Leveraging Generative AI in Knowledge Graphs and vice versa.

Yogesh Haribhau Kulkarni

YHK

Outline

Introduction

Limitations of . . .

Large Language Models in the Enterprise Context

- Hallucinations: LLMs generate plausible but inaccurate responses.
- Misinformation: LLMs unintentionally perpetuate false information.
- Biases: LLMs exhibit biases from training data, impacting decision-making.
- Cut-off: LLMs lack awareness of events after their training.
- Corpus: LLMs lack knowledge of events not in their training data.
- Fine-tuning mitigates but doesn't eliminate hallucinations.
- LLMs don't cite sources, lack access restrictions.

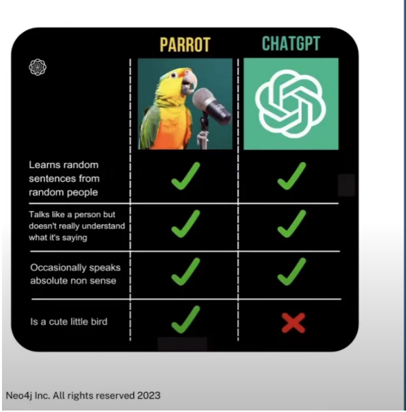
Limitations of . . .

Current AI Reasoning

- LLMs excel in NLP tasks but have limitations in complex reasoning:
  - Lack extensive world knowledge.
  - Implicit reasoning process.
  - Limited mathematical and logical reasoning.
  - Exhibit biases and fallacies.
  - Cannot reason about their own process.
- Human reasoning integrates various forms of inference using structured knowledge.

YHK

Who is Better?



(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

Solution for Limitations

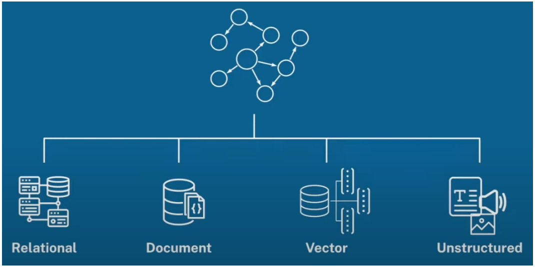
- Fine-tuning: Supervised training phase to optimize LLM performance.
- Use case 1: Updating and expanding internal knowledge.
- Use case 2: Fine-tuning for specific tasks (e.g., summarization, translation).

Knowledge Graphs as Support Mechanisms for LLMs

- Knowledge Graphs: Structured representation using nodes and edges.
- Interface with LLMs: Integrates Knowledge Graphs with LLMs.
- Enables access to structured data.
- Enhances LLM outputs.
- Benefits of Integration: Mitigates LLM limitations.
- Provides accurate, context-specific, unbiased information.

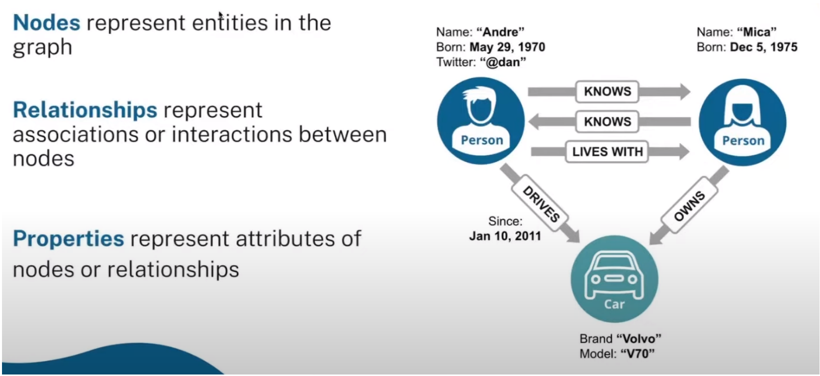
What is a Graph?

