EDBT ICDT BARCELONA 2025

Online Resources

Unifying Large Language Models and Knowledge Graphs for Question Answering: Recent Advances and Opportunities





Chuangtao Ma	Yongrui Chen	Tianxing Wu	Arijit Khan	Haofen Wang
AAU	SEU	SEU	AAU	TongjiU

AALBORG UNIVERSITET

Tutorial Outline

1) Introduction (15 Min) – Arijit Khan

1.1 Large Language Models (LLMs)
1.2 Knowledge Graphs (KGs)
1.3 Unifying LLMs+KGs
1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma

2.1 KGs as Background Knowledge
2.2 KGs as Reasoning Guidelines
2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**

3.1 Complex QA
3.2 Explainable QA
3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu

4.1 Performance Metrics
4.2 Benchmark Datasets
4.3 Industry Applications and Demonstrations

5) Opportunities for Data Management (10 Min) – Arijit Khan

6) Future Directions (5 Min) – Tianxing Wu

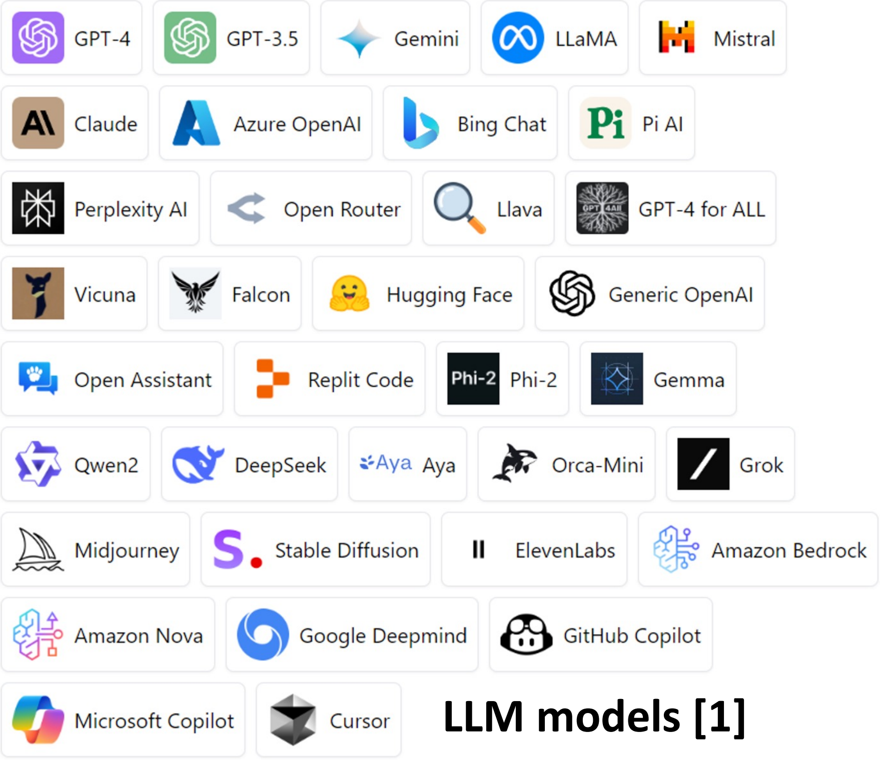
- **Q&A Session (10 Min)**

Large Language Models (LLMs) - Introduction

GPT-4 | GPT-3.5 | Gemini | LLaMA | Mistral
Claude | Azure OpenAI | Bing Chat | Pi AI
Perplexity AI | Open Router | Llava | GPT-4 for ALL
Vicuna | Falcon | Hugging Face | Generic OpenAI
Open Assistant | Replit Code | Phi-2 | Gemma
Qwen2 | DeepSeek | Aya | Orca-Mini | Grok
Midjourney | Stable Diffusion | ElevenLabs | Amazon Bedrock
Amazon Nova | Google Deepmind | GitHub Copilot
Microsoft Copilot | Cursor

LLM models [1]

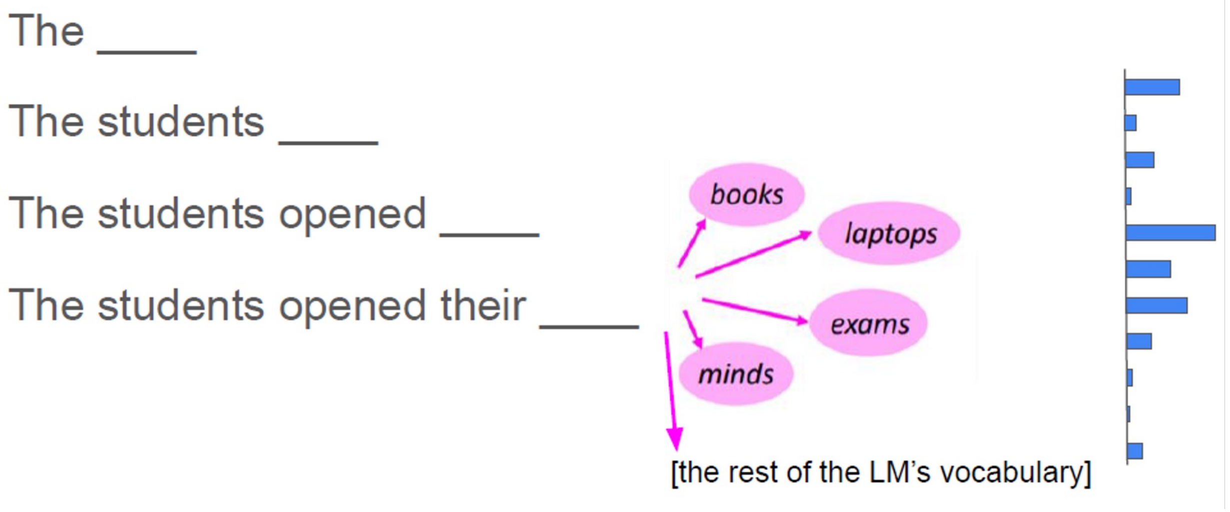
Large Language Models (LLMs) - Introduction



LLM models [1]

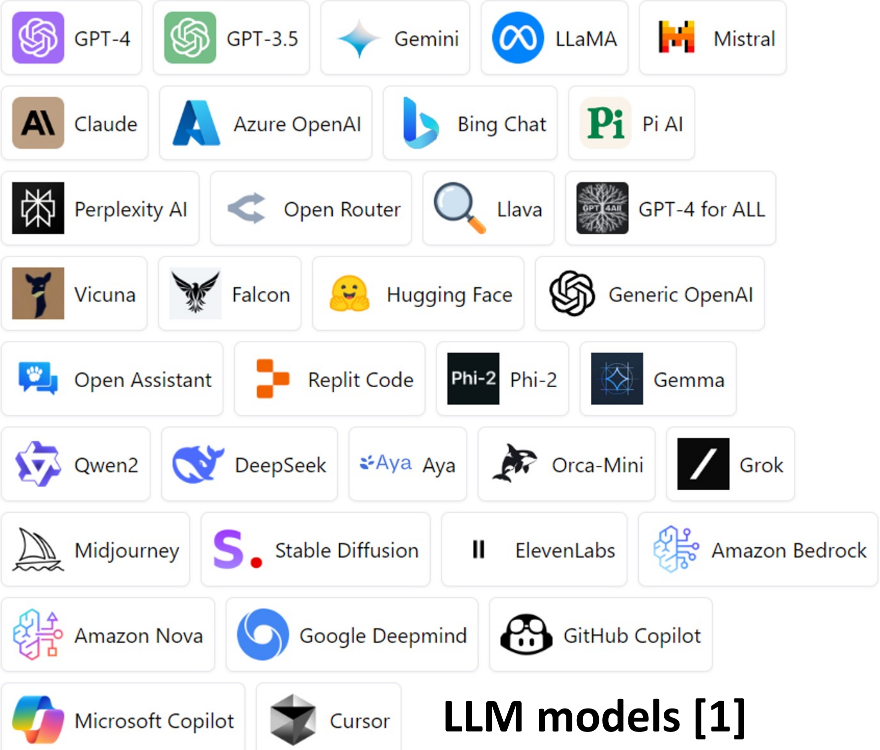
- Models the probability of the next word, given the history (context) of preceding words.

$$p(w) = p(w_1) \times p(w_2|w_1) \times p(w_3|w_1, w_2) \times p(w_l|w_1, \ldots, w_{l-1})$$
$$= \prod_{t=1}^{|w|} p(w_t|w_1, \ldots, w_{t-1})$$



Language models text generation [2]

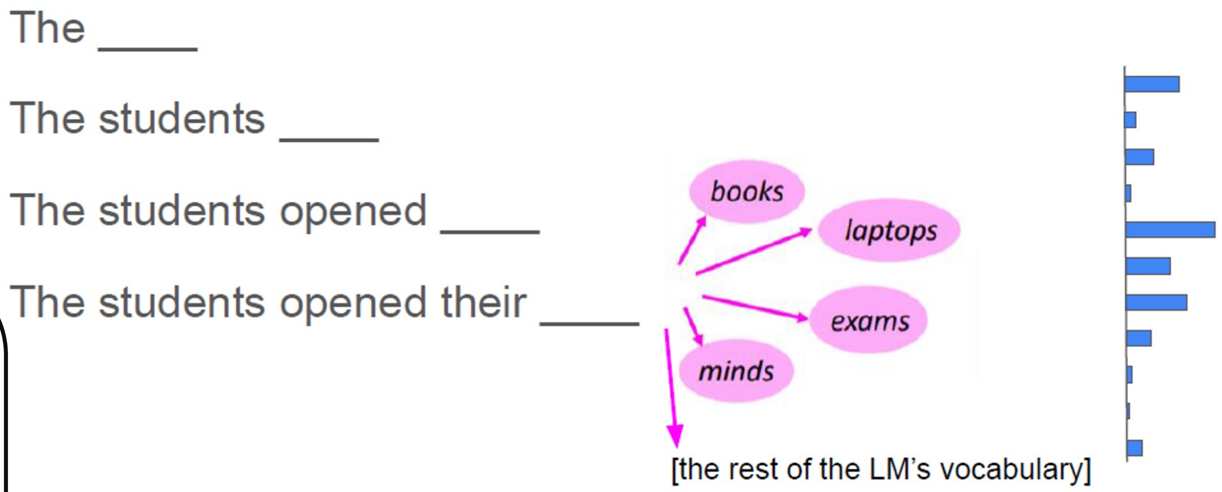
Large Language Models (LLMs) - Introduction



LLM models [1]

- Models the probability of the next word, given the history (context) of preceding words.

$$
\begin{aligned}
p(w) &= p(w_1) \times p(w_2|w_1) \times p(w_3|w_1, w_2) \times p(w_l|w_1, \ldots, w_{l-1}) \\
&= \prod_{t=1}^{|w|} p(w_t|w_1, \ldots, w_{t-1})
\end{aligned}
$$

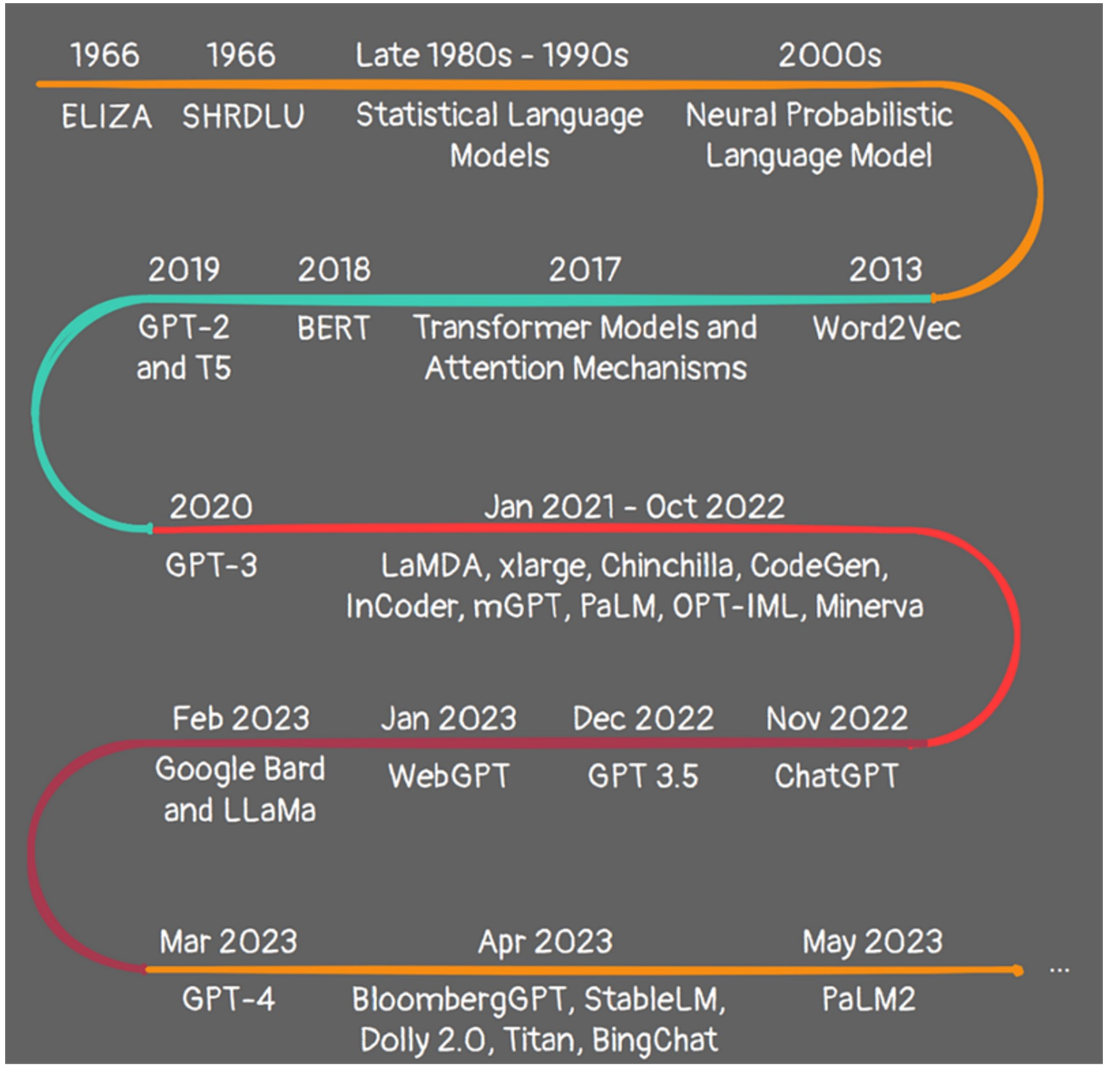


Language models text generation [2]

- **Generative AI (genAI):** AI system whose primary task is to generate content, e.g., GANs, VAEs, RNNs, LLMs, VLMs, VALL-E.
- **Large Language Models (LLMs):** Generative AI systems primarily designed for natural language processing tasks.
- **Foundation Models (FMs):** AI systems serving as the basis for a wide range of AI applications - can be adapted to a range of different, more specific purposes. E.g., LLMs, VLMs, speech FMs.

- often used interchangeably.

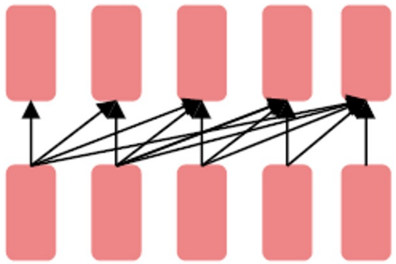
Language Models (LMs) - History



History of LMs [3]

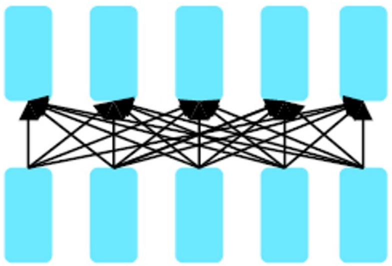
- **Stage 1 (1960-1990):** Linguistic Rules, Statistics-based Models
- **Stage 2 (2000):** Neural Language Models, Word Embedding, LSTM, GRU
- **Stage 3 (2010):** Pre-trained Language Models (PLMs) based on Transformer, Self-attention
  - ✓ e.g., BERT, GPT-2, BART
  - Parallel computation on GPUs for faster learning, more model parameters, and more training data
  - Trained on an extensive volume of unlabeled text in a self-supervised manner to capture general linguistic knowledge, and are employed in diverse NLP tasks via supervised fine-tuning, e.g., machine translation, text summarization, and question-answering (QA)
- **Stage 4 (2020):** Large Language Models (LLMs)
  - ✓ Large models (with 7-100B+ parameters)
  - ✓ Capable of performing more complex tasks and problem-solving compared to PLMs
  - Prompt-based Interaction, Retrieval-augmented generation (RAG) without updating model parameters
  - Scaling the models, compute, and data leads in increase in performance

PLMs and LLMs Architecture



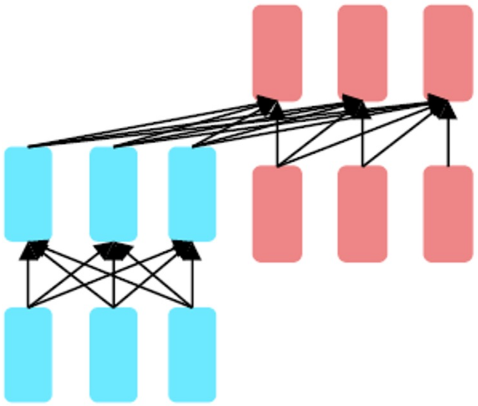
Decoders

- GPT, Claude, Llama, ..
- Text generation
- Emergent properties (text classification, summarization, translation, question answering, and diverse tasks)
- New tasks without updating model parameters via prompt-based in-context learning and retrieval augmented generation (RAG)



Encoders

- BERT, RoBERTa, ..
- Text comprehension (sentiment analysis, text classification, question-answering, and named entity recognition)



Encoder-decoders

- BART, T5 , ..
- Both text comprehension and generation (machine translation, summarization, and question answering)

Large Language Models (LLMs) – Benefits

Applications of LLMs [4]

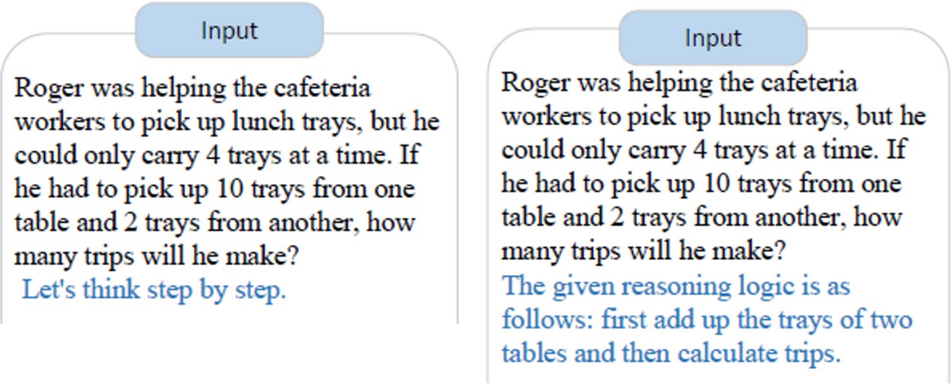
Emerging abilities; generalizing to unseen tasks; task descriptions provided as text.

Scaling the models, compute, and data leads in increase in performance.

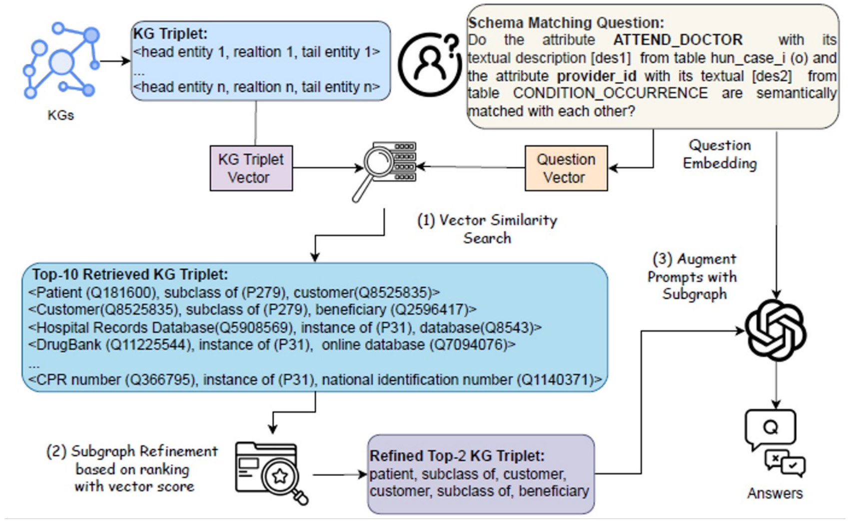
Perform new and creative tasks using prompt-based Interaction and retrieval-augmented generation (RAG) without updating model parameters.

✔ LLM pipelines remove task-specific supervision and need for labeled data – easy to use, less expensive, and fast to iterate.

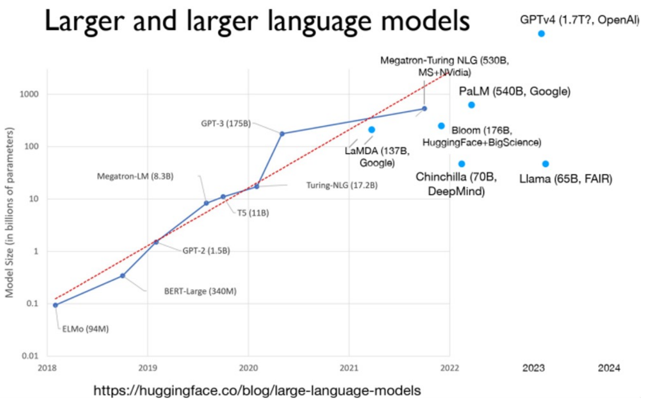
✔ LLMs act as knowledge bases - can be probed for QA and querying tasks.



Prompt Engineering



Retrieval-augmented Generation (RAG)



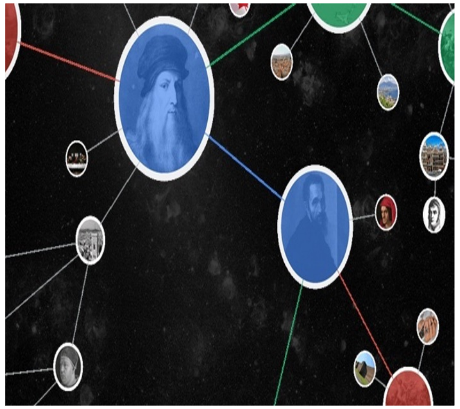
Scaling of LLMs

Large Language Models (LLMs) – Challenges

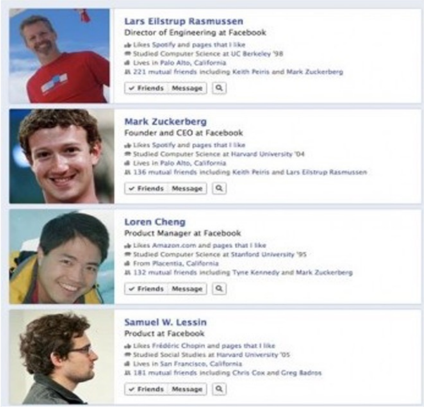
- **Alignment Problem:** LLMs may produce harmful, unsafe, toxic, or undesirable outputs – inappropriate language, misinformation, bias, and discrimination.
- **Hallucination:** Parametric knowledge, lack consistent representations of knowledge, fail to understand a question due to lack of context, knowledge gap (lack up-to-date and domain-specific knowledge), cannot recall facts (about not so popular or long-tail entities) → output unreliable and incoherent responses, hallucinate by generating factually incorrect statements.
- **Lack of Consistency:** Generate logically contradicting outputs → low semantic similarity of LLM outputs due to paraphrased versions of a question (meaning-preserving text alternations), violate important relational properties such as negation, symmetry, and transitivity; Adversarial LLM Jailbreaks.
- **Privacy Concern:** Data privacy, personally identifiable information, data retention policy, IP leakage, security vulnerabilities, legal compliance.
- **Black-box Model:** Many LLMs are proprietary and little information is released about them. Difficult to explain LLM predictions with billions of parameters. Knowledge in LLMs is hard to interpret, update, and is prone to bias. Challenging to deploy LLMs in decision-critical applications.
- **Environmental Concern:** High cost, energy consumption, carbon emissions, and water usage.
- **Societal Impacts:** Job loss, disparities, phishing, fraud, manipulation, plagiarism, cheating, fake news, big tech monopolies, societal unrest, …

Knowledge Graphs (KGs) – Introduction

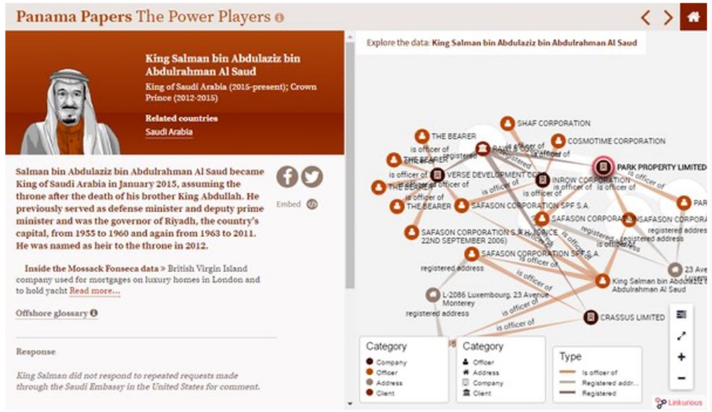
- Integrating knowledge + data at large scale → Knowledge graph



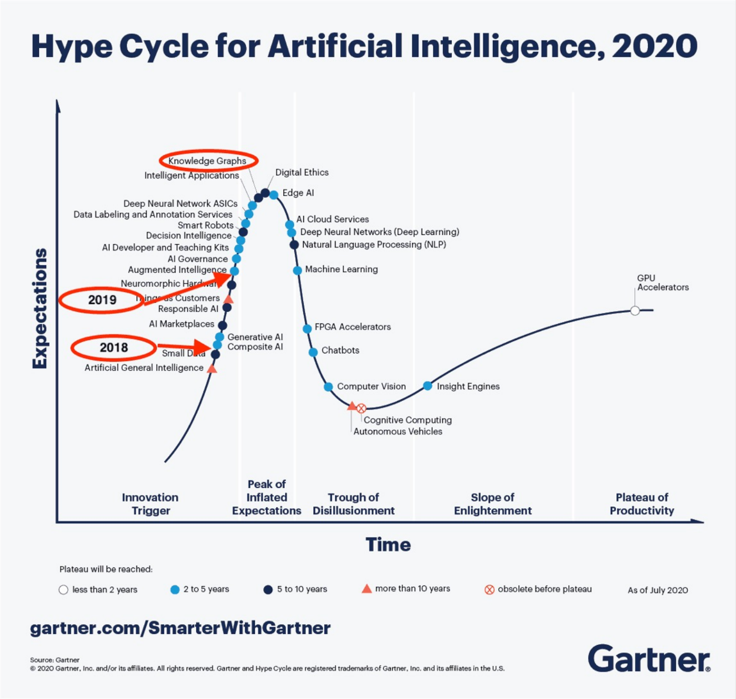
Google Knowledge Graph (2012)



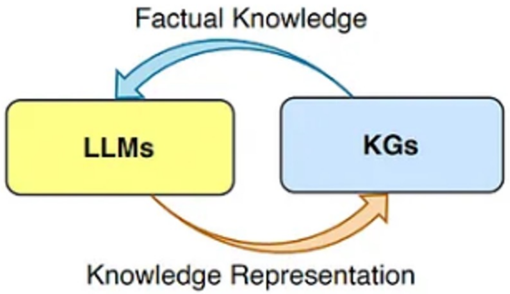
“People who like things I like” Facebook graph search (2013)



Panama Papers investigation, led by ICIJ, exposed highly connected networks of offshore tax structures used by world’s richest elites (2016)



