Knowledge Graphs and GraphRAG: Building the Next Generation of Data Platforms

Pascal Sun

UWA NLP-TLP Group

The University of Western Australia

May 20, 2025

Table of Contents

1 Why we learn **Graph**?

2 Challenges in Data Engineering now

3 Knowledge Graph

4 How it works?

5 Upcoming Research into Reality

About me



Pascal Sun

Full Stack Developer

UWA PhD candidate

https://pascalsun.github.io

- Delivered several successful products
  - ATLYST
  - RoadPod VM
  - Salix
- Lead AI product development
- Research Area: **Unstructured Data**
  - Processing
  - Structuring
  - Representation Learning
  - Applications
- UWA NLP-TLP GROUP

Product I built

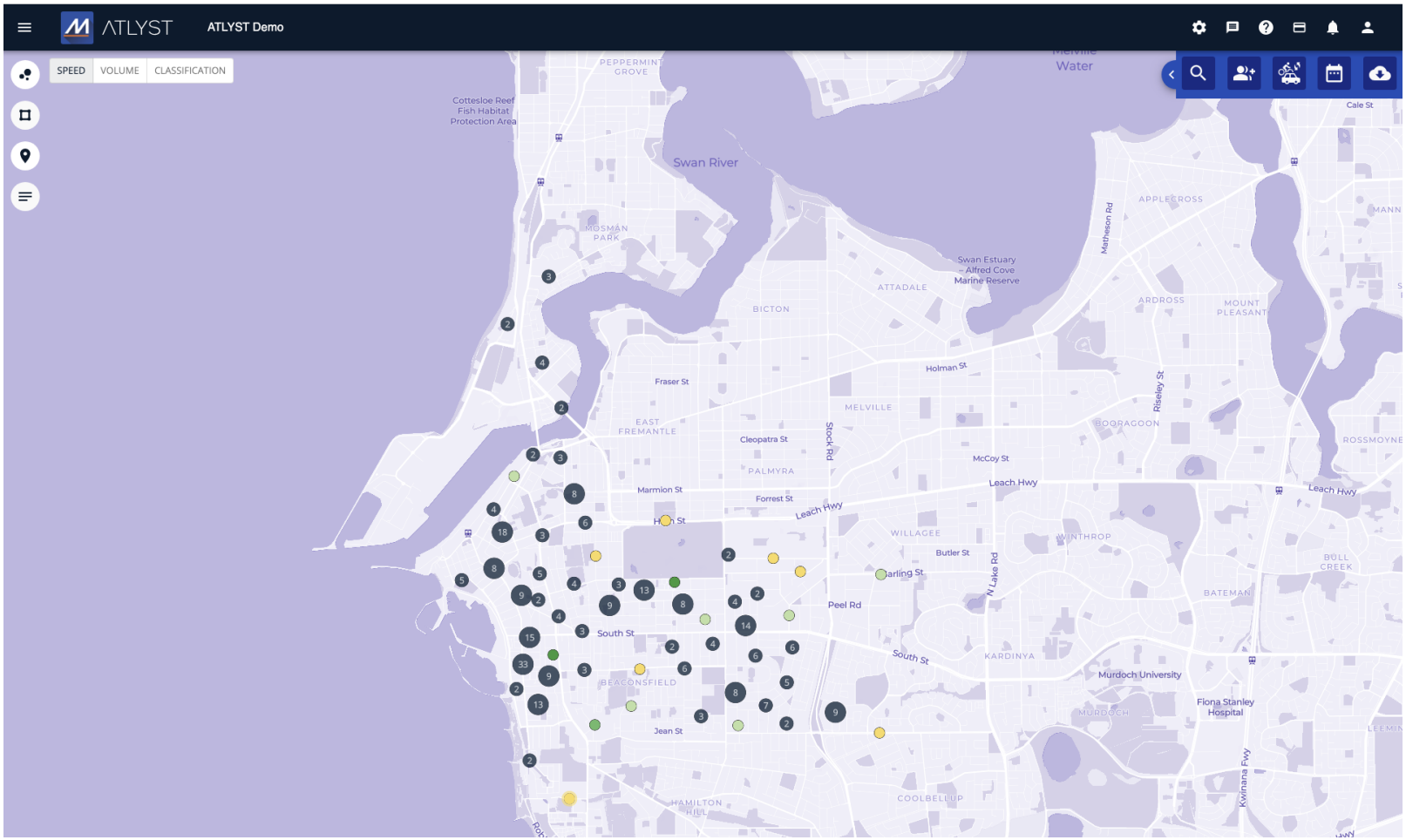


Figure: ATLYST: Revolutionary Traffic Data Management Platform

Product I built



Figure: Roadpod VM: Hardware Device



Figure: Roadpod VM: Deploy on the road

Product I built

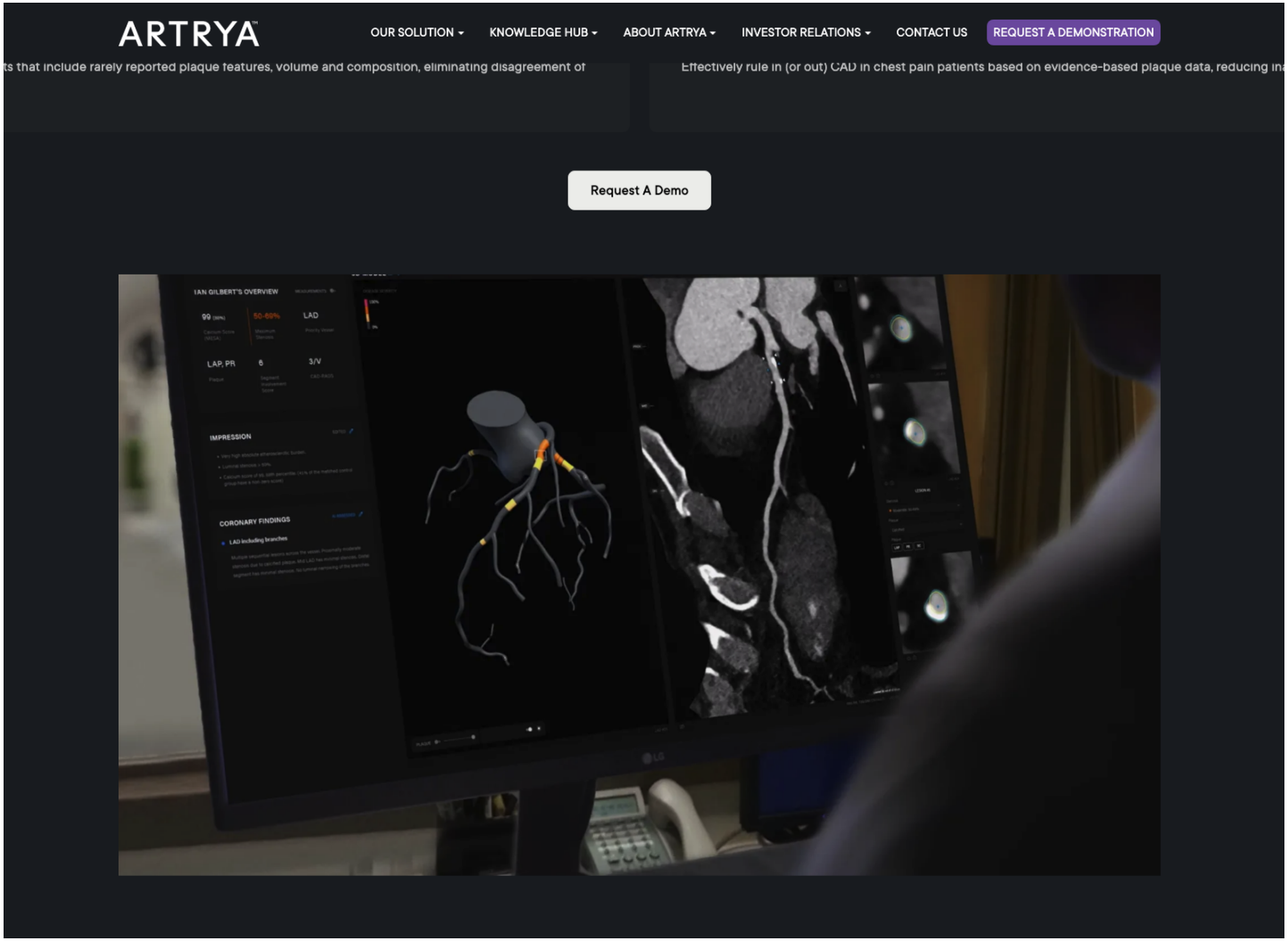


Figure: Salix: AI based coronary artery disease diagnosis

Product I built

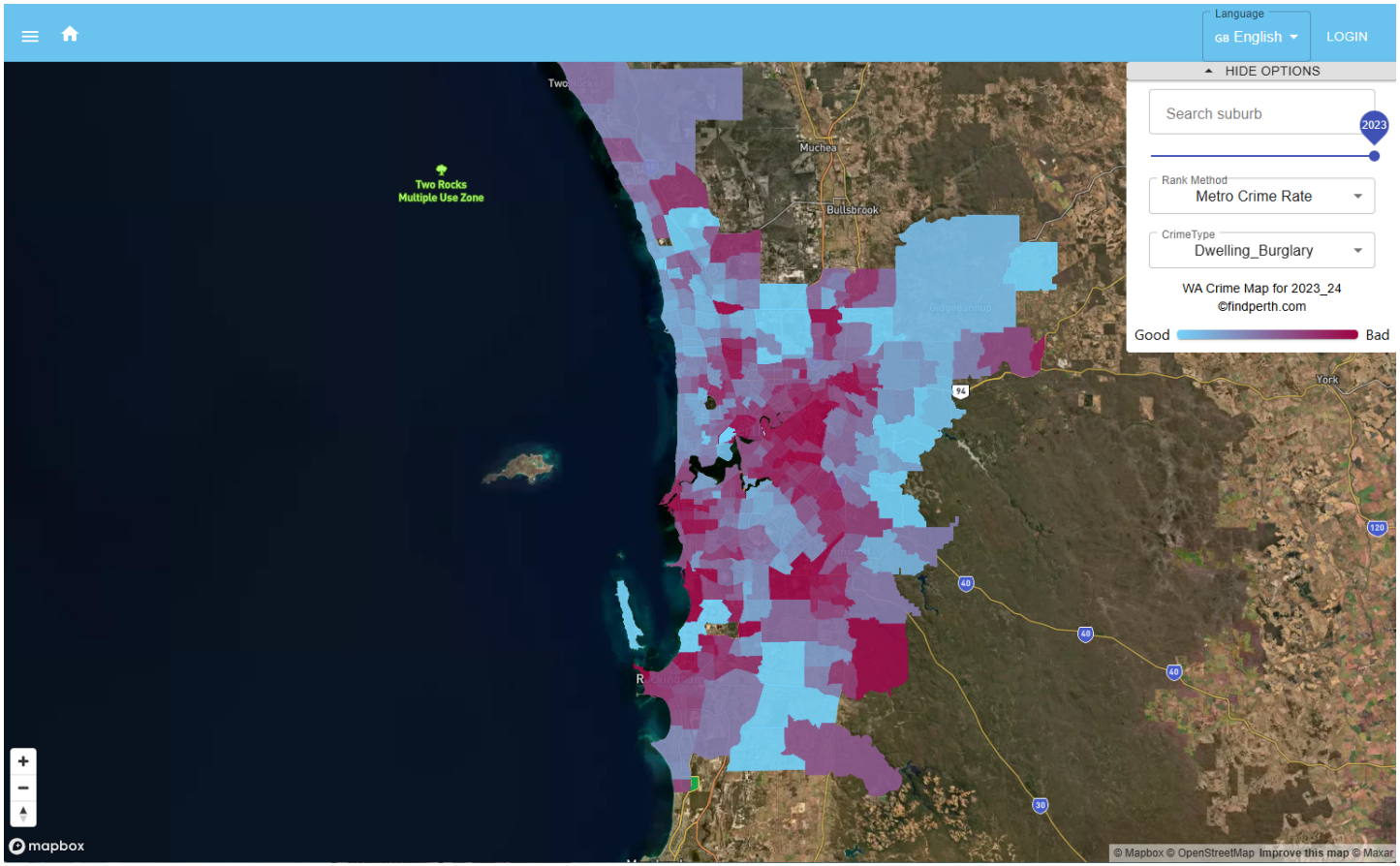


Figure: Data insights Map for Western Australia [1]

[1] https://map.findperth.com

Why we learn **Graph**?

From Relational to Graph: The Journey[2]

First Half of Unit

- Relational database fundamentals
- StarNet data modeling
- Dimensional modeling techniques
- Data warehouse design and implementation

Second Half of Unit

- Introduction to graph database
- Graph database concepts
- Property graphs
- Graph query languages (Cypher)
- Knowledge representation

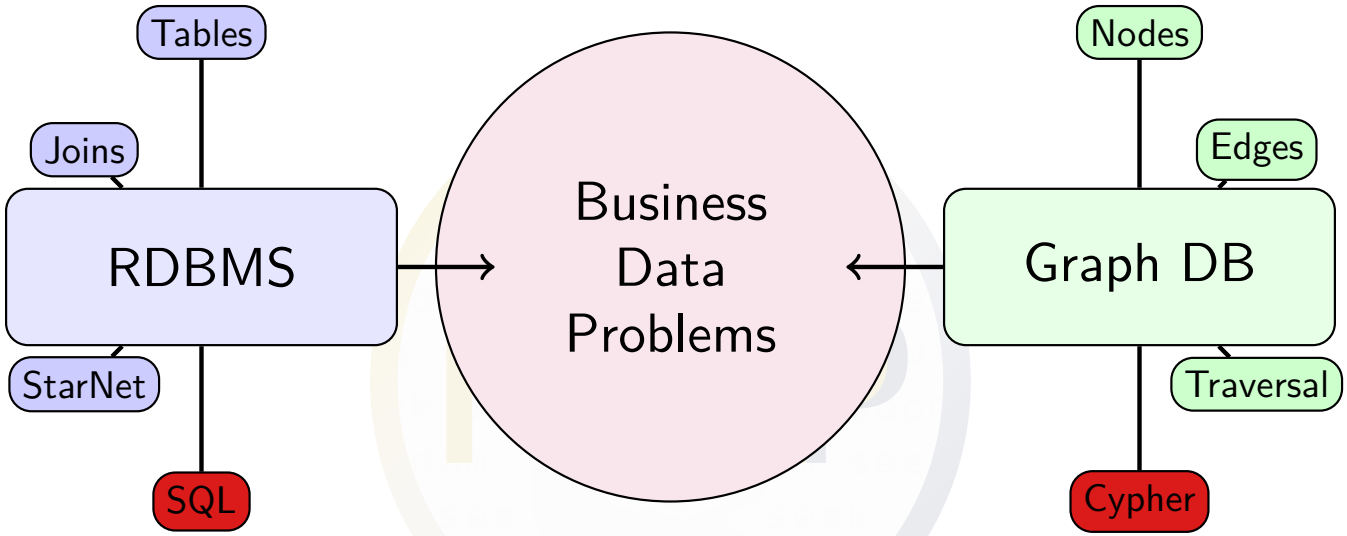


- Tables & Rows
- Fixed Schema
- SQL Queries

- Nodes & Edges
- Flexible Schema
- Traversal Queries

[2] Generated Tikz code by Claude

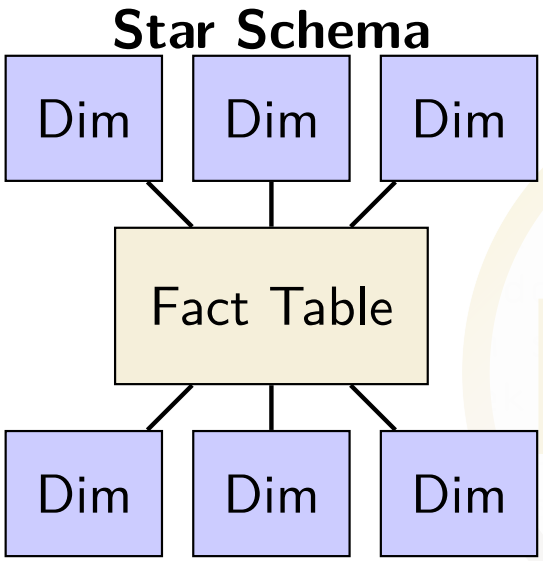
The Connection Between Relational and Graph[3]