- Stock Market chart? Lines plotted on Graph paper?
- Comp Science Data Structure?
- Graph: Nodes connected by edges, concepts connected by relations
- Can you imagine how to write this in code?
- Neo4j is a Graph Database, meaning data is stored in form of graph, and queried by Graph Query language called Cypher.
- As its very generic, its a super-set of all data structures



(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

What is a Property Graph?



(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

What is KG?

- A knowledge graph is a graph-based database that represents knowledge in a structured and semantically rich format.
- This could be generated by extracting entities and relationships from structured or unstructured data, such as text from documents.
- A key requirement for maintaining data quality in a knowledge graph is to base it on standard ontology.
- Having a standardized ontology often involves the cost of incorporating this ontology in the software development cycle.

YHK

Knowledge Graphs for Structured Knowledge

- Knowledge graphs (KGs) overcome knowledge limitations of LLMs:
  - Represent entities and relationships.
  - Conceptual ontology, logical axioms, factual data.
  - Machine-readable semantic representation.
- KGs provide:
  - Extensible semantic fabric.
  - Logical axioms for deduction.
  - Explicit representation of entities.
  - Factual accuracy through provenance.
- KGs have limitations in flexible reasoning and contextual understanding.

YHK

What is LLM?

- Large Language Models (LLMs): Machine learning algorithms for interpreting, translating, and summarizing natural language texts.
- Size: Span hundreds of gigabytes with trillions of parameters.
- Deep neural networks: Learn from extensive training data to generate appropriate outputs.
- Self-attention mechanisms: Capture relationships between words, even with incorrect positioning.
- Transformer architecture: Implements self-attention mechanisms.
- Training data: Accumulated from multiple sources (books, internet, articles, social media, research papers).
- Applications: Conversational AI, content creation engines, search engines, customer service agents, etc. Enhance natural language processing tasks.

Why KG + LLM?

- Reasoning is vital to human intelligence.
- Developing artificial general intelligence requires addressing reasoning challenges.
- Combining Large Language Models (LLMs) with knowledge graphs can enhance AI reasoning.

(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

Why KG + LLM?

- KnowledgeGraphs: Excellent for representing domain data.
- Deliver answers through expert-formulated queries.
- Large Language Models (LLMs): Allow any user to ask questions.
- Retrieve comprehensive answers.
- LLM answers lack user-specific information.

So, combining both will be a Win-Win situation.

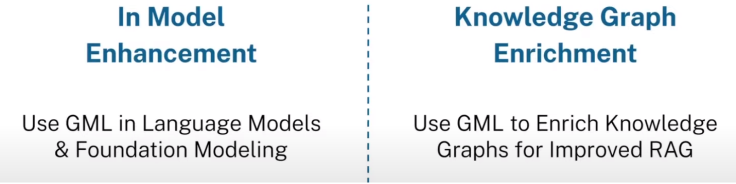
(Ref: Knowledge Graphs + Large Language Models = The ability for users to ask their own questions? - Peter Lawrence)

KG + LLM for?

Possible Modes:

- Use of KG in building better LLM
- Use of KG in better usage of LLM, say, RAG
- Use of LLM in building better KG
- Use of LLMs in better querying of KG

Any ideas?



(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

Leveraging KG for Better LLM

- Large Language Models (LLMs), such as ChatGPT, have demonstrated immense potential in various applications but are limited by hallucinations, misinformation, and biases in enterprise contexts.
- How integrating a graph-based Knowledge Graph as a support mechanism and interface can significantly improve the capabilities of LLMs?

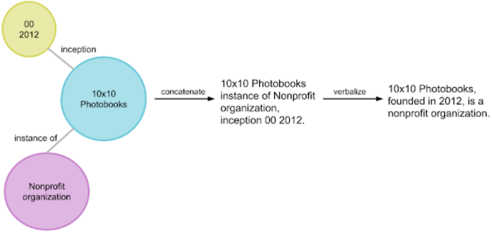
KG as LLM Pre-training Corpora

- KGs: Factual nature, extracted from trusted sources.
- Post-processing and human editors ensure accuracy.
- Advantages of incorporating KGs: Improved factual accuracy, reduced toxicity.
- Integration challenge: Different structural format from pre-training corpora in language models.

