HybridRAG: A Fusion of Graph and Vector Retrieval - Mitesh Patel, NVIDIA

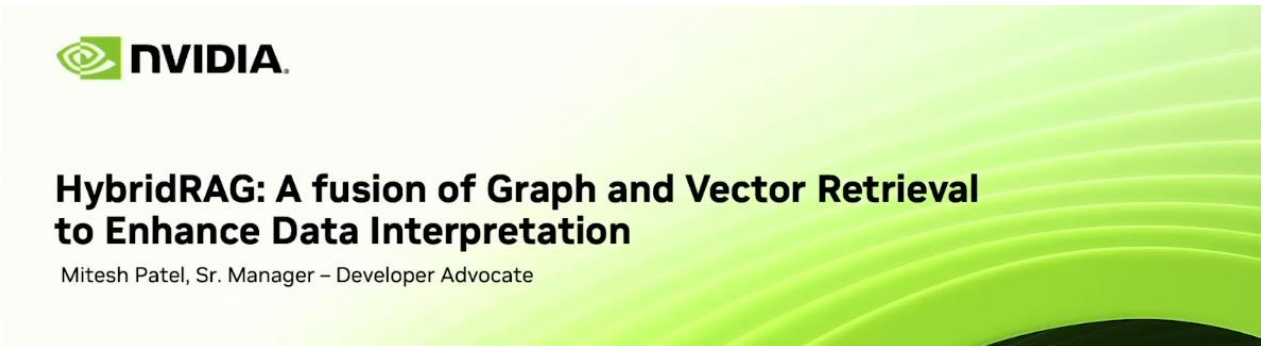
AI Engineer
247K subscribers
Subscribe
235 Share Ask Download ...

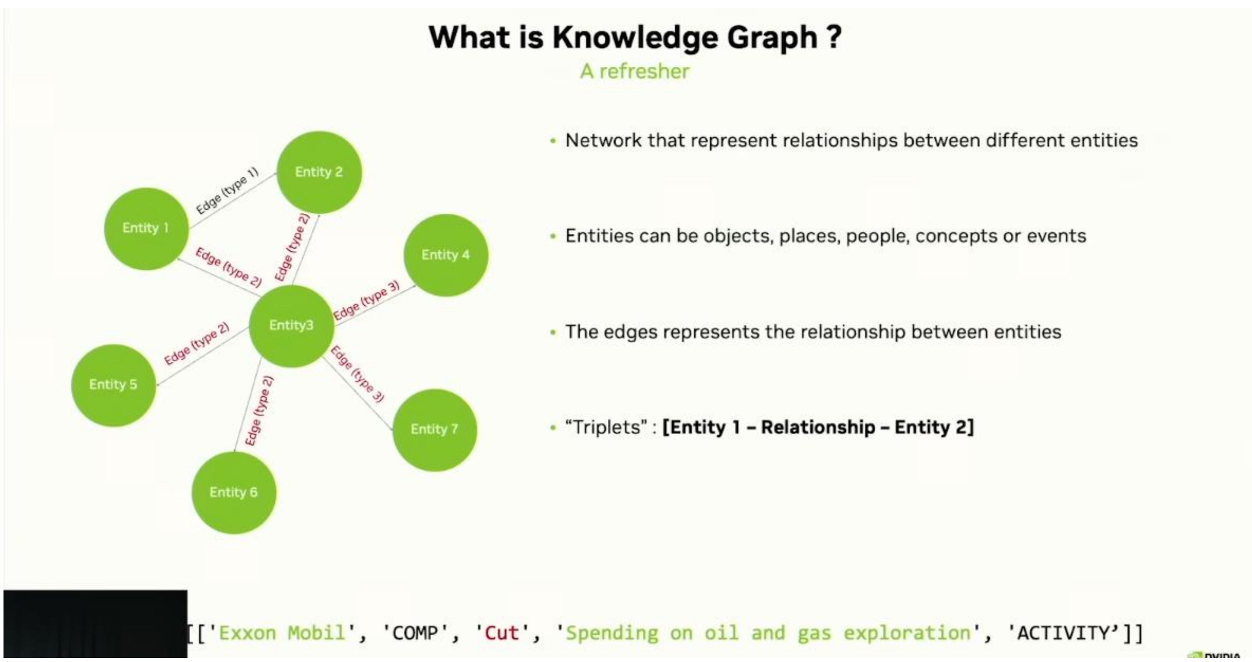
8,818 views Jul 22, 2025 AIEWF 2025 Complete Playlist