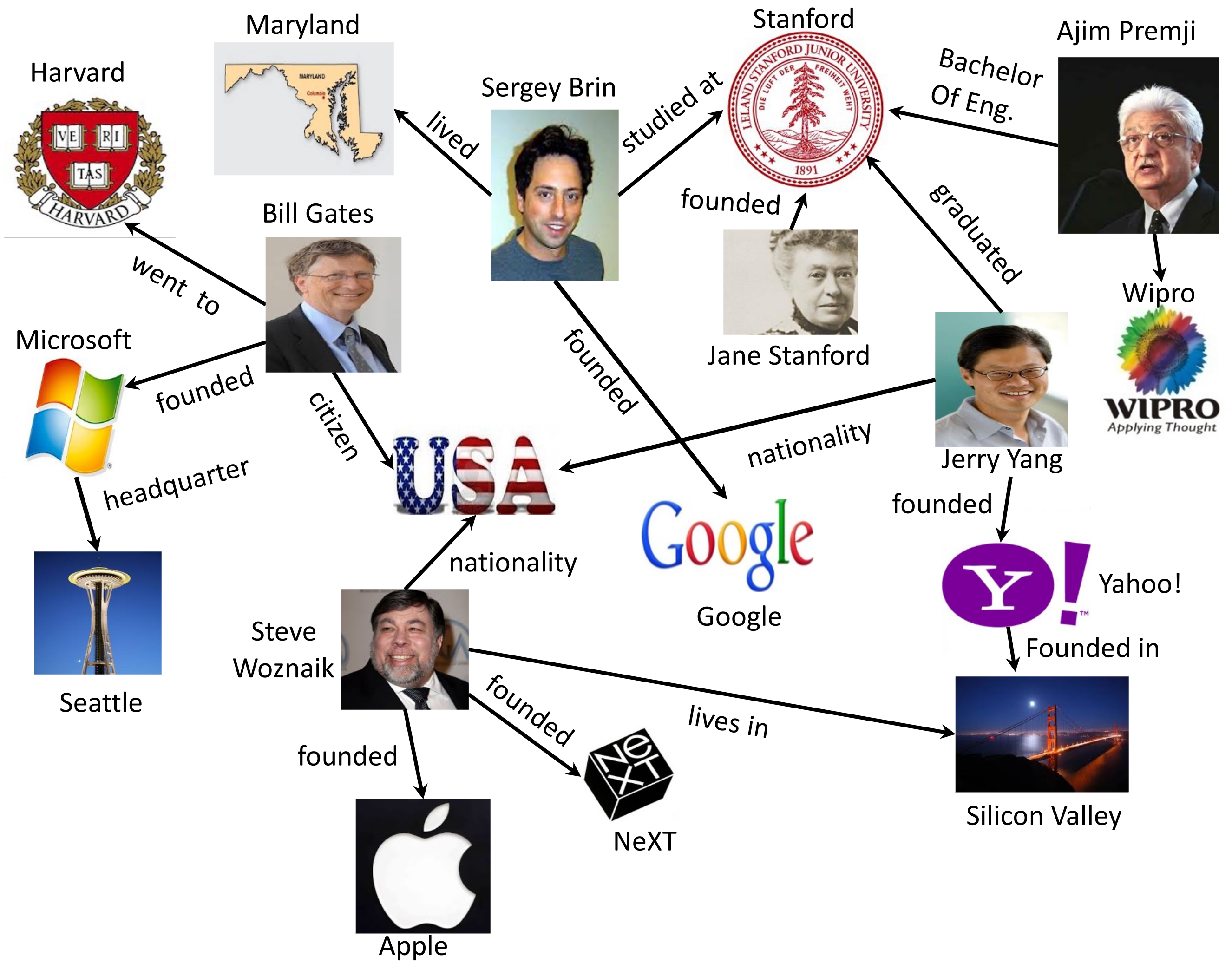
In 2020, Gartner put Knowledge Graphs at the peak of its AI hype cycle (2020)



LLM + KG (2024)

Claudio Gutierrez and Juan F. Sequeda. Knowledge Graphs: A Tutorial on the History of Knowledge Graph's Main Ideas. CIKM 2020 Tutorial

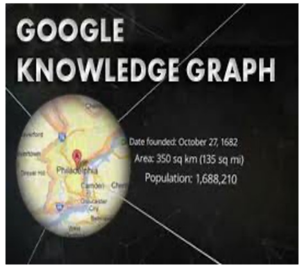
Knowledge Graphs (KGs) – Data Sources and Categories



Knowledge Graph

Freebase

DBpedia



WIKIDATA

yago select knowledge

Factual KG

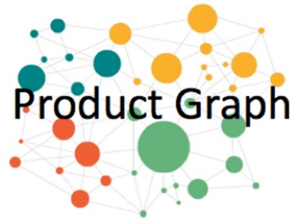
ConceptNet 5

WebChild

Event2Mind

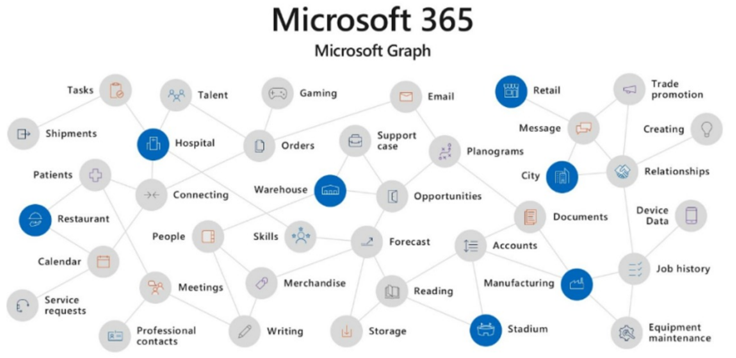
Commonsense KG

Microsoft Academic

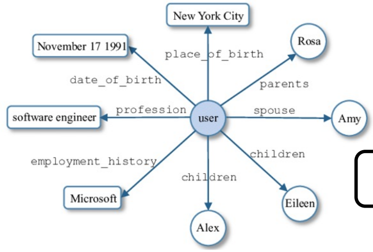


ClaimsKG

Domain-specific KG



Industrial KG



Personal KG (PKG)

Knowledge Graphs (KGs) – Components, Representation, and Usage

KG Components

- **Nodes:** entities, concepts, or instances within a domain, e.g., people, places, organizations, concepts, events, etc.
- **Edges:** relationships and connections between nodes, e.g., Is-a relationship, Part-of relationship, Related-to relationship, etc.
- **Properties:** additional descriptive information and metadata associated with nodes or edges, e.g., attributes, features, labels, Qualifiers, metadata, etc.
- **Ontology:** schema and vocabulary used within the KG, providing a structured framework for representing domain knowledge.

Knowledge Graphs (KGs) – Components, Representation, and Usage

KG Components

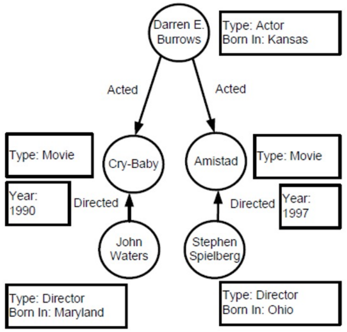
- **Nodes:** entities, concepts, or instances within a domain, e.g., people, places, organizations, concepts, events, etc.
- **Edges:** relationships and connections between nodes, e.g., Is-a relationship, Part-of relationship, Related-to relationship, etc.
- **Properties:** additional descriptive information and metadata associated with nodes or edges, e.g., attributes, features, labels, Qualifiers, metadata, etc.
- **Ontology:** schema and vocabulary used within the KG, providing a structured framework for representing domain knowledge.

KG Representations

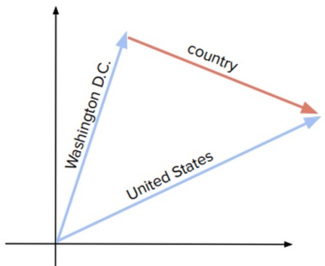
- **RDF Triples:** collection of <subject, predicate, object> triples.
- **Property Graph:** graph model having nodes and edges with arbitrary number of properties, where a node (a subject or an object) denotes an entity and a directed edge (a predicate) is a relationship between two entities.
- **KG Embedding:** Vector representation of KG nodes and edges in low-dimensional space, such that the original structures and relations in the KG are preserved in these learned semantic vectors.

```
Person1 isNamed "John Waters"
Person2 isNamed "Stephen Spielber
Person3 isNamed "Darren E. Burrow
Movie1 hasTitle "Cry-Baby"
Movie1 hasActor Person3
Movie1 hasDirector Person1
Movie2 hasTitle "Amistad"
Movie2 hasActor Person3
Movie2 hasDirector Person2
```

RDF Triples



Property Graph



KG Embedding

Knowledge Graphs (KGs) – Components, Representation, and Usage

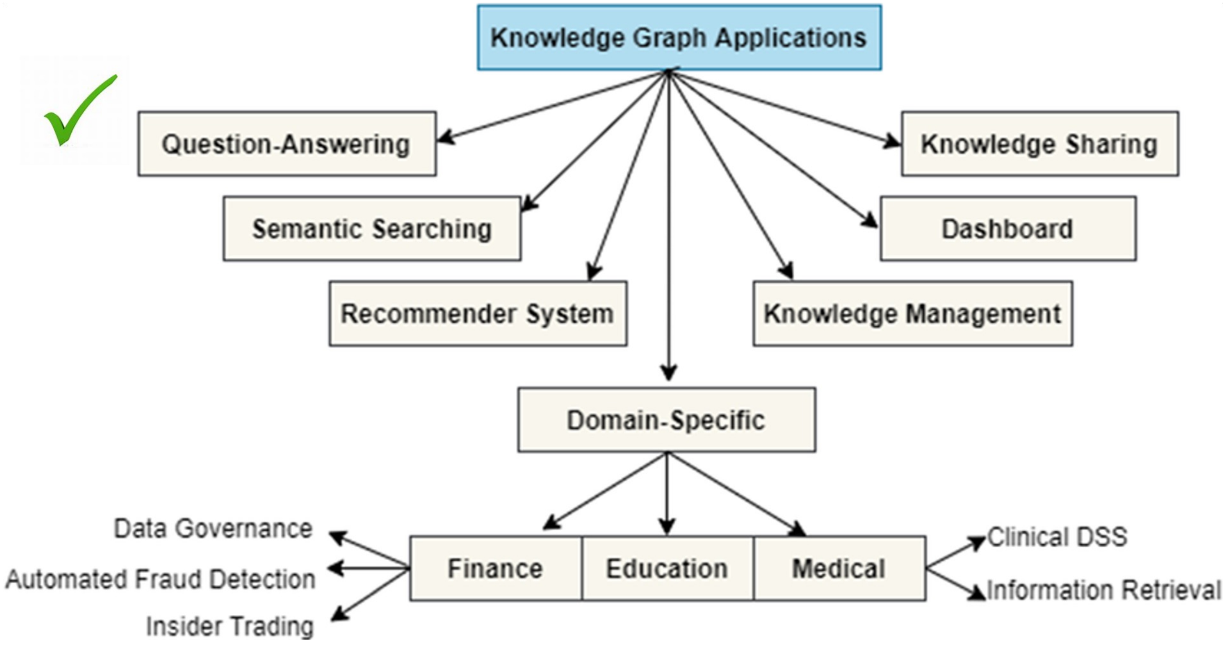
KG Components

- **Nodes:** entities, concepts, or instances within a domain, e.g., people, places, organizations, concepts, events, etc.
- **Edges:** relationships and connections between nodes, e.g., Is-a relationship, Part-of relationship, Related-to relationship, etc.
- **Properties:** additional descriptive information and metadata associated with nodes or edges, e.g., attributes, features, labels, Qualifiers, metadata, etc.
- **Ontology:** schema and vocabulary used within the KG, providing a structured framework for representing domain knowledge.

KG Representations

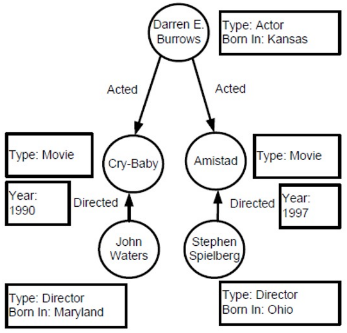
- **RDF Triples:** collection of <subject, predicate, object> triples.
- **Property Graph:** graph model having nodes and edges with arbitrary number of properties, where a node (a subject or an object) denotes an entity and a directed edge (a predicate) is a relationship between two entities.
- **KG Embedding:** Vector representation of KG nodes and edges in low-dimensional space, such that the original structures and relations in the KG are preserved in these learned semantic vectors.

KG Applications

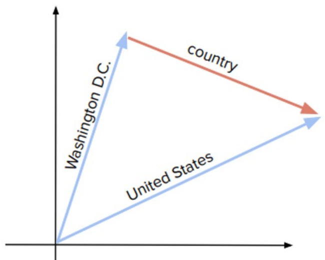



```
Person1 isNamed "John Waters"
Person2 isNamed "Stephen Spielber
Person3 isNamed "Darren E. Burrow
Movie1 hasTitle "Cry-Baby"
Movie1 hasActor Person3
Movie1 hasDirector Person1
Movie2 hasTitle "Amistad"
Movie2 hasActor Person3
Movie2 hasDirector Person2
```

RDF Triples

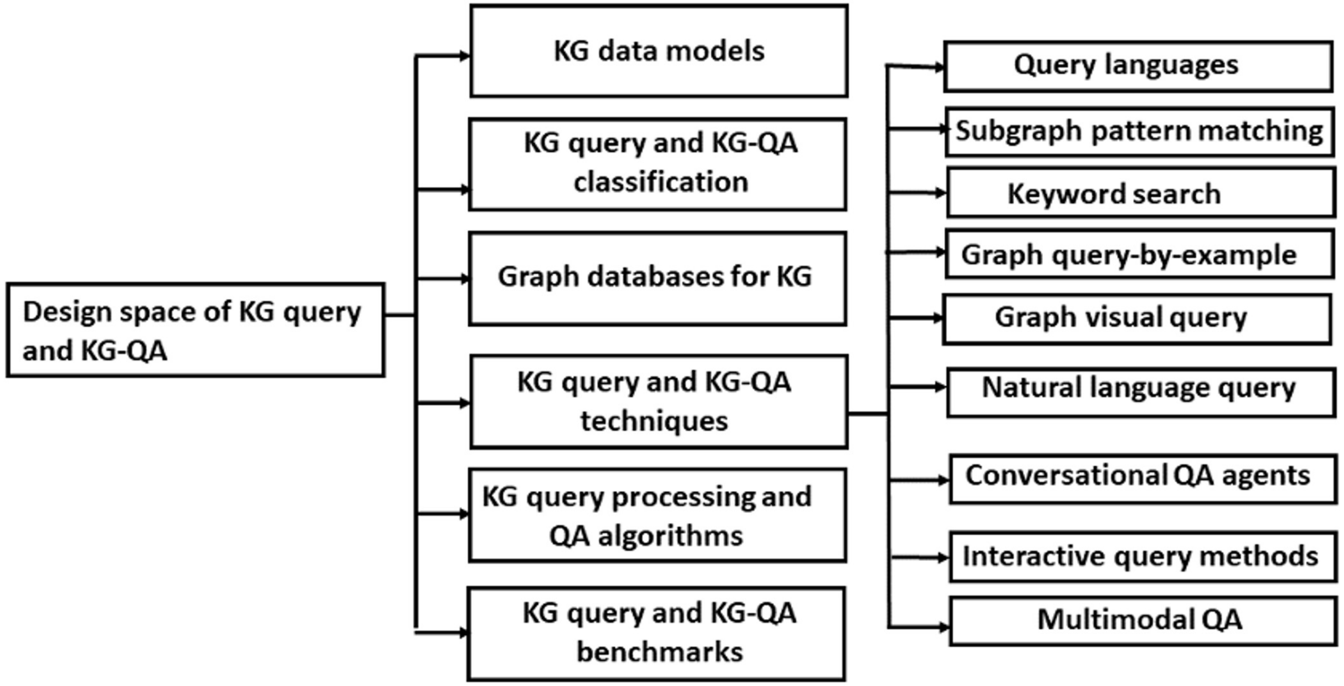


Property Graph



KG Embedding

Knowledge Graphs (KGs) - Query and Question Answering (QA)



Design space of KG query and KG-QA problems

- **Query:** A query has a structure, e.g., a graph pattern, a logic query, an SQL or a SPARQL query.

- **QA:** QA deals with answering unstructured natural language questions (NLQs) – it also includes a natural language understanding task.

Arijit Khan. Knowledge Graphs Querying. ACM SIGMOD Record 52(2): 18-29 (2023).

Knowledge Graphs (KGs) – Benefits and Challenges

KG Benefits

structured, highly curated, and reliable representation of knowledge via explicit relationships.

support symbolic reasoning and inference, with answer validation and explainability.

✓ schema-flexible: updated dynamically with new knowledge via addition or deletion of triples/ nodes & edges.

✓ offer accurate explicit knowledge in many downstream applications, e.g., web search, QA, semantic search, personal assistants, fact-checking, and recommendation.

KG Challenges

- **Difficult to construct**.
- **Difficult to query** due to incompleteness, schema-flexibility, heterogeneity, and massive-scale.
- **Lack of user-friendliness in writing query**: non-professional users find it challenging to write an accurate query, e.g., via SPARQL, Cypher, Gremlin, GSQL, etc., since users must have full knowledge of the query language, schema, and the vocabulary used in a KG. Current KG querying approaches generally lack language understanding, are inadequate to deal with unseen entities and new facts, and often ignore multi-modal information in KGs.
- **Interoperability issue**: existing methods are tailored for specific KGs or downstream tasks.

LLMs+ KGs: Synergy

KG for LLM

KGs offer external knowledge (up-to-date, domain specific, and symbolic knowledge) for enhancing the accuracy, consistency, transparency, and the overall capabilities of LLMs.

KG-enhanced pre-training, fine-tuning, and inference (KG-RAG).

KG-enhanced validation (LLM guardrail) and explainability.

LLM for KG

LLMs augment KGs via knowledge extraction, auto-completion, and incorporating multi-modal information, enhancing usability and performance of downstream tasks with natural language understanding and generalization capabilities.

LLM-enhanced KG creation and completion.

LLM-enhanced KG embedding.

✓ LLM-enhanced KG querying, analytics, and domain-specific applications.

LLM+KG

Downstream applications benefit from the complementarity of LLMs and KGs – LLMs and KGs offer parametric vs. explicit knowledge, respectively.

Question Answering (QA): Introduction

- **QA:** QA deals with answering unstructured natural language questions (NLQs) – it also includes a natural language understanding task.

QA Categories

- **Simple vs. Complex Questions:**
  - ✓ Simple question → a single triple and a single relation, e.g., "*where was Albert Einstein born?*" can be answered based on the relation 'born': <Albert Einstein, born, ?place>.
  - ✓ Complex question → multiple KG relations (multi-hop) and/or additional operations (e.g., aggregate, order, temporal), e.g., *"what was the first movie of James Cameron that own an Oscar?"*

- **Multi-document QA**
- **Multi-lingual QA**
- **Multi-modal QA**
- **Multi-run and conversational QA**
- **Temporal QA**
- **Factoid QA**
- **Explainable QA**

QA Applications

- Text generation, chatbots, dialog generation, web search, entity linking, natural language query, fact-checking, …
- Open-domain QA, domain-specific QA
- AI, NLP, information retrieval, and data management

LLM+KG for QA: Motivation and Challenges

- PLMs & LLMs for QA based on their pre-trained knowledge and natural language understanding capabilities [35]

Challenges of PLMs and LLMs in QA

- Limited reasoning ability for complex QA
- Lack of up-to-date and domain-specific knowledge
- Hallucination and inconsistency

Motivation of KGs+LLMs in QA

KGs can offer external, precise, up-to-date, and domain-specific knowledge to LLMs via pre-training, fine-tuning, and RAG (Graph RAG, KG-RAG)

✓ Improve LLM's accuracy and consistency
✓ Support answer validation (LLM guardrail) and explainability.

Challenges of KGs+LLMs in QA

- Knowledge conflict
- Poor relevance and quality of retrieved data, limited context size of LLMs
- Large-scale and dynamic KGs
- Lack of iterative and multi-hop reasoning:

LLM+KG for QA: Roles of KG in Complex QA

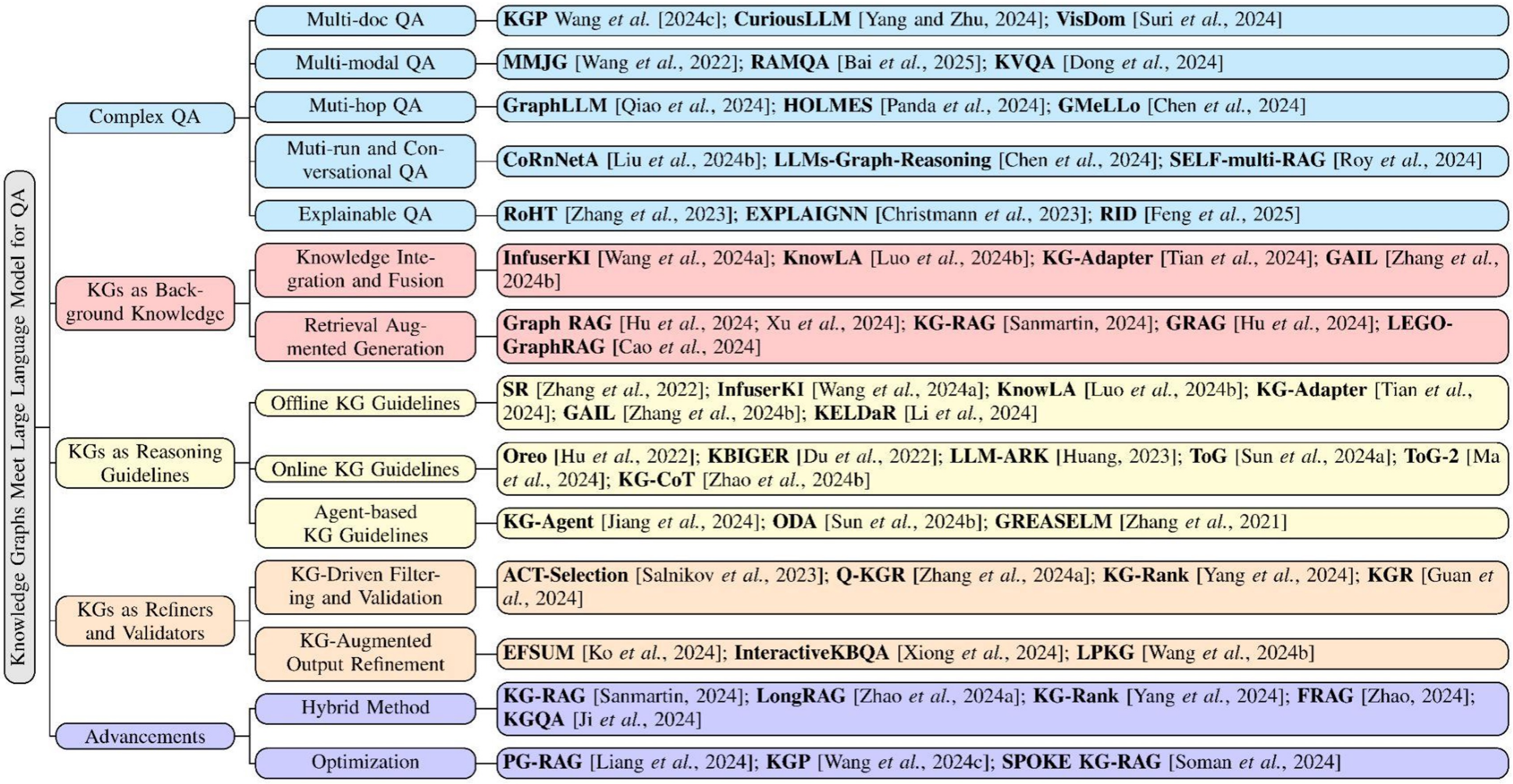
Approach	Strength	Limitation	KG Requirement
KG as Background Knowledge	Broad Coverage	Static Knowledge	High Domain Coverage
KG as Reasoning Guidelines	Multi-hop Capabilities	Computational Overhead	Rich Relational Paths
KG as Refiners and Validator	Hallucination Reduction	Validation Latency	High Accuracy & Recency

Comparison of Approaches with Different Roles of KG

Alignment of Approaches to Complex QA with Different Roles of KG

Approach	Multi-doc QA	Multi-modal QA	Multi-hop QA	Multi-run QA	XQA
KG as Background Knowledge	✓	✓	✓	✓	X
KG as Reasoning Guidelines	✓	✓	✓	X	✓
KG as Refiners and Validator	X	X	✓	✓	X
Hybrid Methods	✓	✓	✓	✓	✓

LLM+KG for QA - Timeline



A Structured Taxonomy of Synthesizing LLMs and KGs for QA

Relevant Tutorials

QA, LLMs, KG

- Danqi Chen and Wen tau Yih. 2020. Open-domain question answering. In ACL. 34–37.
- Lihui Liu, Zihao Wang, Jiaxin Bai, Yangqiu Song, and Hanghang Tong. 2024. New frontiers of knowledge graph reasoning: Recent advances and future trends. In WWW. 1294–1297.
- Jizhi Zhang, Keqin Bao, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2024. Large language models for recommendation: Progresses and future directions. In WWW Companion (2024). 1268–1271.

LLMs+KGs/Graphs, RAG

- Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on RAG meeting LLMs: Towards retrieval-augmented large language models. In SIGKDD. 6491–6501.
- Chao Huang, Xubin Ren, Jiabin Tang, Dawei Yin, and Nitesh Chawla. 2024. Large language models for graphs: Progresses and directions. In WWW. 1284-1287.
- Qiang Zhang, Jiaoyan Chen, Zaiqiao Meng. 2024. Integrating Knowledge Graphs and Large Language Models for Advancing Scientific Research. Learning on Graph Conference (LoG).

Online Resources

Unifying LLMs with KGs for QA
Part - 2

Chuangtao Ma

Aalborg University

AALBORG UNIVERSITET

Tutorial Outline

1) **Introduction (15 Min) – Arijit Khan**
- 1.1 Large Language Models (LLMs)
- 1.2 Knowledge Graphs (KGs)
- 1.3 Unifying LLMs+KGs
- 1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma
- 2.1 KGs as Background Knowledge
- 2.2 KGs as Reasoning Guidelines
- 2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**
- 3.1 Complex QA
- 3.2 Explainable QA
- 3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu
- 4.1 Performance Metrics
- 4.2 Benchmark Datasets
- 4.3 Industry Applications and Demonstrations

5) Opportunities for Data Management (10 Min) – Arijit Khan

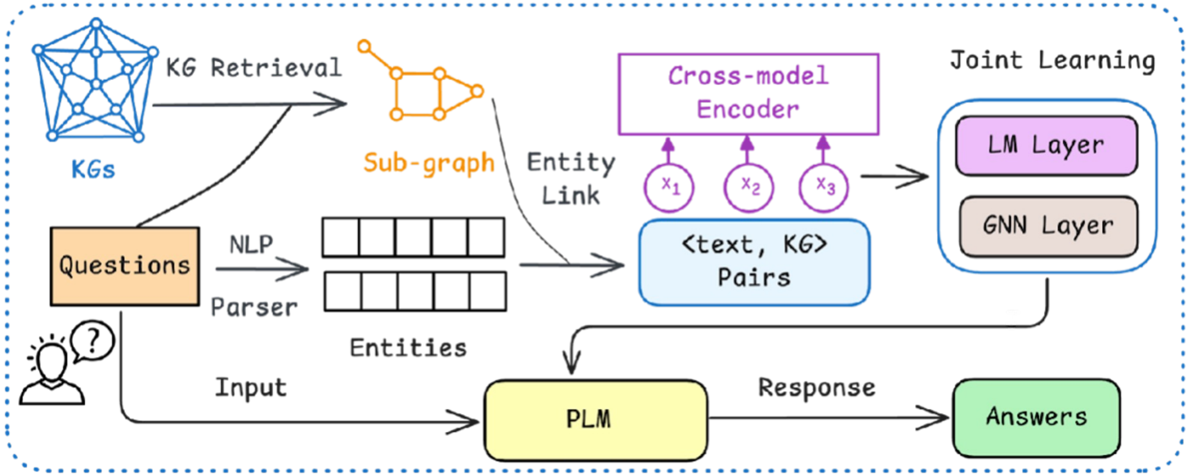
6) Future Directions (5 Min) – Tianxing Wu

- **Q&A Session (10 Min)**

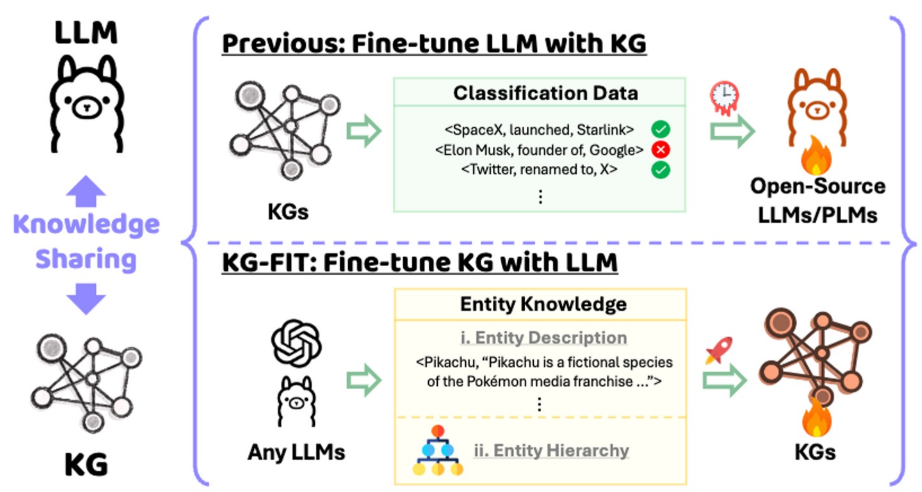
KGs as Background Knowledge

- KGs and Text Alignment
  - Are the KGs available for text?
    - \- KGs and text data are stored separately (Common scenario for QA task)
    - \- KGs (entities or relations) having the textual description (Text-KG pair)
  - How to align the KGs and text?

