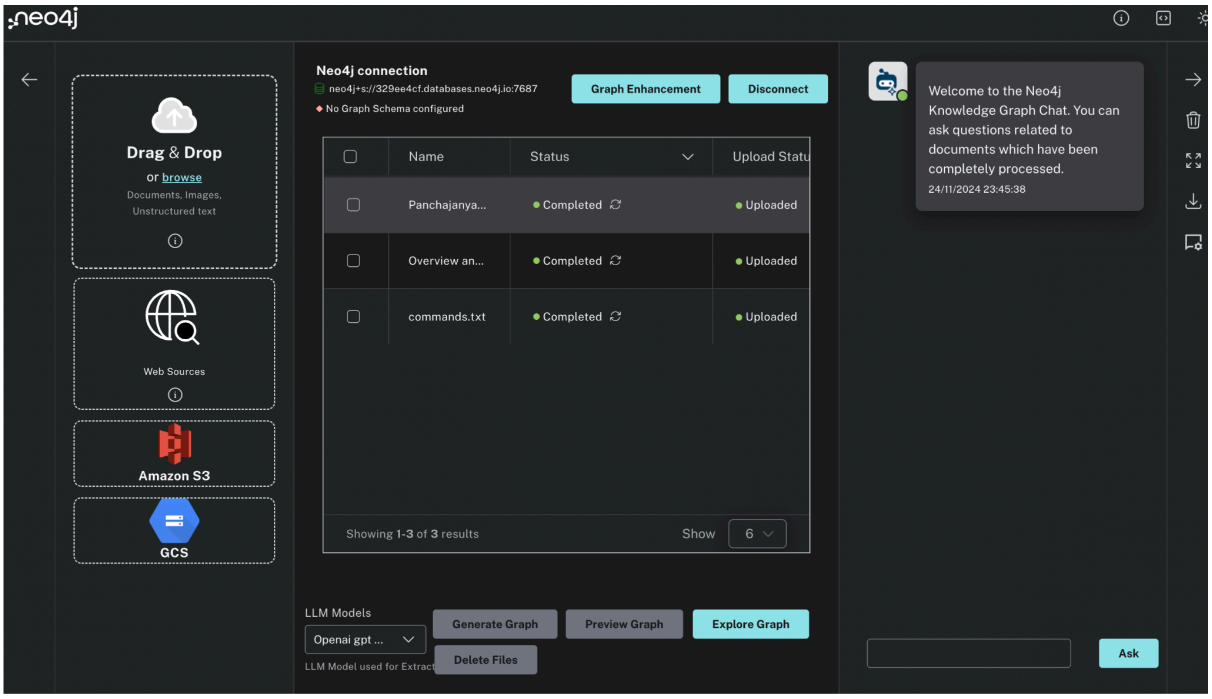
Don't have a Neo4j instance? Start for free today

Drop your neo4j credentials file here
or browse

Protocol: neo4j+s
URI: 329ee4cf.databases.neo4j.io
Database: neo4j
Username: neo4j
Password: ••••••••••••••••••••••••••••••••••••

Connect

Welcome to the Neo4j Knowledge Graph Chat. You can ask questions related to documents which have been completely processed.
24/11/2024 23:15:10



Neo4j connection
neo4j+s://329ee4cf.databases.neo4j.io:7687
No Graph Schema configured

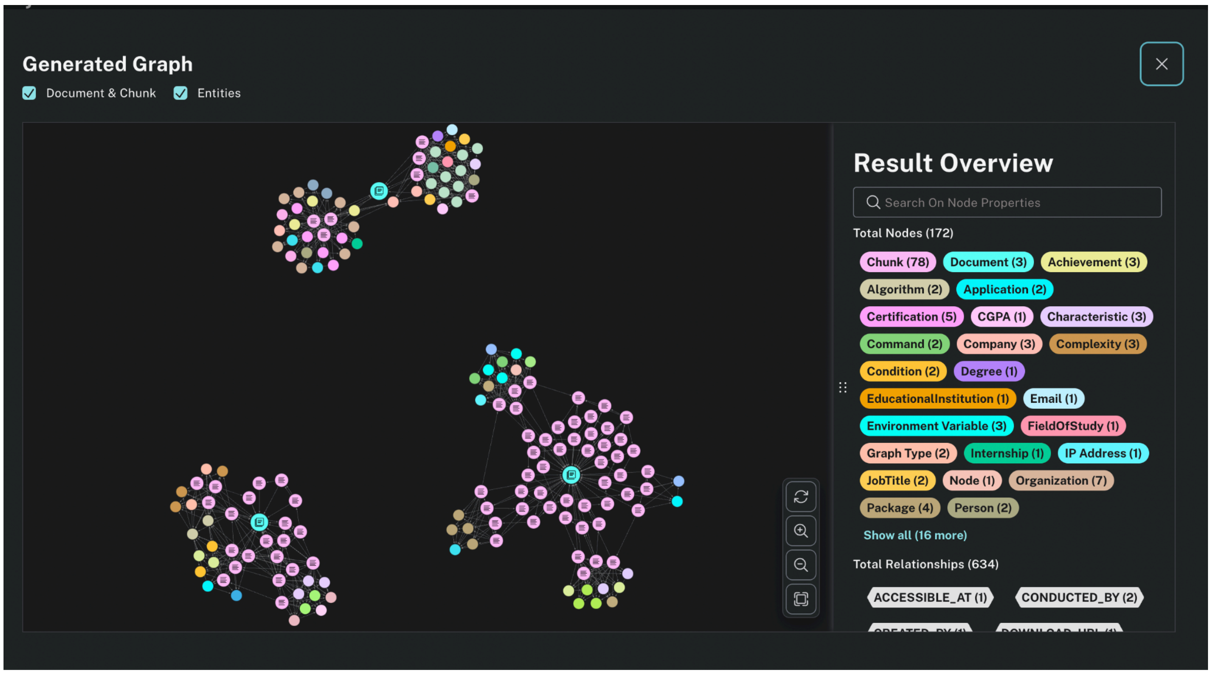
	Name	Status	Upload Statu
	Panchajanya...	Completed	Uploaded
	Overview an...	Completed	Uploaded
	commands.txt	Completed	Uploaded