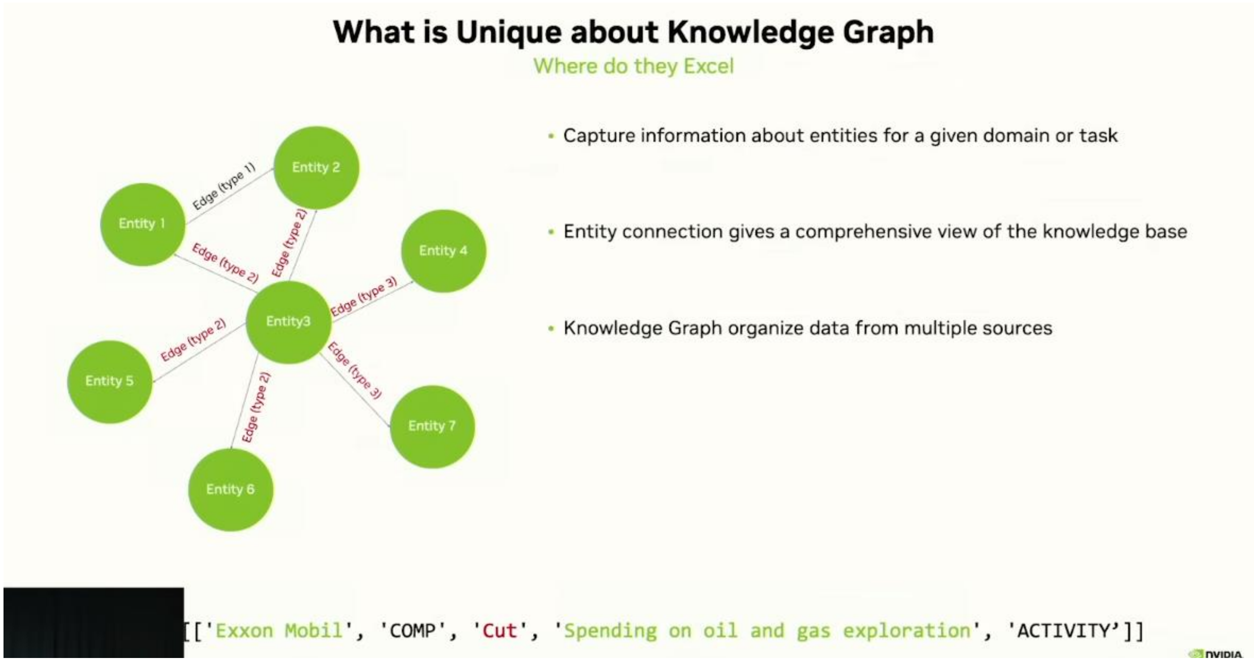
Interpreting complex information from unstructured text data poses significant challenges to Large Language Models (LLM), with difficulties often arising from specialized terminology and the multifaceted relationships between entities in document architectures. Conventional Retrieval Augmented Generation (RAG) methods face limitations in capturing these nuanced interactions, leading to suboptimal performance. In our talk, we introduce a novel approach integrating Knowledge Graph-based RAG (GraphRAG) with VectorRAG, designed to refine question-answering (Q&A) systems for more effective information extraction from complex texts. Our approach employs a dual retrieval strategy that harnesses both knowledge graphs and vector databases, enabling the generation of precise and contextually appropriate answers, thereby setting a new standard for LLMs in processing sophisticated data.