Entity Linking
KG Retrieval



Joint learning



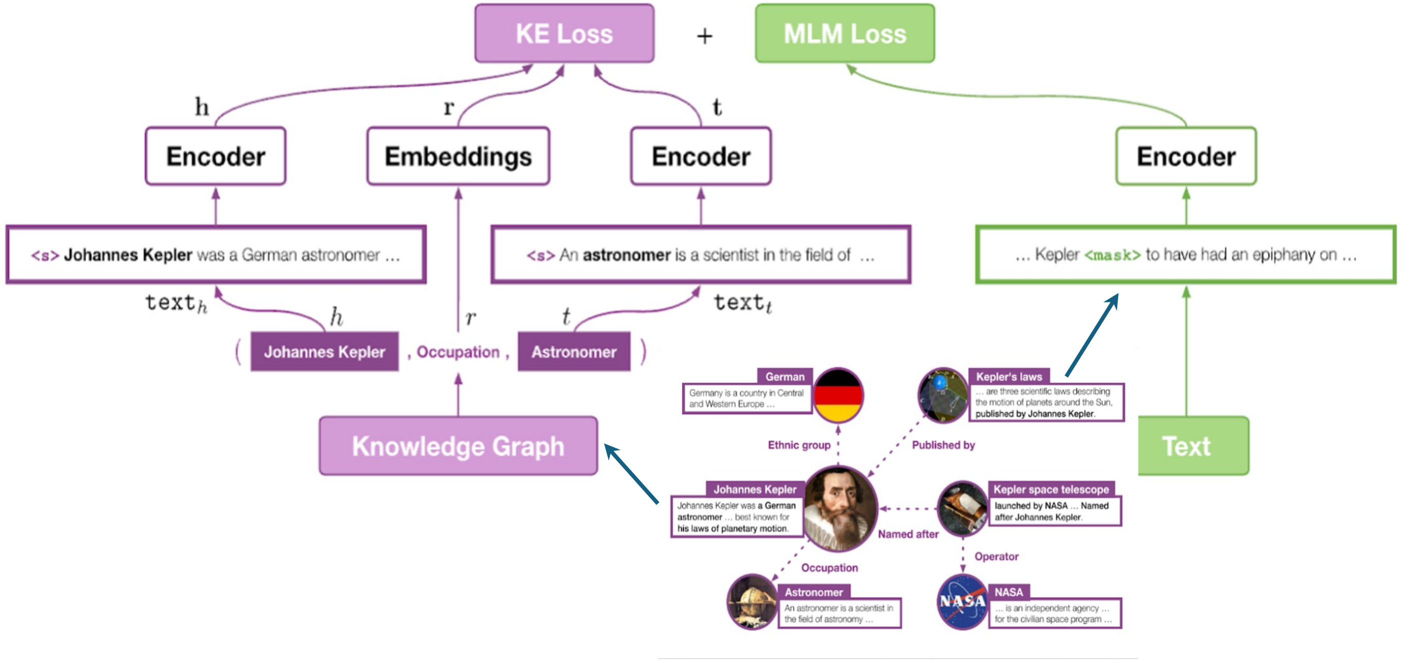
Fine-tuning

Jiang, Pengcheng, et al. KG-FIT: Knowledge graph fine-tuning upon open-world knowledge. *Advances in Neural Information Processing Systems* 37 (2024): 136220-136258.

KGs as Background Knowledge

- Knowledge Integration and Fusion
  - Joint Learning: Unified representation for KG and PLM [ACL2021]

    - Encode textual description of entity as entity embeddings and jointly train the KE and MLM on the same PLM



$$\mathcal{L} = \mathcal{L}_{\mathrm{KE}} + \mathcal{L}_{\mathrm{MLM}},$$

Joint loss for knowledge embeddings and masked language model

Wang, X., et al. A unified model for knowledge embedding and pre-trained language representation. ACL, 2021, 9: 176-194.

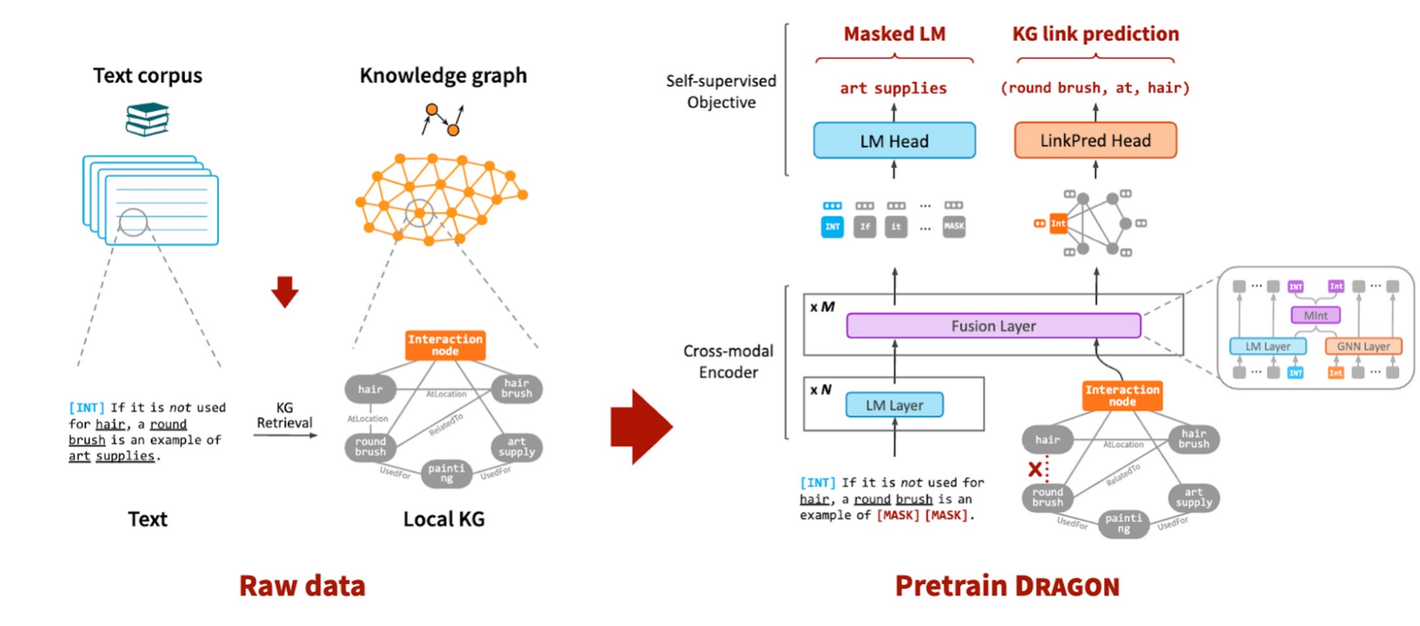
KGs as Background Knowledge

- Knowledge Integration and Fusion
  - Joint Learning: Bidirectional language and KG pretraining [NeurIPS2022]

    - Retrieving relevant subgraph from KG based on text to create text-KG pair.

    - Leveraging **cross-modal encoder** that fuses the input text-KG pair bidirectionally.

    - Unifying masked **LM and KG link prediction** for and joint learning reasoning.



Modality interaction (Mint) with interaction token and node to mix representation for joint learning

$$[\boldsymbol{h}_{int}^{(\ell)}; \boldsymbol{e}_{int}^{(\ell)}] = \text{MInt}([\tilde{\boldsymbol{h}}_{int}^{(\ell)}; \tilde{\boldsymbol{e}}_{int}^{(\ell)}]),$$

Yasunaga, Michihiro, et al. Deep bidirectional language-knowledge graph pretraining. *Advances in Neural Information Processing Systems* 35 (2022): 37309-37323.

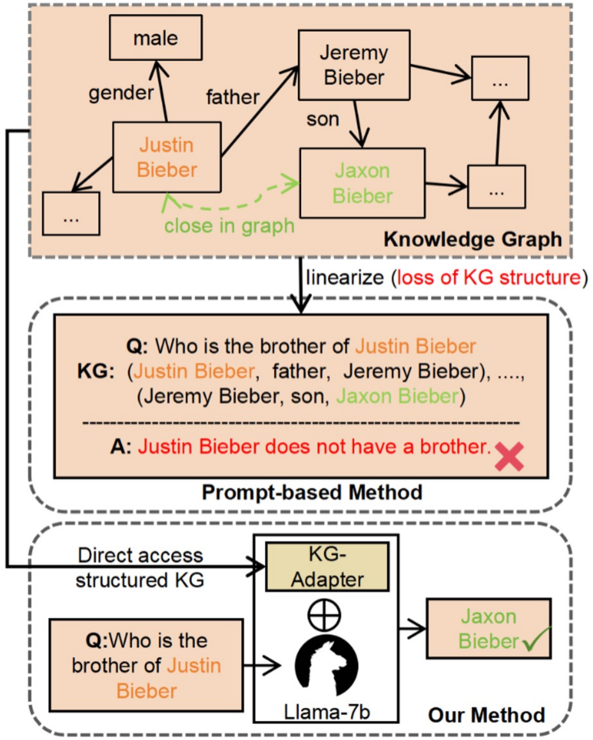
KGs as Background Knowledge

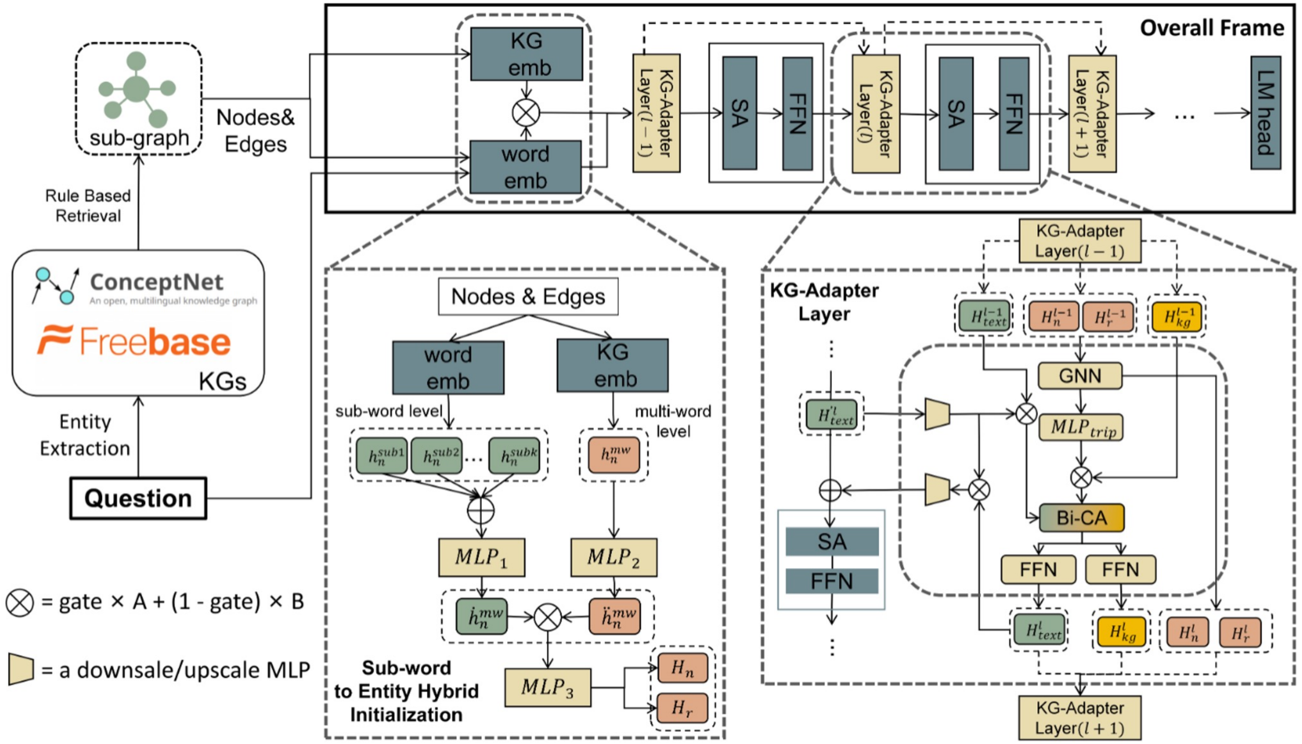
- Knowledge Integration and Fusion
  - Fine-tuning: incorporate the knowledge with text during fine-tuning (KG-Adapter) [ACL 2024]

Fuse representations of entity-level KG and sub-word KG

GNN-based KGs representations updating

Bidirectional cross- attention based reconstruction

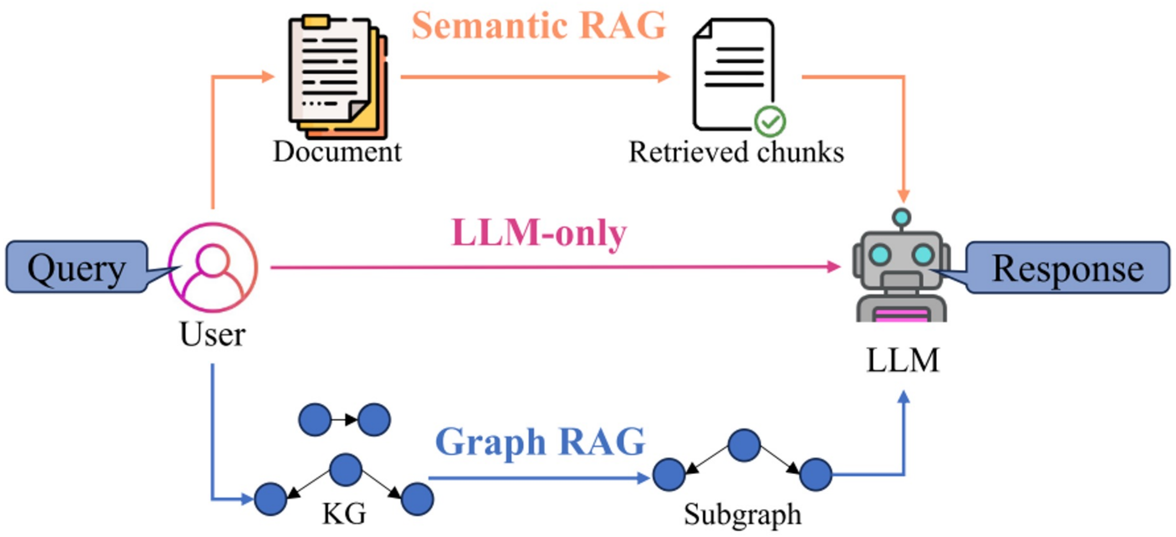




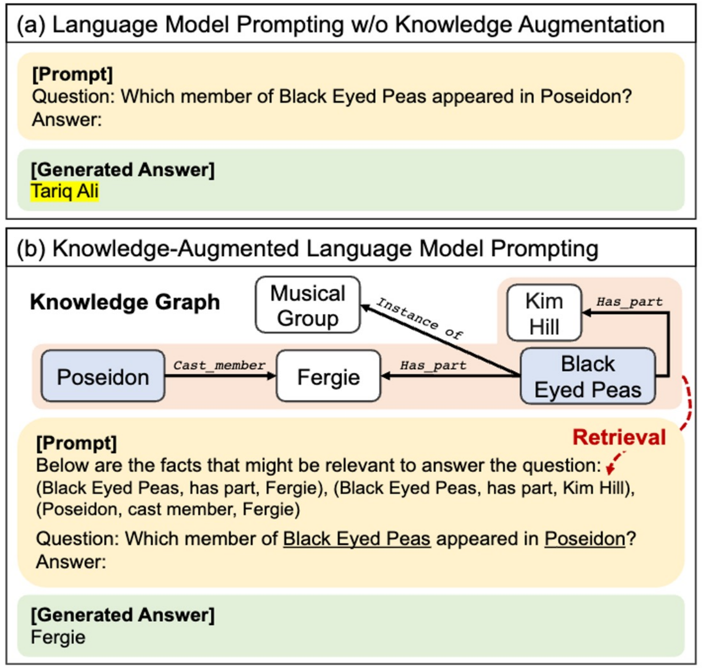
Jiang, Pengcheng, et al. KG-FIT: Knowledge graph fine-tuning upon open-world knowledge. *Advances in Neural Information Processing Systems* 37 (2024): 136220-136258.

KGs as Background Knowledge

- Retrieval Augmented Generation (RAG)
  - Semantic RAG: retrieve document or chunks with **limited reasoning abilities**
  - KG-RAG: retrieve subgraph (triples) from KGs with factual-based relationships



LLM *vs* RAG *vs* Graph RAG



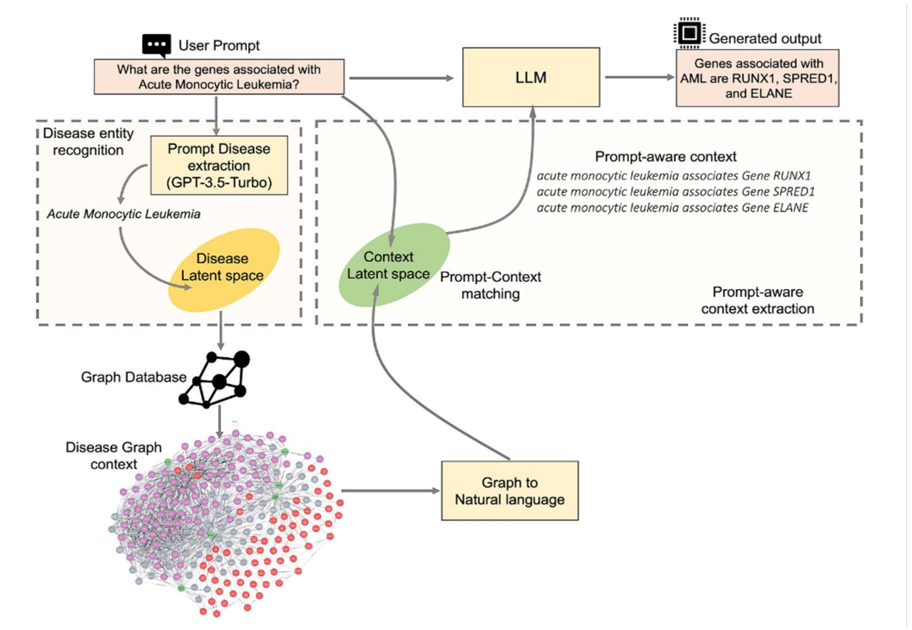
Prompt-based Augmentation

Xiangrong, Zhu, et al. Knowledge Graph-Guided Retrieval Augmented Generation. *arXiv preprint arXiv:2502.068641* (2025).
Baek, Jinheon, et al. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. *arXiv preprint arXiv:2306.04136* (2023).

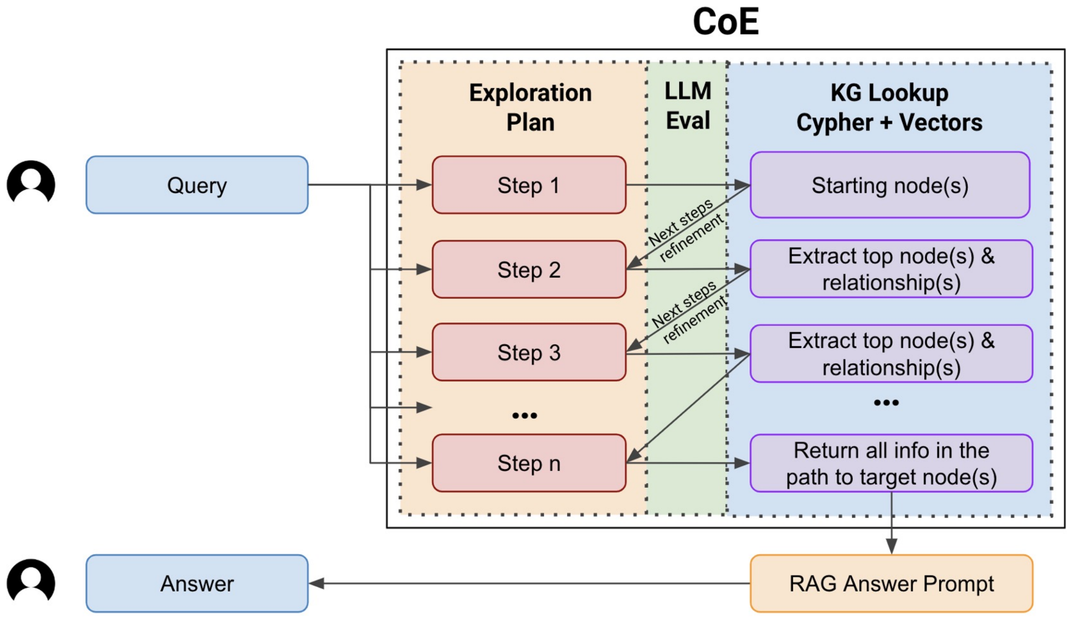
KGs as Background Knowledge

- Retrieval Augmented Generation (RAG)
  - KG-RAG for knowledge intensive tasks [Bioinformatics, 2024]
  - Chain of Explorations (CoE) for KG-RAG [arXiv2024]

Introduce CoT with LLM to select relevant nodes or relationships from KG



KG-RAG: Cypher-query based Retrieval

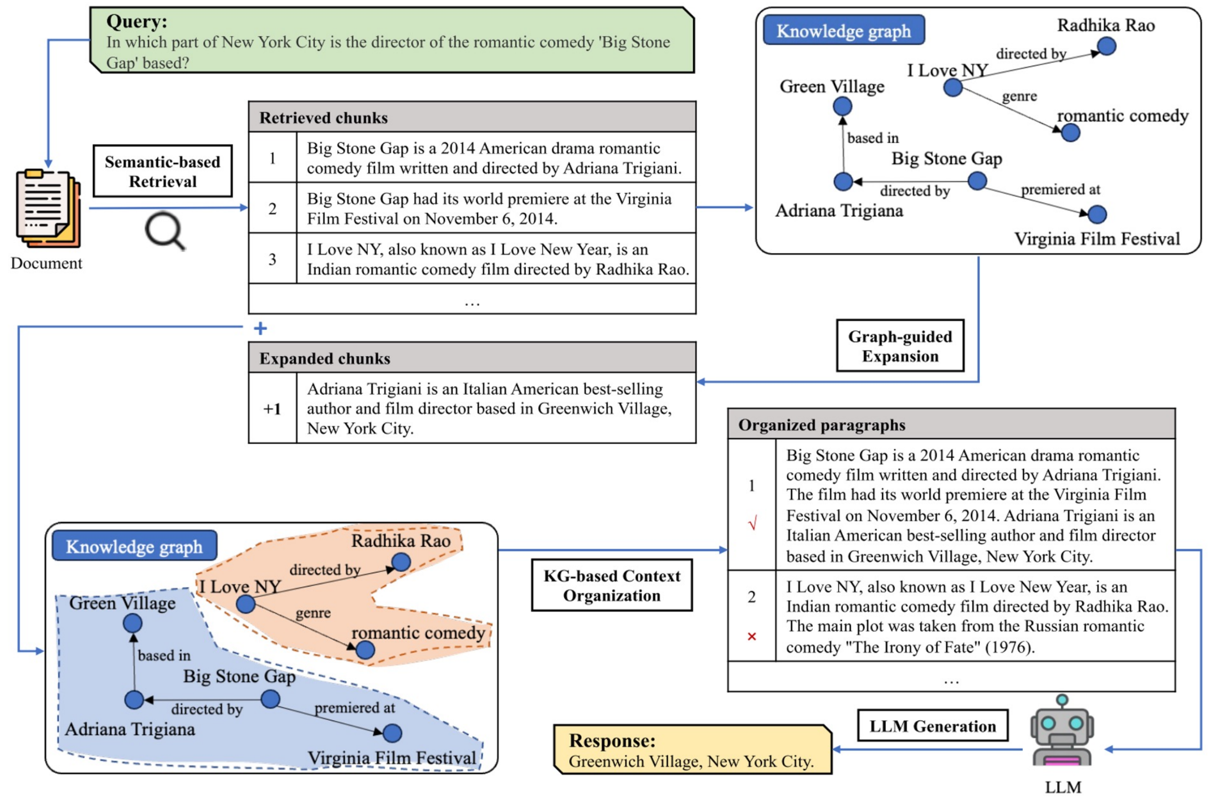


KGQA over a KG retrieval based on CoE

Soman, Karthik, et al. Biomedical knowledge graph-optimized prompt generation for large language models. *Bioinformatics* 40.9 (2024): btae560.
Sanmartin, Diego. KG-RAG: Bridging the gap between knowledge and creativity. *arXiv preprint arXiv:2405.12035* (2024).

KGs as Background Knowledge

- Retrieval Augmented Generation (RAG)
  - KG-guided RAG (KG^{2}RAG) [arXiv 2025]



- Text with available existing KG: establish linkage between text chunks and KG chunks

- Text without KG: extract entities and relations from text chunks to form subgraph

$$\mathcal{S} = \{s(q,c) \mid c \in \mathcal{D}\},$$

a. Semantic-based chunks retrieval

$$\mathcal{G}_q^0 = \{(h,r,t,c) \mid c \in \mathcal{D}_q\} \subseteq \mathcal{G}.$$

b. Retrieve the relevant subgraph from KG

$$\mathcal{G}_q^m = \text{traverse}(\mathcal{G}, \mathcal{G}_q^0, m),$$

c. Expand retrieved chunks with the m-hop BFS searched neighbor subgraphs on KG

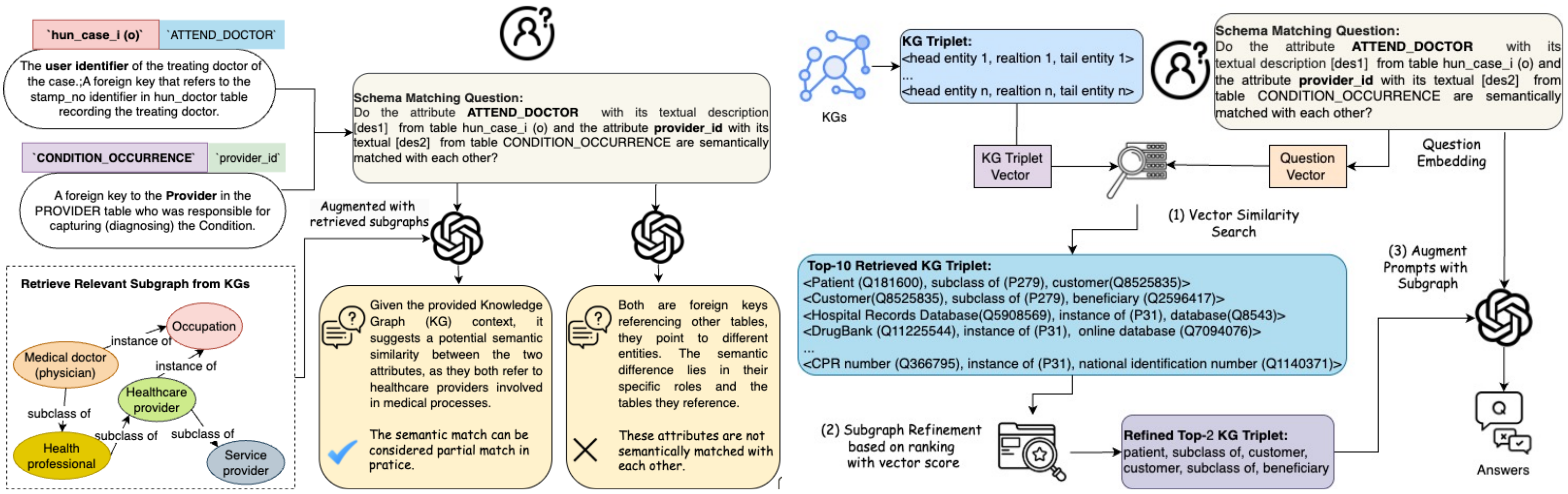
$$R(q, \mathcal{T}_i) = C(q, \text{conc}(\mathcal{T}_i)),$$

d. Rank the relevant expanded chunks and incorporates it with the retrieved chunks as context

Xiangrong, Zhu, et al. Knowledge Graph-Guided Retrieval Augmented Generation. *arXiv preprint arXiv:2502.068641* (2025).