Showing 1-3 of 3 results Show 6

LLM Models
Openai gpt ...
LLM Model used for Extract

Welcome to the Neo4j Knowledge Graph Chat. You can ask questions related to documents which have been completely processed.
24/11/2024 23:45:38

2. Generated Knowledge Graphs



Result Overview

Total Nodes (172)

Chunk (78) Document (3) Achievement (3) Algorithm (2) Application (2) Certification (5) CGPA (1) Characteristic (3) Command (2) Company (3) Complexity (3) Condition (2) Degree (1) EducationalInstitution (1) Email (1) Environment Variable (3) FieldOfStudy (1) Graph Type (2) Internship (1) IP Address (1) JobTitle (2) Node (1) Organization (7) Package (4) Person (2)

Show all (16 more)

Total Relationships (634)

ACCESSIBLE_AT (1) CONDUCTED_BY (2)

3. Questions and Answers

Document 1: Overview and Strengths/Weaknesses of Uninformed Search Techniques

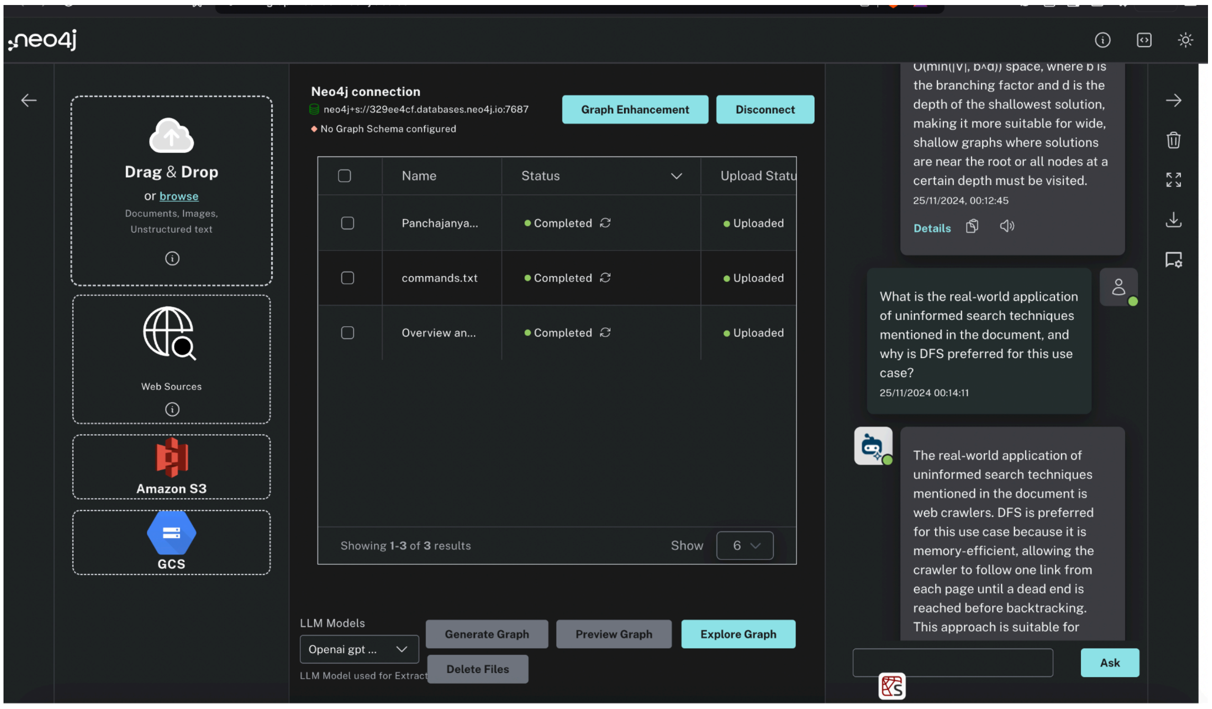
1. Q: What are the key differences between DFS and BFS in terms of memory efficiency and path finding capabilities? A: DFS (Depth-first search) is memory-efficient as it only stores nodes on the current path, making it suitable for scenarios where memory is limited. However, it can get stuck in infinite loops with cyclic graphs. DFS excels in finding solutions deep in the graph, such as finding the longest path in a maze. On the other hand, BFS (Breadth-first search) requires more memory because it stores all nodes at the current level, but it guarantees finding the shortest path in unweighted graphs. BFS is preferable when solutions are near the root or when all nodes at a certain depth must be visited.
2. Q: How do DFS and BFS compare in terms of time and space complexity, and what makes each suitable for different scenarios? A: DFS and BFS both have a time complexity of O(|V| + |E|), where |V| is the number of vertices and |E| is the number of edges. However, they differ in space complexity. DFS uses O(|V|) space in the worst case, making it suitable for deep, narrow graphs where memory is limited. BFS, on the other hand, requires O(min(|V|, b^d)) space, where b is the branching factor and d is the depth of the shallowest solution, making it more suitable for wide, shallow graphs where solutions are near the root or all nodes at a certain depth must be visited.
3. Q: What is the real-world application of uninformed search techniques mentioned in the document, and why is DFS preferred for this use case? A: The real-world application of uninformed search techniques mentioned in the document is web crawlers. DFS is preferred for this use case because it is memory-efficient, allowing the crawler to follow one link from each page until a dead end is reached before backtracking. This approach is suitable for finding distant pages without the need to store all links from each page, which would be impractical for BFS due to its higher memory requirements.