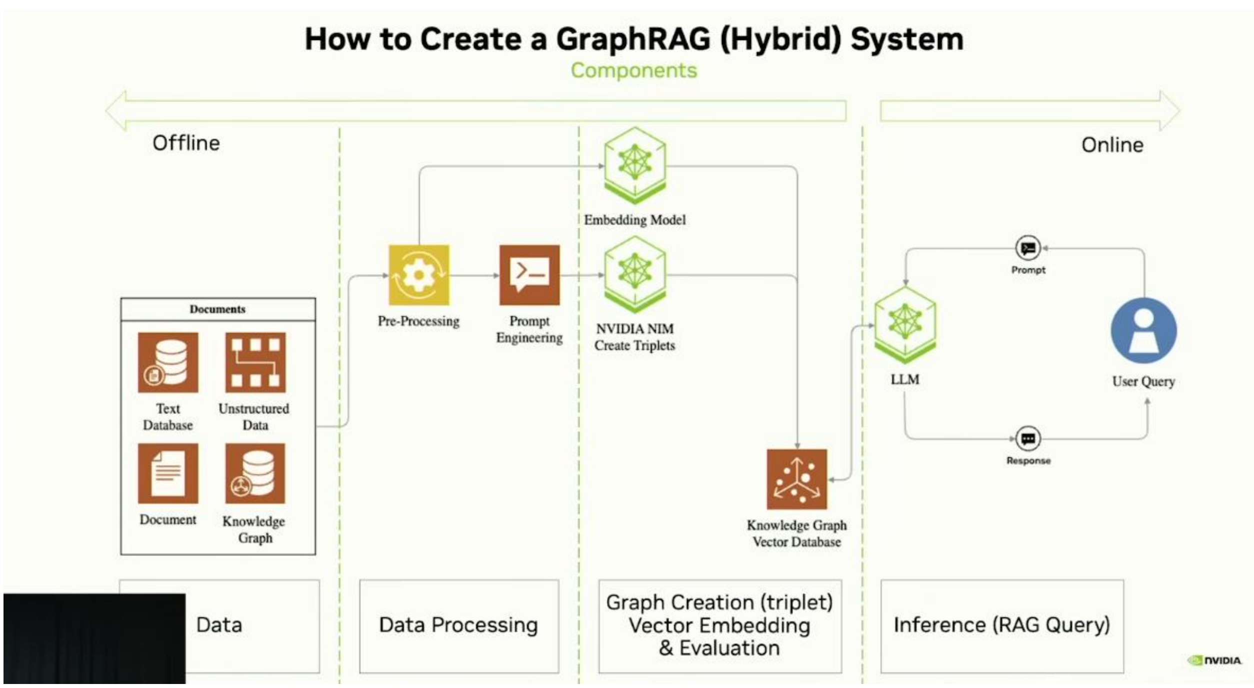
About Mitesh Patel

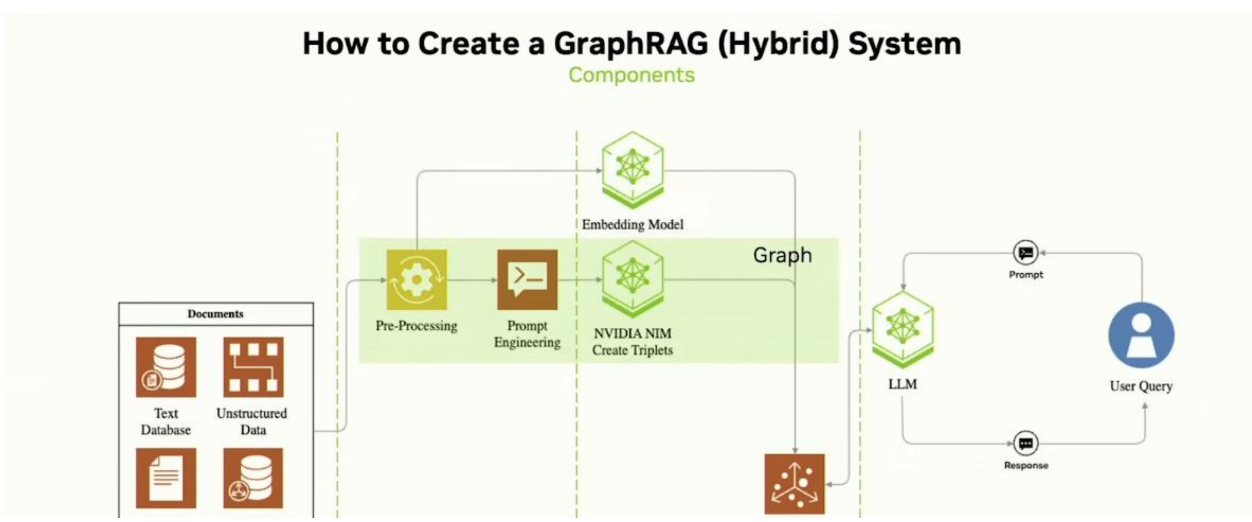
Mitesh Patel is a developer advocate manager at NVIDIA. His team is responsible for creating workflows to showcase how developers can harness GPU acceleration in their workflows using tools and frameworks popular in the developer community. Before NVIDIA, he was a senior research scientist at Fuji Xerox Palo Alto Laboratory Inc. (a research subsidiary of Fuji Xerox), where he worked on developing indoor localization technologies for applications such as asset tracking in hospitals and delivery cart tracking in manufacturing facilities. Mitesh received his Ph.D. in Robotics from the Center of Autonomous Systems (CAS) at the University of Technology Sydney, Australia in 2014.