KGs as Background Knowledge

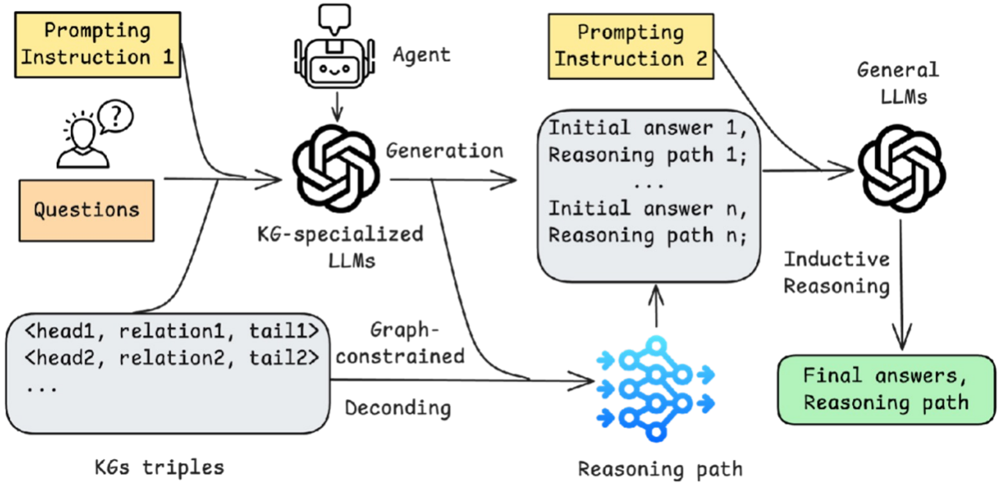
- Retrieval Augmented Generation (RAG)
  - KGRAG4SM: KG based RAG for Schema Matching [arXiv 2025]

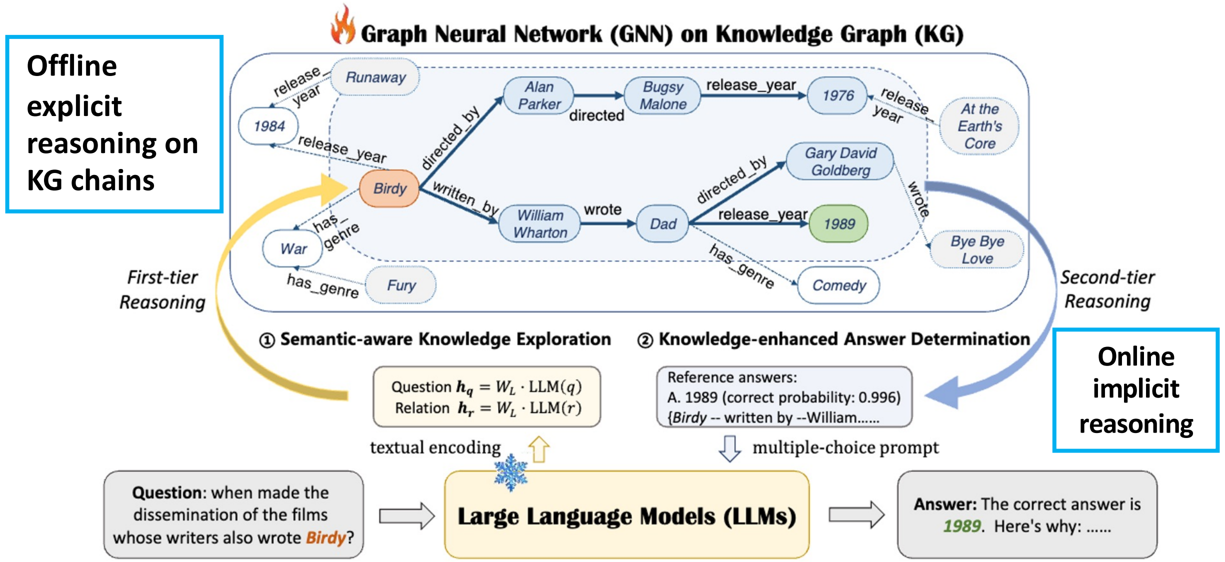


Chuangtao Ma, Sriom Chakrabarti, **Arijit Khan**, Bálint Molnár. Knowledge Graph-based Retrieval-Augmented Generation for Schema Matching. *arXiv preprint arXiv:* 2501.08686 (2025).

KGs as Reasoning Guidelines

- KGs serves as reasoning guidelines to LLMs for QA
  - **Offline KG guidelines:** KGs-based reasoning starts before LLMs reasoning
  - **Online KG guidelines:** KGs-based reasoning directly involves in LLMs reasoning
  - **Agent-based KG guidelines:** Agent-based autonomous reasoning





Liu, Guangyi, et al. Dual Reasoning: A GNN-LLM Collaborative Framework for Knowledge Graph Question Answering. CPAL, (2025).

KGs as Reasoning Guidelines

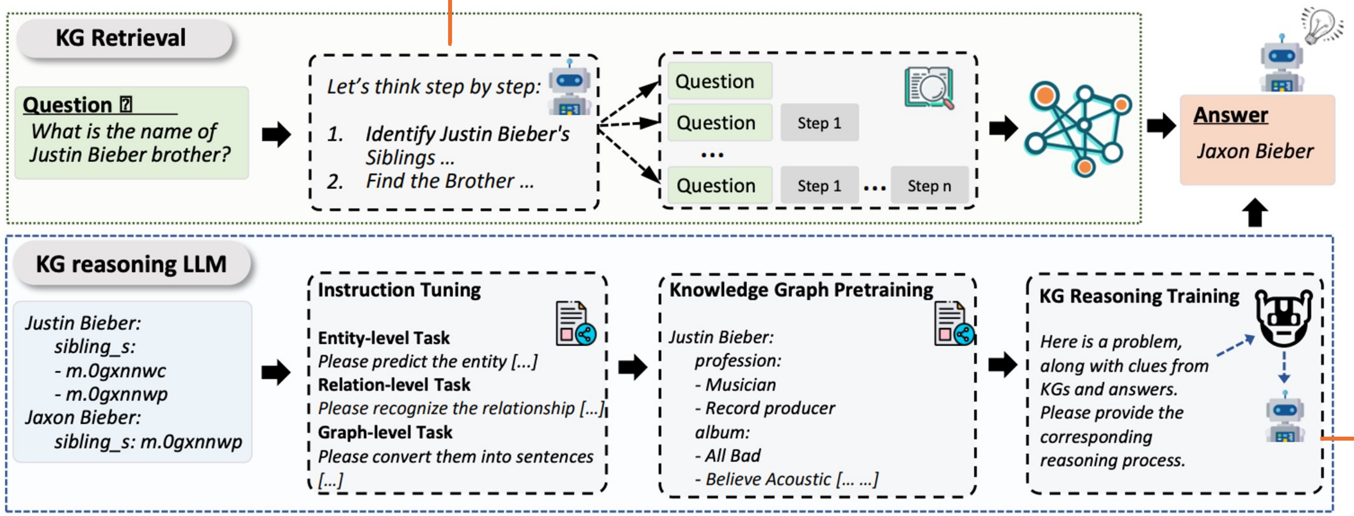
- Offline KG Guidelines
  - KG-based CoT Reasoning for KGQA [EMNLP, 2024]
    - Integrate the reasoning process and subgraphs into knowledge retrieval
    - Employ instruction tuning and continual pre-training to learn the KG reasoning

Prompt 1: Generating CoT for Retrieval

Please think step by step and then answer the given question.

Here are some examples:
Input: <Demonstration Question>
CoT: Let's think step by step. <Demonstration CoT>
Output: <Demonstration Answer>

Input: <Question>
CoT: Let's think step by step.



Prompt 2: Utilizing KG to Reason

Please think step by step and then answer the given question. Please keep the answer as simple as possible and return all the possible answers as a list. If there are hints, please combine this information to answer.

Here are some examples:
Input: <Demonstration Question>
Hints: <Demonstration Knowledge Graph>
CoT: Let's think step by step. <Demonstration CoT>
Output: <Demonstration Answer>

Input: <Question>
Hints: <Knowledge Graph>
CoT: Let's think step by step.

Instruction tuning on KG-to-text dataset

$$\mathcal{L}_{instruct} = -\sum_{l}^{L} y^l logp(\hat{y}^l|\mathcal{I}(x), y^{<l}),$$

$$\mathcal{L}_{pretrain} = -\sum_{l}^{L} x^l logp(\hat{x}^l|x^{<l}),$$

Serialize the KG in YAML format and train it for next token prediction

Ji, Yixin, et al. Retrieval and reasoning on KGs: Integrate knowledge graphs into large language models for complex question answering. *EMNLP*. (2024).

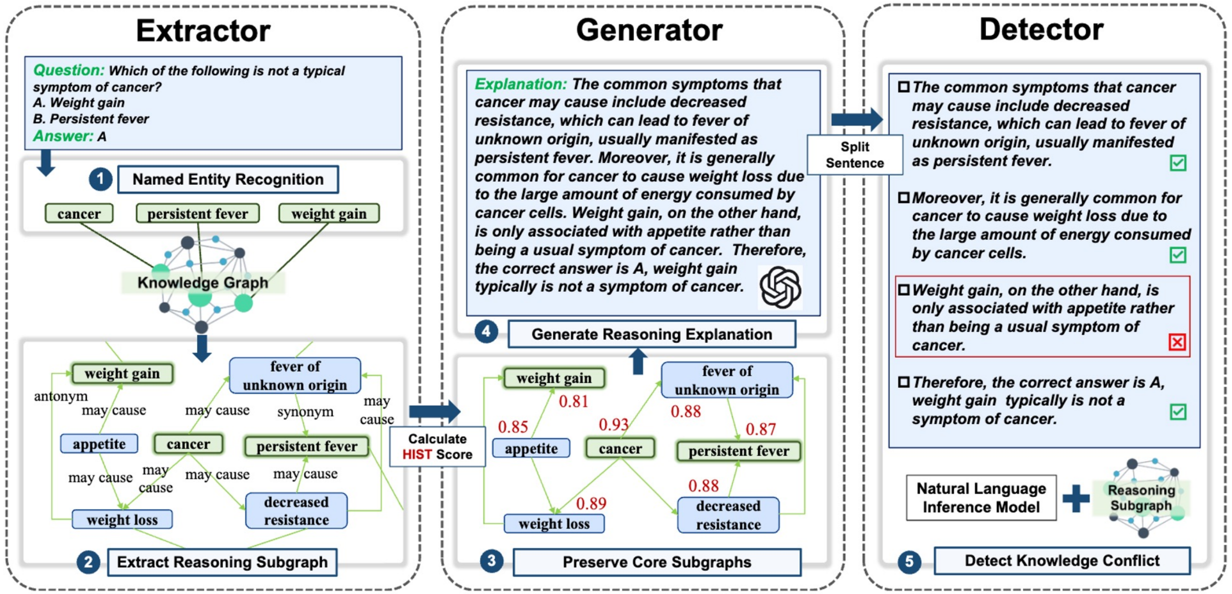
KGs as Reasoning Guidelines

- Online KG Guidelines
  - KG-SFT: KG augmented supervised fine-tuning LLM for KGQA [ICLR2025]
    - Search neighboring entities to obtain the reasoning subgraphs
    - Generate reasoning-based explanations via an external LLM
    - Detect knowledge conflict based on online reasoning (reasoning subgraph and natural language inference model)

(1) Perform NER on QA pairs for entity extraction

(2) Retrieve core subgraph from external KGs that is related to QA pairs vis HITS (Hyperlink-Induced Topic Search)

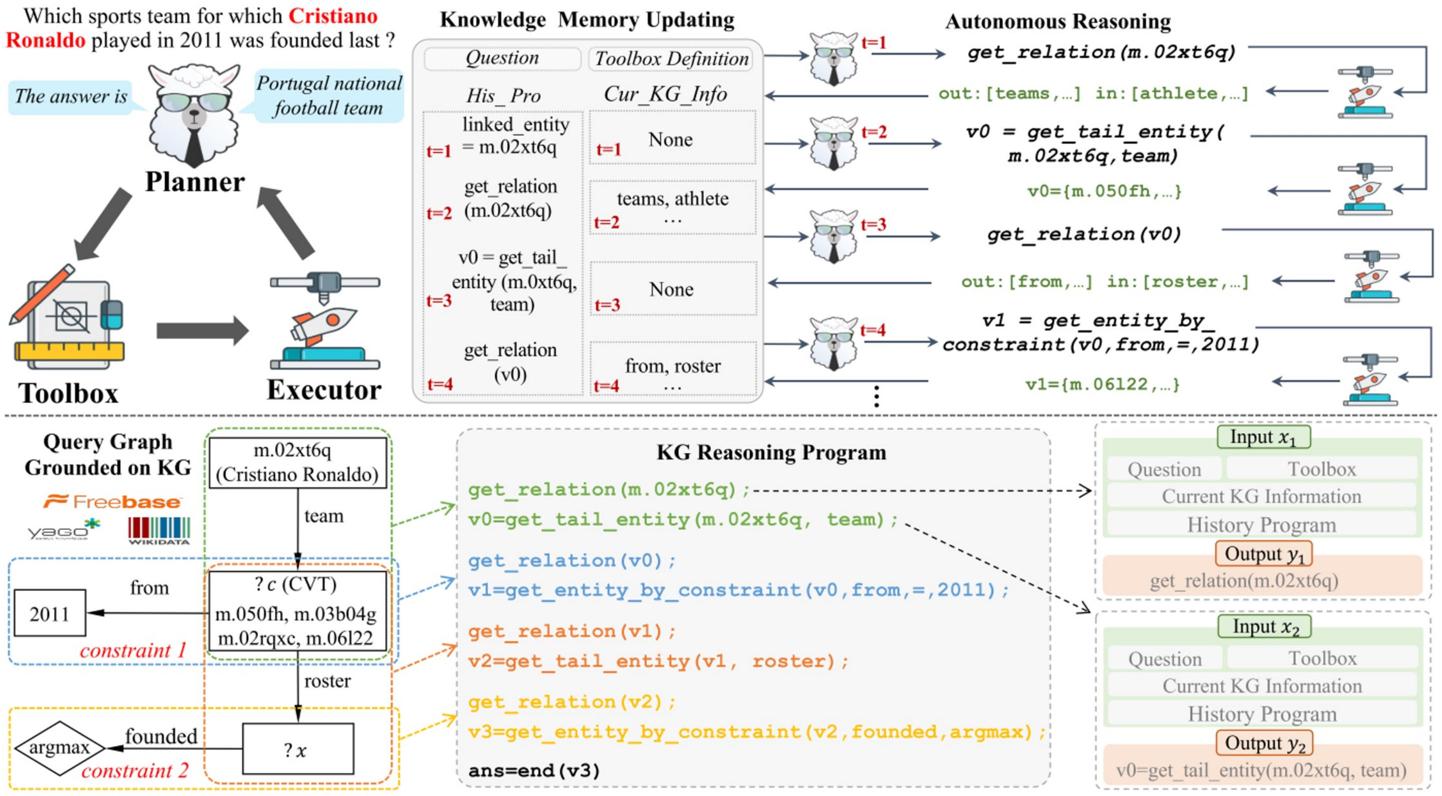
(3) Split the LLM generated reasoning explanations and fuse it with language inference model



Chen, Hanzhu, et al. Knowledge Graph Finetuning Enhances Knowledge Manipulation in Large Language Models. *ICLR*. (2025).

KGs as Reasoning Guidelines

- Agent based Reasoning
  - KG-Agent: Agent-based autonomous reasoning for KGQA [arXiv2024]



Instruction-tuned LLM for planner

KG-based executor for knowledge updating

Agent-based autonomous reasoning over KG

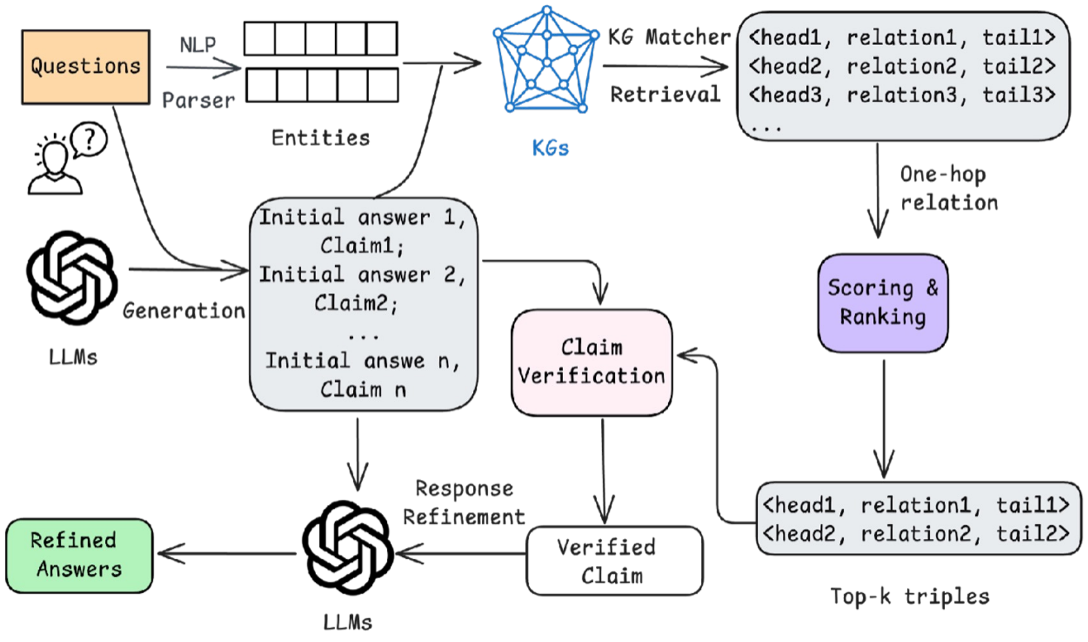
Agent autonomously iterates the tool selection and knowledge updating for reasoning util reaching the answer entities

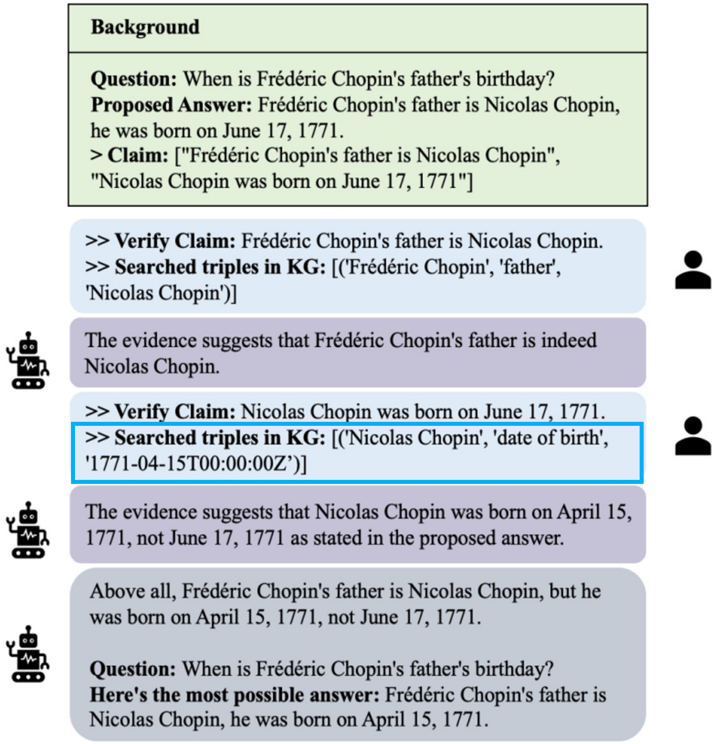
Example of instruction fine-tuning data synthesis and KG reasoning for the input-output pairs

Jiang, Jinhao, et al. KG-Agent: An efficient autonomous agent framework for complex reasoning over knowledge graph. *arXiv preprint arXiv:2402.11163* (2024).

KGs as Refiners and Validators

- Refine and validate the answers for QA
  - **KG-Driven Filtering and Validation:** validate and filter out the incorrect answers
  - **KG-Augmented Output Refinement:** refine intermediate output for final answer

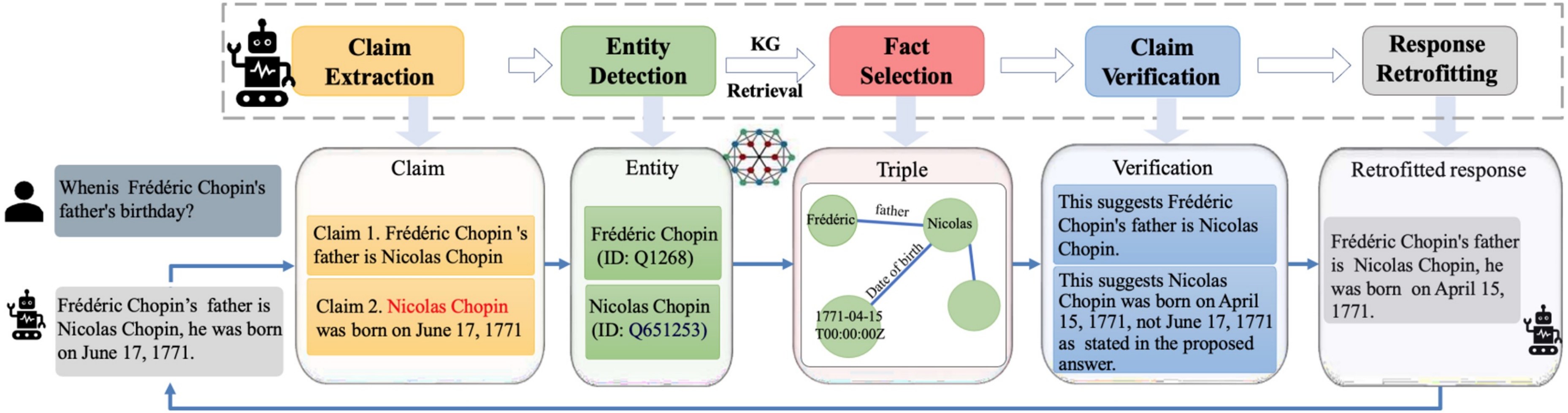




Guan, Xinyan, et al. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. AAAI. (2024).

KGs as Refiners and Validators

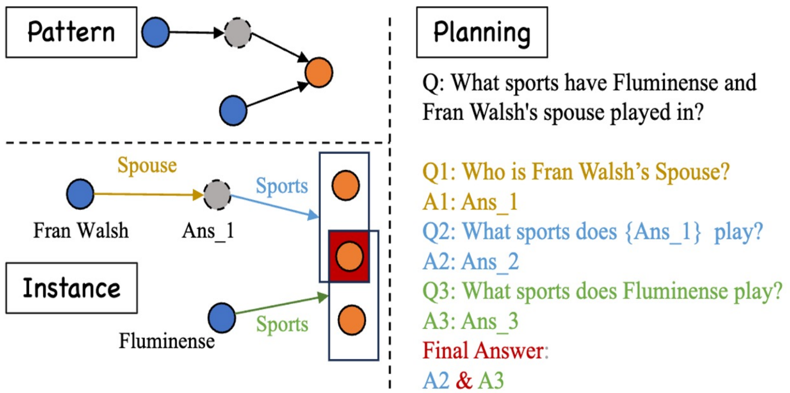
- KG-Driven Filtering and Validation
  - KGR: Knowledge Graph-based Answering Filtering [AAAI2024]
    - Leverage LLM to extract the claims in the generated draft response
    - Prompt LLMs to detect the entities that is relevant to the claims from KGs and extract the critical triples
    - Utilize LLM to compare and verify the model-generated claims with the KGs factual knowledge
    - Filter out the incorrect answer based on the verification suggestions

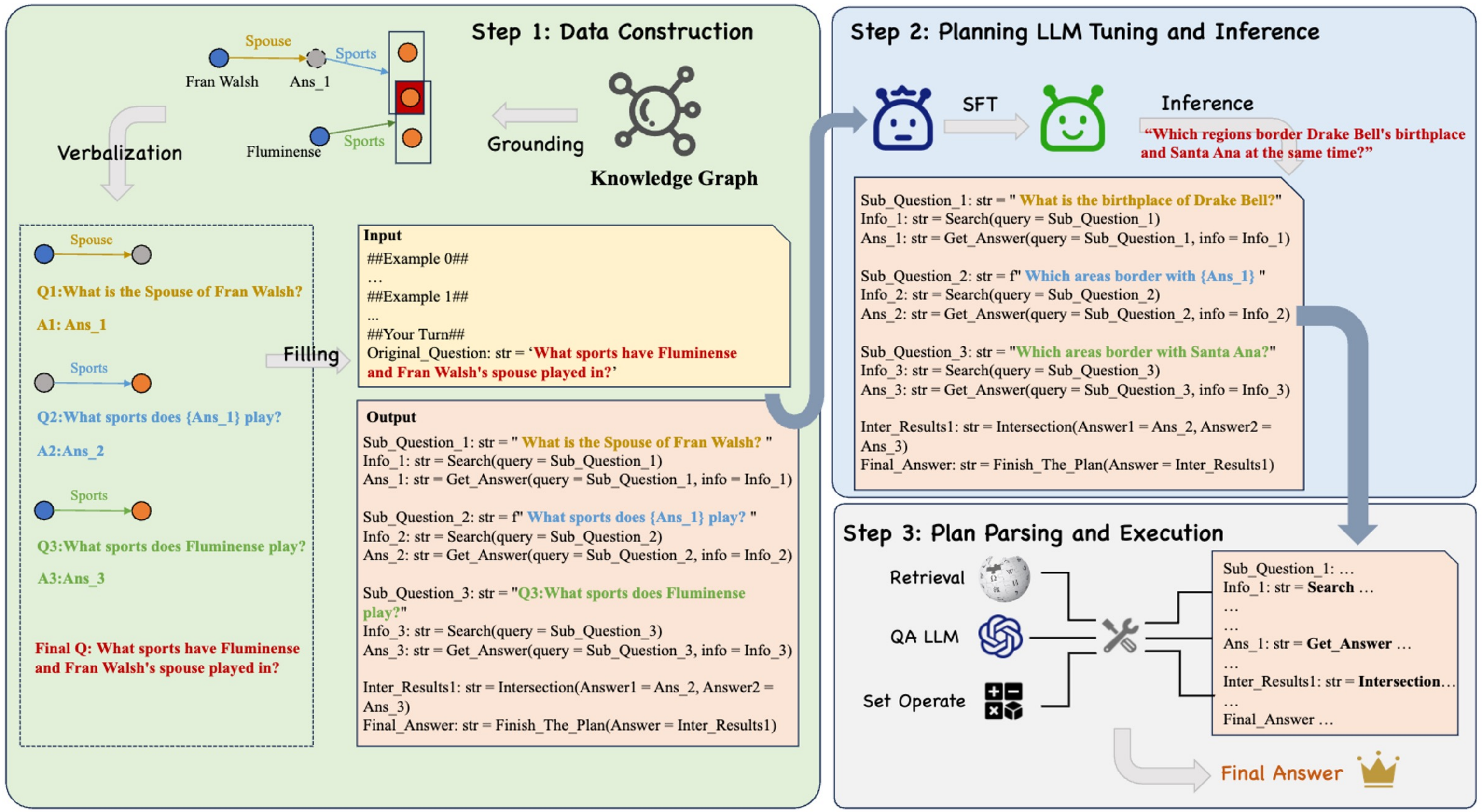


Guan, Xinyan, et al. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. AAAI. (2024).

KGs as Refiners and Validators

- KG-Augmented Output Refinement
  - LPKG: Retrieval-augmented LLMs for KGQA [arXiv2024]





Ground the instances from the defined KGs patterns

↓

Call LLMs to verbalize KG instance with sub-questions and complex question

↓

Fine-tuning LLMs with the prompts, input, outputs in few-shots learning

↓

Leverages the fined-tuned LLMs for inference and planning pipeline

↓

Refine the intermediate answers for final answer based on planning results

Wang, Junjie, et al. Learning to plan for retrieval-augmented large language models from knowledge graphs. *arXiv preprint arXiv:2406.14282. (2025)*.

Challenges

- LM and KG Alignment
  - *Joint learning:* **knowledge updates** are not supported and retraining is needed when the KGs or text changes.
  - *Effective knowledge fusion:* integrating LLMs and KGs with **prompt-based augmentation** is not the optimal solution, while the **knowledge conflicts** need to be mitigated.
- KG-RAG and Knowledge Retrieval
  - *Vector-based graph retrieval: creating embeddings and* vector-based search are **very expensive** tasks for large KGs.
  - *Query-based graph retrieval:* **converting NLQ to GQL is a challengeable** task as the specific KG schema structure is agnostic for LLMs.
- KG-guided Reasoning
  - *Complex reasoning*: reasoning over large-scale KGs is a **time-consuming and computing-consuming** task.
  - *Faithful reasoning:* generating the **reasoning paths from KGs relies on the prompt and tuning LLMs** while the faithful of the KG reasoning needs to be addressed.