(Ref: KELM: Integrating Knowledge Graphs with Language Model Pre-training Corpora - Siamak Shakeri, Oshin Agarwal)

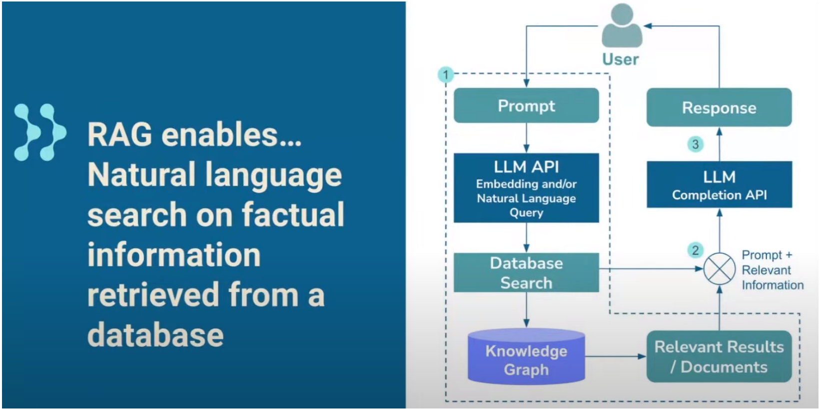
Steps

- KGs: Factual info in structured [subject, relation, object] triples.
- Entity subgraph: Group of related triples.
- Data-to-text generation: Converts subgraphs to natural language text.
- NLP task for KG-based text generation.



(Ref: KELM: Integrating Knowledge Graphs with Language Model Pre-training Corpora - Siamak Shakeri, Oshin Agarwal)

Use of KG in better usage of LLM, say, RAG



(Ref: NODES 2023 - Graph Machine Learning for 2024 - Zach Bluenfeld)

Use of KG in better usage of LLM, say, RAG

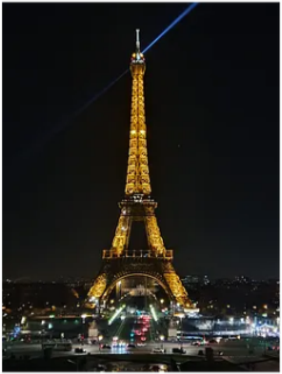
Augmenting company information from KG and adding it into the prompt for better results.

```
import re
import requests

def kg_enhance(company, url):
  res = requests.get('https://kg.diffbot.com/kg/v3/enhance', params={
      'token': DIFFBOT_TOKEN,
      'type': 'Organization',
      'name': company,
      'url': url
    })
  data = res.json()['data']
  return data[0]['entity'] if data else None

def kg_enhanced_recommendations(company, url):
  prompt = f"Recommend me five companies that are similar to {company}. For each
answer, include the name, (url), and up to 5 word summary."
  # Lookup company in knowledge graph
  input = kg_enhance(company, url)
  if input:
    prompt = f"About {company}:\n {input['description']}\n\n" + prompt
  # Ask LLM using enhanced input
  answers = generate_recommendations(prompt)
  # Validate answers against KG
  pattern = r'(\d+)?\.?\s?([\w\s]+)\s\(([^)]+)\)'
  return [line for line in answers if (matches := re.match(pattern, line)) and
kg_enhance(matches.group(2), matches.group(3))]
```

(Ref: Generating Company Recommendations using Large Language Models and Knowledge Graphs - Mike Tung)

YHK

Use of LLMs in building KG



The Eiffel Tower **is** a **wrought-iron lattice tower** on the Champ de Mars in Paris, France. It **is named** after the engineer Gustave Eiffel, whose company **designed** and **built** the tower. Locally **nicknamed** "La dame de fer" (French for "Iron Lady"), it **was constructed** from 1887 to 1889 as the centerpiece of the 1889 World's Fair. The tower **is** 330 metres (1,083 ft) tall, about the same height as an 81-storey building, and the tallest structure in Paris. Its base **is** square, **measuring** 125 metres (410 ft) on each side.