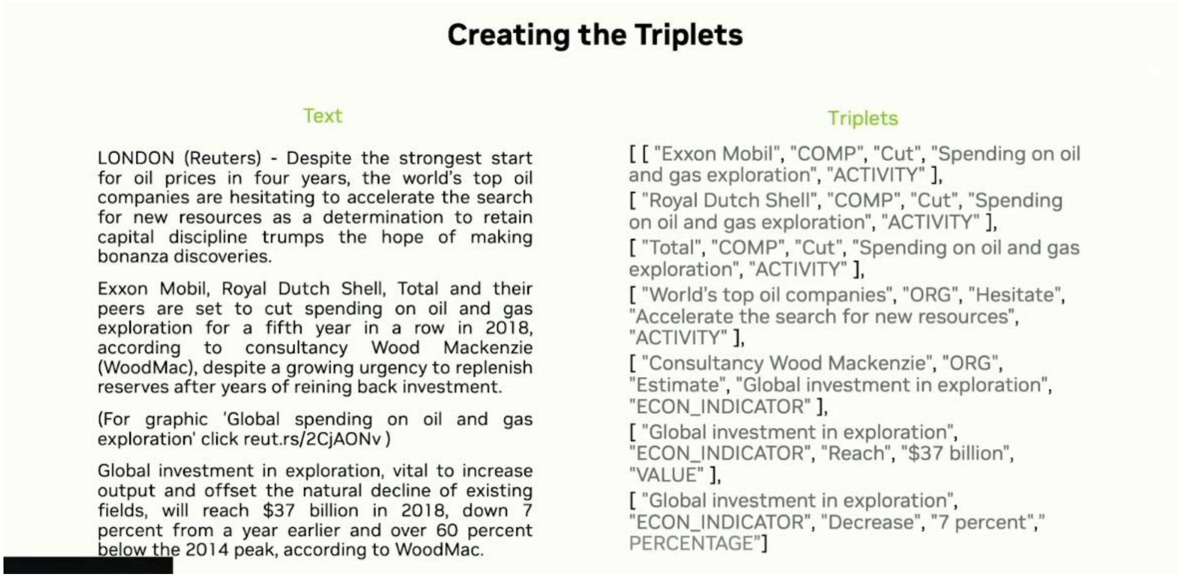




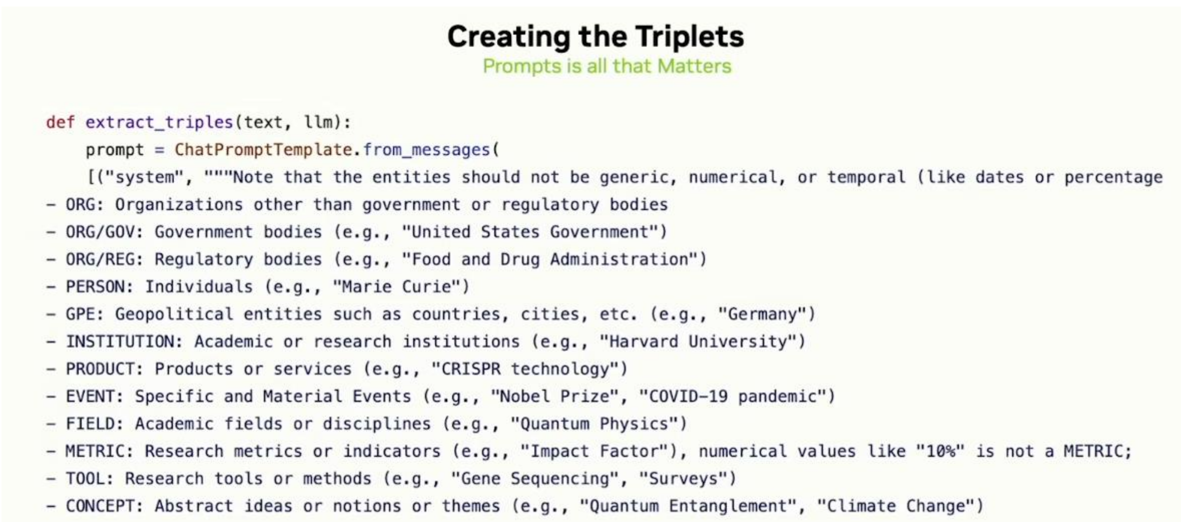

What is Unique about Knowledge Graph
Where do they Excel
Entity 1
Entity 2
Entity3
Entity 4
Entity 5
Entity 6
Entity 7
Edge (type 1)
Edge (type 2)
Edge (type 2)
Edge (type 3)
Edge (type 2)
Edge (type 3)
Edge (type 2)
• Capture information about entities for a given domain or task
• Entity connection gives a comprehensive view of the knowledge base
• Knowledge Graph organize data from multiple sources
[['Exxon Mobil', 'COMP', 'Cut', 'Spending on oil and gas exploration', 'ACTIVITY']]
NVIDIA

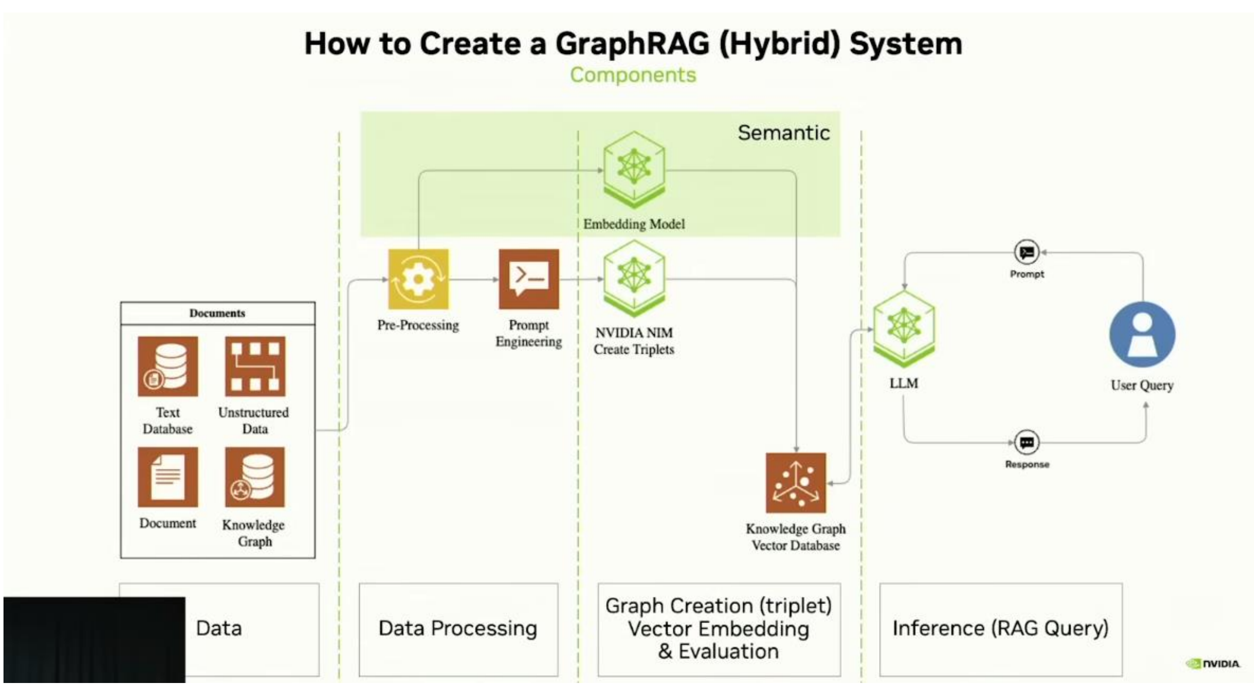

How to Create a GraphRAG (Hybrid) System
Components
Offline
Online
Documents
Text Database
Unstructured Data
Document
Knowledge Graph
Pre-Processing
Prompt Engineering
Embedding Model
NVIDIA NIM Create Triplets
Knowledge Graph Vector Database
LLM
Prompt
Response
User Query
Data
Data Processing
Graph Creation (triplet) Vector Embedding & Evaluation
Inference (RAG Query)
NVIDIA