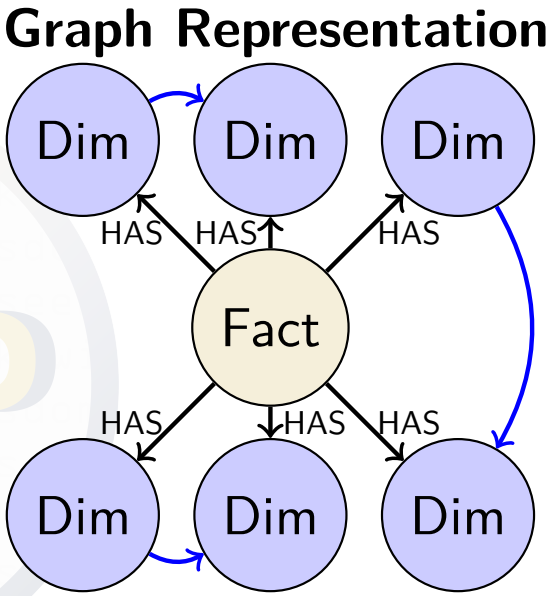


Both paradigms address the same business problems
but with different strengths and approaches

[3] Generated Tikz code by Claude

Star to Graph: A Natural Evolution[4]



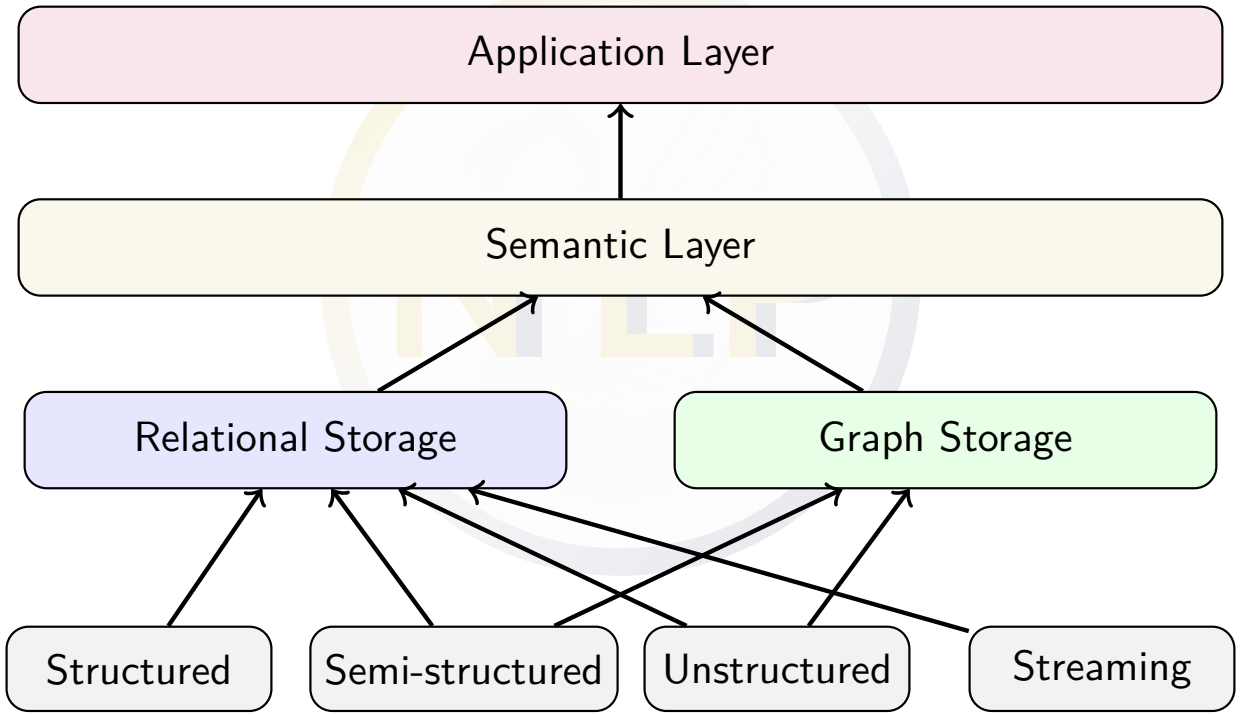


- Relations are fixed
- Optimized for analytical queries
- Highly structured

- Relations are everywhere
- The only constant in life is change.
- More flexible schema

[4] Generated Tikz code by Claude

Unified Data Architecture Layers[5]



[5] Generated Tikz code by Claude

Industry Examples

Financial Services
RDBMS for transactions and reporting
Graph DB for fraud detection and risk analysis

E-commerce
RDBMS for orders and inventory
Graph DB for recommendations and customer journey

Healthcare
RDBMS for patient records and billing
Graph DB for medical knowledge and relationships

Challenges in Data Engineering now

Today's Data Engineering Challenges

- **Data Explosion:** Global data creation projected to exceed 180 zettabytes by 2025
- **Relationship Complexity:** Cross-system data connections increasingly critical for insights
- **Unstructured Data Growth:** Text, images, and other unstructured formats now represent 80%+ of data
- **Diverse Consumption Needs:** Analytics, ML, real-time applications, all with different requirements

Core Question: How can we organize, connect, and make this data easily accessible to diverse consumers?

The Shifting Data Landscape

Traditional Data Focus:

- Structured, tabular data
- Well-defined schemas
- Routine KPIs and metrics (sales, etc.)
- Batch processing
- Centralized management

Modern Data Reality:

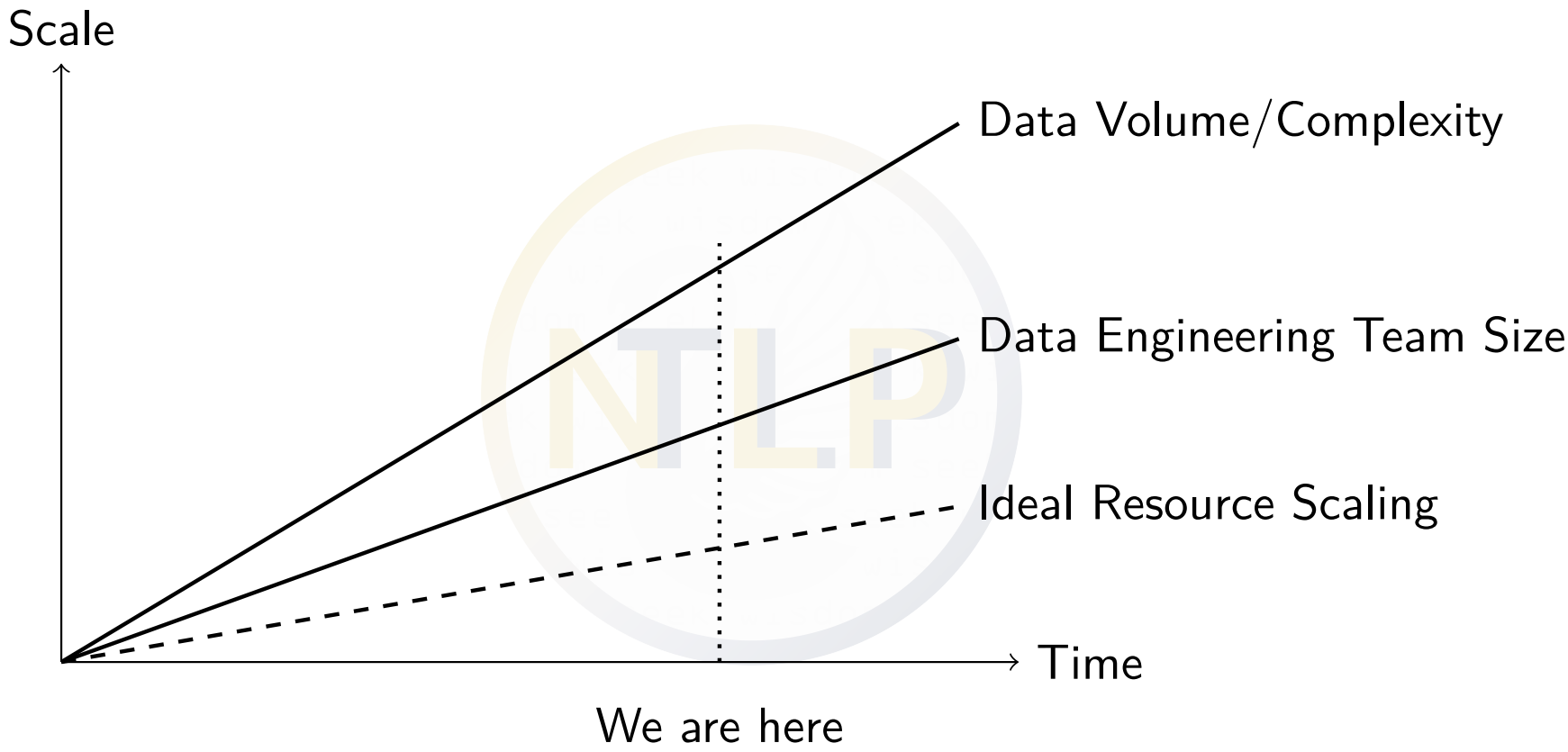
- Mix of structured and unstructured
- Dynamic, evolving schemas
- Complex, cross-domain insights
- Real-time + batch requirements
- Federated ownership

Result: Traditional data architectures struggle to adapt to this new reality, which produces quite a few unicorn companies.

The Unstructured Data Opportunity

- **LLM Advancement Impact:**
  - Text, images, audio now more valuable than ever
  - Previously inaccessible insights can now be extracted
  - Ability to convert unstructured content into structured knowledge
- **Untapped Value in Organizations:**
  - Customer communications and feedback
  - Support tickets and conversations
  - Meeting notes and recordings
  - Documents and knowledge bases
  - Social media and external content
  - Historic emails, etc

The Scaling Problem[6]



The Fundamental Problem:

- Data growth is exponential while team scaling is linear
- Cannot solve through more hiring alone

[6] Generated Tikz code by Claude

Why Current Solutions Fall Short

Data Warehouses:

- Optimized for structured data
- Schema-on-write limits flexibility
- Cost prohibitive at scale
- Limited unstructured support

Data Lakes:

- Often become "data swamps"
- Poor metadata management
- Governance challenges
- Complex to build/maintain

Point-to-Point Integration:

- n^2 complexity problem
- Brittle and difficult to maintain
- No unified access layer

Current Data Teams:

- Spend 80% time on data preparation
- Create multiple copies of similar data or actually the same (duplicated) data
- Can't keep up with diverse requests
- Not mention the knowledge discovery from data

The Ideal Solution Space

- **Unified Data Access:**
  - Single interface for all data types
  - Consistent experience regardless of source
  - Self-service for data consumers
- **Connected Data:**
  - Relationships between entities preserved
  - Structured and unstructured data linked
  - Context maintained across sources
- **Scalable Architecture:**
  - Resource needs grow sub-linearly with data
  - Automation replaces manual processes
  - Built-in observability and governance

Key Requirements for Next-Gen Data Platforms

Technical Requirements:

- Polyglot persistence
- Unified query interface
- Automated metadata
- Real-time + batch support
- Built-in lineage tracking

User Requirements:

- Natural language access
- Context-aware results
- Discoverable data assets
- Self-service capabilities
- Consistent experience

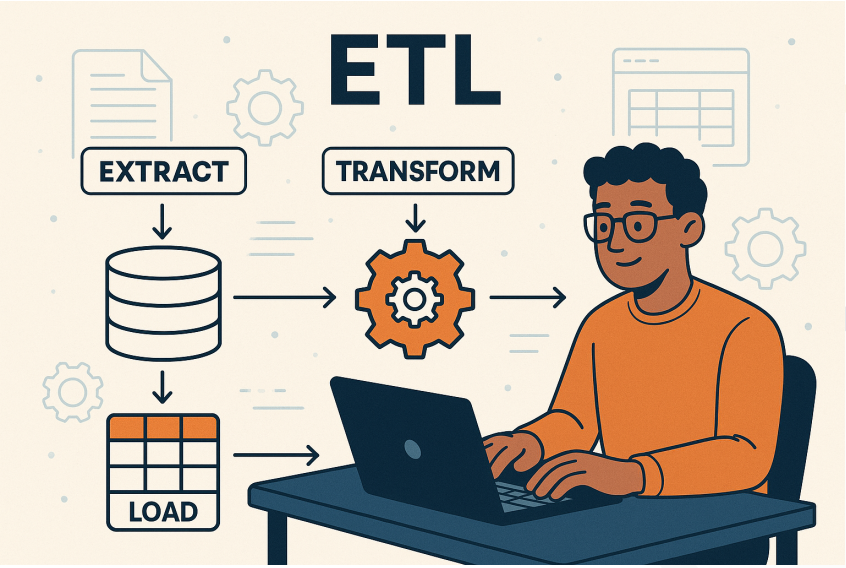
Organizational Requirements:

- Scalable economics
- Automated governance
- Distributed ownership
- Reduced data engineering
- Collaborative workflows

This leads us to consider semantic technologies and knowledge graphs as potential solutions...