Source : Wikipedia (https://en.wikipedia.org/wiki/Eiffel_Tower)

(Ref: Automatic Knowledge Graphs: The Impossible Grail - Patrick Meyer)

YHK

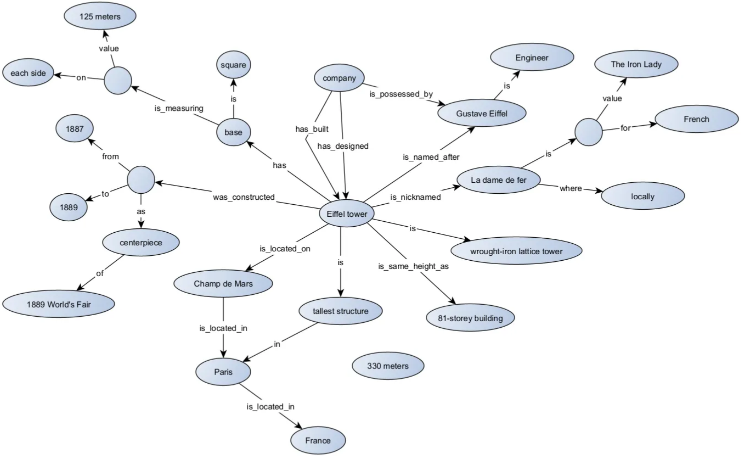
Unscrambling the Information

Representing conceptual understanding, establishing connections between the LLM's numerical vectors and the KG's ontological classes.

- The *Eiffel Tower* **is** a *wrought-iron lattice tower*.
- The *Eiffel Tower* **is located** on the *Champ de Mars*.
- The *Champ de Mars* **is located** in *Paris*.
- *Paris* **is located** in *France*.
- The *Eiffel Tower* **is named** after the *engineer Gustave Eiffel*.
- *Gustave Eiffel's company* **designed** the *Eiffel Tower*.
- *Gustave Eiffel's company* **built** the *Eiffel Tower*.
- The *Eiffel Tower* **is locally nicknamed** *"La dame de fer"*.
- *"La dame de fer"* **is the French for** *"The Iron Lady"*.
- The *Eiffel Tower* **was constructed** from *1887 to 1889*.
- The *Eiffel Tower* **was constructed as** *centerpiece* of *1889 World's Fair*.
- The *Eiffel Tower* **is** *330 meters high*.
- The *Eiffel Tower* **is** the *same height* as an *81-storey building*.
- The *Eiffel Tower* **is the** *tallest structure* in *Paris*.
- The *base of the Eiffel Tower* **is** a *square*.
- The *base of the Eiffel Tower* **measures** *125 meters on each side*.

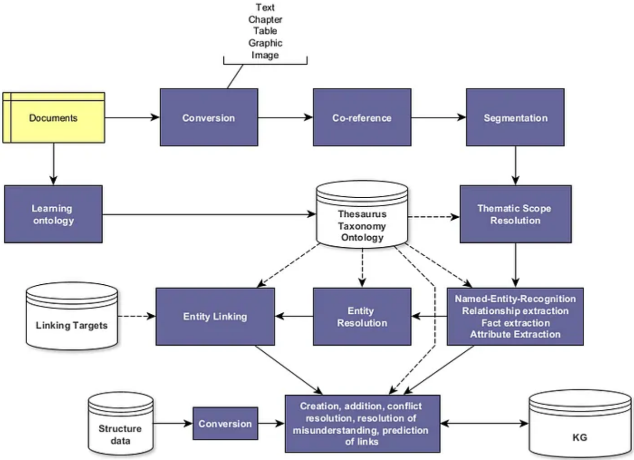
(Ref: Automatic Knowledge Graphs: The Impossible Grail - Patrick Meyer)