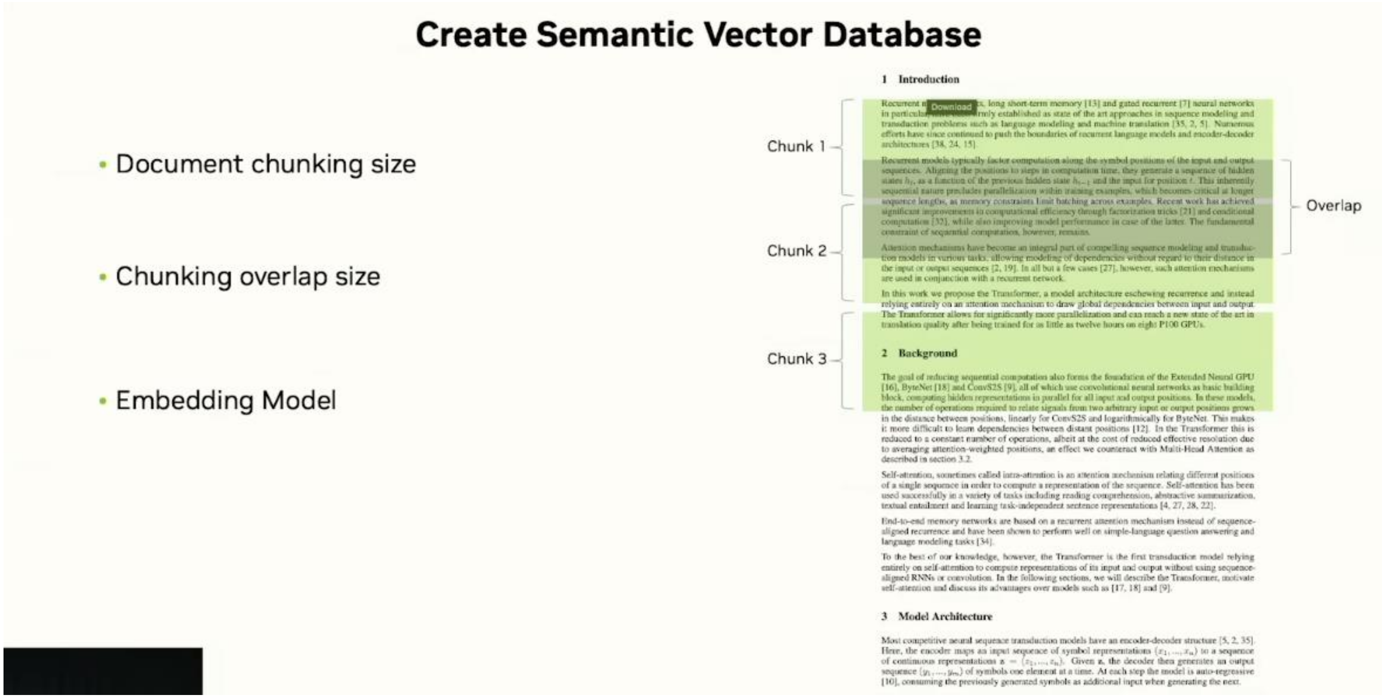

How to Create a GraphRAG (Hybrid) System
Components
Embedding Model
Graph
Pre-Processing
Prompt Engineering
NVIDIA NIM Create Triplets
Documents
Text Database
Unstructured Data
LLM
Prompt
Response
User Query



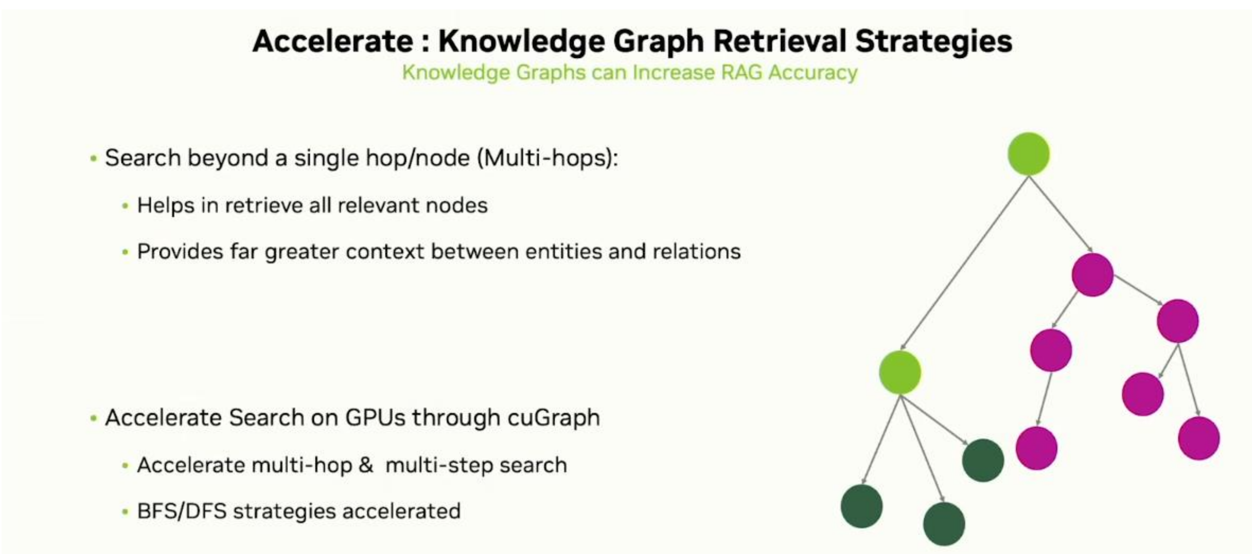
Triplets expose the relationship/information between 2 entities, this information is helpful. We can use LLMs to extract this information for us to save it in our KB in triplet format.