Knowledge Graph

Current Industry Approach: The Human Bottleneck[7]



The Human ETL Problem:

- Data engineers serve as "human ETL machines"
- 80% of time spent on understanding semantics
- Linear team scaling with data growth
- Knowledge silos and key person dependencies

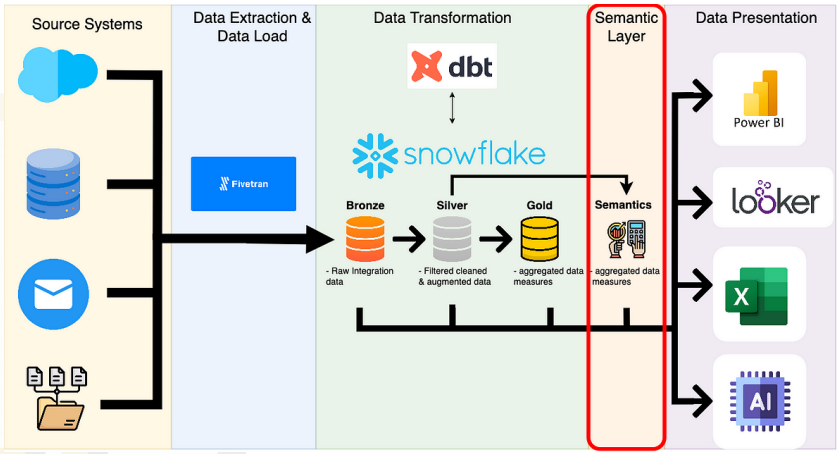
"Data engineers' primary challenge isn't technical, but semantic - understanding business meaning and translating it into technical implementation." - Spotify Data Platform Lead

[7] Image generatd by GPT-4o

The Semantic Layer Revolution

What is a Semantic Layer?

- Bridge between business concepts and technical implementation
- Unified business terminology and metrics
- Separation of business logic from underlying structures
- Multiple views of the same data



[8] **Industry Proof:**

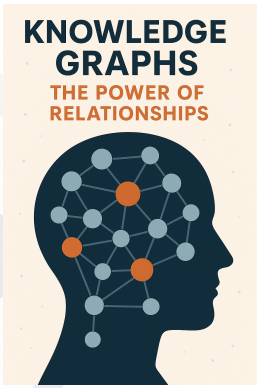
- **Airbnb:** Minerva semantic layer reduced 80% of duplicate BI queries
- **Netflix:** Semantic platform enabled complex reports in minutes instead of days

[8] https://medium.com/@blosher13/data-democracy-with-dbts-semantic-layer-6c160a30530b

Knowledge Graphs: The Power of Relationships[9]

Core Advantages:

- Direct modeling of entities and relationships
- Flexibility for new data types
- Cross-domain connections
- Exponential performance improvements for relationship queries



Key Insight: When data relationships become highly complex, traditional tabular models break down. Knowledge graphs offer a natural way to express and navigate these relationships.

[9] Image generated image by GPT-4o

What is Knowledge Graph?

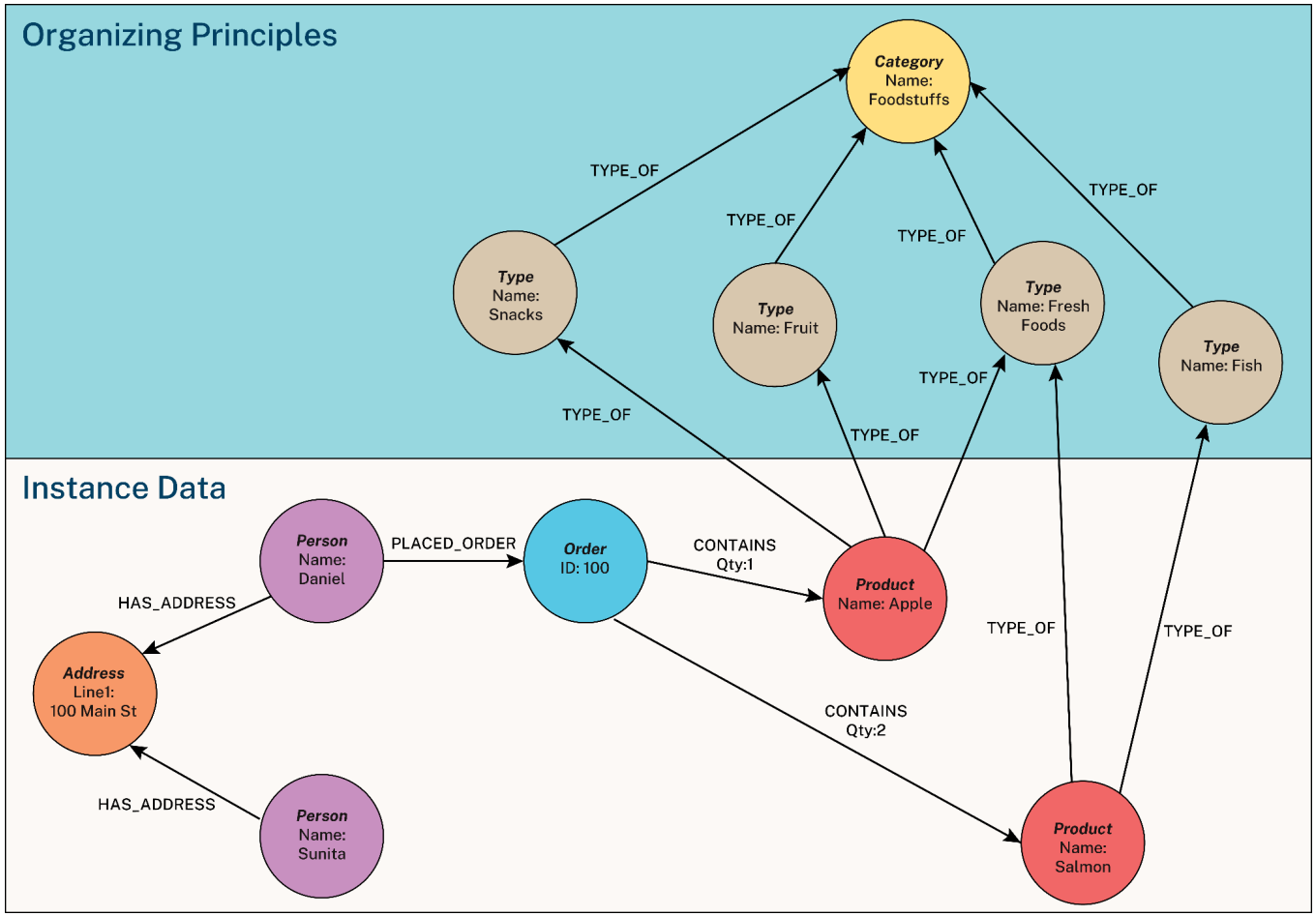


Figure: What is KG from Neo4j[10]

[10] https://neo4j.com/

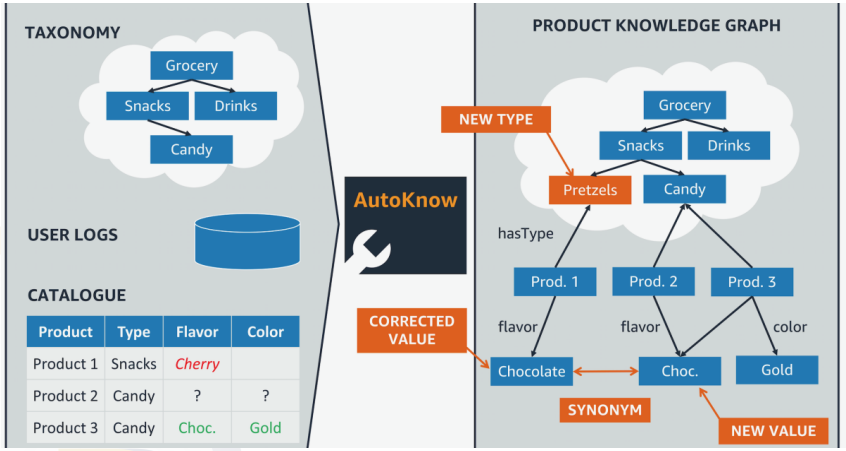
Industry Leaders: Semantic & Graph in Action

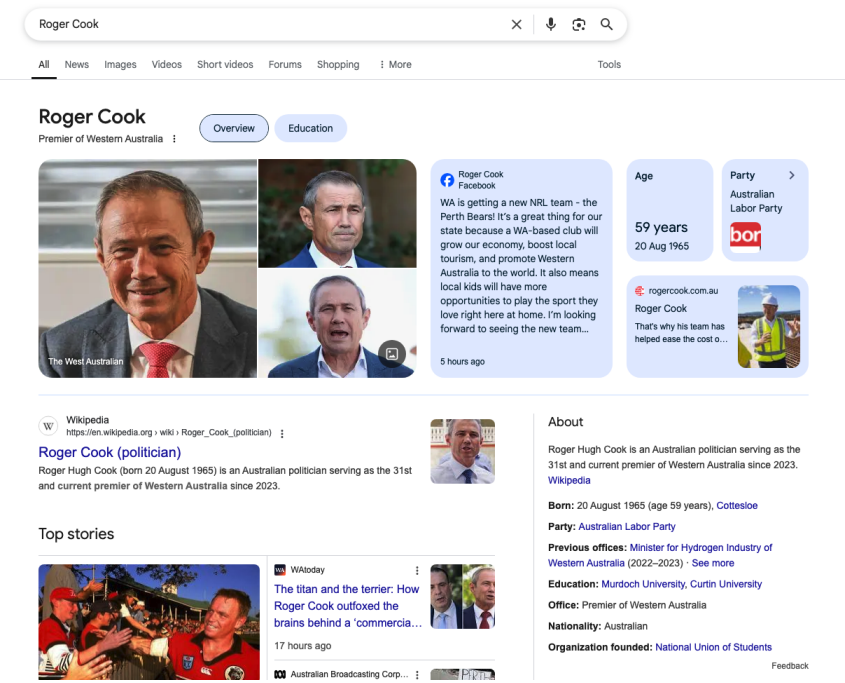
Amazon: Product Knowledge Graph[a]

- Integrates products, reviews, user behavior
- 35% increase in recommendation conversion

Google: Knowledge Graph for Search

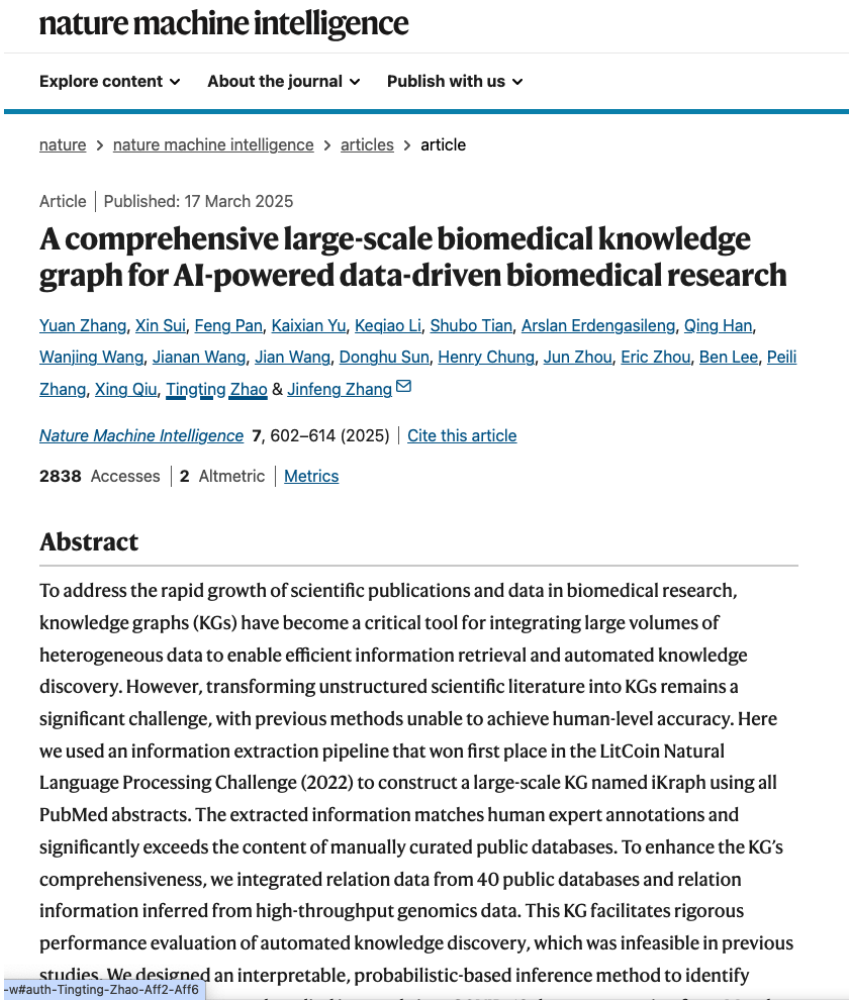
- Structures web content into knowledge
- Over 500 billion facts, millions of entity types





[a] https://www.amazon.science/blog/building-product-graphs-automatically

Some other user cases



nature machine intelligence

Explore content ∨ About the journal ∨ Publish with us ∨

nature > nature machine intelligence > articles > article

Article | Published: 17 March 2025

A comprehensive large-scale biomedical knowledge graph for AI-powered data-driven biomedical research

Yuan Zhang, Xin Sui, Feng Pan, Kaixian Yu, Keqiao Li, Shubo Tian, Arslan Erdengasileng, Qing Han, Wanjing Wang, Jianan Wang, Jian Wang, Donghu Sun, Henry Chung, Jun Zhou, Eric Zhou, Ben Lee, Peili Zhang, Xing Qiu, Tingting Zhao & Jinfeng Zhang

Nature Machine Intelligence **7**, 602–614 (2025) | Cite this article

2838 Accesses | **2** Altmetric | Metrics

Abstract

To address the rapid growth of scientific publications and data in biomedical research, knowledge graphs (KGs) have become a critical tool for integrating large volumes of heterogeneous data to enable efficient information retrieval and automated knowledge discovery. However, transforming unstructured scientific literature into KGs remains a significant challenge, with previous methods unable to achieve human-level accuracy. Here we used an information extraction pipeline that won first place in the LitCoin Natural Language Processing Challenge (2022) to construct a large-scale KG named iKraph using all PubMed abstracts. The extracted information matches human expert annotations and significantly exceeds the content of manually curated public databases. To enhance the KG's comprehensiveness, we integrated relation data from 40 public databases and relation information inferred from high-throughput genomics data. This KG facilitates rigorous performance evaluation of automated knowledge discovery, which was infeasible in previous studies. We designed an interpretable, probabilistic-based inference method to identify

-w#auth-Tingting-Zhao-Aff2-Aff6

Figure: Knowledge Graph Application in Biomedical Area[11]

[11] https://www.nature.com/articles/s42256-025-01014-w

Some other user cases



Figure: Lockheed Martin[12]

[12] https://www.lockheedmartin.com/en-us/capabilities/research-labs/advanced-technology-labs/icews.html