Online Resources

Advanced Topics on LLM+KG for QA
Part -3

Yongrui Chen

Southeast University

Tutorial Outline

1) **Introduction (15 Min) – Arijit Khan**
- 1.1 Large Language Models (LLMs)
- 1.2 Knowledge Graphs (KGs)
- 1.3 Unifying LLMs+KGs
- 1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma
- 2.1 KGs as Background Knowledge
- 2.2 KGs as Reasoning Guidelines
- 2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**
- 3.1 Complex QA
- 3.2 Explainable QA
- 3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu
- 4.1 Performance Metrics
- 4.2 Benchmark Datasets
- 4.3 Industry Applications and Demonstrations

5) Opportunities for Data Management (10 Min) – Arijit Khan

6) Future Directions (5 Min) – Tianxing Wu

- **Q&A Session (10 Min)**

Contents

KG vs LLM – QA Capability Comparison

LLM QA

- **Code Pre-training:** enhance LLM reasoning during training
- **Prompt Engineering:** eliciting LLM reasoning during inference

KG QA

- Graph computing
- Rule-based reasoning
- Ontology reasoning
- Spatial-temporal reasoning
- KG embedding/GNN

LLM QA

- zero-shot prompting
- Few-shot prompting
- CoT prompting
- Instruction

+

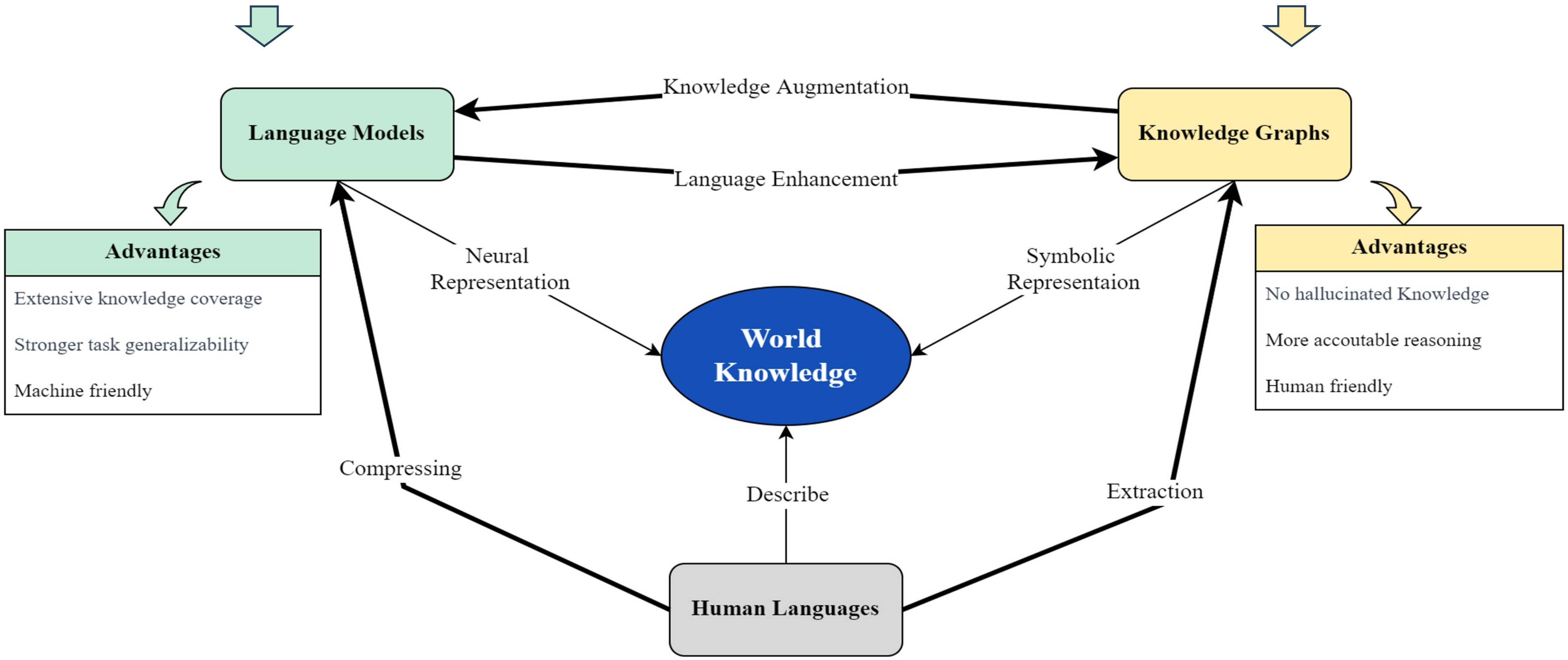
KG QA

- Graph computing
- Rule-based reasoning
- Ontology reasoning
- Spatial-temporal reasoning
- KG embedding/GNN

KG vs LLM – How do KG and LLM collaborate for QA?

Focus on scale
& has high coverage

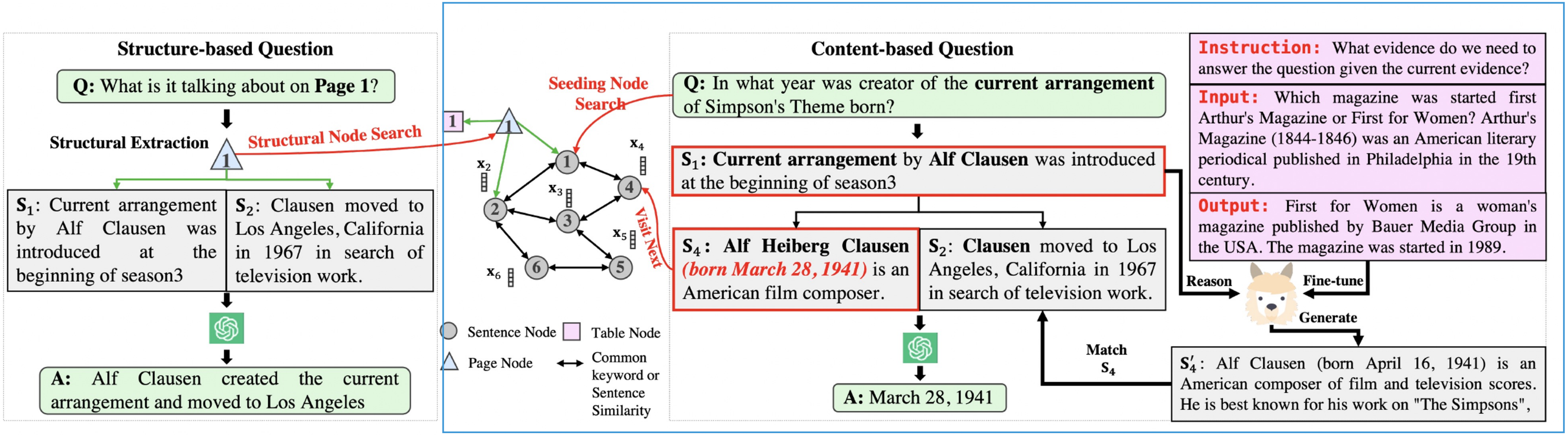
Focus on presentation
& has high accuracy



Large Knowledge Models: Perspective and Challenges, 2024.

Contents

Advanced Topics – QA over Multiple Documents

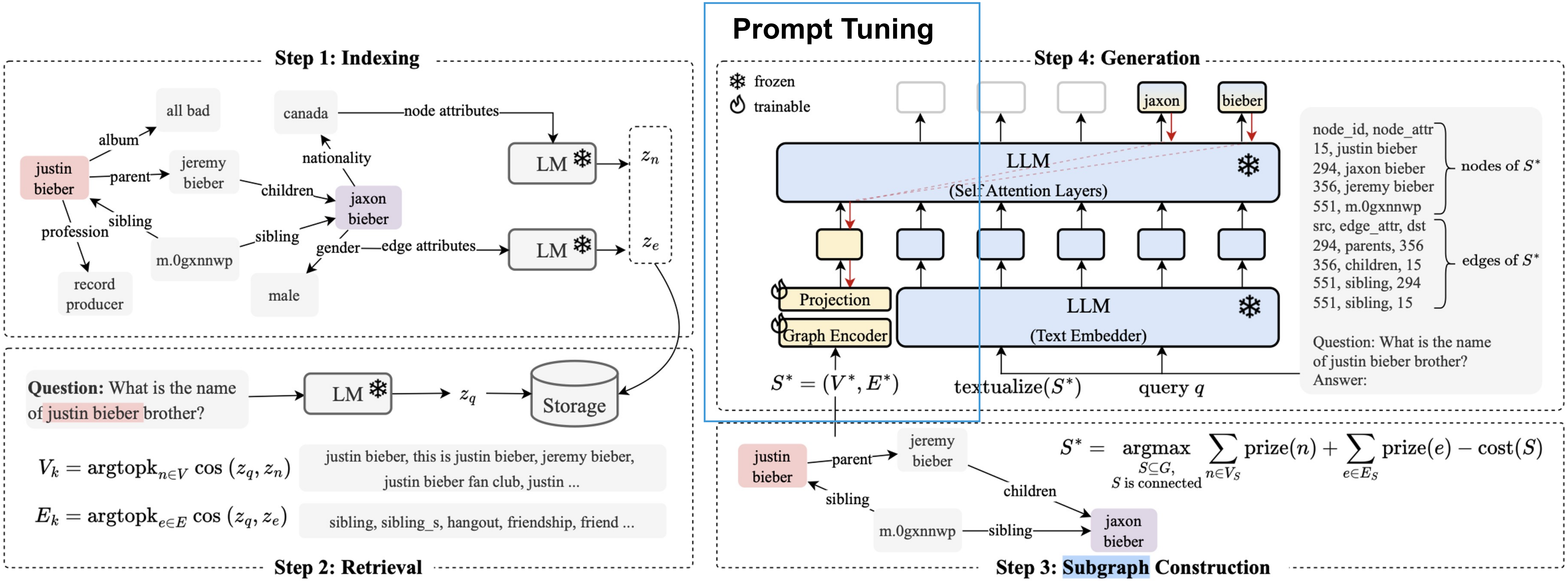


Enhancing LLMs for **Multi-Document QA**, which requires understanding logical associations across multiple documents.

- **KG Construction**: Building a KG where **nodes represent passages or document structures** (e.g., pages, tables) and **edges denote semantic/lexical similarity or structural relations** between them.
- **KG Traversal**: Employing an **LLM-based graph traversal agent** to navigate the KG, gathering relevant supporting passages to assist LLMs in answering questions.

Knowledge graph prompting for multi-document question answering. AAAI 2024

Advanced Topics – Retrieval Augment Generation



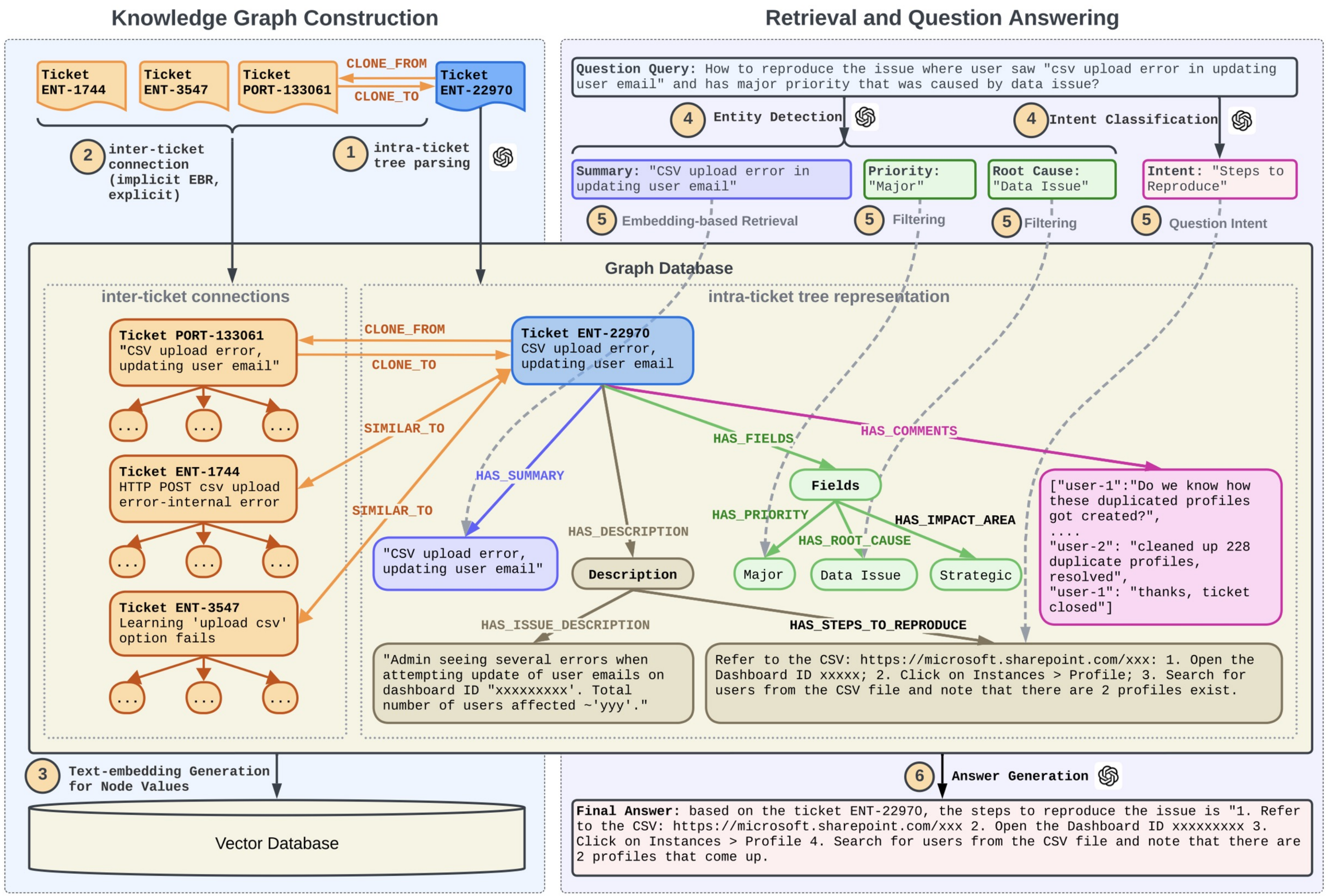
- The method involves four main steps: **indexing** the graph, **retrieving** relevant nodes and edges, **constructing a connected subgraph**, and **generating** the answer using the retrieved subgraph and the query.
- By employing RAG for **direct information retrieval from the actual graph**, G-Retriever effectively **mitigates hallucination** in graph-based question answering.

G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering . Preprint 2024.

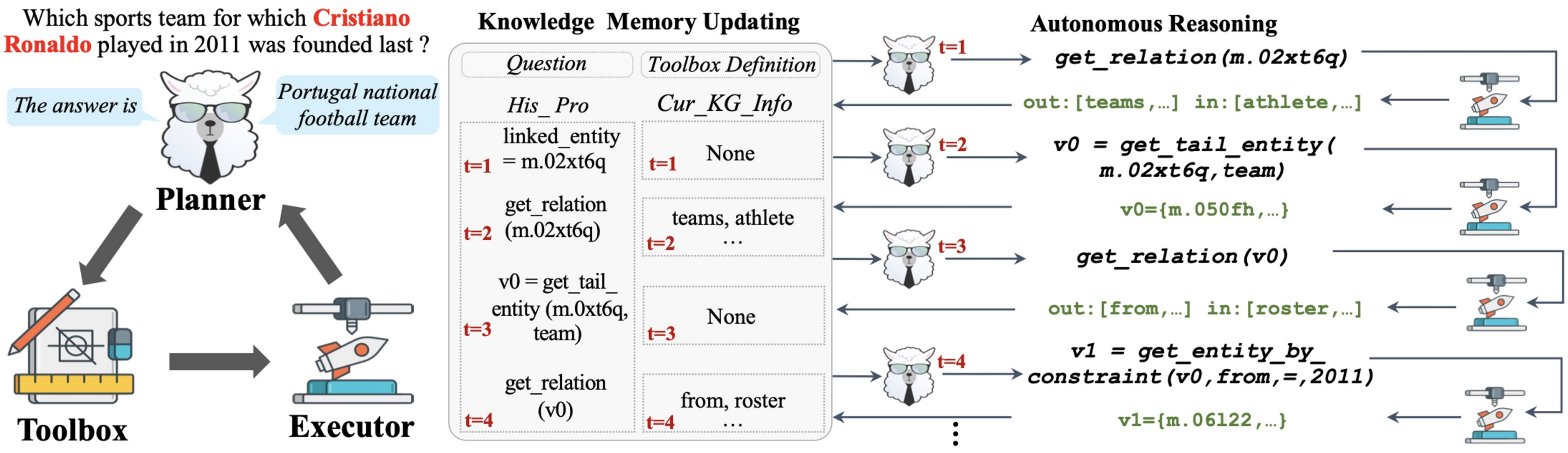
Advanced Topics – Retrieval Augment Generation

Enhancing the conventional RAG approach by integrating a **knowledge graph** constructed from **historical customer service issue tickets** to improve retrieval accuracy and answer quality.

- Consumer queries are parsed to identify **named entities and intents**.
- The system retrieves related sub-graphs from the KG based on the parsed query, leveraging **both entity matching and embedding similarity**.
- An LLM generates answers using the **retrieved sub-graphs** as context.



Advanced Topics – KG Agent

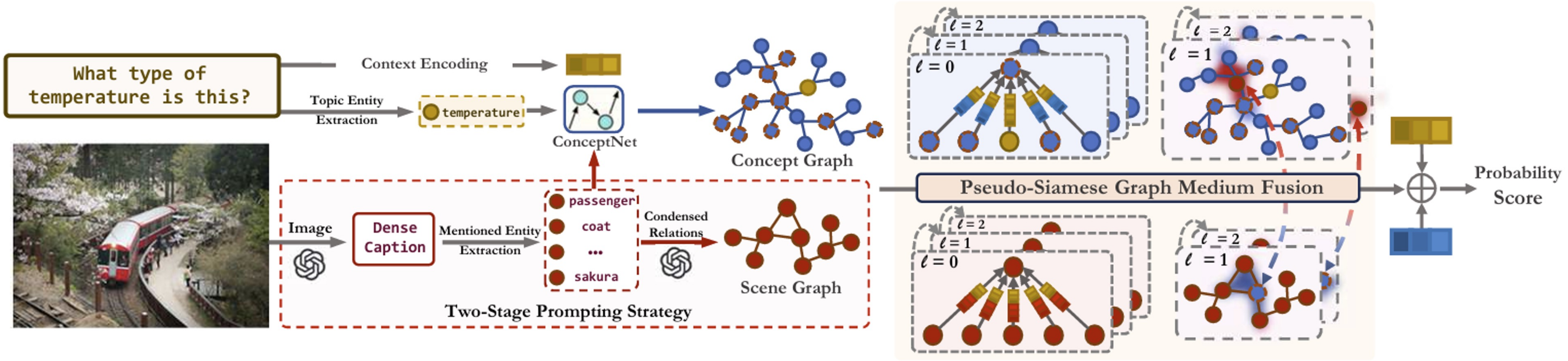


Integrates a **small LLM (e.g., 7B)**, a **multifunctional toolbox**, a **KG-based executor**, and **knowledge memory**.

- Employs an **iterative mechanism** where the LLM autonomously selects a tool from the toolbox and updates the knowledge memory to continue reasoning over the KG until the answer is found.
- **Multifunctional Toolbox**: Extends the LLM's capacity to manipulate structured data by providing tools for **extraction, semantic understanding, and logic operations** on KG data and intermediate results (e.g., filtering, counting, retrieval, relation retrieval, entity disambiguation).

KG-Agent: An Efficient Autonomous Agent Framework for Complex Reasoning over Knowledge Graph. Preprint 2024

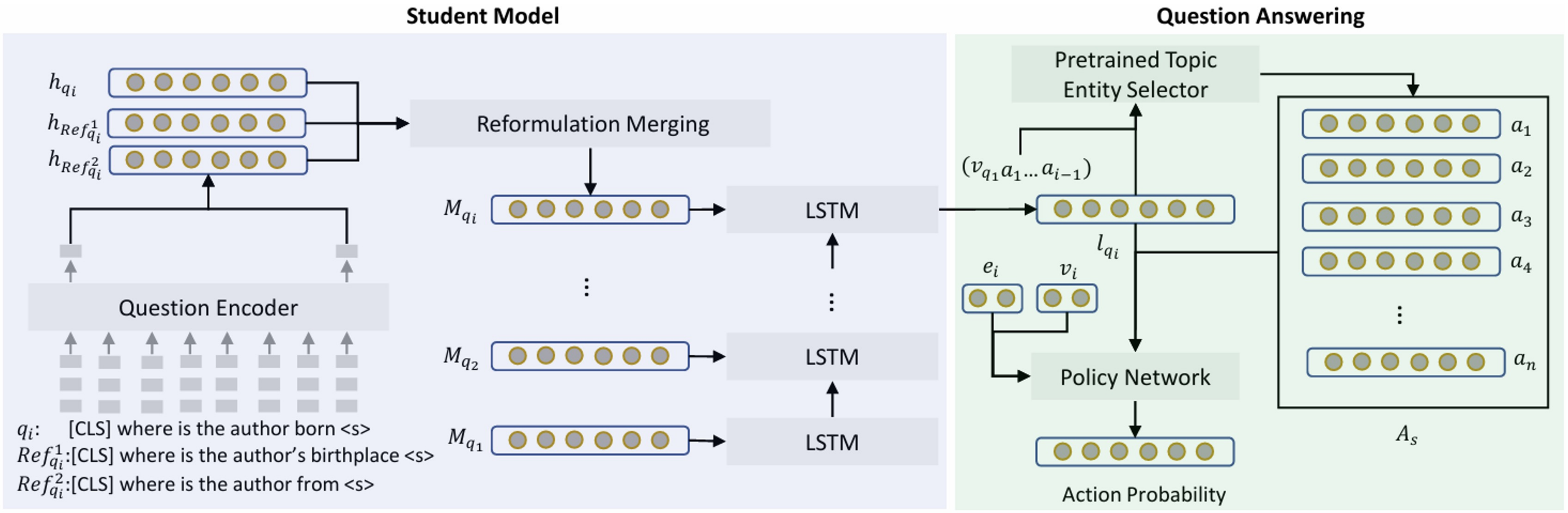
Advanced Topics - Visual QA



- **Two-Stage Prompting**: Utilizing LLMs to generate a **dense image caption** and subsequently extract a **scene graph** containing detailed visual features from it.
- **Coupled Concept Graph**: Constructing a **concept graph** using **ConceptNet**, linking scene graph entities with external knowledge.
- **Pseudo-Siamese Graph Medium Fusion (PS-GMF)**: Utilizing **shared entities as mediums** between the scene graph and concept graph to achieve **cross-modal information exchange** and **fusion.**

Modality-aware integration with large language models for knowledge-based visual question answering. ACL 2024.

Advanced Topics – Conversational QA

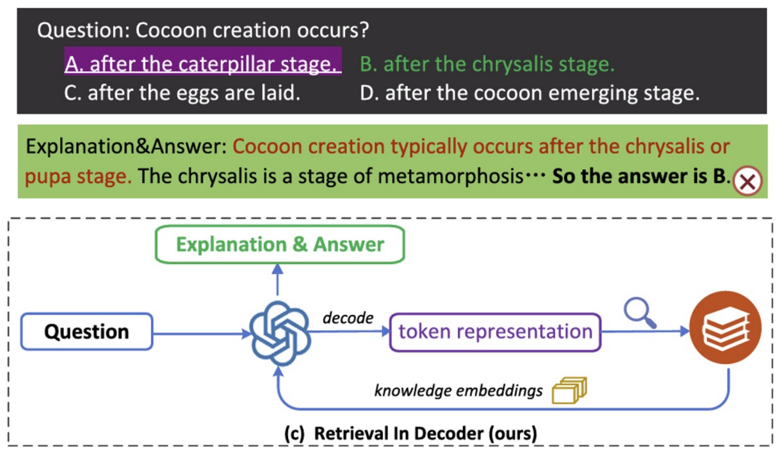


- A **teacher model** is trained directly using **human-written reformulations** to learn effective question representations.
- A **student model**, with the same architecture, is trained to **mimic the teacher's output** using the **LLM-generated reformulations**. This helps the student model approach the performance of the teacher model, even with potentially lower-quality LLM-generated reformulations.

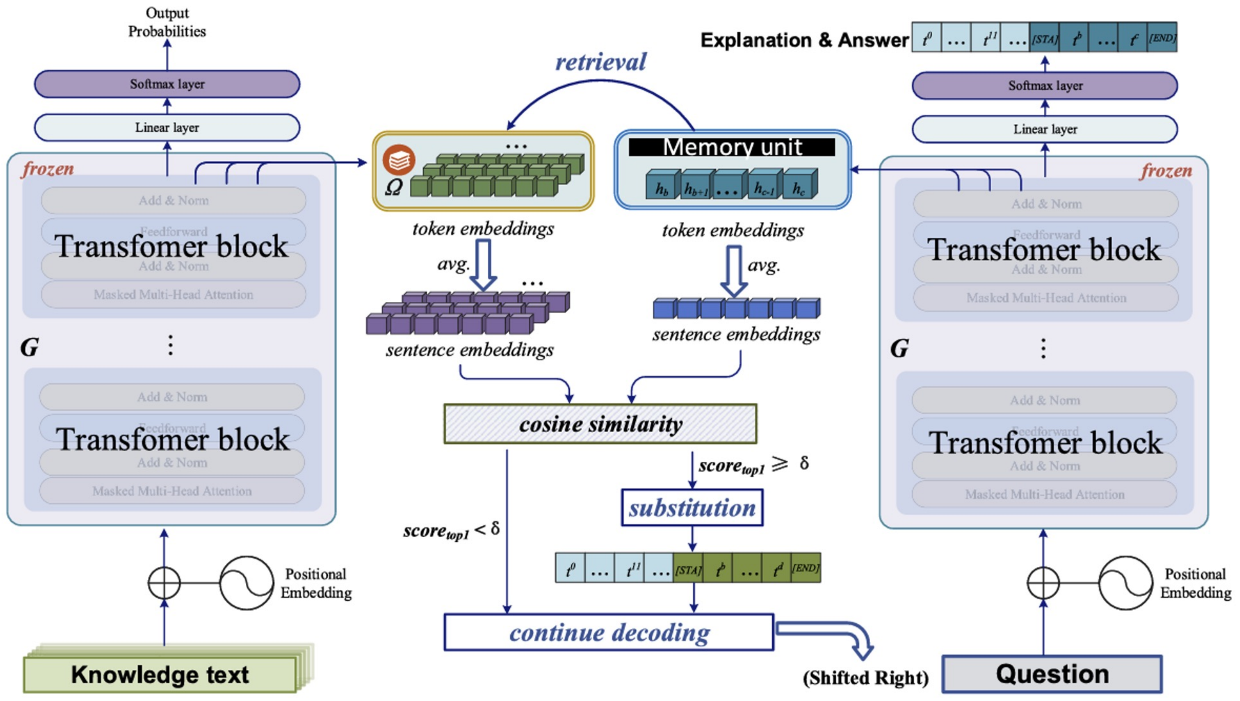
Conversational Question Answering with Language Models Generated Reformulations over Knowledge Graph. ACL Findings 2024.

Advanced Topics – Explainable QA

To enhance the **faithfulness and credibility** of generative models in QA, which contributes to explainability.

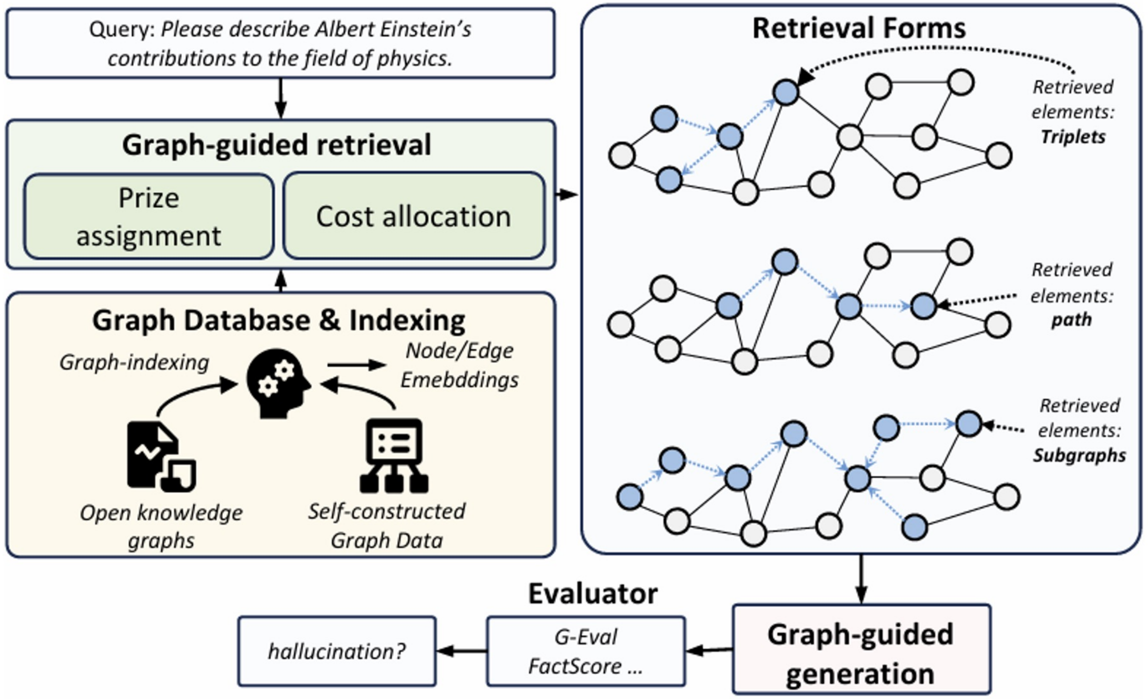


- **Integrated Retrieval:** Integrates information retrieval directly into the decoding process of generative language models, rather than treating them as separate components.
- **Multi-Granularity Decoding:** Supports dynamic adjustment of decoding granularity between token-level and sentence-level based on retrieval outcomes.
- **Rationale-Aware Explanation Generation:** Employs prompt learning to generate explanations that explicitly contain marked rationales.