Populating KG



(Ref: Automatic Knowledge Graphs: The Impossible Grail - Patrick Meyer)

Populating KG is not easy

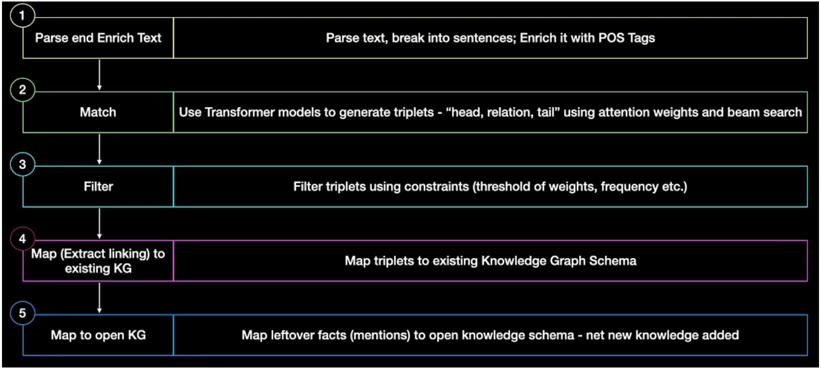


(Ref: Automatic Knowledge Graphs: The Impossible Grail - Patrick Meyer)

How?

- Knowledge graph generation: Systematic approach for organizations.
- Step 1: Ingest standard ontology (e.g., insurance risk).
- Use LLM (e.g., GPT-3) to script and populate graph database.
- Step 2: LLM acts as intermediate layer for natural language queries.
- Customized creation and search queries on the graph.
- Platform customization (e.g., Neo4j).

Steps

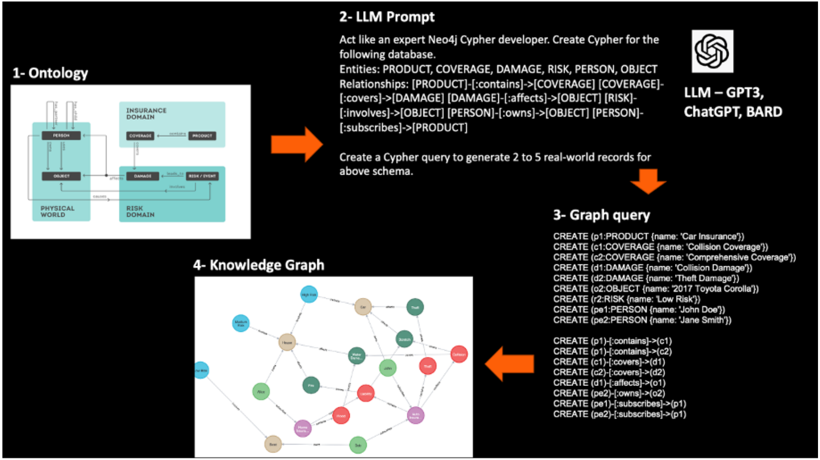


(Ref: Language Models are Open Knowledge Graphs .. but are hard to mine! - Nikhil Dharap)

Steps

- Study ontology, identify entities and relations.
- Build text prompt for LLM to generate schema and database.
- Prompt includes ontology description, entities, relationships.
- Clearly described in natural language for LLM understanding.
- Include constraints (data types, unique/foreign keys).
- Text prompt used as input for LLM to generate Cypher query.

Work-flow



(Ref: How to use large language models and knowledge graphs to manage enterprise data - Dattaraj Rao, Persistent Systems)

Pros and Cons

- Standardization: Ensures consistent and standardized data.
- Efficiency: Saves time and resources, ensures correct queries.
- Intuitive querying: Reduces need for deep graph understanding.
- Productivity: Updates knowledge graph with new data.
- Modified Cypher query adds new nodes and edges.
- Updated query ingested into graph database.