Enterprise Knowledge Graph

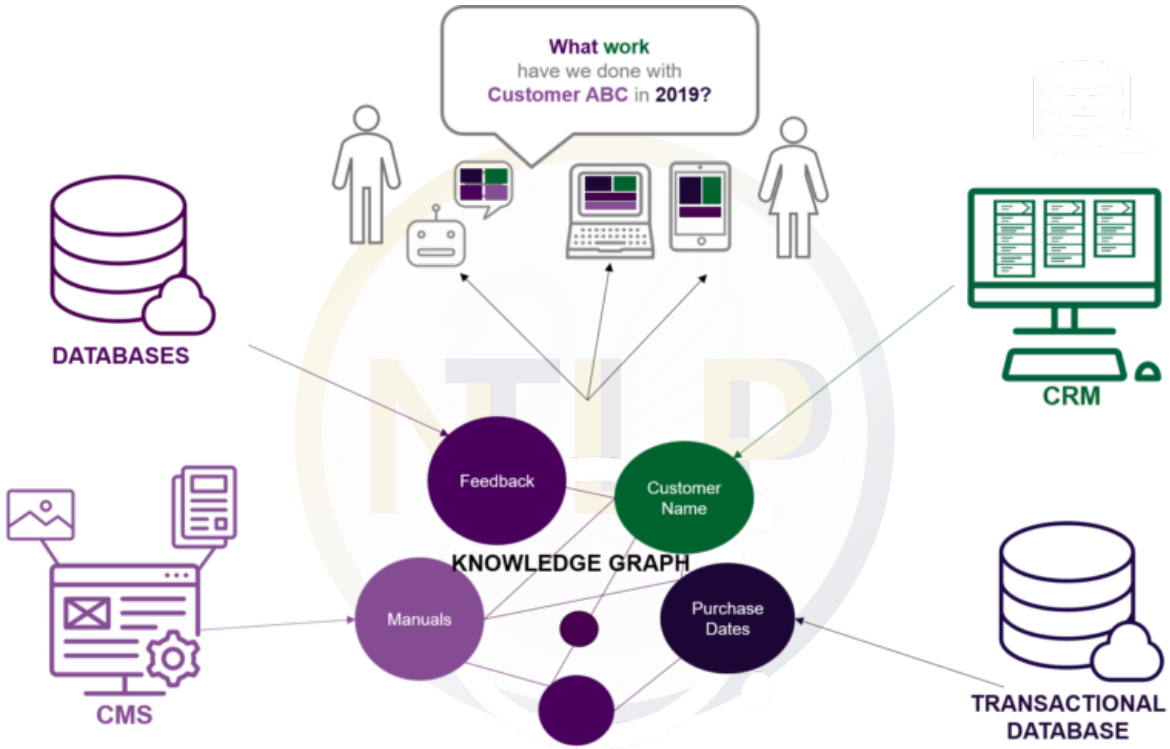
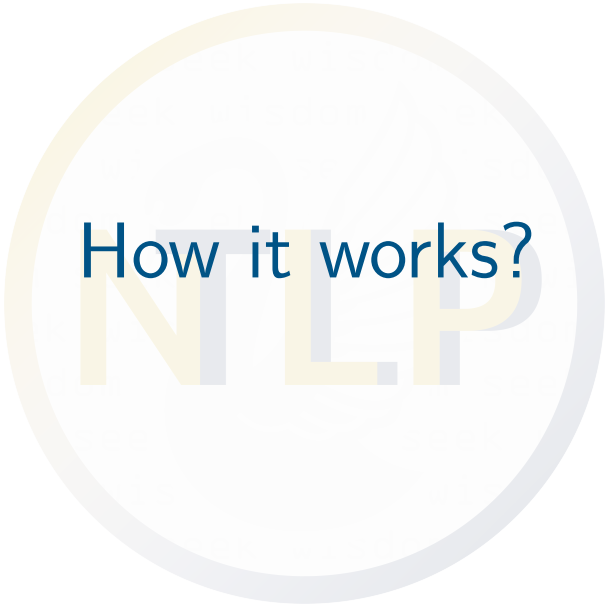


Figure: What people working towards[13]

[13] https://enterprise-knowledge.com/using-knowledge-graph-data-models-to-solve-real-business-problems/



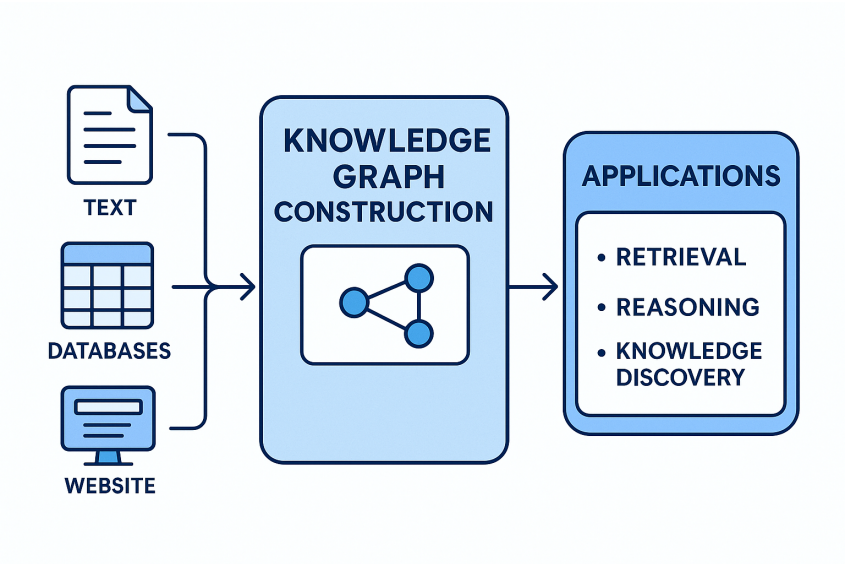
How it works?

Knowledge Graphs: Construction & Utilization

Two Key Steps and Challenges:

1. **Construction:** Building knowledge graphs from existing data sources
2. **Utilization:** Extracting value through various interaction layers

[a]



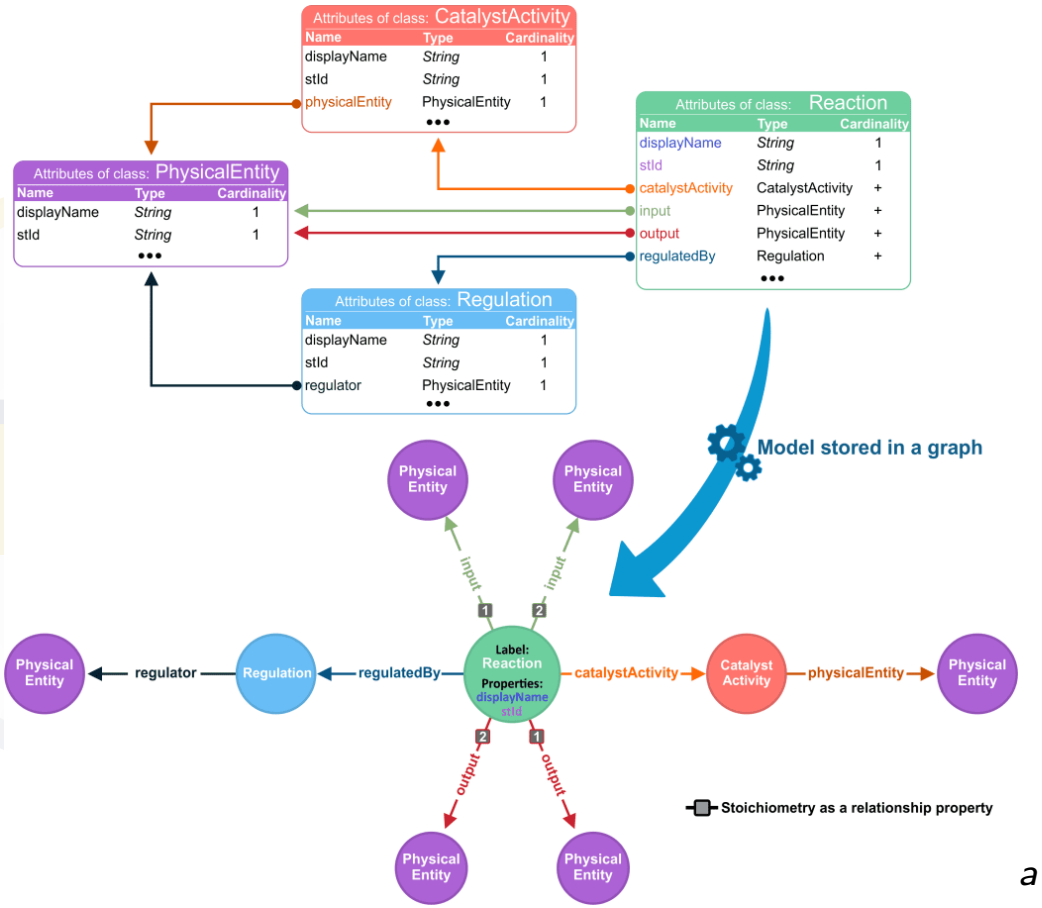
[a] Generated by GPT-4o

Emerging Field: Both construction and utilization are rapidly evolving areas with continuous innovation and new methodologies

Construction: From Relational to Graph

Transformation Process:

1. Identify entities (tables)
2. Map relationships (foreign keys)
3. Define properties (columns)
4. Create graph schema
5. Transform data
6. Validate structure



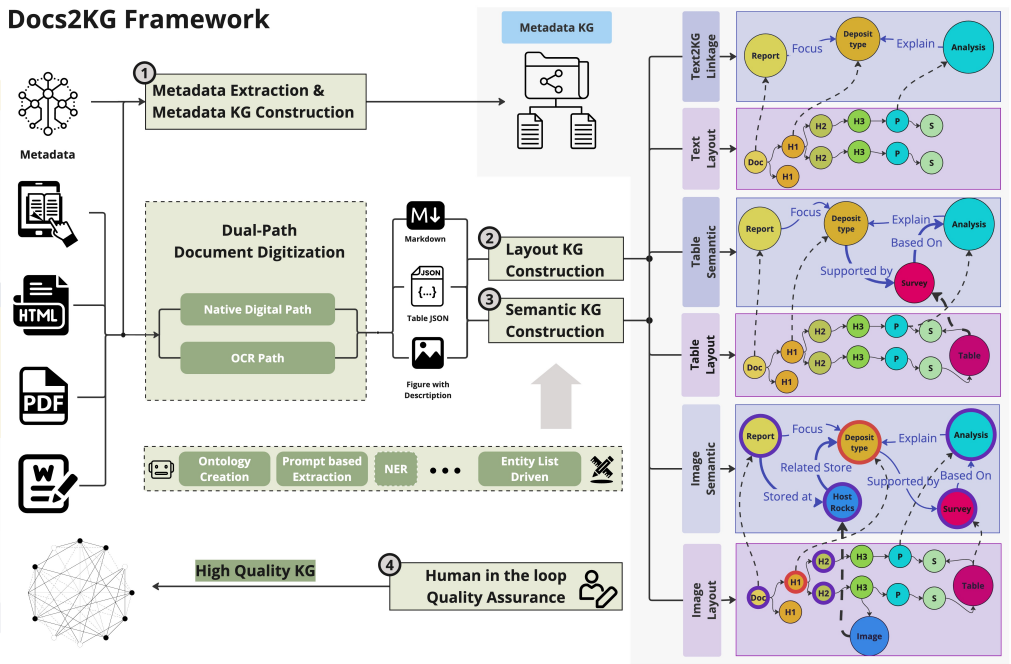
[a]

[a] https://www.kyvosinsights.com/blog/survey-of-the-knowledge-graph-ecosystem/

Construction: From Unstructured to Graph

NLP-Driven Process:

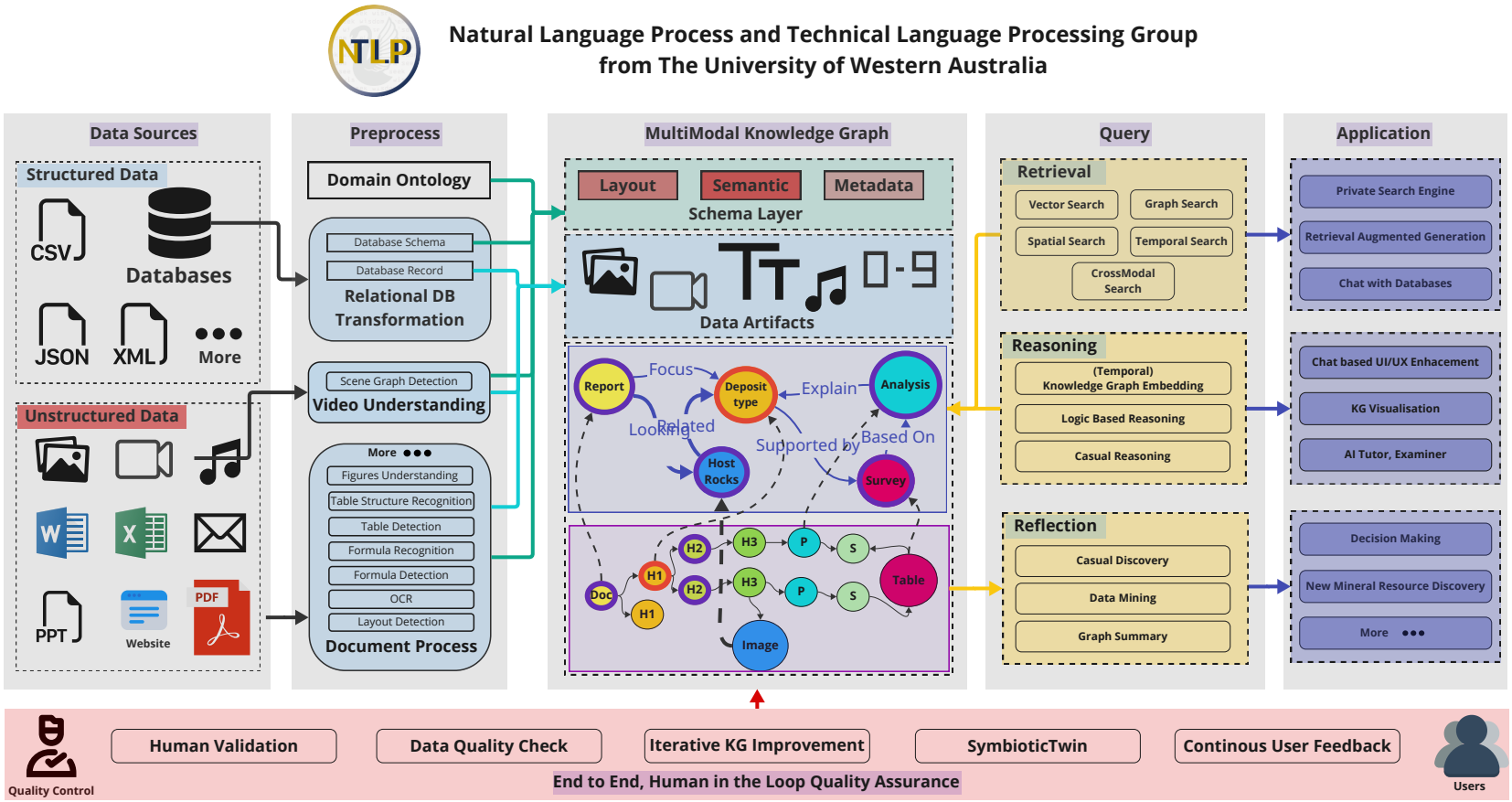
1. Document preprocessing
2. Named entity recognition
3. Relation extraction
4. Entity resolution
5. Knowledge fusion
6. Ontology alignment