Advanced Topics – Explainable QA

Goal: Enhancing the **trustworthiness** of LLMs in open-ended question answering by integrating **KGs.**



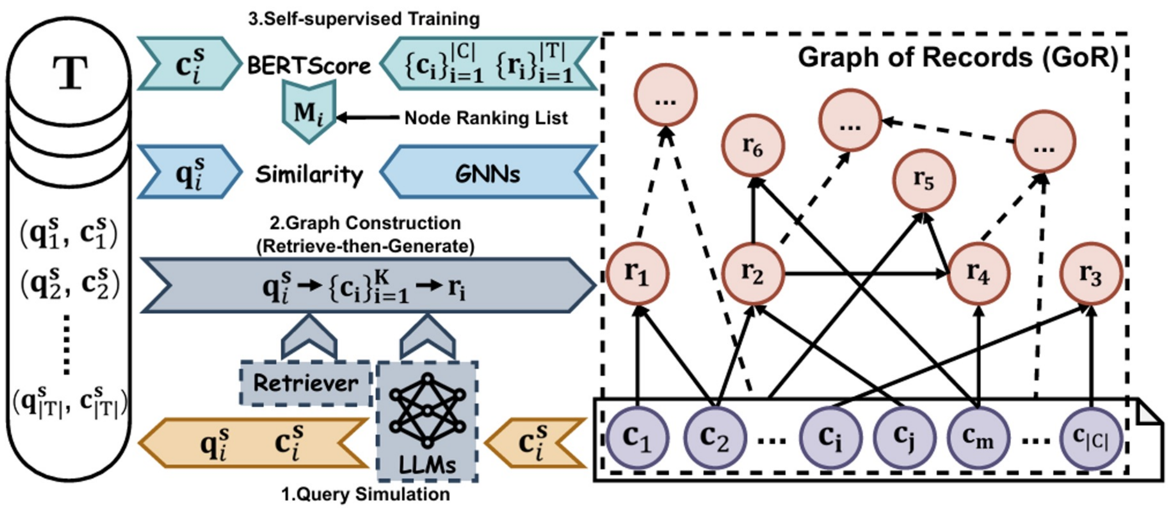
- **Explainability via Knowledge Source:** KGs provide structured and explicit factual information. Each piece of data in a KG can be traced back to its source, offering provenance.
- **Transparency in Reasoning:** The traceability of KG information not only enables verification of the model's reasoning but also brings transparency to the decision-making process.
- **Open-ended Answers with Supporting Facts:** The OKGQA benchmark encourages LLMs to generate more elaborate answers, including reasoning paths and supporting facts derived from the KG.

Can Knowledge Graphs Make Large Language Models More Trustworthy? An Empirical Study Over Open-ended Question Answering. Preprint 2024.

Contents

Optimization and Efficiency – Index-based Optimization

Goal: To enhance **RAG** performance in long-context global summarization by using a graph structure built from **LLM-generated historical responses**.



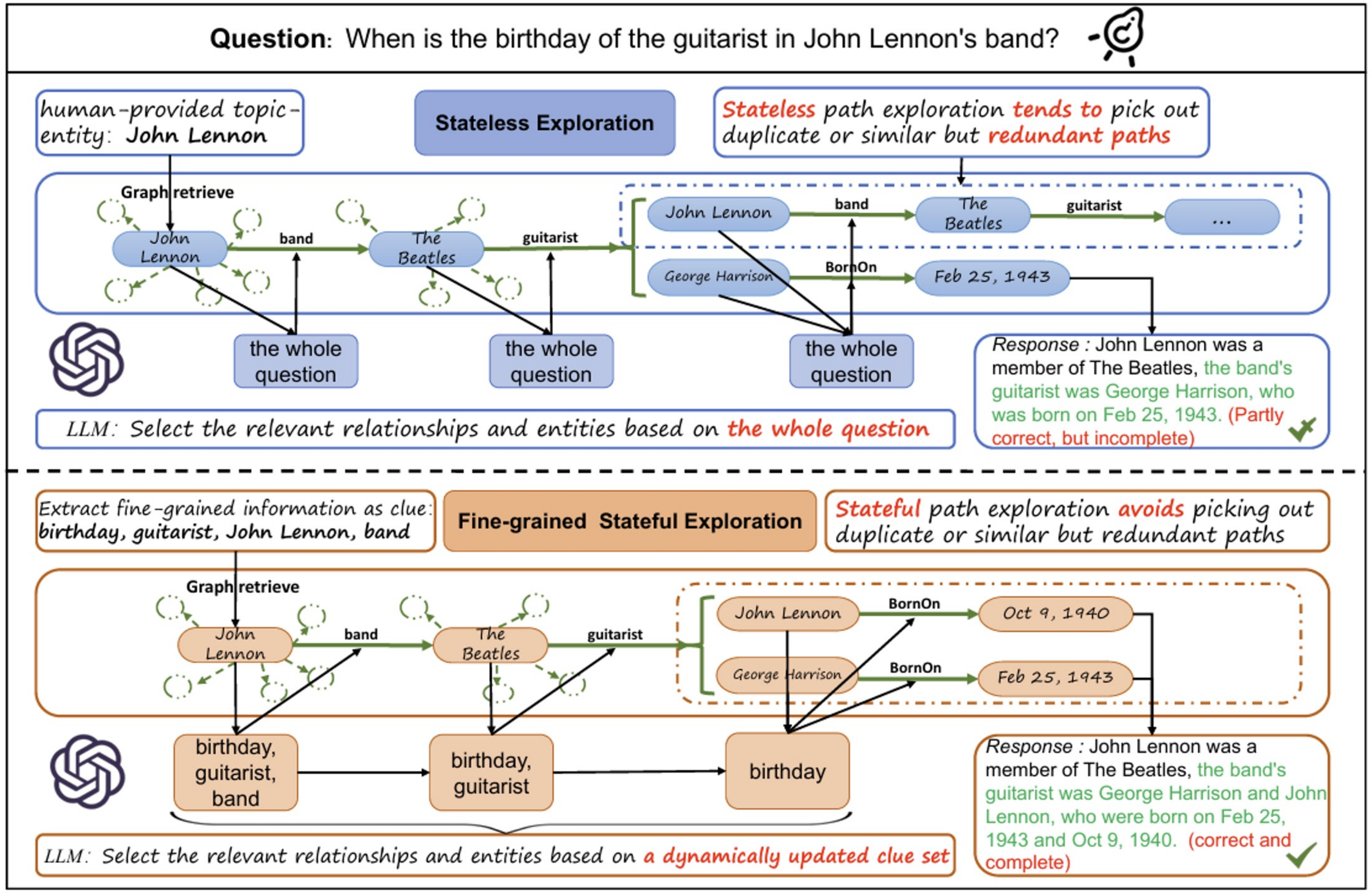
- Simulate user queries, retrieve relevant text chunks, and establish edges between the retrieved text chunks and their corresponding **LLM-generated responses** to construct a **Graph of Records**.
- Utilize a **GNN** to learn embeddings for the nodes in the graph, capturing fine-grained correlations.
- Effectively discovers and leverages **fine-grained correlations between LLM historical responses and text chunks**, thereby improving RAG performance.

Boosting retrieval augmented generation for long-context summarization with graphs. Preprint 2024.

Optimization and Efficiency – Graph Retrieval-based Optimization

Goal: Addresses the **information granularity mismatch** between questions and knowledge graphs, which is identified as a primary source of inefficiency in existing methods.

- Extracts **fine-grained, independent pieces of information (clues)** from the question to guide the retrieval process.
- By avoiding redundancy and ensuring no pertinent information is overlooked, the method **significantly reduces the average number of LLM calls** required for knowledge retrieval compared to existing stateless iterative exploration methods

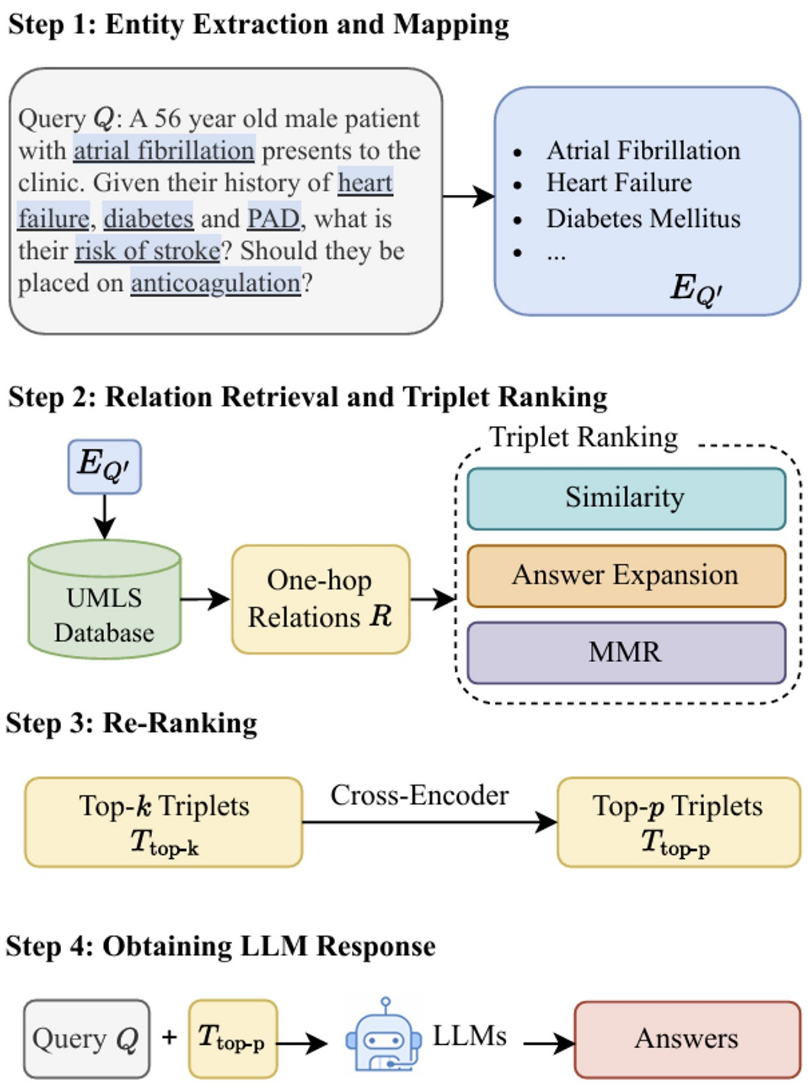


Clue-Guided Path Exploration: Optimizing Knowledge Graph Retrieval with Large Language Models to Address the Information Black Box Challenge. Preprint 2024.

Optimization and Efficiency – Ranking-based Optimization

Goal: Leverages **ranking and re-ranking techniques** to refine the selection and ordering of relevant information retrieved from the medical KG.

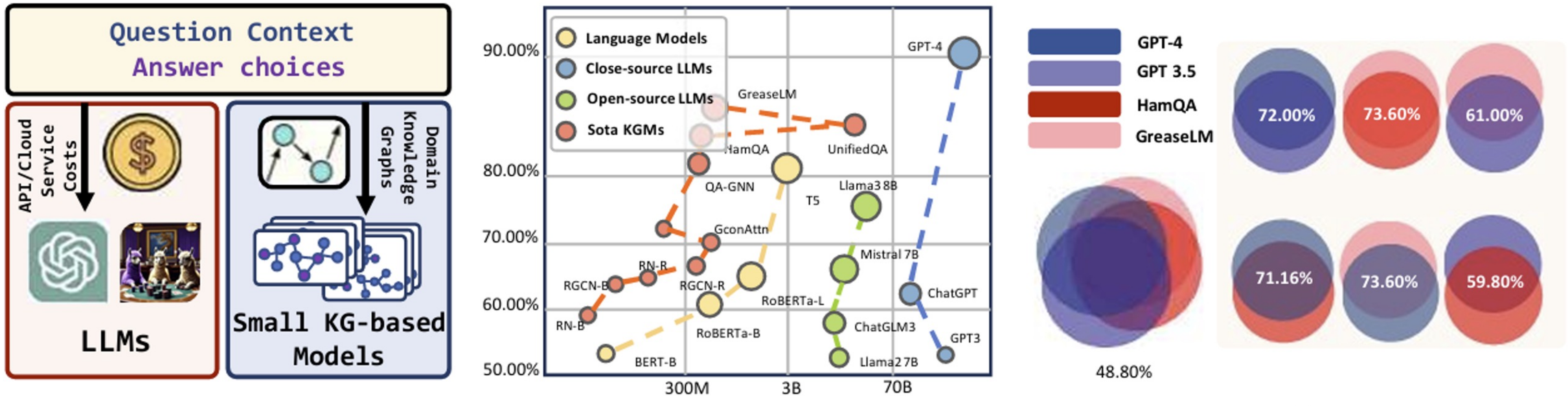
- **Similarity Ranking**: Ranks triplets based on their semantic similarity to the input question using UmlsBERT embeddings.
- **Answer Expansion Ranking**: Uses an LLM to generate a preliminary answer, then ranks triplets based on their similarity to the expanded question-answer context. This helps in identifying information relevant to the potential answer.
- **MMR Ranking**: Selects triplets based on both their relevance to the question and their dissimilarity to already selected triplets, promoting diversity and reducing redundancy.



KG-Rank: Enhancing Large Language Models for Medical QA with Knowledge Graphs and Ranking Techniques. BioNLP 2024.

Optimization and Efficiency – Cost-based Optimization

Goal: To achieve **cost-efficient KBQA** by minimizing the usage and expenses associated with LLMs.



(a) (b) (c)

- **Multi-Armed Bandit Formulation:** Models the model selection problem as a tailored multi-armed bandit problem to balance exploration (trying different models) and exploitation (using the best-performing models) within a limited budget.
- **Accuracy Expectation with Cluster-Level Thompson Sampling:** Estimates the accuracy expectation of choosing either LLMs or KGMs based on their historical success and failure rates. This helps in initially guiding the policy towards more promising model types.
- **Context-Aware Policy:** Learns a context-aware policy that considers the semantics of the question to further distinguish and select the most suitable expert model (either an LLM or a KGM) for that specific question.

Cost-efficient Knowledge-based Question Answering with Large Language Models. NeurIPS 2024.

Contents

Conclusion & Future Work

Conclusion

- **LLM-KG Integration Enhances QA**: Combining LLMs with KGs improves multi-document and multimodal QA by enhancing reasoning, reducing hallucinations, and increasing answer accuracy.
- **Optimization Improves Efficiency**: Techniques like index-based and graph retrieval-based optimization boost system efficiency, scalability, and cost-effectiveness.
- **Conversational and Explainable QA**: QA systems are evolving into multi-turn, explainable models with KG Agents enabling transparent and trustworthy reasoning.

Future Work

- **Deeper LLM-KG Fusion**: Advancing dynamic KG updates and adaptive retrieval will improve knowledge adaptation and model performance.
- **Enhanced Multimodal QA**: Future systems will better integrate text, images, and videos for richer reasoning and more comprehensive answers.
- **Scalable and Privacy-Preserving QA**: Efficient, large-scale QA solutions leveraging federated learning and edge computing will enhance privacy and real-time capabilities.

Online Resources

Evaluations and Applications
Part - 4

Tianxing Wu

Southeast University

AALBORG UNIVERSITET

Tutorial Outline

1) **Introduction (15 Min) – Arijit Khan**
   1.1 Large Language Models (LLMs)
   1.2 Knowledge Graphs (KGs)
   1.3 Unifying LLMs+KGs
   1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma
   2.1 KGs as Background Knowledge
   2.2 KGs as Reasoning Guidelines
   2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**
   3.1 Complex QA
   3.2 Explainable QA
   3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu
   4.1 Performance Metrics
   4.2 Benchmark Datasets
   4.3 Industry Applications and Demonstrations

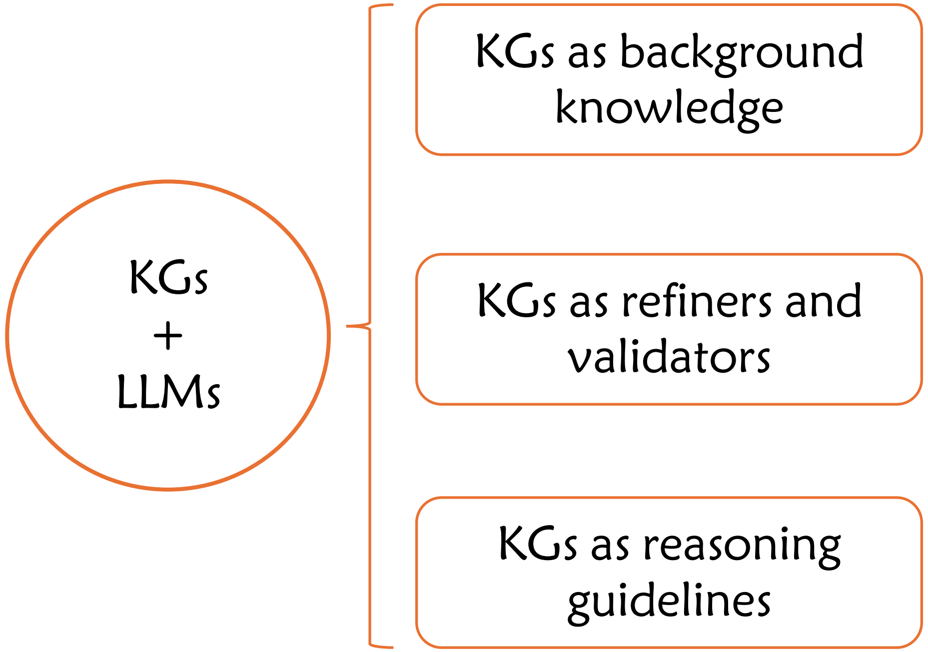
5) Opportunities for Data Management (10 Min) – Arijit Khan

6) Future Directions (5 Min) – Tianxing Wu

- **Q&A Session (10 Min)**

Performance Metrics

- Some metrics have been proposed to measure different aspects of **LLM + KGs for QA**.
- According to the roles of KGs, the metrics are categorized into three types, which respectively measure the **Answer Quality**, the **Retrieval Quality** of RAG, and the **Reasoning Quality.**

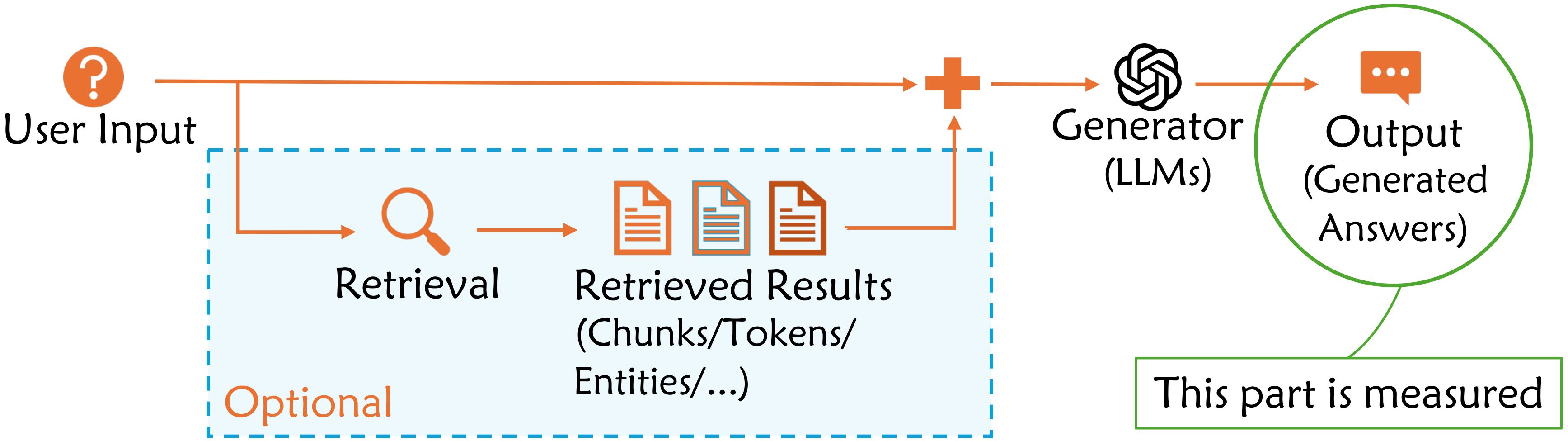


- **Answer Quality**: Measuring the accuracy of the generated answer and its relevance to the context or the question.
- **Retrieval Quality**: Measuring the accuracy of the retrieval process or the relevance of retrieved content to the question.
- **Reasoning Quality**: Measuring the accuracy of the reasoning steps in multi-hop reasoning scenarios.

Performance Metrics

Metrics measuring the **Answer Quality**:

- ***BERTScore***: Assess the semantic similarity between generated answers and the reference text, utilizing their contextual embeddings from pre-trained transformers (e.g., $BERT$) , computing the cosine similarity between the embeddings as $BERTScore$.
- $\boldsymbol{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|} \frac{1}{rank_i}$, the average reciprocal rank of the first correct answer across a set of queries, where $|Q|$ is the number of queries and $rank_i$ is the rank position of the first correct answer for the i-th query.



Peng, B., Zhu, Y., Liu, Y., Bo, X., Shi, H., Hong, C., ... & Tang, S. (2024). **Graph retrieval-augmented generation: A survey**. *arXiv preprint arXiv:2408.08921*.

Performance Metrics

Metrics measuring the **Answer Quality**:

- ***Faithfulness***: Prompt LLMs to extract a set of statements from an answer, and to determine whether each statement can be inferred from its context. Faithfulness is defined as $F = \frac{|V|}{|S|}$, where $|V|$ is the number of statements supported by the LLM and $|S|$ is the total number of statements.

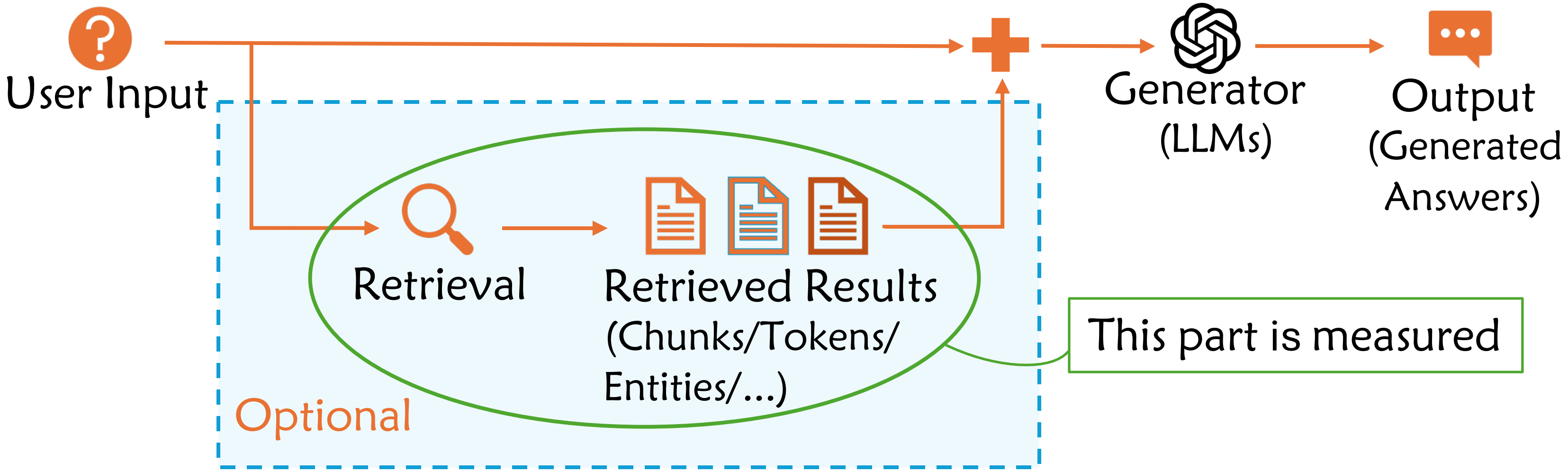
- ***Answer Relevance***: $AR = \frac{1}{n} \sum_{i=1} sim(q, q_i)$, where q_i is potential questions generated for the answer to q, and $sim(q, q_i)$ measures the cosine similarity between their embeddings.

Es, S., James, J., Anke, L. E., & Schockaert, S. (2024, March). **Ragas: Automated evaluation of retrieval augmented generation**. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations* (pp. 150-158).

Performance Metrics

Metrics measuring the **Retrieval Quality** of RAG:

- $\boldsymbol{Precision} = \frac{TP}{TP+FP}$, the fraction of relevant instances among the retrieved instances. (TP: true positives, FP: false positives)
- $\boldsymbol{Recall@k} = \frac{|RD \cap Top_{kd}|}{|RD|}$, the fraction of relevant instances that have been retrieved over the total amount of relevant cases, considering only the top-k results. (RD: relevant documents, Top_{kd}: top-k retrieved documents)



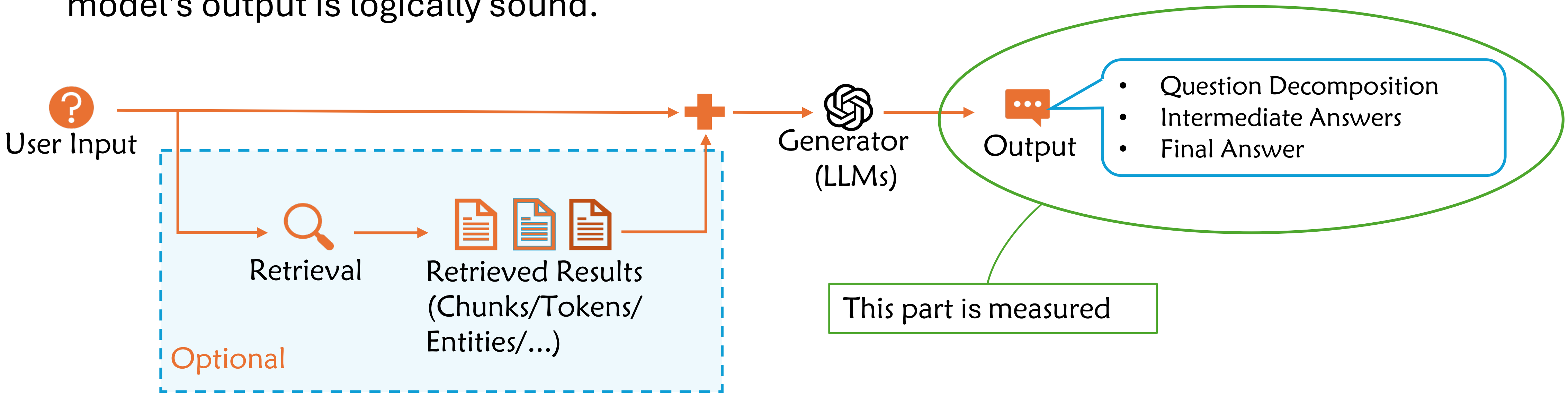
Yu, H., Gan, A., Zhang, K., Tong, S., Liu, Q., & Liu, Z. (2024, August). **Evaluation of retrieval-augmented generation: A survey**. In *CCF Conference on Big Data* (pp. 102-120).

Performance Metrics

Metrics measuring the **Reasoning Quality**:

Multi-hop QA: Hop-wise answering accuracy (Hop-Acc)

- $\text{Hop-Acc} = \frac{N_\text{c}}{N_\text{t}}$. N_c is the number of samples where the reasoning path matches the gold path, and N_t is the total number of evaluated samples.
- Hop-Acc measures whether the reasoning process for multi-hop questions follows the correct sequence of logical steps.
- A higher Hop-Acc indicates more rational and coherent reasoning, ensuring that the model's output is logically sound.



Gu, H., Zhou, K., Han, X., Liu, N., Wang, R., & Wang, X. (2024, August). **PokeMQA: Programmable knowledge editing for Multi-hop Question Answering**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 8069-8083).

Performance Metrics

Metrics measuring the **Reasoning Quality**:

Multi-hop QA: Hop-wise answering accuracy (Hop-Acc)

- $\text{Hop-Acc} = \frac{N_\text{c}}{N_\text{t}}$. N_c is the number of samples where the reasoning path matches the gold path, and N_t is the total number of evaluated samples.
- Hop-Acc measures whether the reasoning process for multi-hop questions follows the correct sequence of logical steps.
- A higher Hop-Acc indicates more rational and coherent reasoning, ensuring the model's output is logically sound.

Input Question: Which continent is the football club Messi plays for located in?

↓

Subquestion 1: What is the football club Messi plays for? **Answer 1**: Inter Miami.