Use of LLMs in better querying of KG

- LLM converts verbose query to GraphQL syntax.
- GraphQL performs information retrieval.
- LLM converts GraphQL response to verbose answer.

(Ref: Knowledge Graphs + Large Language Models = The ability for users to ask their own questions? - Peter Lawrence)

How?

- Query : `How is Valve-104 connected to Reflux-401?`
- Using the OpenAI gpt-3.5-turbo LLM, provide the ontology of the RDF graph as the context prompt for the LLM.
- Ask it to prepare some SPARQL

```
Sure! Here is an example of a SPARQL query that returns the sequence of nodes
      that connect two particular nodes, assuming that there is a unique path
      between them:
‘‘‘
PREFIX connectivity: <http://example.org/ontology#>
SELECT ?path
WHERE {
  {
    <http://example.org/node1> (connectivity:connectedTo)+ ?path .
    ?path connectivity:connectedTo <http://example.org/node2> .
  }
}
‘‘‘
```

How?

- Property path syntax: Find paths connecting two nodes.
- `(connectivity:connectedTo)+`: Traverses arbitrary length path with "connectedTo" property.
- Query result: `"?path"` variable represents sequence of connecting nodes (as URIs).
- Assumes unique path between nodes.
- Multiple paths or cycles may result in multiple/incorrect results.

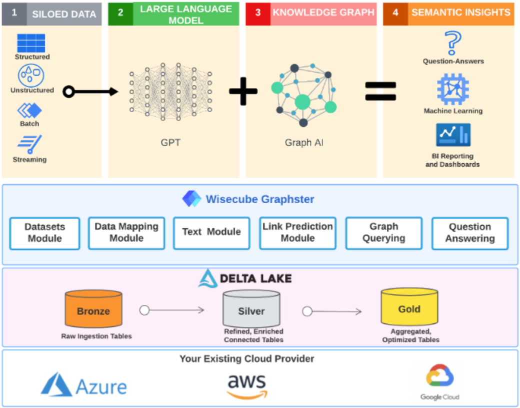
YHK

KG + LLM @ Neo4j

- https://github.com/neo4j/NaLLM
- Explore, develop, and showcase practical uses of LLMs with Neo4j data science algorithms.
- Strategy can yield substantial breakthroughs for businesses.
- Enhance accuracy, transparency, and predictability of model output.
- Open up new use-cases for LLMs and databases.
- Real-world use-cases:
  - Natural Language Interface: "Talk to your database" with generated queries based on user questions and inferred database schema.
  - Creating Knowledge Graph from Unstructured Data: LLMs decipher entities, discern relationships, and eliminate redundancies by recognizing duplicates.

Example

Wisecube's Unified Approach for Leveraging Large Language Models to Augment its Biomedical Knowledge Graph



(Ref: Combining Large Language Models and Knowledge Graphs - Haziqa Sajid)

YHK

Conclusions

Conclusions

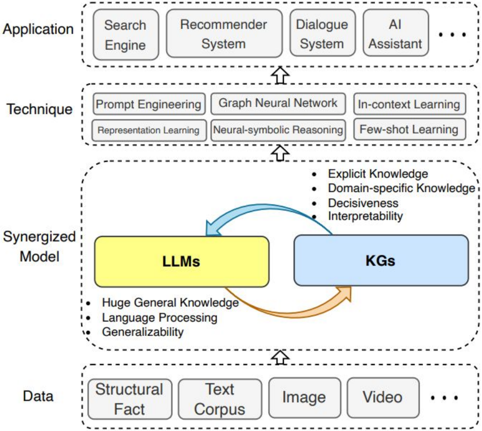
- Graph-based Knowledge Graphs enhance LLMs in the enterprise context.
- Leveraging Knowledge Graphs improves AI accuracy and relevance.
- Benefits: Better data comprehension, refined compliance procedures, accelerated research and development.
- CIOs should explore applying LLMs to internal data stores and constructing knowledge graphs using graph data science algorithms for substantial business breakthroughs.data science algorithms, as this strategy could yield substantial breakthroughs for their businesses.