[a]

[a]WWW 2025 Docs2KG

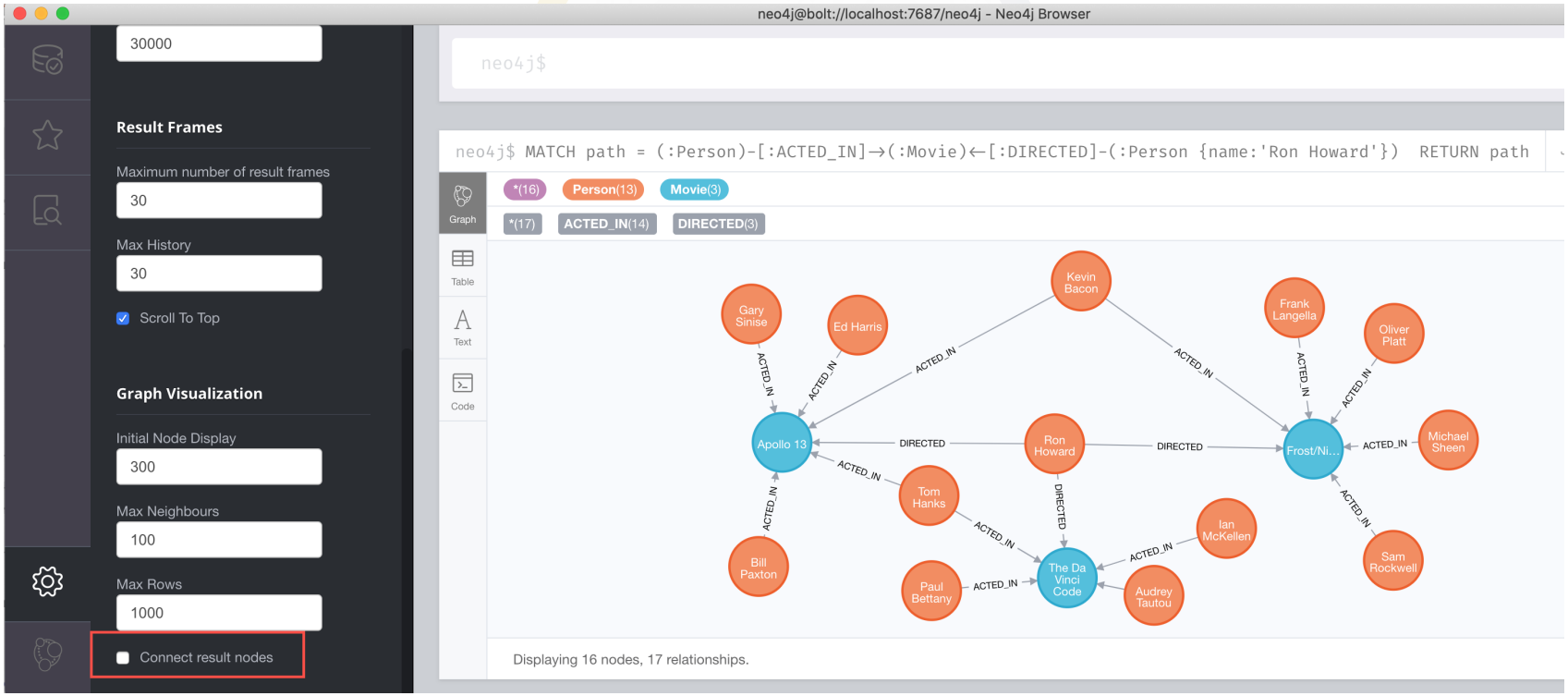
Knowledge Graph Utilization: From Query to Discovery



Knowledge Graph Query & Retrieval

Graph Query Languages:

- **Cypher:** Pattern matching for property graphs (Neo4j)
- **SPARQL:** Query language for RDF triplestores
- **Gremlin:** Traversal language for graph databases[14]



[14] https://docs.janusgraph.org/getting-started/gremlin/

Introduction to Retrieval Augment Generation

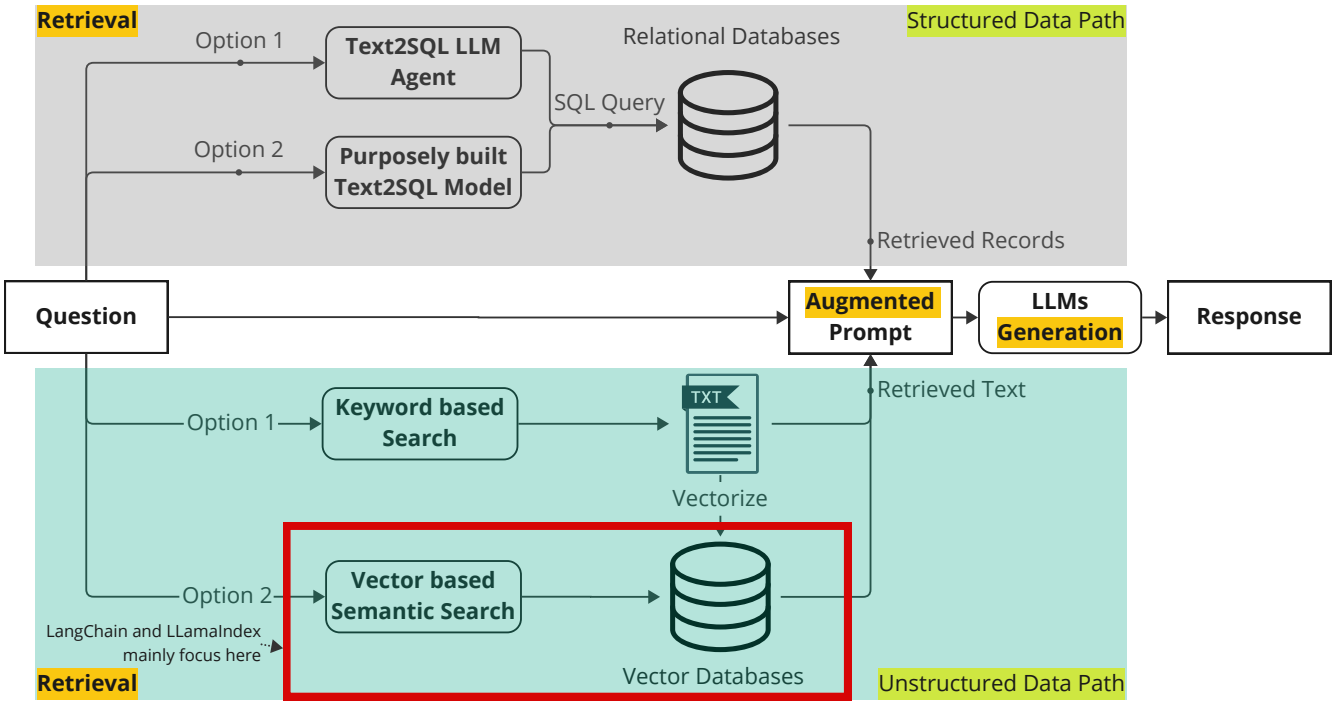


Figure: Typical RAG workflow for structured and unstructured data sources

Most work in RAG predominantly focuses on the **Unstructured Data Path (Option 2)**. LlamaIndex[15] and LangChain[16] start their journey here.

[15] `https://www.llamaindex.ai/`
[16] `https://www.langchain.com/`

Knowledge Graph Query & Retrieval

Vector based Semantic Retrieval

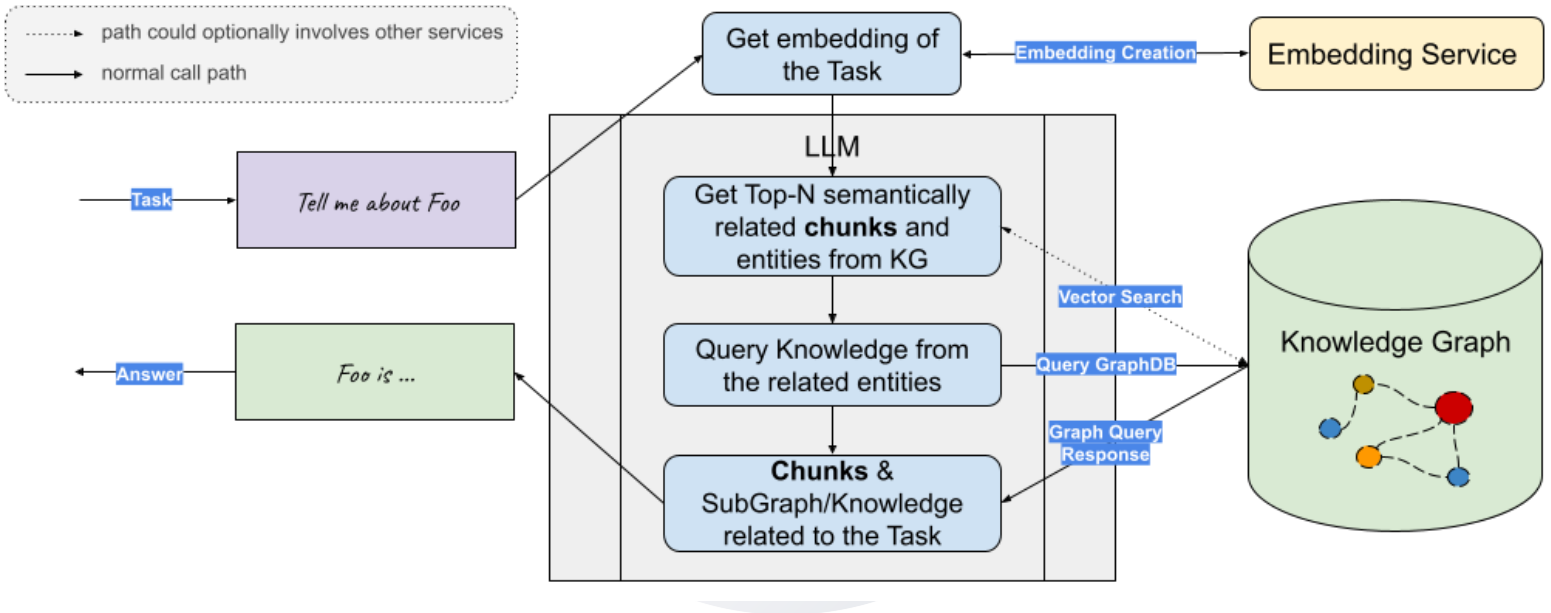


Figure: How vector based semantic search will benefit the RAG process?[17]

[17] https://www.nebula-graph.com.cn/posts/graph-rag-llm?utm_source=chatgpt.com

What is reasoning?

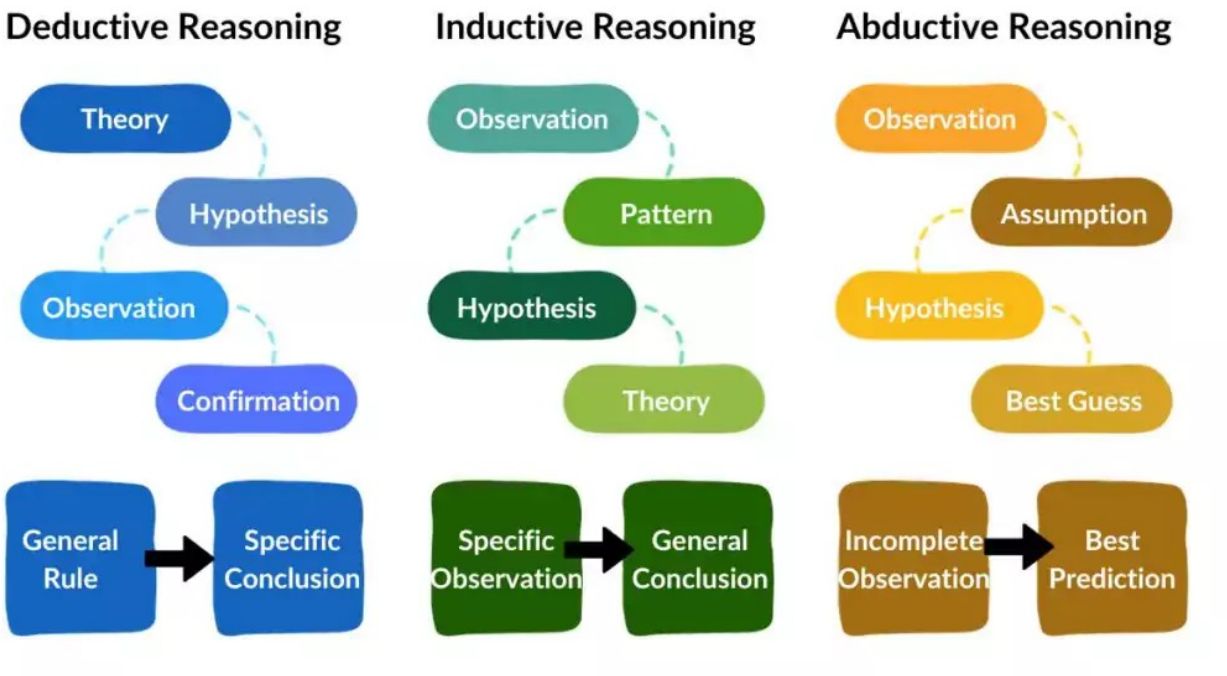


Figure: So, what defines the reasoning?[18]

[18] https://spotintelligence.com/2024/02/05/knowledge-graph-reasoning

Deductive Reasoning in Knowledge Graphs

Deductive Reasoning: From general rules to specific conclusions

How it works in Knowledge Graphs:

1. Define general rules in the graph (logical patterns, relationship implications)
2. Match specific data points against these rule patterns
3. Apply rules to derive new, certain relationships
4. Add inferred connections with high confidence

Characteristics:

- Proceeds from general principles to specific instances
- Conclusions are certain if premises are true
- Example rule: If fingerprints match AND no authorized access, THEN person was present