Document 2: Commands.txt (Big Data Project Setup)

1. Q: What is the complete process of setting up the Spark cluster, including master and worker node configuration? A: [Include the detailed answer from the chat]
2. Q: What performance testing scenarios are configured in the script, and what parameters are being varied? A: The performance testing scenarios configured in the script involve a "Performance Tests Scale-up Analysis." This analysis is conducted by varying the number of virtual machines (VMs) used, specifically scaling up from 2 to 6 VMs. The parameters being varied in this scenario are the total number of executor cores, which corresponds to the number of VMs used in each test iteration.
3. Q: What are the different monitoring and troubleshooting commands available for checking cluster health, and what aspects do they monitor? A: The context provides a command for monitoring cluster access through the Spark Web UI, which can be accessed at the URL `http://$MASTER_IP:8080`. This tool monitors the overall health and performance of the Spark cluster. The Spark Web UI typically monitors aspects such as job progress, executor status, and resource usage.

Document 3: Panchajanya_Thummala_AI_Op.pdf

1. Q: What is the candidate's experience with monitoring tools and incident management, and how does it relate to AI implementation? A: [Include the detailed answer from the chat]
2. Q: What are all the AI-related certifications and skills mentioned in the resume, and how recent are they? A: Panchajanya Thummala is certified in Weaviate vector databases, which is an AI-related certification. Additionally, he possesses skills in AI-related areas such as Langchain and RAG (Retrieval-Augmented Generation).
3. Q: What projects has the candidate worked on that demonstrate their transition from support to AI engineering? A: Panchajanya Thummala has worked on developing an Incident Management RAG chatbot for his current team. This project demonstrates his transition from a support role to AI engineering, as it involves leveraging AI to find similar issues during incident calls, thereby enhancing the support process.



Neo4j connection
neo4j+s://329ee4cf.databases.neo4j.io:7687
No Graph Schema configured

	Name	Status	Upload Statu
	Panchajanya...	● Completed	● Uploaded
	commands.txt	● Completed	● Uploaded
	Overview an...	● Completed	● Uploaded

Showing 1-3 of 3 results   Show 6

O(min(|V|, b^d)) space, where b is the branching factor and d is the depth of the shallowest solution, making it more suitable for wide, shallow graphs where solutions are near the root or all nodes at a certain depth must be visited.
25/11/2024, 00:12:45

What is the real-world application of uninformed search techniques mentioned in the document, and why is DFS preferred for this use case?
25/11/2024 00:14:11

The real-world application of uninformed search techniques mentioned in the document is web crawlers. DFS is preferred for this use case because it is memory-efficient, allowing the crawler to follow one link from each page until a dead end is reached before backtracking. This approach is suitable for