Benefits of Combining Knowledge Graphs and LLMs

- LLMs:
  - Hallucinate.
  - Have general knowledge.
  - Understand natural language.
  - Do not provide structured responses.
- KGs:
  - Do not understand language.
  - Provide structured responses.
  - Are accurate and interpretable.
  - Have domain-specific knowledge.
- LLMs+KGs have the potential to overcome some of the weaknesses of LLMs.

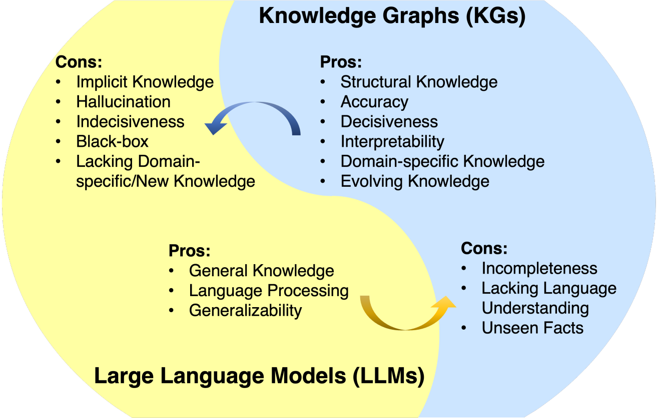
(Ref: LinkedIn post by Miguel Fierro)

Benefits of Combining Knowledge Graphs and LLMs



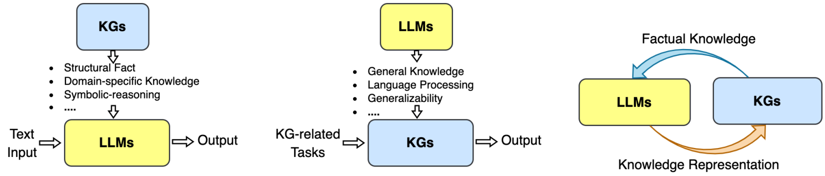
(Ref: LinkedIn post by Miguel Fierro)

Awesome-LLM-KG Github repo



(Ref: https://github.com/RManLuo/Awesome-LLM-KG)

Awesome-LLM-KG Github repo



a. KG-enhanced LLMs **b. LLM-augmented KGs** **c. Synergized LLMs + KGs**

(Ref: https://github.com/RManLuo/Awesome-LLM-KG)

Benefits of Combining Knowledge Graphs and LLMs

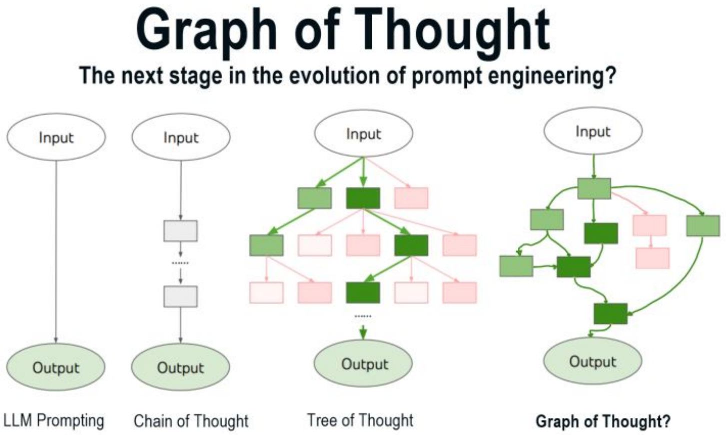
- Centralized source of accurate knowledge
- Structured knowledge fusion of information in different formats
- Increased informative value of collected data
- Gives LLMs a human reference frame of the real-world

What's next . . .