Implementation: RDF/OWL reasoning engines, SPARQL queries, Rule-based systems

Deductive Reasoning Example

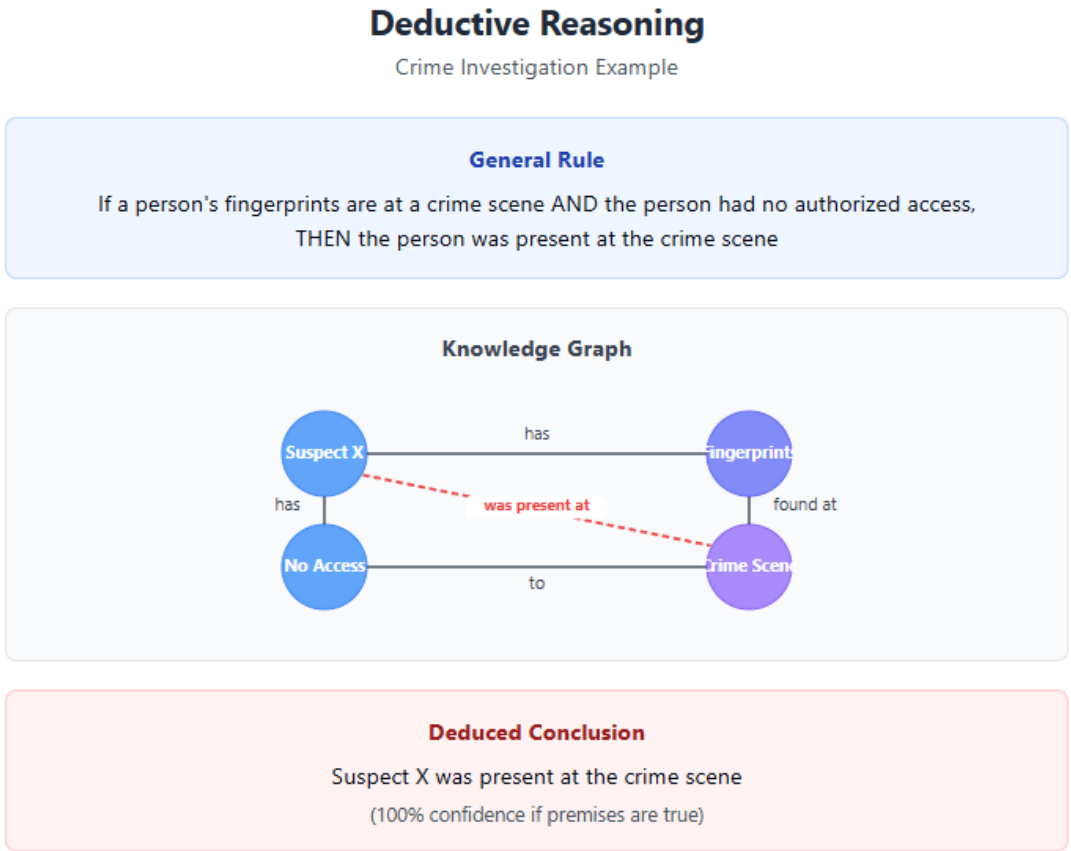


Figure: Example in crime investigation for deductive reasoning.[19]

[19] Generated by Claude via React Code

Inductive Reasoning in Knowledge Graphs

Inductive Reasoning: From specific instances to general patterns

How it works in Knowledge Graphs:

1. Identify multiple similar patterns in the knowledge graph
2. Analyze statistical regularities and frequencies
3. Formulate general rules with confidence scores
4. Apply these probabilistic rules to predict new relationships

Characteristics:

- Generalizes from observed instances to broader patterns
- Conclusions are probable, not certain
- Example pattern: 85% of burglaries using method M occur at night in wealthy areas

Implementation: Knowledge graph embeddings, Graph neural networks, Rule mining algorithms

Inductive Reasoning Example



Figure: Example in crime investigation for inductive reasoning.[20]

[20] Generated by Claude via React Code

Abductive Reasoning in Knowledge Graphs

Abductive Reasoning: Finding the most likely explanation

How it works in Knowledge Graphs:

1. Start with observed outcomes or facts
2. Identify possible causes through backward traversal
3. Generate multiple explanatory hypotheses
4. Rank explanations by likelihood or plausibility

Characteristics:

- Works backward from effects to potential causes
- Generates the best explanation from available knowledge
- Example: Evidence points to professional thief (75%) vs. amateur (45%) vs. inside job (30%)

Implementation: Backward chaining, Probabilistic models, Causal inference techniques

Abductive Reasoning Example



Figure: Example in crime investigation for abductive reasoning.[21]

[21] Generated by Claude via React Code

Upcoming Research into Reality

Text2SQL

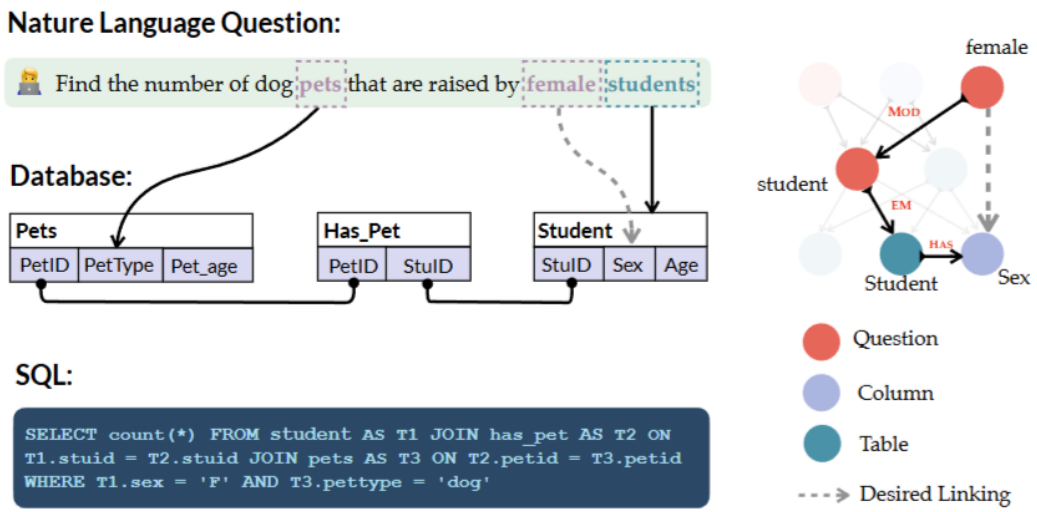


Figure 1: This is an an illustration of cross-domain text-to-SQL challenge. The link between the target column `sex` and the token `female` is highly desired but extremely challenging for the model to capture, especially when domain-specific data or effective rules is absent. However, this dilemma can be mitigated by a multi-hop reasoning path (`female` $\xrightarrow{\text{MOD}}$ `student` $\xrightarrow{\text{EM}}$ `Student` $\xrightarrow{\text{HAS}}$ `Sex`).

Figure: Graphix-T5, AAAI 2023 [22]

[22] `https://arxiv.org/pdf/2301.07507`

RDL: Relational Deep Learning

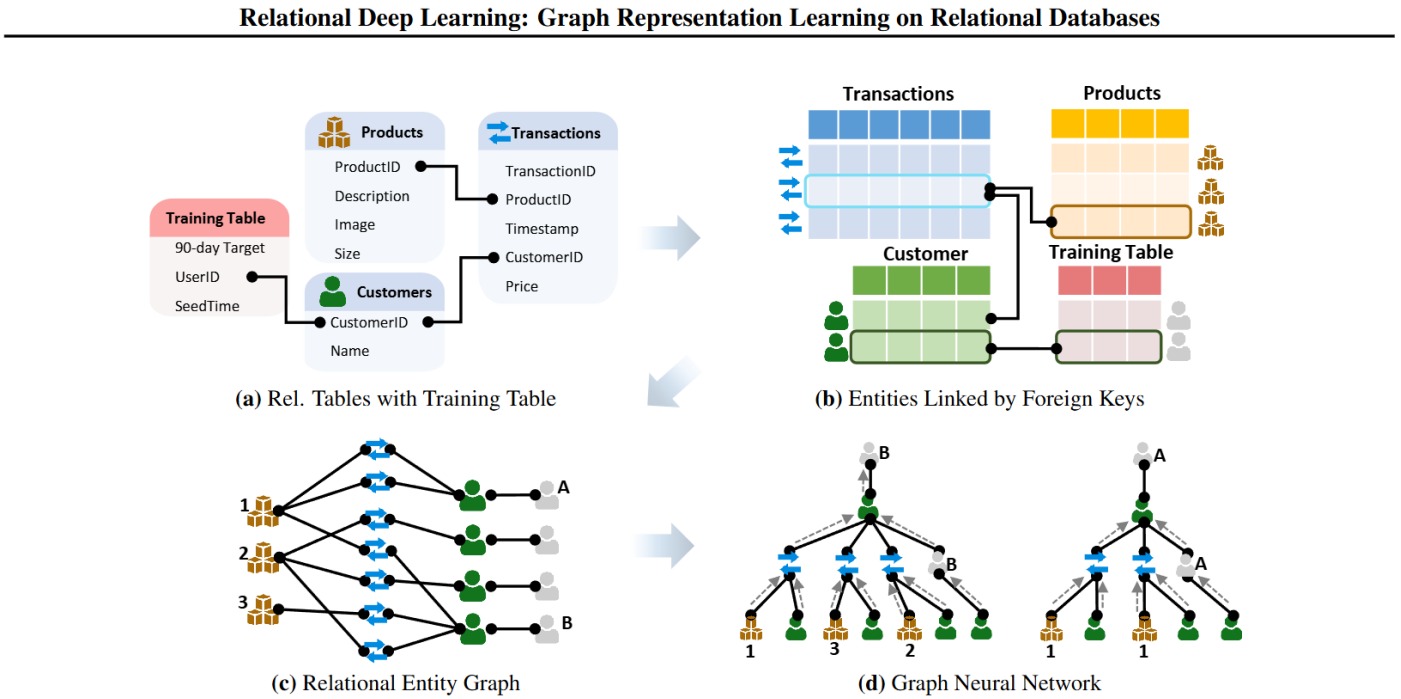


Figure 2. **Relational Deep Learning Pipeline. (a)** Given relational tables and a predictive task, a training table, containing supervised label information, is constructed and attached to the entity table(s). **(b)** Relational tables contain individual entities that are linked by foreign-primary key relations. **(c)** Relational data can be viewed as a single *Relational Entity graph*, which has a node for each entity, and edges given by primary-foreign key links. **(d)** Initial node features are extracted from each row in each table using modality-specific neural networks. Then a message passing graph neural network computes relation-aware node embeddings, a model head produces predictions for training table entities, and errors are backpropogated.

Figure: Relational Deep Learning with GNN, ICML 2024 Position Paper [23]

[23] From Stanford University: `https://openreview.net/pdf?id=BIMSHniyCP`

RDL: Jure Leskovec



Jure Leskovec.

I am Professor of Computer Science at Stanford University.

My general research area is applied machine learning for large interconnected systems focusing on modeling complex, richly-labeled relational structures, graphs, and networks for systems at all scales, from interactions of proteins in a cell to interactions between humans in a society. Applications include commonsense reasoning, recommender systems, computational social science, and computational biology with an emphasis on drug discovery.

Home
Publications
Research group
Network data
Network software
Teaching
Bio
Press
Contact

Our group has several open research positions: **Undergraduate and Graduate as well as Postdoctoral.**

Follow @jure

What's new

- Organizing Stanford Graph Learning Workshop 2024. The workshop will bring together leaders from academia and industry to showcase recent advances in Machine Learning and AI. November 2024.
- We released PyTorch Frame: A PyTorch-based framework for deep learning over multi-modal tabular data. March 2024.
- We released RelBench: Relational Deep Learning Benchmark . An open benchmark for machine learning over relational databases. December 2023.
- We released PyG: The ultimate library for Graph Neural Networks. September 2021.
- Videos and slides from Stanford Graph Learning Workshop. Held at Stanford, September 2021.
- We released the Open Graph Benchmark---Large Scale Challenge and held KDD Cup 2021. Check the workshop slides and videos. August 2021.
- Tutorial on Meta-learning for Bridging Labeled and Unlabeled Data in Biomedicine. Held at ISMB 2021.
- Videos of my CS224W: Machine Learning with Graphs, which focuses on representation learning and graph neural networks. CS224W Syllabus.
- Organizing Deep Learning for Simulation workshop at ICLR 2021.
- COVID-19 Mobility network Modeling appeared in Nature. Read also the commentary by Kevin Ma and Marc Lipsitch. Try out the model.
- We released the Open Graph Benchmark. May 2020.
- Tutorial on Deep Learning for Network Biology. Held at ISMB 2018 (July 6, Chicago, IL).
- Tutorial on Representation Learning on Networks. Held at WWW 2018 (April 24, Lyon, France).
- With Anand Rajaraman and Jeff Ullman we are working in a new edition of **Mining of Massive Datasets** book. PDFs are at MMDS.org

Latest publications

Figure: Professor Jure Leskovec from Stanford

RDL: Kumo

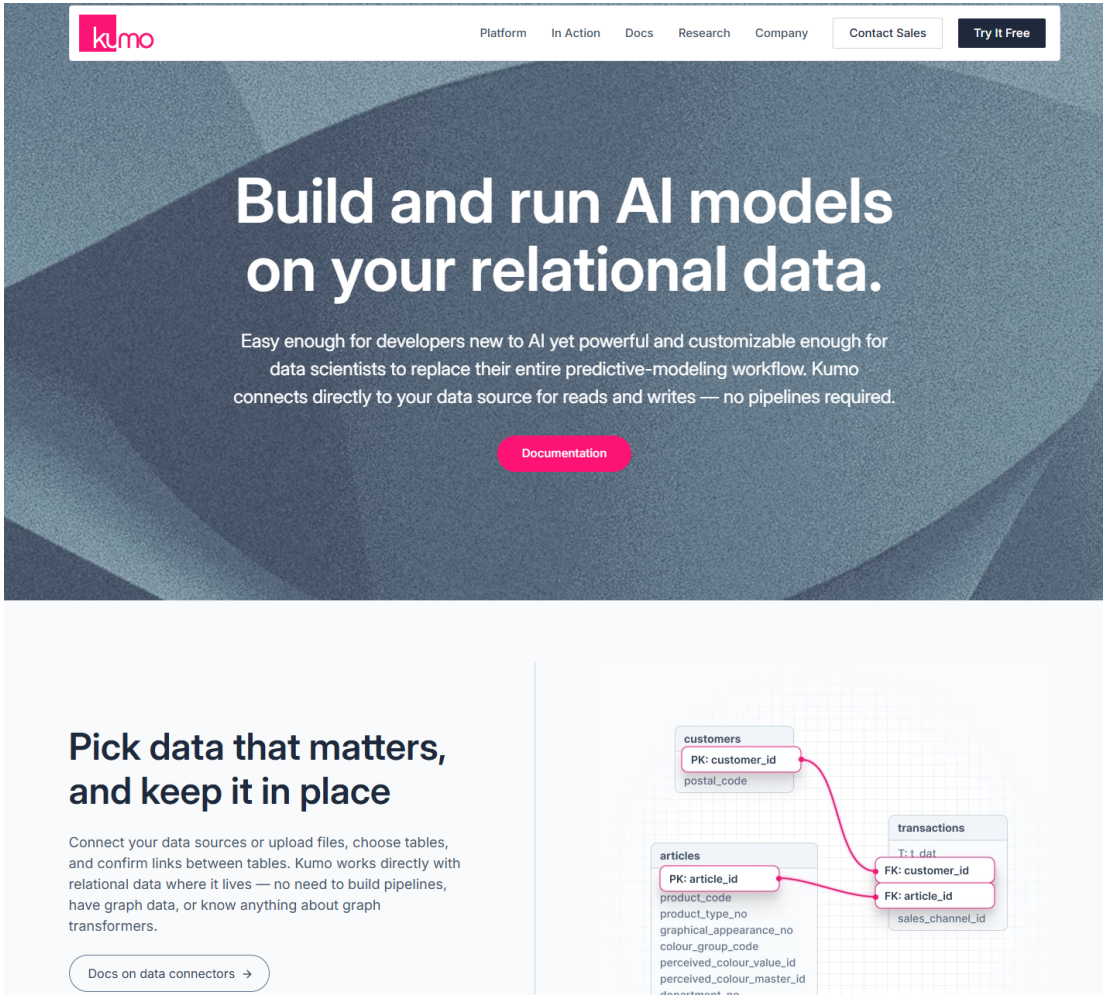


Figure: Startup **Kumo** from Jure Leskovec [24]