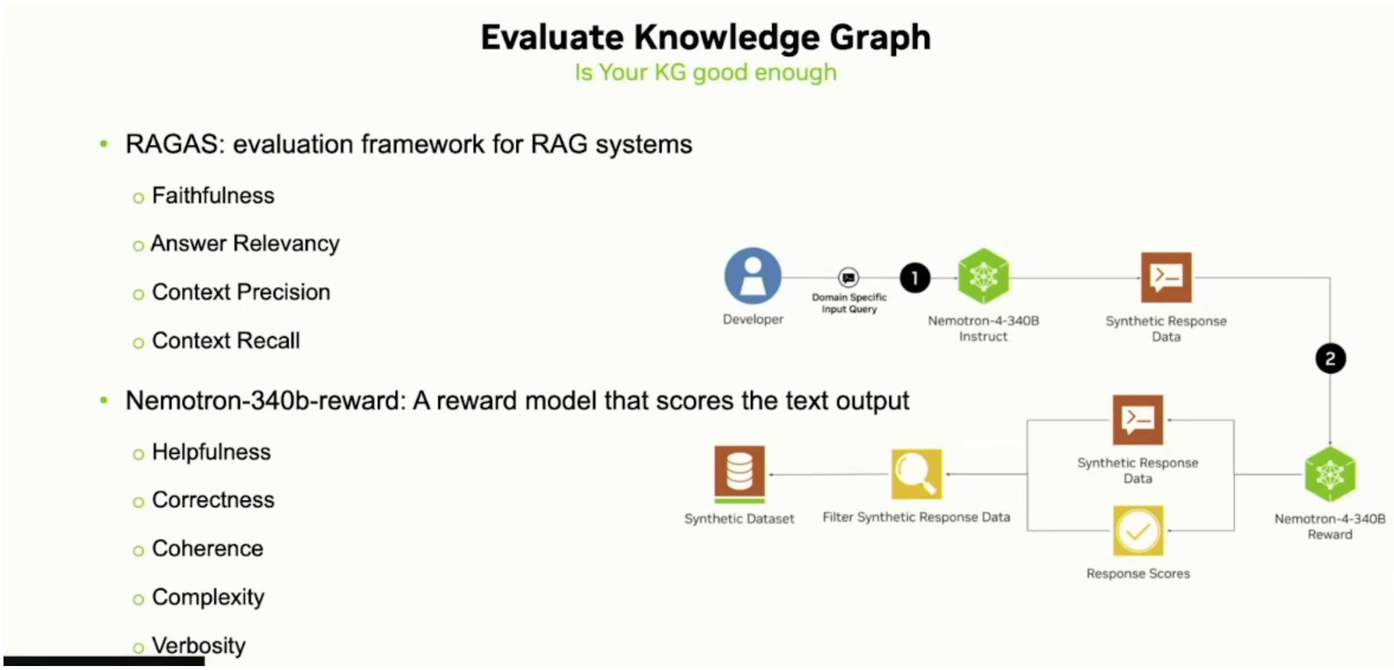




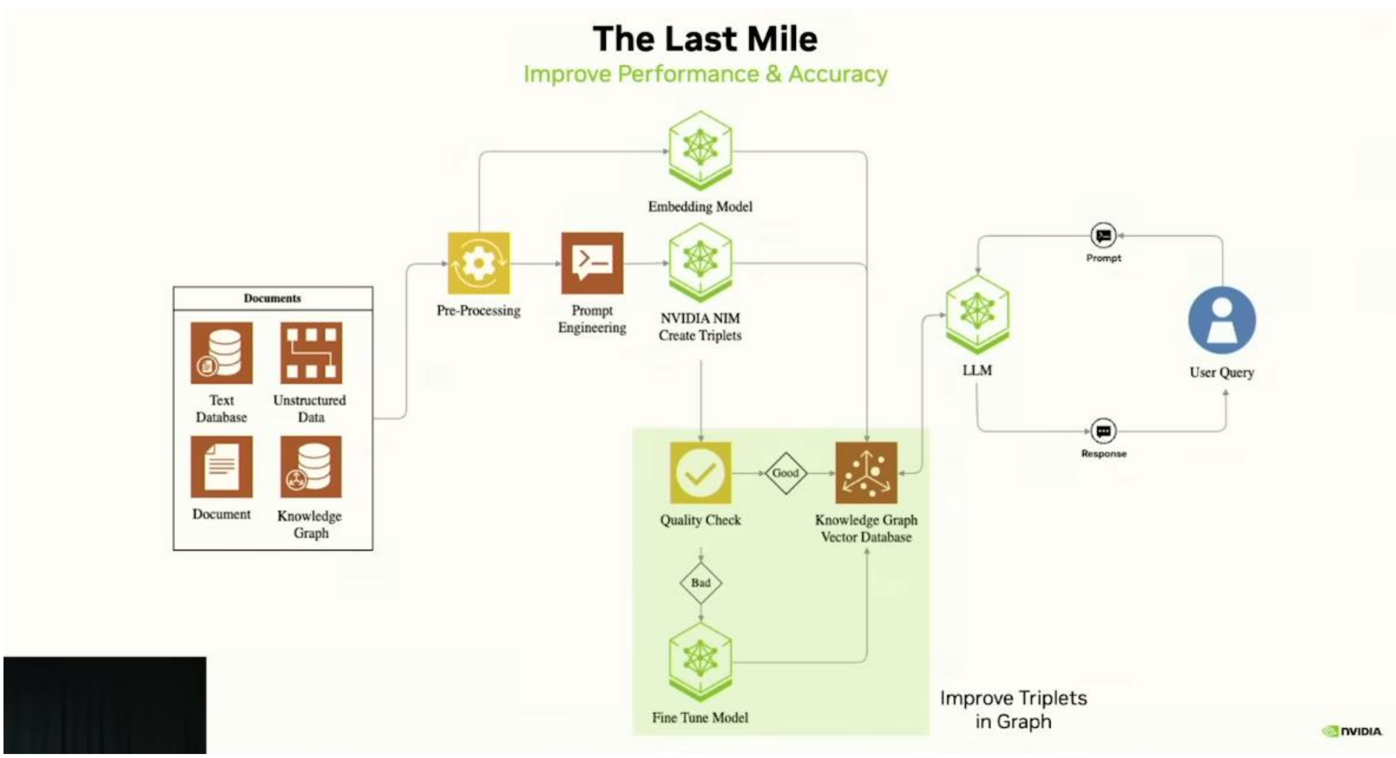
This is where you convert your documents into chunks and put in a vector DB