- Chain of Thought:
  - LLM determines necessary steps
  - Executes them sequentially
- Tree of Thought:
  - Combines thought with symbolic tree search algorithm
  - Enables optimal 'thought path' selection
  - Advances LLM's planning complexity
- Graph of Thought:
  - Thoughts modeled as nodes connected by edges
  - Directed Acyclic Graphs (DAGs) used
  - Revolutionizes data pipeline orchestration
  - Models dependencies without circular loops

(Ref: LinkedIn post by Tony Seale)

What's next ...



(Ref: LinkedIn post by Tony Seale)

References

- Knowledge Graphs & LLMs: Fine-Tuning Vs. Retrieval-Augmented Generation - Tomaz Bratanic
- Building knowledge graphs from unstructured data with OpenAI by Priyanka Shah
- LLM's Closing the KG Gap - Dean Allemang
- The Future of Knowledge Graphs in a World of Large Language Models - Denny Vrandecic
- Generating Company Recommendations using Large Language Models and Knowledge Graphs by Mike Tung
- Large Language Model = Knowledge Graph Store? Yes, by Fine-Tuning LLM With KG - Peter Lawrence
- Knowledge Graphs + Large Language Models = The ability for users to ask their own questions? - Peter Lawrence
- Automatic Knowledge Graphs: The Impossible Grail - Patrick Meyer
- Project NaLLM - https://github.com/neo4j/NaLLM
- KELM: Integrating Knowledge Graphs with Language Model Pre-training Corpora
- How to use large language models and knowledge graphs to manage enterprise data
- Enhancing Large Language Models with Graph-based Knowledge Graphs: Overcoming Limitations and Unlocking Enterprise Potential - Nicholas Moss
- "Unifying Large Language Models and Knowledge Graphs: A Roadmap" - Shirui Pan et al
- LLM Ontology-prompting for Knowledge Graph Extraction - Peter Lawrence
- Exploring ChatGPT for Learning, Code, Data, NLP & Fun
- Create Neo4j Database Model with ChatGPT
- ChatGPT and Neo4j for Creating and Importing Sample Datasets
- Doctor.ai+GPT-3+Kendra = An Ensemble Chatbot for Healthcare
- Creating a Knowledge Graph From Video Transcripts With ChatGPT
- Context-Aware Knowledge Graph Chatbot With GPT-4 and Neo4j

References

- Generating Company Recommendations using LLMs & Knowledge Graphs
- Integrating Neo4j into the LangChain ecosystem
- Fine-tuning an LLM model with H2O LLM Studio to generate Cypher statements
- Generating Cypher Queries With ChatGPT 4 on Any Graph Schema
- ChatGPT Generated Star Wars Data: Let's Explore With Neo4j
- LangChain has added Cypher Search
- Knowledge Graphs & LLMs: Harnessing Large Language Models with Neo4j
- Jump into Graph: Creating Mock Data to get started with Graph Databases
- Neo4j Live: GraphGPT
- OpenAI API Access - APOC Extended Documentation
- LangChain Cypher Search: Tips & Tricks
- Neo4j Integrations with Generative AI Features in Google Cloud Vertex AI
- Neo4j Knowledge Graphs and Google Generative AI
- Integrate LLM workflows with Knowledge Graph using Neo4j and APOC
- Unifying Large Language Models and Knowledge Graphs: A Roadmap
- Neo4j's Role in Fueling Generative AI with Graph Technology
- Knowledge Graphs & LLMs: Fine-Tuning Vs. Retrieval-Augmented Generation
- Knowledge Graphs & LLMs: Multi-Hop Question Answering
- Neo4j Knowledge Graphs and Google Generative AI
- Personal Movie Recommendation Agent with GPT4 + Neo4j
- Creating a Knowledge Graph From Video Transcripts With ChatGPT

Thanks . . .

- Search **"Yogesh Haribhau Kulkarni"** on Google and follow me on LinkedIn and Medium
- Office Hours: Saturdays, 2 to 5pm (IST); Free-Open to all; email for appointment.
- Email: yogeshkulkarni at yahoo dot com

(Generated by Hugging Face QR-code-AI-art-generator, with prompt as "Follow me")

YHK