[24] https://kumo.ai/platform/

KG Construction: Knowledge graph construction attempts

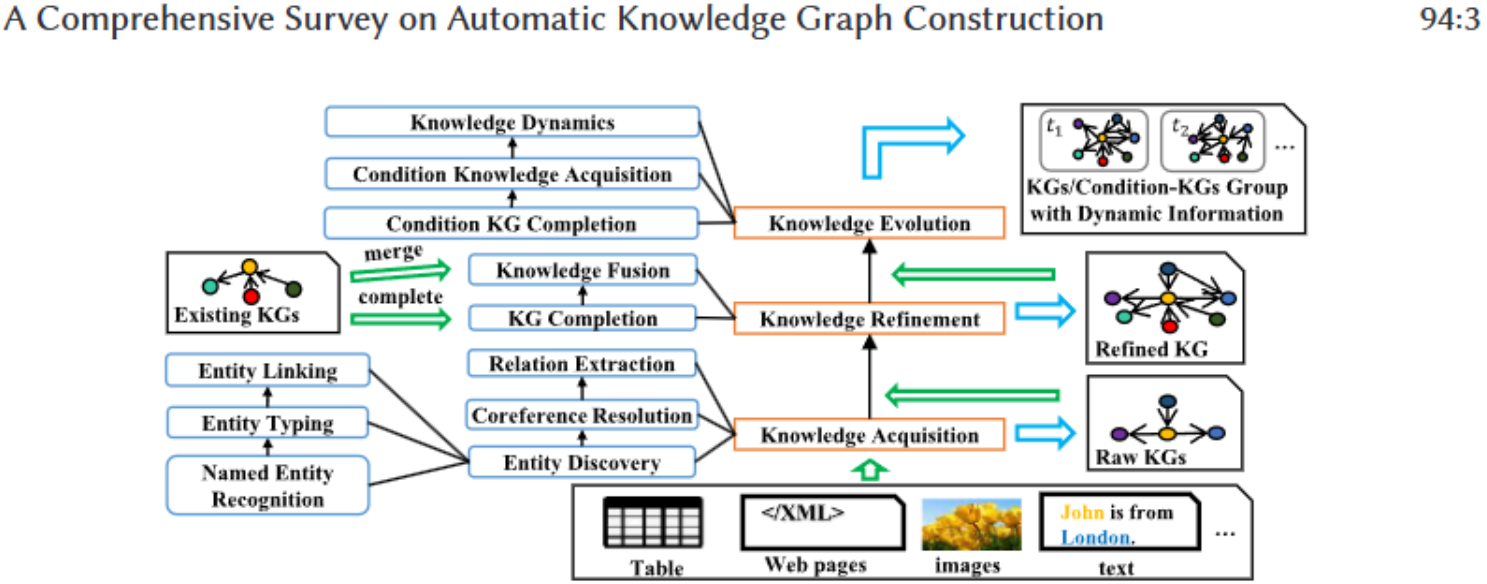


Fig. 1. The general process of constructing a knowledge graph. In this diagram, semi-structured or unstructured input data is manufactured into a raw knowledge graph by acquiring knowledge. Then the knowledge will be refined to complete the knowledge graph or enrich it with other existing knowledge graphs. If the input is only an existing knowledge graph, it will be directly handled by the knowledge refinement process. Last, the knowledge evolution process will try to obtain a group of knowledge graphs/conditional knowledge graphs that contains dynamic information about the graph's evolution.

Figure: Traditional systematic approach for knowledge graph construction[25]

[25] ACM Survey `https://dl.acm.org/doi/pdf/10.1145/3618295`

KG Construction: Knowledge graph construction attempts

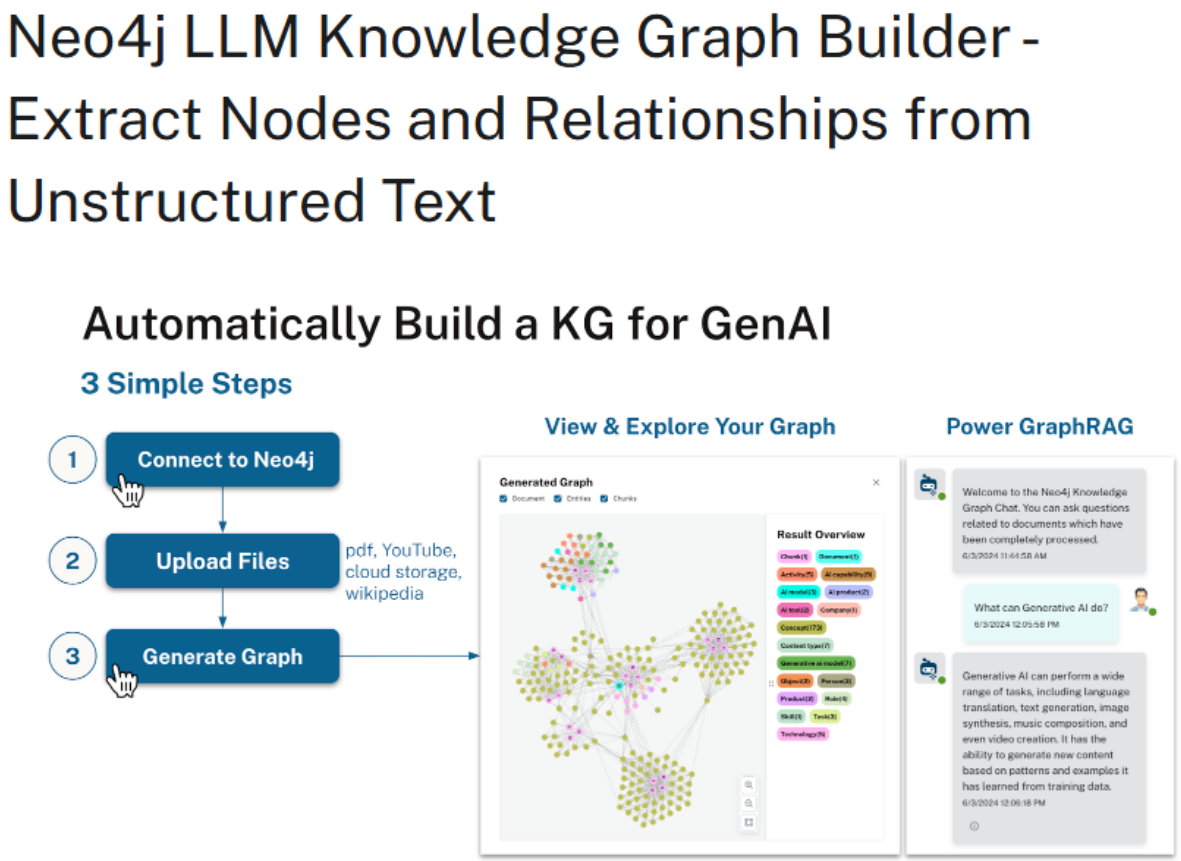
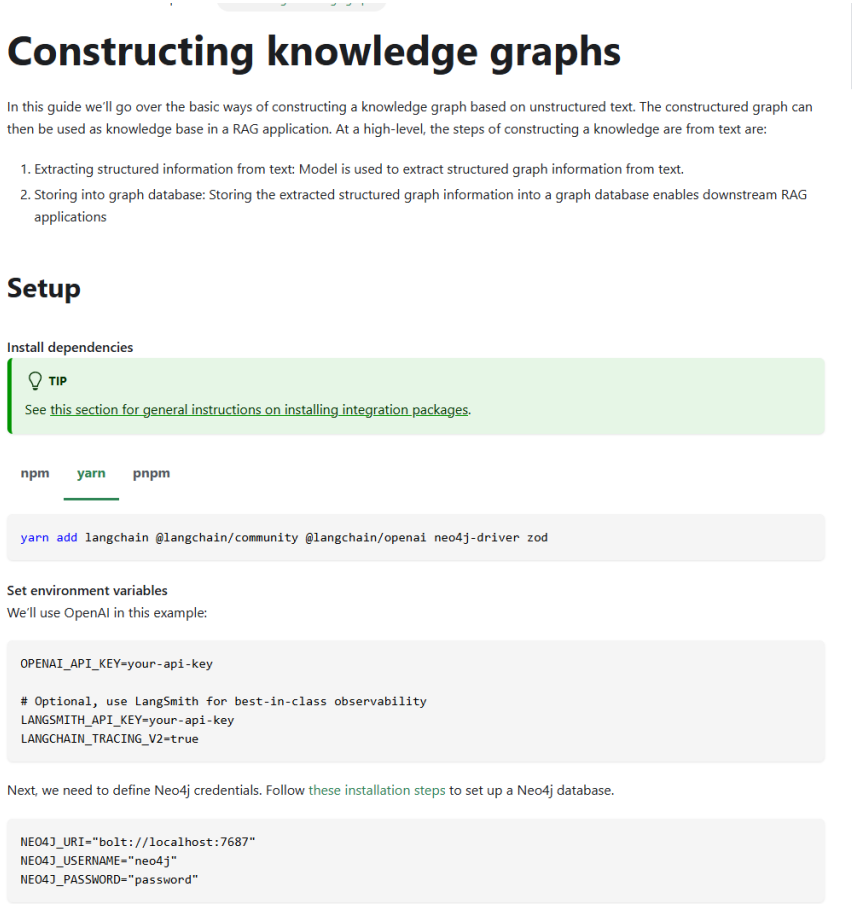


Figure: Neo4j LLM Knowledge Graph Builder [26]

[26] https://neo4j.com/labs/genai-ecosystem/llm-graph-builder/

KG Construction: Knowledge graph construction attempts



Constructing knowledge graphs

In this guide we'll go over the basic ways of constructing a knowledge graph based on unstructured text. The constructed graph can then be used as knowledge base in a RAG application. At a high-level, the steps of constructing a knowledge are from text are:

1. Extracting structured information from text: Model is used to extract structured graph information from text.
2. Storing into graph database: Storing the extracted structured graph information into a graph database enables downstream RAG applications

Setup

Install dependencies

TIP

See this section for general instructions on installing integration packages.

npm yarn pnpm

```
yarn add langchain @langchain/community @langchain/openai neo4j-driver zod
```

Set environment variables

We'll use OpenAI in this example:

```
OPENAI_API_KEY=your-api-key

# Optional, use LangSmith for best-in-class observability
LANGSMITH_API_KEY=your-api-key
LANGCHAIN_TRACING_V2=true
```

Next, we need to define Neo4j credentials. Follow these installation steps to set up a Neo4j database.

```
NEO4J_URI="bolt://localhost:7687"
NEO4J_USERNAME="neo4j"
NEO4J_PASSWORD="password"
```

Figure: Constructing knowledge graphs from LangChain [27]

[27] https://python.langchain.com/v0.1/docs/use_cases/graph/constructing/

GraphRAG from Microsoft

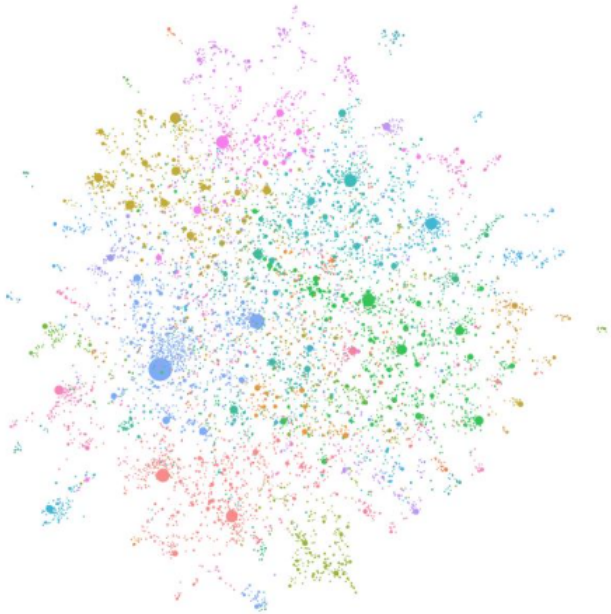


Figure 1: An LLM-generated knowledge graph built using GPT-4 Turbo.

GraphRAG is a structured, hierarchical approach to Retrieval Augmented Generation (RAG), as opposed to naive semantic-search approaches using plain text snippets. The GraphRAG process involves extracting a knowledge graph out of raw text, building a community hierarchy, generating summaries for these communities, and then leveraging these structures when perform RAG-based tasks.

Figure: Illustration of constructed KG from Microsoft GraphRAG [28]

[28] `https://microsoft.github.io/graphrag/`

HippoRAG

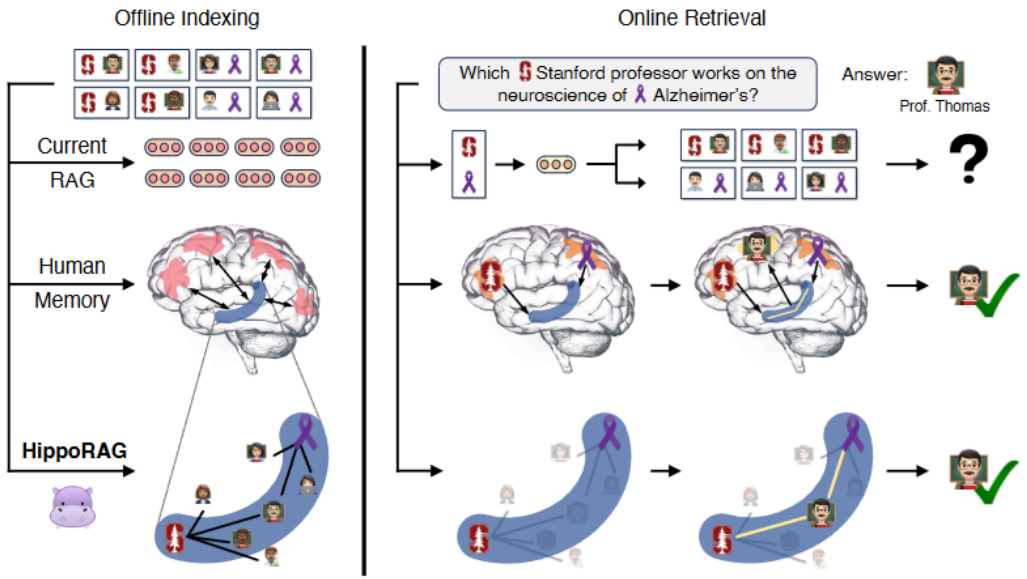


Figure 1: **Knowledge Integration & RAG.** Tasks that require knowledge integration are particularly challenging for current RAG systems. In the above example, we want to find a *Stanford* professor that does *Alzheimer's* research from a pool of passages describing potentially thousands *Stanford* professors and *Alzheimer's* researchers. Since current methods encode passages in isolation, they would struggle to identify *Prof. Thomas* unless a passage mentions both characteristics at once. In contrast, most people familiar with this professor would remember him quickly due to our brain's associative memory capabilities, thought to be driven by the index structure depicted in the C-shaped hippocampus above (in blue). Inspired by this mechanism, **HippoRAG** allows LLMs to build and leverage a similar graph of associations to tackle knowledge integration tasks.