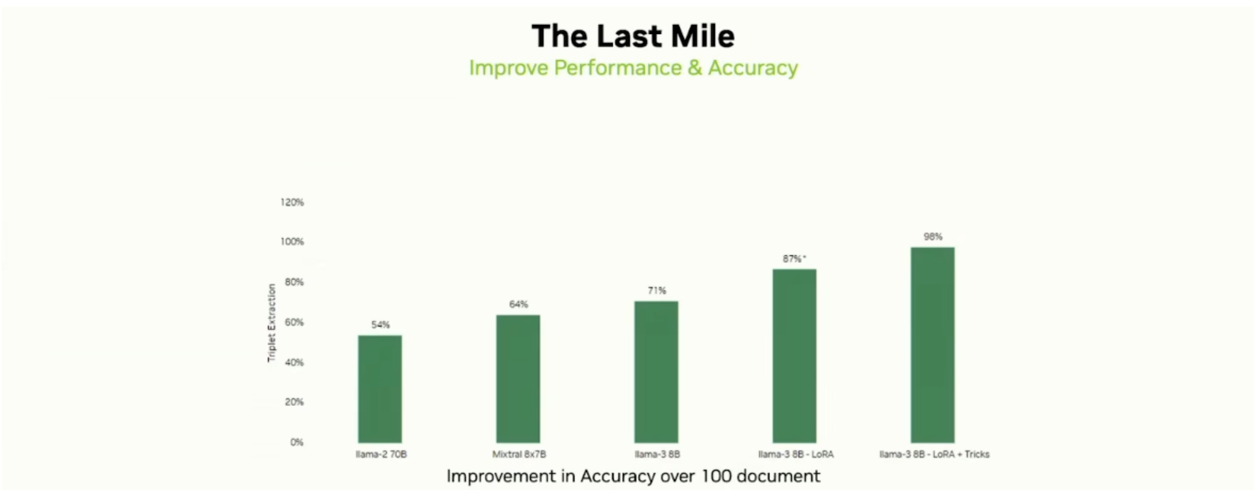
You can query your data using different strategies like depth or breadth approaches