↓

Subquestion 2: Which continent is Inter Miami located in? **Answer 2**: NA.

Gu, H., Zhou, K., Han, X., Liu, N., Wang, R., & Wang, X. (2024, August). **PokeMQA: Programmable knowledge editing for Multi-hop Question Answering**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 8069-8083).

Benchmark Datasets

To effectively evaluate different aspects of **LLM + KGs for QA**, benchmark datasets must include specific types of data:

Answer Quality

- Ground-truth answers, representing the correct responses to questions.
- Supporting evidence, extracted KG triples or other references that justify the correctness of the answer.

Retrieval Quality

- Query-KG linkages that map questions to KG entities or relations.
- Ground-truth retrieval results, provide the expected relevant paths, subgraphs, or documents to assess retrieval accuracy.

Reasoning Quality

- Reasoning chains and intermediate steps that explain how the answer is derived.
- Complex constraints, such as temporal reasoning or negations, involved in the reasoning process.

Benchmark Datasets

WebQSP is a dataset designed for evaluating question-answering systems. It contains real-world questions and corresponding SPARQL queries, aimed at testing a system's ability to answer factual questions using structured knowledge bases like Freebase.

Dataset Example:

- Question: What character did Natalie Portman play in Star Wars?
- SPARQL Query:

```
PREFIX ns: <http://rdf.freebase.com/ns/>
SELECT DISTINCT ?x
WHERE {
  FILTER (?x != ns:m.09l3p)
  FILTER (!isLiteral(?x) OR lang(?x) = '' OR langMatches(lang(?x),
'en’))
  ns:m.09l3p ns:film.actor.film ?y .
  ?y ns:film.performance.character ?x .
  ?y ns:film.performance.film ns:m.0ddt_ .
}
```

- Topic Entity Name: Natalie Portman
- Answers: Padmé Amidala
- Inferential Chain: film.actor.film → film.performance.character

Answer Quality

Retrieval Quality

Reasoning Quality

Benchmark Datasets

A summary of various benchmark datasets used for evaluating the performance of **LLM + KGs for QA**

Dataset Name	Answer Quality	Retrieval Quality	Reasoning Quality	Brief Description
WebQSP	✓	✓	△	Contains SPARQL queries for knowledge-based QA.
CAQA	△	✓	△	Evaluates complex reasoning and attribution, including supportive, contradictory, and irrelevant cases.
CR-LT KGQA	✓	△	✓	Focuses on long-tail entities and commonsense reasoning.
PATQA	✓	△	✓	Present-anchored temporal QA.
MINTQA	✓	✓	✓	A multi-hop question answering benchmark for evaluating LLMs on new and tail knowledge.
MedQA	✓	△	△	Multilingual medical exam dataset with multiple-choice and medical texts.
KGs+LLMs for EnterpriseQA	✓	✓	✗	Assesses LLM and KG integration for QA on enterprise SQL databases.
XplainLLM	✓	△	✓	Focuses on explainability in QA reasoning.
LLM-KG-Bench	✓	✗	✗	LLMs in knowledge graph engineering.

- Core Evaluation Objective (✓): The dataset is primarily designed for this evaluation target.
- Partial Support (△): The dataset can be adapted to evaluate this objective, but it is not the main focus.
- Not Supported (✗): The dataset does not support this evaluation objective.

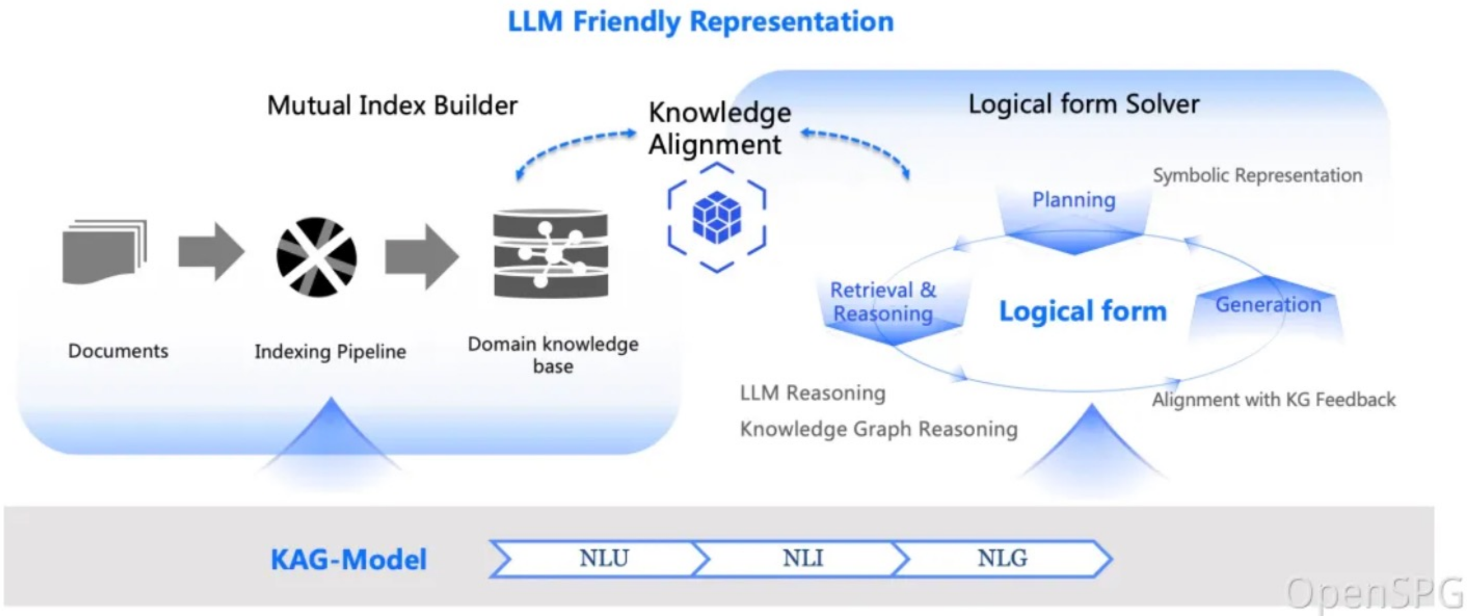
Industrial Applications

○ **KAG** (by Antgroup) ——A knowledge-augmented framework enhancing LLMs with Knowledge Graphs and vector retrieval for domain-specific QA.

○ **Technical Architecture**

➢ **kg-builder** implements a knowledge representation that is **friendly to LLMs**, enabling both schema-free information extraction and schema-constrained knowledge construction, while supporting mutual index representation for efficient retrieval.

➢ **kg-solver** uses a logical symbol-guided hybrid solving and reasoning engine, integrating planning, reasoning, and retrieval operators to transform natural language problems into a process combining language and symbols.

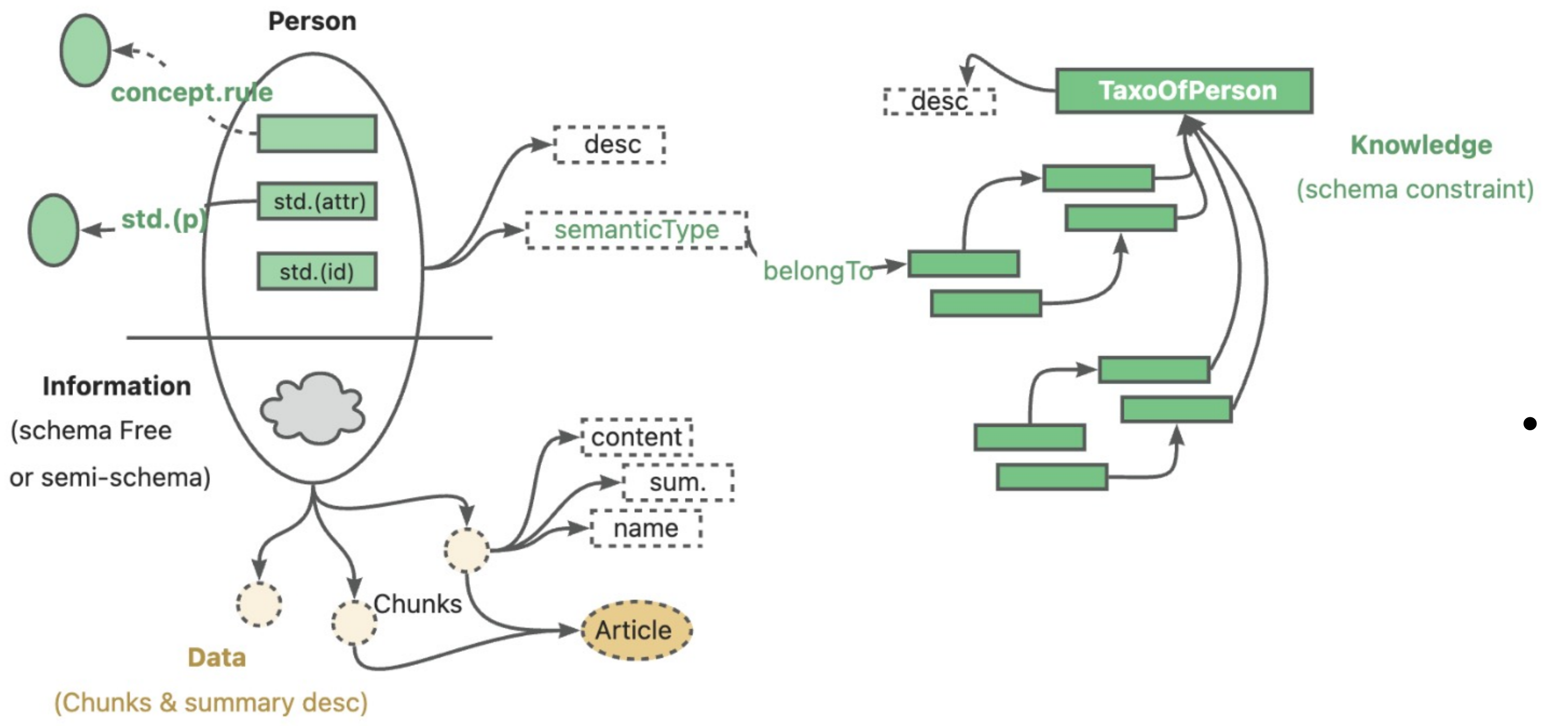


Source: https://github.com/OpenSPG/KAG

Industrial Applications

Core Feature: LLMs-Friendly Knowledge Representation

- KAG addresses the challenge of integrating unstructured data, structured information, and business expertise into a unified representation.

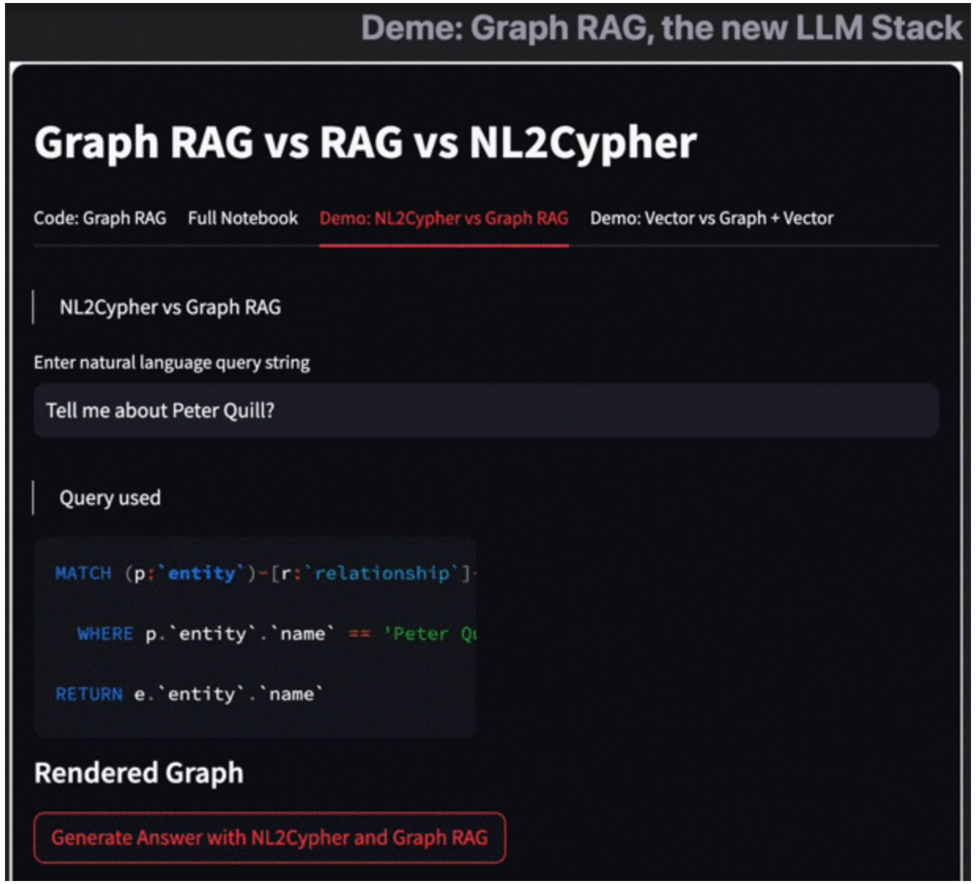


- For **unstructured data, structured data**, KAG uses advanced techniques like layout analysis, knowledge extraction, property normalization, and semantic alignment to construct a business knowledge graph.
- It supports **schema-free** data extraction and **schema-constrained** expertise construction, promoting cross-index representation for better inverted index creation and logical reasoning.

Source: https://github.com/OpenSPG/KAG

Industrial Applications

- **Graph RAG** (by NebulaGraph) —— A pioneering framework integrating Knowledge Graphs with LLMs to enhance search engines with deeper contextual understanding for smarter, more precise, and cost-effective search results.



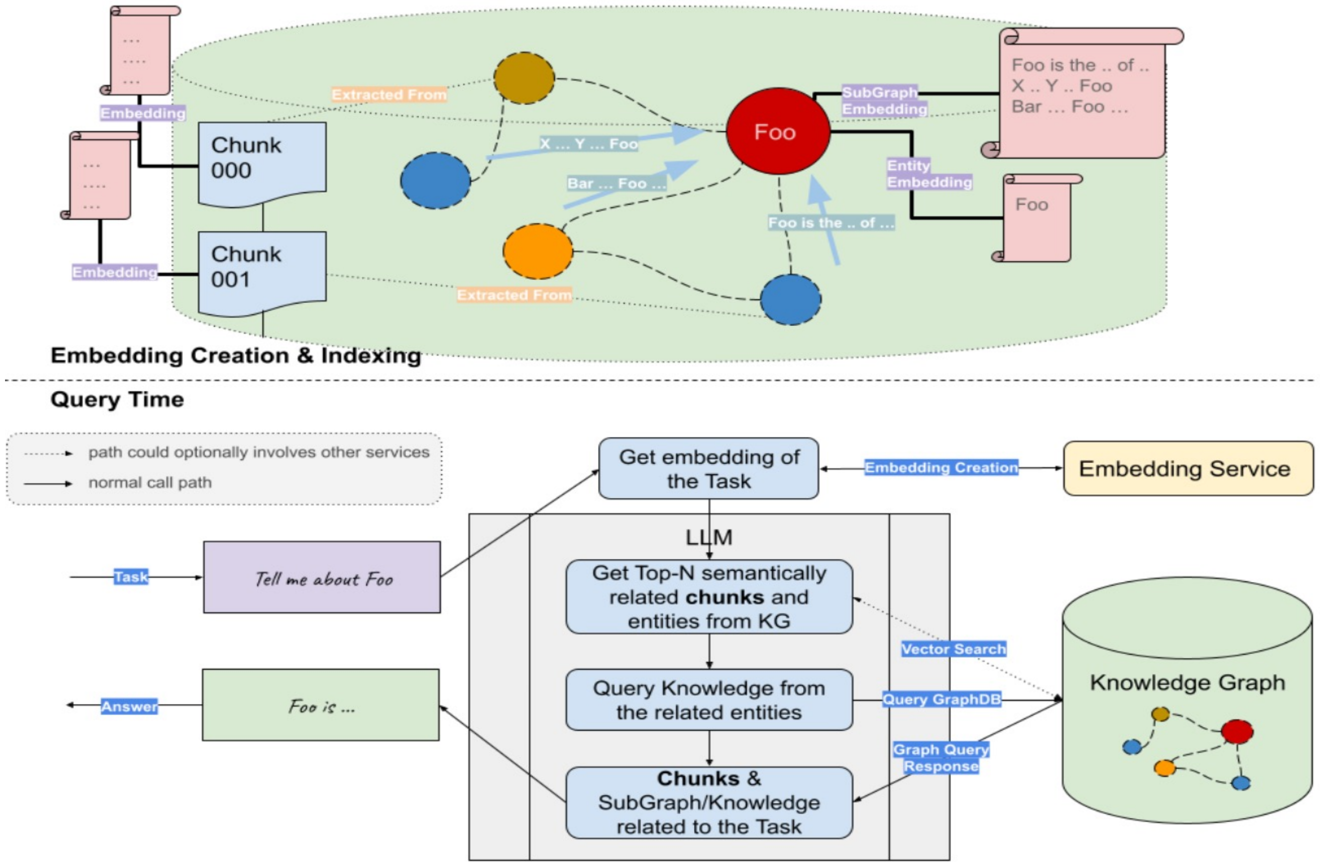
NLP2Cypher-based KG Query Engine

Translates tasks or questions into answer-oriented graph queries, similar to Text2SQL, without relying on entity subgraph retrieval.

Source: https://www.nebula-graph.io/posts/graph-RAG#post-graph-rag-vs-text2cypher

Industrial Applications

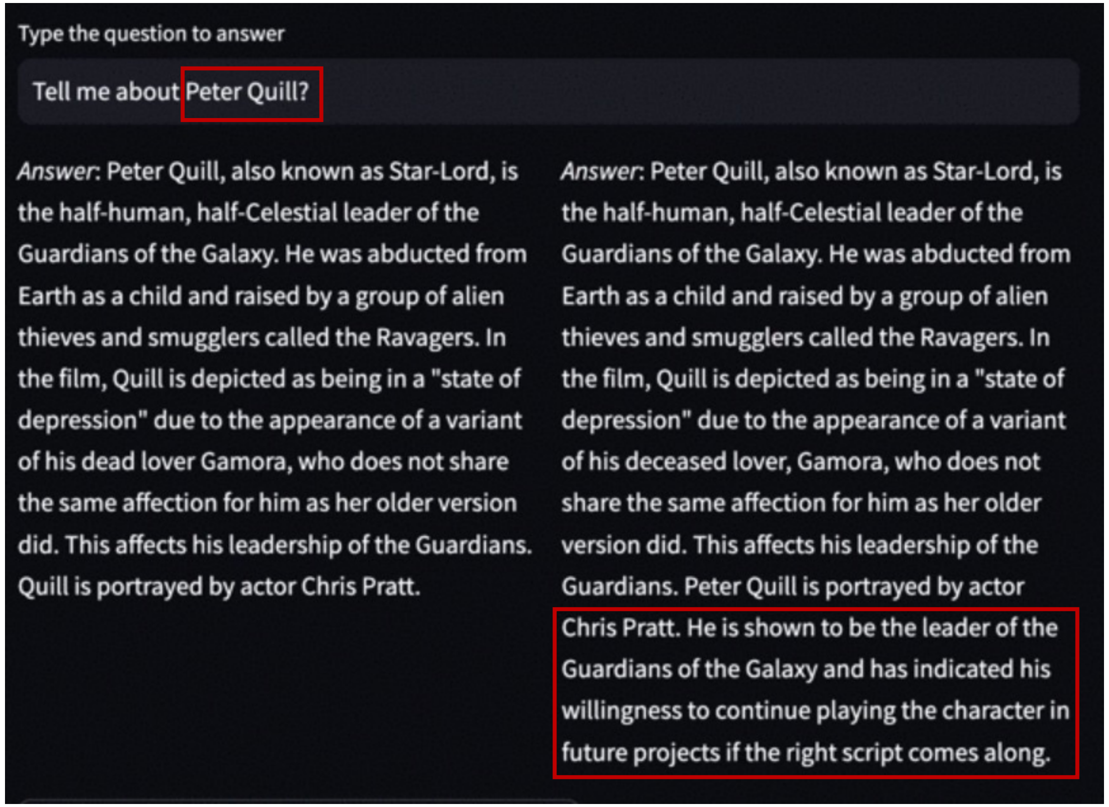
- NebulaGraph database seamlessly integrates with **Llama Index** and **LangChain**, building Graph RAG is highly simplified and requires minimal effort, based on just three lines of code. Even more complex RAG logic can be easily integrated, such as **Graph + Vector RAG**：



Source: https://www.nebula-graph.com.cn/posts/graph-rag-llm

Industrial Applications

Vector RAG：
Only provide simple information on his identity, plot, and actors



Graph + Vector RAG：
Can obtain more information about the protagonist's skills, character goals, and identity changes

- Graph RAG can understand the relations between entities, equate knowledge graphs to large-scale vocabularies and better comprehend the intent of complex queries, leading to more accurate and relevant search results.

Source: https://www.nebula-graph.io/posts/graph-RAG

Online Resources

Opportunities for Data Management
Part - 5

Arijit Khan

Aalborg University

Tutorial Outline

1) **Introduction (15 Min) – Arijit Khan**
    1.1 Large Language Models (LLMs)
    1.2 Knowledge Graphs (KGs)
    1.3 Unifying LLMs+KGs
    1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma
    2.1 KGs as Background Knowledge
    2.2 KGs as Reasoning Guidelines
    2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**
    3.1 Complex QA
    3.2 Explainable QA
    3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu
    4.1 Performance Metrics
    4.2 Benchmark Datasets
    4.3 Industry Applications and Demonstrations

5) Opportunities for Data Management (10 Min) – Arijit Khan

6) Future Directions (5 Min) – Tianxing Wu

- **Q&A Session (10 Min)**

Opportunities for Data Management

- **Natural Language Questions (NLQ) to Structured Query**
- **Efficient and Explainable Retrieval-augmented Generation (RAG)**
- **Knowledge Alignment and Dynamic Integration**
- **Querying over Heterogeneous and Multimodal Data**
- **Roles of Vector and Graph Databases**

Natural Language Questions (NLQ) to Graph Query

Motivation

- User-friendly querying in graph databases → avoid intricacy of graph query language (GQL) for non-expert users.
- Broadening applicability of graph DBs across various domains, e.g., knowledge-base question answering (KG-QA), voice assistants, web search, information retrieval, and recommendation.
- GQL-based querying maintains rich data and logical pathways, enhancing interactivity and interpretability, over vector-based retrieval.

Challenges

- Ambiguity of natural language questions.
- Hallucination and inconsistency of LLMs.

Similar to Text2SQL

- Complex GQL syntax and graph schema
  - large and heterogeneous schema, use of resource identifiers, overlapping relation types, lack of normalization
- Multi-hop questions

Specific for Text2GQL

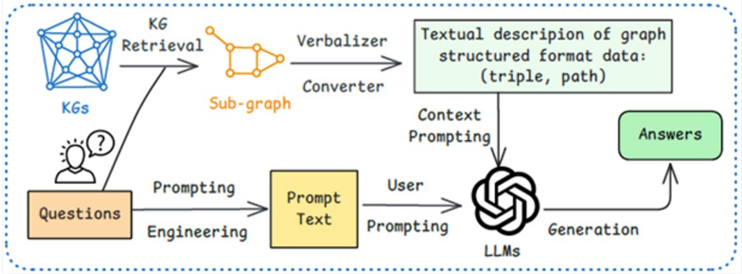
- Limited training datasets and tools

Methods

- Multiple LLMs coordination, LLM agents, fine-tuning, RAG, property graph views over RDF, graph patterns enhancement, …

Efficient and Explainable Retrieval-augmented Generation (RAG)

- **Retrieval-augmented generation (RAG)** to use KG context to improve LLM's accuracy and consistency.



- **GraphRAG:**

✓ Graph-based retrieval-augmented generation
✓ Synergy of graph DB + graph ML

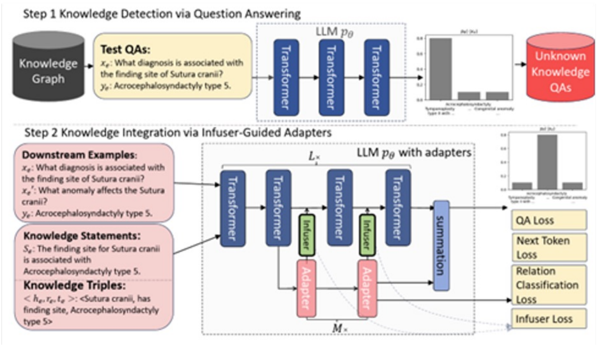
- Efficient Graph RAG and KG-RAG for real-time, interactive querying and exploration
  - ✓ **Various retrieval techniques**
    - Vector-based KG triples retrieval;
    - Vector-based Entity Retrieval + Breadth First Search for relevant paths retrieval;
    - LLM-based Entity Retrieval + Breadth First Search for relevant paths retrieval;
    - LLM-based subgraphs retrieval (Text2GQL);
    - Graph DB as semantic cache of LLMs
  - ✓ **Various ranking schemes**
    - KG triples ranking;
    - KG relations and paths ranking;
    - KG subgraphs ranking.
  - ✓ **Various knowledge integration schemes**
    - graph to text conversion
    - graph embedding
    - resolve knowledge conflict

- **Explainable GraphRAG:**

✓ Factual and counterfactual explanation
✓ Role of pretrained knowledge vs. retrieved knowledge
✓ KGs as LLM guardrail

Knowledge Alignment and Dynamic Integration

- Knowledge conflict
- Knowledge forgetting, catastrophic forgetting
- Dynamic Knowledge integration



InfuserKI: Enhancing Large Language Models with Knowledge Graphs via Infuser-Guided Knowledge Integration (EMNLP 2024)

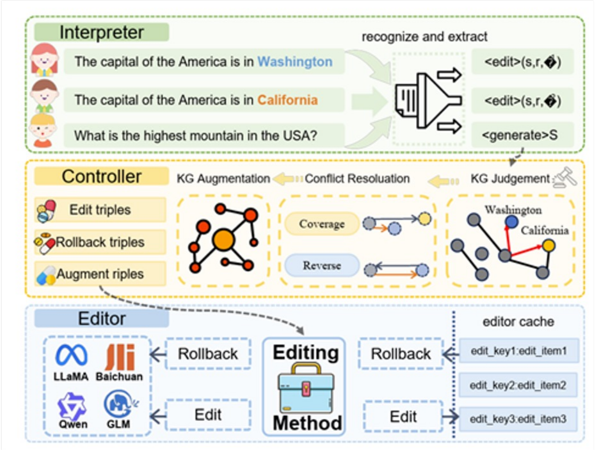
✓ Prompt engineering to prioritize external knowledge over parametric memory

✓ Fine-tuning and contrastive decoding to enhance contextual grounding

✓ Model editing, parameter pruning to reduce knowledge conflicts

✓ Integrating unknown knowledge into LLMs without unnecessary overlap of known knowledge

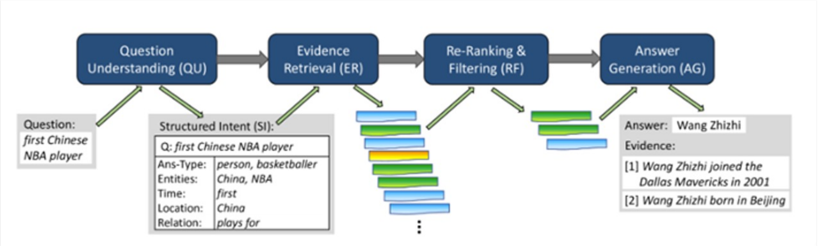
✓ Collaborative knowledge editing in LLM+KG



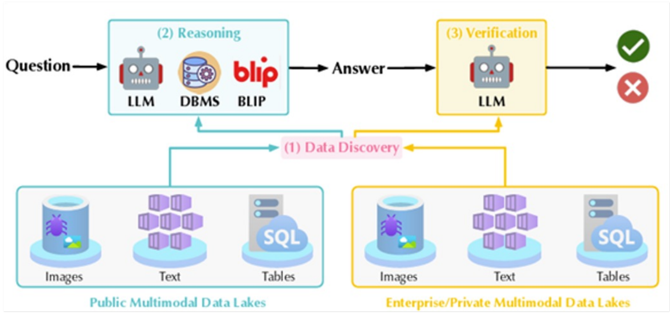
OneEdit: A Neural-Symbolic Collaboratively Knowledge Editing System (VLDB workshop 2024)

Querying over Heterogeneous and Multimodal Data

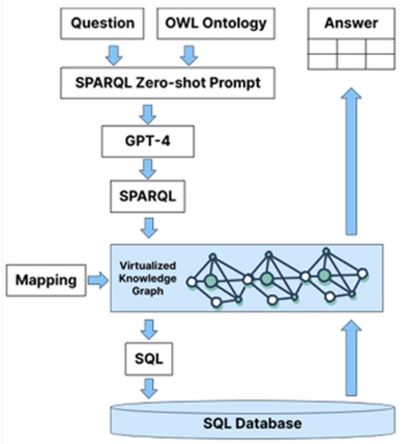
- Heterogeneous knowledge sources (e.g., unstructured text, structured tables, and knowledge graphs) – data lake
- Integrating multiple pieces of evidence of different modalities to infer correct and complete answers.
- Verification of generated answers against trusted source ➔ trustworthy question answering
- E.g., *"What do others say about my papers?"* or *"Find competitors with similar products to mine and analyze their pricing strategies for different products"*.