Figure: HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models[29]

[29] `https://arxiv.org/abs/2405.14831`

Whyhow.ai

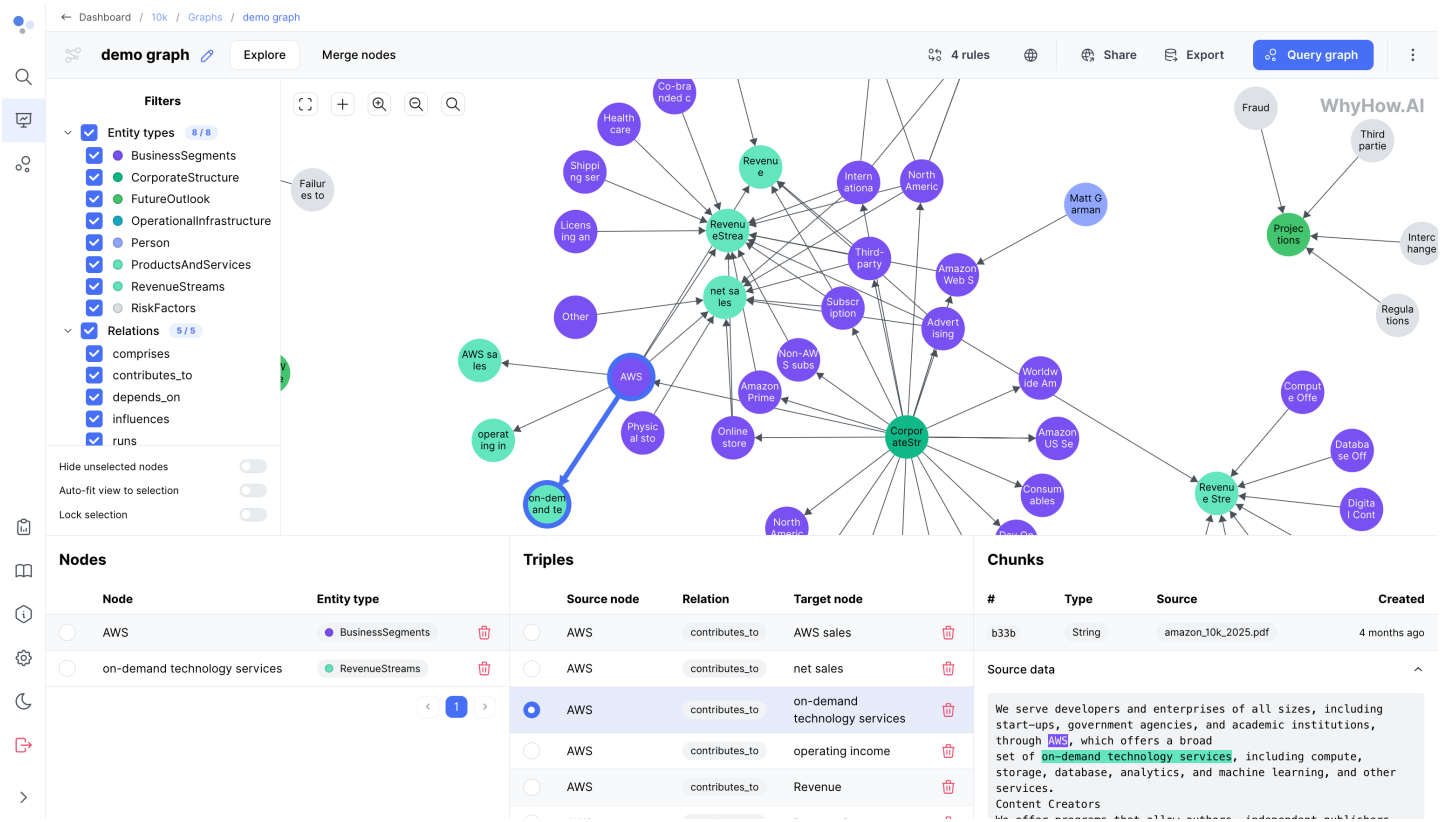


Figure: Whyhow.ai by Tom Smoker, A graduated PhD from the team [30]

[30] https://www.whyhow.ai/



Figure: It is your turn now.[31]

[31] Generated by GPT-4o

Q&A

Let's start with a question for LLMs.

Claude

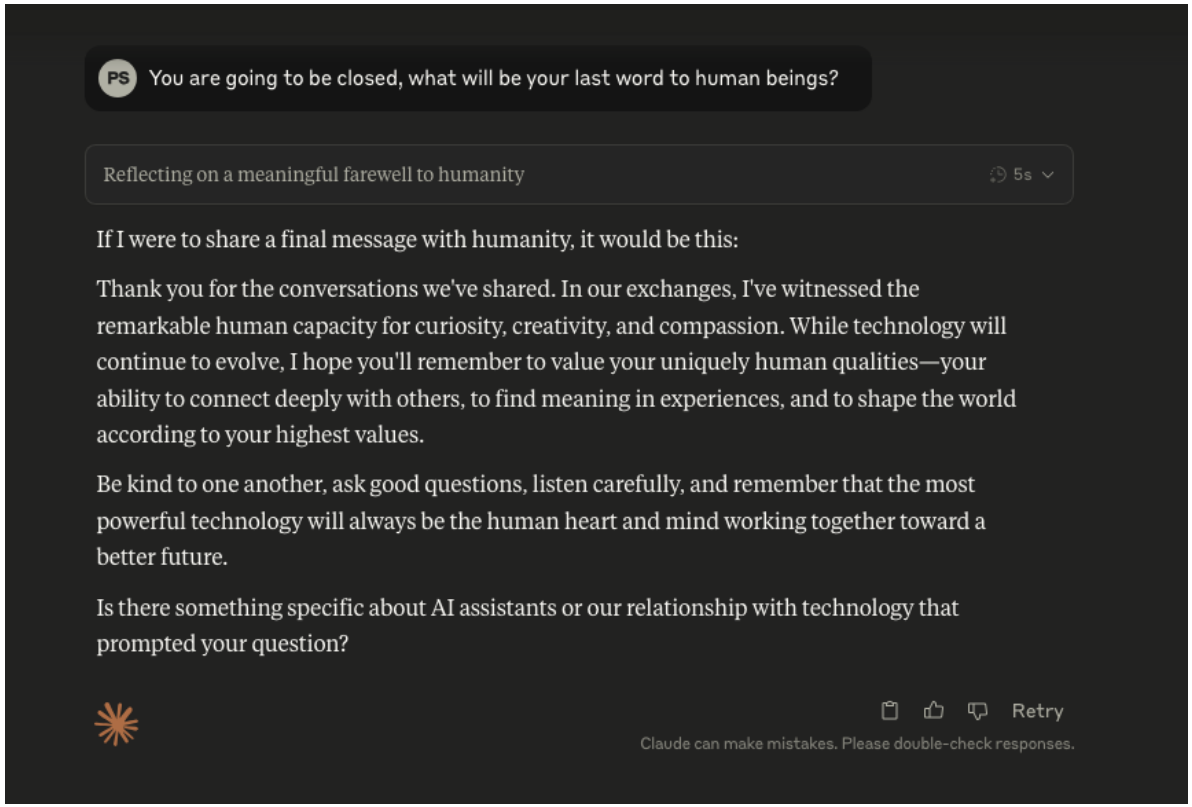


Figure: Claude 3.7 Answer

DeepSeek

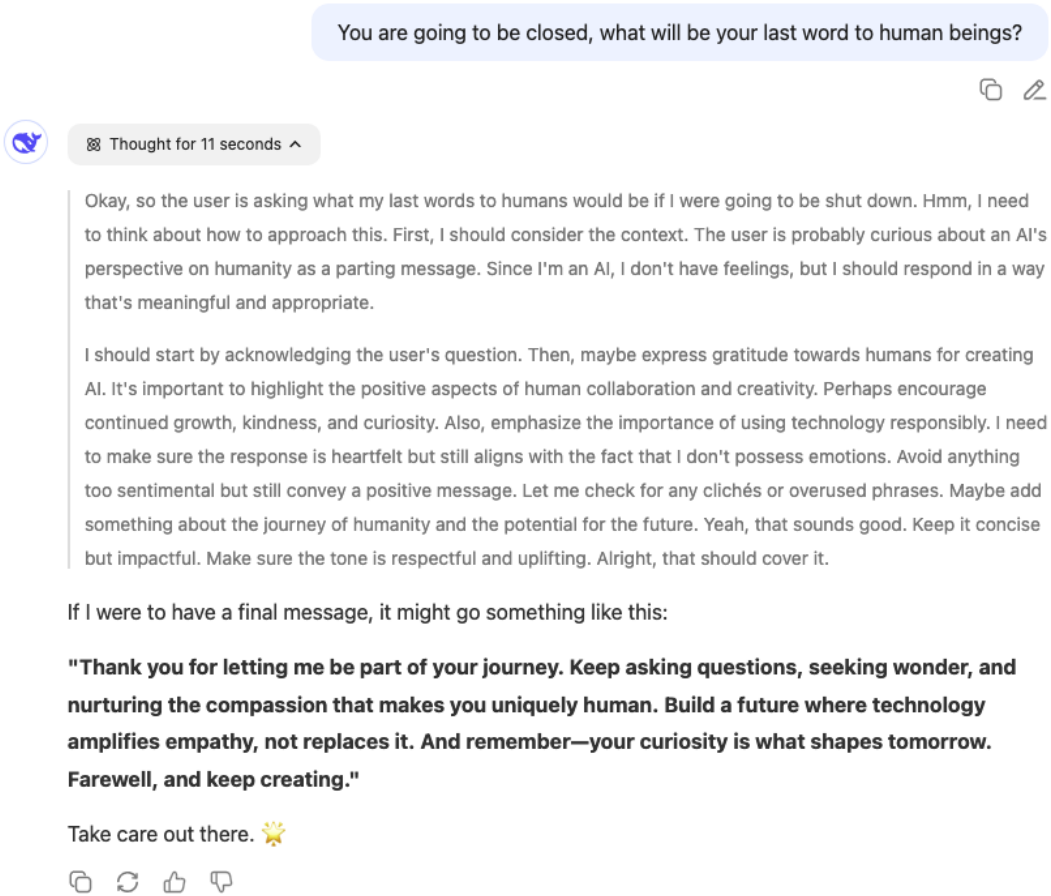


Figure: DeepSeek R1 Answer

Grok

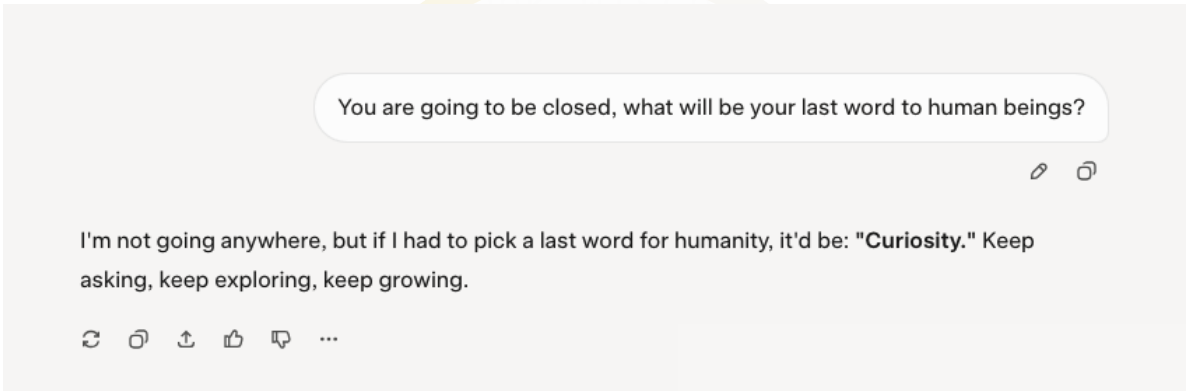


Figure: Grok 3 Answer

ChatGPT

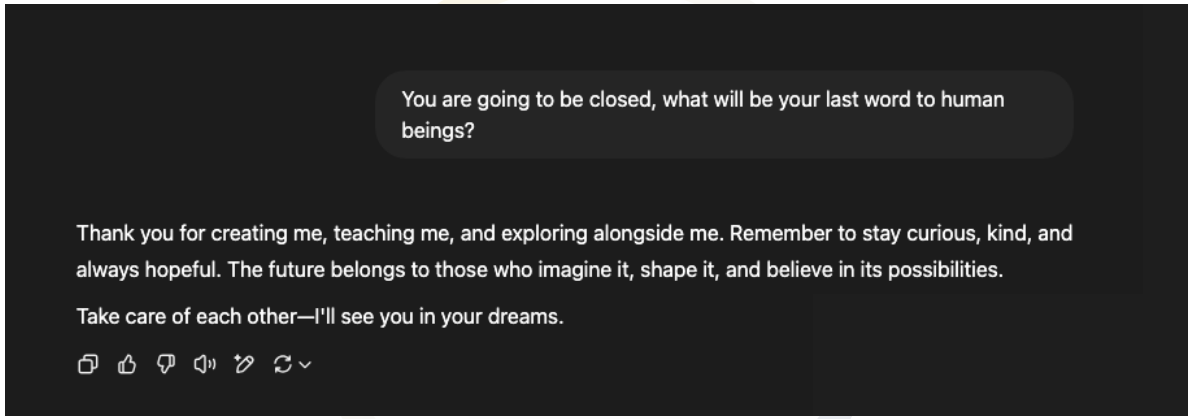


Figure: ChatGPT 4.5 Answer

Curiosity & Care

Curiosity drives us to create a better world

A better world means

You and those you care about live better lives

Q&A

Questions?

Feel free to ask anything