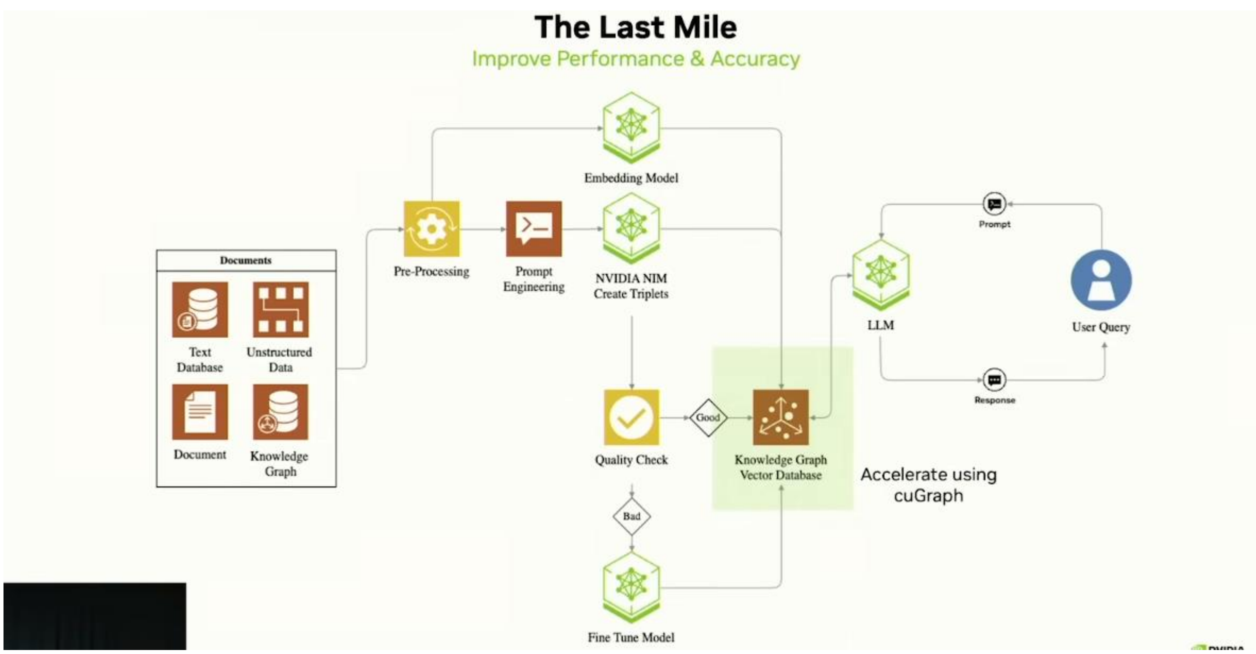


RAGAS is a pip install library that allows you to bring your own model



The way you create your Knowledge Graph (KG) can help improve your system performance and results





If your graph gets really big with lots of nodes, you might start having latency and network issues. Tweak your logic and measure the network performance

The Last Mile

Improve Performance & Accuracy

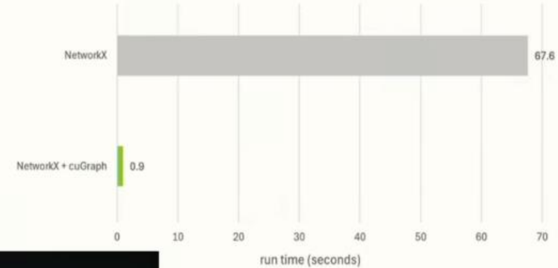
The full end-to-end accelerated NetworkX experience is now enabled by setting the `NX_CUGRAPH_AUTOCONFIG` environment variable to True.

```
%env NX_CURGAPH_AUTOCONFIG=True

import pandas as pd
import networkx as nx

url = "https://data.rapids.ai/cugraph/datasets/cit-Patents.csv"
df = pd.read_csv(url, sep=" ", names=["src", "dst"], dtype="int32")
G = nx.from_pandas_edgelist(df, source="src", target="dst")

%time result = nx.betweenness_centrality(G, k=10)
```



[redacted]m run on a citation graph of U.S. patents (4M nodes, 16M edges) is 70x faster than NetworkX on CPU;

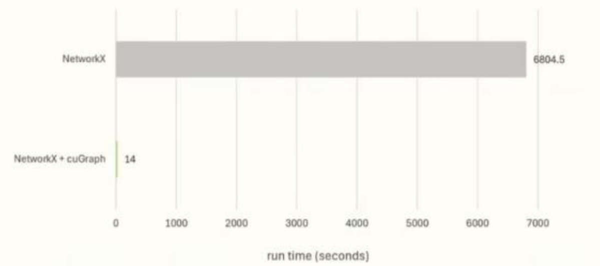


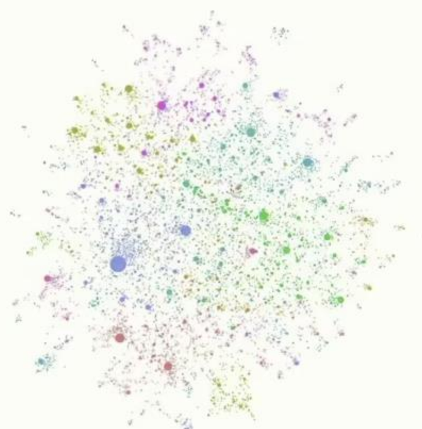
Figure 2. The betweenness centrality algorithm run on the Live Journal social network (5M nodes, 69M edges) is 485x faster than NetworkX on CPU for number of samples (k) set to 100.; SW: NetworkX 3.4.1, cuGraph/nx-cugraph 24.10; GPU: NVIDIA A100 80GB; CPU: Intel Xeon w9-3495X (56 cores) 250GB

NVIDIA

Should I Use Graph or Semantic or Hybrid

What to use ?

- Your data
  - Structured data
  - Semantic/Unstructured data
  - Can graph be created
- Application/Usecase
  - Complex relationship
  - Semantic relationship
  - Latency



Developer Tools and Resources

Accelerate innovation and growth

Learn more: developer.nvidia.com

Individuals

Software
100s of APIs, models, SDKs, microservices, and early access to NVIDIA tech

Training
Hands-on self-paced courses, instructor-led workshops, and certifications

GPU Sandbox
Approval basis, multi-GPU and multi-node

Learning
Tutorials, self-paced courses, blogs, documentation, code samples

Community
Dedicated developer forums, meetups, hackathons

Ecosystem
GTC, NVIDIA Partner Network

Organizations

Startups
Cloud credits, engineering resources, technology discounts, exposure to VCs

Venture Capital
Deal flow and portfolio support for Venture Capital firms

Higher Education
Teaching kits, training, curriculum co-development, grants

ISVs and SIs
Engineering guidance, discounts, marketing opportunities

Research
Grant programs, collaboration opportunities

Enterprises
Tailored developer training, skills certification, technical support