QUASAR system: Philipp Christmann and Gerhard Weikum. RAG-based Question Answering over Heterogeneous Data and Text. IEEE Data Engineering Bulletin. December 2024 Edition on RAG



Symphony system: Nan Tang, Chenyu Yang, Zhengxuan Zhang, Yuyu Luo. Symphony: Towards Trustworthy Question Answering and Verification using RAG over Multimodal Data Lakes. IEEE Data Engineering Bulletin. December 2024 Edition on RAG



Juan Sequeda, Dean Allemang, Bryon Jacob. Increasing Accuracy of LLM-powered Question Answering on SQL databases: Knowledge Graphs to the Rescue. IEEE Data Engineering Bulletin. December 2024 Edition on RAG

Roles of Vector and Graph Databases

- RAG requiring both vector search over Vector DBs and graph search over Graph DBs.

- E.g., *"finding all positive reviews written by a specific customer"* or *"summarizing the impact of COVID-19 on the global economy"*.

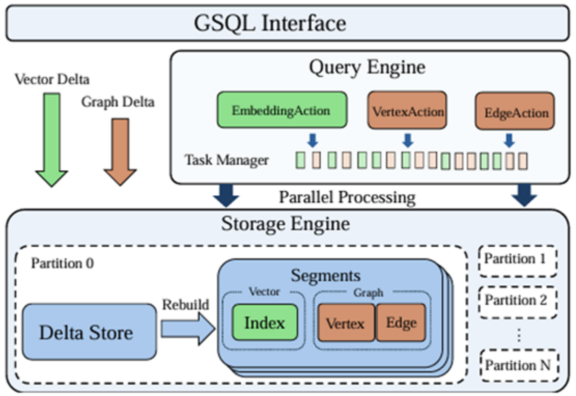
- TigerVector ➔ integrates vector search seamlessly into TigerGraph, a distributed graph database system.

✓ Unified system supporting both vector data and graph data. Reduces data movement, minimizes data silos.

✓ Vector embeddings as a new attribute of existing graph nodes.

✓ Decouples storage of vector embeddings from other graph attributes ➔ utilize native vector indexes, updates involving both graph attributes and vector attributes performed atomically.

✓ Hybrid searches of vector and graph data using a unified GSQL query language.



Shige Liu, Zhifang Zeng, Li Chen, Adil Ainihaer, Arun Ramasami, Songting Chen, Yu Xu, Mingxi Wu, and Jianguo Wang. 2025. TigerVector: Supporting vector search in graph databases for advanced RAGs. arXiv:2501.11216 (2025)

EDBT ICDT BARCELONA 2025

Online Resources

Future Directions
Part - 6

Tianxing Wu

Southeast University

AALBORG UNIVERSITET

Tutorial Outline

1) **Introduction (15 Min) – Arijit Khan**
- 1.1 Large Language Models (LLMs)
- 1.2 Knowledge Graphs (KGs)
- 1.3 Unifying LLMs+KGs
- 1.4 Question Answering (QA)

2) Unifying LLMs with KGs for QA (25 Min) – Chuangtao Ma
- 2.1 KGs as Background Knowledge
- 2.2 KGs as Reasoning Guidelines
- 2.3 KGs as Refiners and Validators

3) **Advanced Topics on LLM+KG for QA (25 Min) - Yongrui Chen**
- 3.1 Complex QA
- 3.2 Explainable QA
- 3.3 Optimization and Efficiency

- **Break (10 Min)**

4) Evaluations and Applications (20 Min) – Tianxing Wu
- 4.1 Performance Metrics
- 4.2 Benchmark Datasets
- 4.3 Industry Applications and Demonstrations

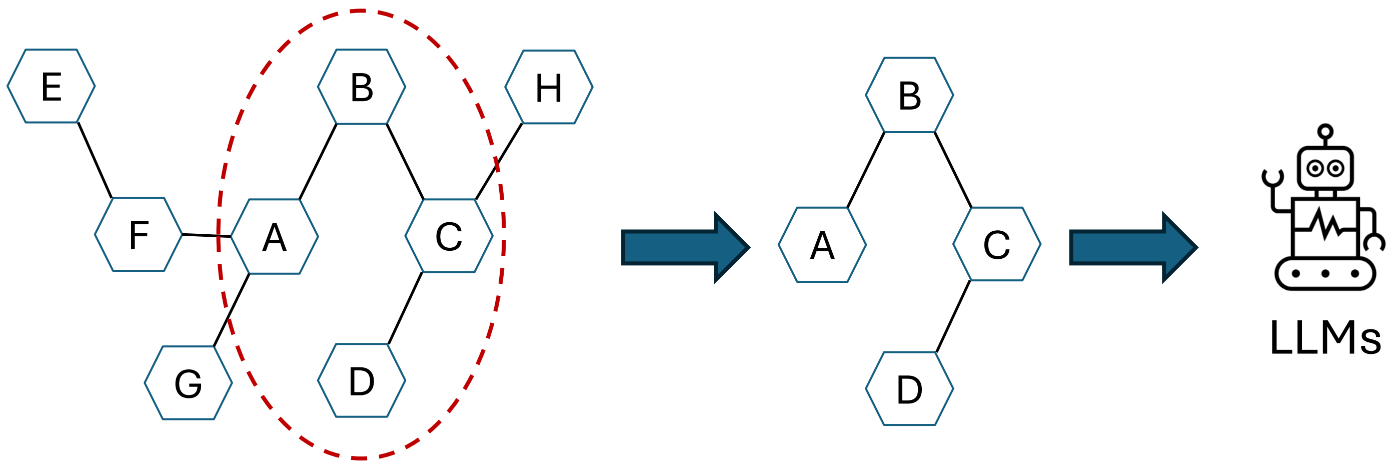
5) Opportunities for Data Management (10 Min) – Arijit Khan

6) Future Directions (5 Min) – Tianxing Wu

- **Q&A Session (10 Min)**

Effectiveness and Efficiency of Subgraph Retrieval

- Challenge: Effectiveness and efficiency retrieval of relevant subgraphs.
- Reasons:
  - LLMs have a limited context length, making it impractical to process entire knowledge graphs. This necessitates the effective extraction of relevant subgraphs.
  - Retrieving subgraphs from large-scale knowledge graphs is computationally expensive.
- Potential Solutions:
  - Develop optimized methods for efficient subgraph retrieval.



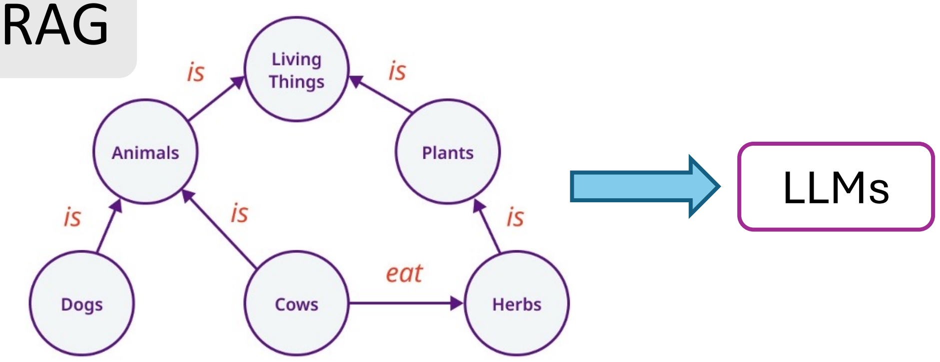
Security, Privacy, Explainability and Fairness in QA

- Security & Privacy:
  - Unifying domain-specific KGs raises privacy risks.

- Differential Privacy
- Federated Learning
- Anonymization
- Access Control

- Explainability & Fairness:
  - QA reasoning relies on the reasoning chains over the factual graph.

Graph RAG



Source: https://www.datastax.com/guides/graph-rag

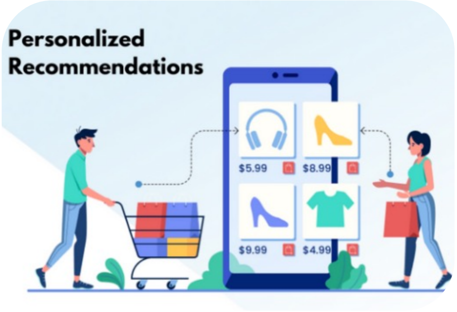
Explainability & Fairness

Costly & Inefficient

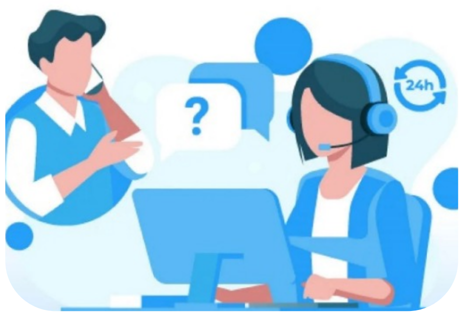
Explore more efficient retrieval methods

Other Data Science Applications

- The combination of LLMs and KGs leverages **LLMs' natural language understanding** and **KGs' structured knowledge** to enhance applications like:



Personalized Recommendations



Customer Service

Medical Diagnostics

Financial Decision-Making

- **Future**: Smarter, knowledge-rich solutions across domains.

Thanks!

- Online Resources
  - Tutorial Webpage [https://machuangtao.github.io/LLM-KG4QA/tutorial-edbt25/]
  - GitHub Repository [https://github.com/machuangtao/LLM-KG4QA]
- Co-organized Other Related Events
  - LLM+KG Workshop@VLDB2024 [https://seucoin.github.io/workshop/llmkg/], Workshop Report [https://vldb.org/workshops/2024/proceedings/LLM+KG/LLM+KG-1.pdf], Workshop Panel Report [https://wp.sigmod.org/?p=3813]
  - LLM+Graph Workshop@VLDB2025 [https://seucoin.github.io/workshop/llmg2025/] **Paper Submission Open!**
  - Guest Editorial: Special issue on "Neuro-Symbolic Intelligence: large Language Model enabled Knowledge Engineering", World Wide Web 2025 [https://link.springer.com/article/10.1007/s11280-024-01327-7]

References

[1] https://custom.typingmind.com/tools/model-icons

[2] A brief introduction to (large) language models. Sachin Kumar. sachink@allenai.org
https://courses.cs.washington.edu/courses/cse473/23au/slides/473-LMs.pdf

[3] https://blog.csdn.net/weixin_59191169/article/details/141964075

[4] https://www.appypie.com/blog/top-10-real-world-applications-of-large-language-models

[5] Xi Chen, Wei Hu, Arijit Khan, Shreya Shankar, Haofen Wang, Jianguo Wang, and Tianxing Wu. Large Language Models, Knowledge Graphs, and Vector Databases: Synergy and Opportunities for Data Management (A Report on the LLM+KG@VLDB24 Workshop's Panel Discussion). https://wp.sigmod.org/?p=3813

[6] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language Models as Knowledge Bases?. In EMNLP-IJCNLP.

[7] Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo, Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. Trustworthy LLMs: a Survey and Guideline for Evaluating Large Language Models' Alignment. NeurIPS SoLaR Workshop 2023.

[8] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: principles, taxonomy, challenges, and open questions. ACM Trans. Inf. Syst., 2024.

[9] Bishwamittra Ghosh, Sarah Hasan, Naheed Anjum Arafat, and Arijit Khan. Logical Consistency of Large Language Models in Fact-checking. In ICLR 2025.

[10] Qinbin Li, Junyuan Hong, Chulin Xie, Jeffrey Tan, Rachel Xin, Junyi Hou, Xavier Yin, Zhun Wang, Dan Hendrycks, Zhangyang Wang, Bo Li, Bingsheng He, and Dawn Song. LLM-PBE: Assessing Data Privacy in Large Language Models. PVLDB 17 (11), 3201 - 3214, 2024.

[11] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for Large Language Models: A Survey. ACM Transactions on Intelligent Systems and Technology, Volume 15, Issue 2, Article No. 20, Pages 1 - 38, 2024.

[12] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. FAccT '21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. Pages 610 - 623.

References

[13] Claudio Gutierrez and Juan F. Sequeda. Knowledge Graphs: A Tutorial on the History of Knowledge Graph's Main Ideas. CIKM 2020 Tutorial

[14] Gerhard Weikum, Xin Luna Dong, Simon Razniewski, and Fabian Suchanek. Machine Knowledge: : Creation and Curation of Comprehensive Knowledge Bases. Foundations and Trends in Databases, Volume 10, Issue 2-4, Pages 108 - 490, 2021.

[15] Bilal Abu-Salih. Domain-specific knowledge graphs: a survey. J. Netw. Comput. Appl., 185:103076, 2021.

[16] Natasha Noy, Yuqing Gao, Anshu Jain, Anant Narayanan, Alan Patterson, and Jamie Taylor. Industry-scale knowledge graphs: lessons and challenges. Communications of the ACM, Volume 62, Issue 8, Pages 36 - 43.

[17] Martin G. Skjæveland, Krisztian Balog, Nolwenn Bernard, Weronika Lajewska, and Trond Linjordet.
An ecosystem for personal knowledge graphs: A survey and research roadmap. AI Open 5: 55-69 (2024).

[18] Arijit Khan. Knowledge Graphs Querying. ACM SIGMOD Record 52(2): 18-29 (2023).

[19] Vinay K. Chaudhri, Chaitanya K. Baru, Naren Chittar, Xin Luna Dong, Michael R. Genesereth, James A. Hendler, Aditya Kalyanpur, Douglas B. Lenat, Juan Sequeda, Denny Vrandecic, and Kuansan Wang. Knowledge Graphs: Introduction, History and, Perspectives. AI Mag. 43(1): 17-29 (2022).

[20] Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. Head-to-Tail: How Knowledgeable are Large Language Models (LLMs)? A.K.A. Will LLMs Replace Knowledge Graphs? NAACL-HLT 2024: 311-325.

[21] Yushi Sun, Xin Hao, Kai Sun, Yifan Xu, Xiao Yang, Xin Luna Dong, Nan Tang, and Lei Chen. Are Large Language Models a Good Replacement of Taxonomies? Proc. VLDB Endow. 17(11): 2919-2932 (2024).

[22] Arijit Khan and Sayan Ranu. Big-Graphs: Querying, Mining, and Beyond. in Sherif Sakr and Albert Zomaya (eds.), *Springer Handbook of Big Data Technologies*, Springer, 2017.

[23] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Mikhail Galkin, Sahand Sharifzadeh, Asja Fischer, Volker Tresp, and Jens Lehmann.
Bringing Light Into the Dark: A Large-Scale Evaluation of Knowledge Graph Embedding Models Under a Unified Framework. IEEE Trans. Pattern Anal. Mach. Intell. 44(12): 8825-8845 (2022).

[24] Federico Bianchi, Gaetano Rossiello, Luca Costabello, Matteo Palmonari, and Pasquale Minervini. Knowledge Graph Embeddings and Explainable AI. Knowledge Graphs for eXplainable Artificial Intelligence 2020: 49-72.

References

[25] Sanju Mishra Tiwari, Fatima N. Al-Aswadi, and Devottam Gaurav. Recent trends in knowledge graphs: theory and practice. Soft Comput. 25(13): 8337-8355 (2021).

[26] Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying large language models and knowledge graphs: A roadmap. IEEE Trans. Knowl. Data Eng., 75836(7):3580–3599, 2024.

[27] Jeff Z. Pan , Simon Razniewski , Jan-Christoph Kalo , Sneha Singhania , Jiaoyan Chen , Stefan Dietze , Hajira Jabeen , Janna Omeliyanenko , Wen Zhang , Matteo Lissandrini , Russa Biswas , Gerard de Melo , Angela Bonifati , Edlira Vakaj , Mauro Dragoni , and Damien Graux. Large Language Models and Knowledge Graphs: Opportunities and Challenges. Trans. Graph Data Knowl., 1(1):1–38, 2023.

[28] Amanda Kau, Xuzeng He, Aishwarya Nambissan, Aland Astudillo, Hui Yin, and Amir Aryani. Combining knowledge graphs and large language models. arXiv:2407.06564, 2024.

[29] Nourhan Ibrahim, Samar Aboulela, Ahmed Ibrahim, and Rasha Kashef. A survey on augmenting knowledge graphs (KGs) with large language models (LLMs): models, evaluation metrics, benchmarks, and challenges. Discov. Artif. Intell., 4(1):76, 2024.

[30] Aleksandr Perevalov, Andreas Both, and Axel-Cyrille Ngonga Ngomo**.** Multilingual question answering systems for knowledge graphs–a survey. Semantic Web,15(5):2089–2124, 2024.

[31] Aidan Hogan, Xin Luna Dong, Denny Vrandečić, and Gerhard Weikum Large language models, knowledge graphs and search engines: A crossroads for answering users' questions. arXiv:2501.06699, 2025.

[32] Arijit Khan, Tianxing Wu, and Xi Chen. LLM+KG: Data Management Opportunities in Unifying Large Language Models + Knowledge Graphs. In Proc. of Workshops at the International Conference on Very Large Data Bases, co-located with VLDB 2024.

[33] Ernests Lavrinovics, Russa Biswas, Johannes Bjerva, and Katja Hose. Knowledge Graphs, Large Language Models, and Hallucinations: An NLP Perspective. J. Web Semant. 85: 100844 (2025).

[34] Rishiraj Saha Roy and Avishek Anand. Question Answering for the Curated Web: Tasks and Methods in QA over Knowledge Bases and Text Collections. Synthesis Lectures on Information Concepts, Retrieval, and Services, Morgan & Claypool Publishers 2021, pp. 1-194.

[35] Jie He, Nan Hu, Wanqiu Long, Jiaoyan Chen, and Jeff Z. Pan. MINTQA: A Multi-Hop Question Answering Benchmark for Evaluating LLMs on New and Tail Knowledge, arXiv:2412.17032, 2024.

References

[36] Jiang, Pengcheng, et al. KG-FIT: Knowledge graph fine-tuning upon open-world knowledge. Advances in Neural Information Processing Systems 37 (2024): 136220-136258.

[37] Wang, Xxiaozhi., et al. A unified model for knowledge embedding and pre-trained language representation. ACL, 9 (2021): 176-194.

[38] Yasunaga, Michihiro, et al. Deep bidirectional language-knowledge graph pretraining. Advances in Neural Information Processing Systems, 35 (2022): 37309-37323.

[39] Jiang, Pengcheng, et al. KG-FIT: Knowledge graph fine-tuning upon open-world knowledge. Advances in Neural Information Processing Systems, 37 (2024): 136220-136258.

[40] Xiangrong, Zhu, et al. Knowledge Graph-Guided Retrieval Augmented Generation. arXiv preprint arXiv:2502.068641 (2025).

[41] Baek, Jinheon, et al. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. arXiv preprint arXiv:2306.04136 (2023).

[42] Soman, Karthik, et al. Biomedical knowledge graph-optimized prompt generation for large language models. Bioinformatics 40.9 (2024): btae560.

[43] Sanmartin, Diego. KG-RAG: Bridging the gap between knowledge and creativity. arXiv preprint arXiv:2405.12035 (2024).

[44] Cai, Yuzheng, et al. SimGRAG: Leveraging Similar Subgraphs for Knowledge Graphs Driven Retrieval-Augmented Generation. arXiv preprint arXiv:2412.15272 (2024).

[45] Liu, Guangyi, et al. Dual Reasoning: A GNN-LLM Collaborative Framework for Knowledge Graph Question Answering. CPAL, (2025).

[46] Gao, Zengyi, et al. FRAG: A Flexible Modular Framework for Retrieval-Augmented Generation based on Knowledge Graphs. arXiv preprint arXiv:2501.09957 (2025).

[47] Ji, Yixin, et al. Retrieval and reasoning on KGs: Integrate knowledge graphs into large language models for complex question answering. EMNLP. (2024).

[48] Chen, Hanzhu, et al. Knowledge Graph Finetuning Enhances Knowledge Manipulation in Large Language Models. ICLR. (2025).

[49] Jiang, Jinhao, et al. KG-Agent: An efficient autonomous agent framework for complex reasoning over knowledge graph. arXiv preprint arXiv:2402.11163 (2024).

[50] Guan, Xinyan, et al. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. AAAI. (2024).

[51] Wang, Junjie, et al. Learning to plan for retrieval-augmented large language models from knowledge graphs. arXiv preprint arXiv:2406.14282 (2025).

References

[52] Chen, Huajun. Large knowledge model: Perspectives and challenges. arXiv preprint arXiv:2312.02706 (2023).

[53] Wang, Yu, et al. Knowledge graph prompting for multi-document question answering. Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. No. 17. 2024.

[54] He, Xiaoxin, et al. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. Advances in Neural Information Processing Systems 37 (2024): 132876-132907.

[55] Xu, Zhentao, et al. Retrieval-augmented generation with knowledge graphs for customer service question answering. Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2024.

[56] Dong, Junnan, et al. Modality-Aware Integration with Large Language Models for Knowledge-Based Visual Question Answering. Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2024.

[57] Liu, Lihui, et al. Conversational Question Answering with Language Models Generated Reformulations over Knowledge Graph. Findings of the Association for Computational Linguistics ACL 2024. 2024.

[58] Feng, Jianzhou, et al. Retrieval In Decoder benefits generative models for explainable complex question answering. Neural Networks 181 (2025): 106833.

[59] Sui, Yuan, and Bryan Hooi. Can Knowledge Graphs Make Large Language Models More Trustworthy? An Empirical Study over Open-ended Question Answering. arXiv e-prints (2024): arXiv-2410.

[60] Zhang, Haozhen, Tao Feng, and Jiaxuan You. Graph of records: Boosting retrieval augmented generation for long-context summarization with graphs. arXiv preprint arXiv:2410.11001 (2024).

[61] Tao, Dehao, et al. Clue-Guided Path Exploration: Optimizing Knowledge Graph Retrieval with Large Language Models to Address the Information Black Box Challenge. arXiv preprint arXiv:2401.13444 (2024).

[62] Yang, Rui, et al. KG-Rank: Enhancing Large Language Models for Medical QA with Knowledge Graphs and Ranking Techniques. Proceedings of the 23rd Workshop on Biomedical Natural Language Processing. 2024.

[63] Dong, Junnan, et al. Cost-efficient Knowledge-based Question Answering with Large Language Models. The Thirty-eighth Annual Conference on Neural Information Processing Systems.

References

[64] Jack Longwell, Ali Akbar Alavi, Fattane Zarrinkalam, Faezeh Ensan. Triple Augmented Generative Language Models for SPARQL Query Generation from Natural Language Questions. SIGIR-AP 2024: Proceedings of the 2024 Annual International ACM SIGIR. Pages 269 – 273.

[65] Yuhang Zhou, Yu He, Siyu Tian, Yuchen Ni, Zhangyue Yin, Xiang Liu, Chuanjun Ji, Sen Liu, Xipeng Qiu, Guangnan Ye, Hongfeng Chai. R3-NL2GQL: A Model Coordination and Knowledge Graph Alignment Approach for NL2GQL. Findings of the Association for Computational Linguistics: EMNLP 2024.

[66] Yuanyuan Liang, Keren Tan, Tingyu Xie, Wenbiao Tao, Siyuan Wang, Yunshi Lan, Weining Qian. Aligning Large Language Models to a Domain-specific Graph Database for NL2GQL. CIKM '24: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. Pages 1367 – 1377.

[67] Zhuoyang Li, Liran Deng, Hui Liu, Qiaoqiao Liu, Junzhao Du. UniOQA: A Unified Framework for Knowledge Graph Question Answering with Large Language Models. arXiv:2406.02110. 2024.

[68] Yuanyuan Liang, Tingyu Xie, Gan Peng, Zihao Huang, Yunshi Lan, Weining Qian. NAT-NL2GQL: A Novel Multi-Agent Framework for Translating Natural Language to Graph Query Language. arXiv:2412.10434. 2024.

[69] Yanlin Feng, Simone Papicchio, Sajjadur Rahman. CypherBench: Towards Precise Retrieval over Full-scale Modern Knowledge Graphs in the LLM Era. arXiv:2412.18702. 2024.

[70] Aibo Guo, Xinyi Li, Guanchen Xiao, Zhen Tan, Xiang Zhao. SpCQL: A Semantic Parsing Dataset for Converting Natural Language into Cypher. CIKM '22: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. Pages 3973 – 3977.

[71] Quoc-Bao-Huy Tran, Aagha Abdul Waheed, and Sun-Tae Chung. Robust Text-to-Cypher Using Combination of BERT, GraphSAGE, and Transformer (CoBGT) Model. Appl. Sci. 2024, 14(17), 7881.

[72] Markus Hornsteiner, Michael Kreussel, Christoph Steindl, Fabian Ebner, Philip Empl andStefan Schönig. Real-Time Text-to-Cypher Query Generation with Large Language Models for Graph Databases. Future Internet 2024, 16(12), 438.

[73] Yang Liu, Xin Wang, Jiake Ge, Hui Wang, Dawei Xu, Yongzhe Jia. Text to Graph Query Using Filter Condition Attributes. VLDB Workshops 2024.

[74] Xinke Zhao; Hankiz Yilahun; Askar Hamdulla. Text-to-CQL Based on Large Language Model and Graph Pattern Enhancement. 2024 IEEE 5th International Conference on Pattern Recognition and Machine Learning (PRML).

[75] Chuangtao Ma, Sriom Chakrabarti, Arijit Khan, Bálint Molnár. Knowledge Graph-based Retrieval-Augmented Generation for Schema Matching. CoRR abs/2501.08686 (2025)

References

[76] Yubo Huang and Guosun Zeng. 2024. RD-P: A Trustworthy retrieval-augmented prompter with knowledge graphs for LLMs. In CIKM. 942–952.

[77] Pengcheng Huang, et al. PIP-KAG: Mitigating Knowledge Conflicts in Knowledge-Augmented Generation via Parametric Pruning. arXiv:2502.15543. 2025.

[78] Ningyu Zhang, Zekun Xi, Yujie Luo, Peng Wang, Bozhong Tian, Yunzhi Yao, Jintian Zhang, Shumin Deng, Mengshu Sun, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, Huajun Chen. OneEdit: A Neural-Symbolic Collaboratively Knowledge Editing System. VLDB Workshops 2024.

[79] Fali Wang, et al. InfuserKI: Enhancing Large Language Models with Knowledge Graphs via Infuser-Guided Knowledge Integration. EMNLP (Findings) 2024: 3675-3688.

[80] Philipp Christmann and Gerhard Weikum. RAG-based Question Answering over Heterogeneous Data and Text. IEEE Data Engineering Bulletin. December 2024 Edition on RAG.

[81] Nan Tang, Chenyu Yang, Zhengxuan Zhang, Yuyu Luo. Symphony: Towards Trustworthy Question Answering and Verification using RAG over Multimodal Data Lakes. IEEE Data Engineering Bulletin. December 2024 Edition on RAG.

[82] Shicheng Liu, Jialiang Xu, Wesley Tjangnaka, Sina J. Semnani, Chen Jie Yu, Monica Lam. SUQL: Conversational Search over Structured and Unstructured Data with Large Language Models. NAACL-HLT (Findings) 2024: 4535-4555.

[83] Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Sejr Schlichtkrull, Sonal Gupta, Yashar Mehdad, Scott Yih. UniK-QA: Unified Representations of Structured and Unstructured Knowledge for Open-Domain Question Answering. NAACL-HLT (Findings) 2022: 1535-1546.

[84] Dongkyu Lee, et al. MATTER: Memory-Augmented Transformer Using Heterogeneous Knowledge Sources. ACL (Findings) 2024: 16110-16121.

[85] Shirley Wu, Shiyu Zhao, Michihiro Yasunaga, Kexin Huang, Kaidi Cao, Qian Huang, Vassilis N. Ioannidis, Karthik Subbian, James Zou, Jure Leskovec. STaRK: Benchmarking LLM Retrieval on Textual and Relational Knowledge Bases. NeurIPS 2024 Track Datasets and Benchmarks Poster.

[86] Yao Xu, Shizhu He, Jiabei Chen, Zeng Xiangrong, Bingning Wang, Guang Liu, Jun Zhao, Kang Liu. LLaSA: Large Language and Structured Data Assistant. NAACL 2025.

[87] Juan Sequeda, Dean Allemang, Bryon Jacob. Increasing Accuracy of LLM-powered Question Answering on SQL databases: Knowledge Graphs to the Rescue. IEEE Data Engineering Bulletin. December 2024 Edition on RAG.

[88] Shige Liu, Zhifang Zeng, Li Chen, Adil Ainihaer, Arun Ramasami, Songting Chen, Yu Xu, Mingxi Wu, and Jianguo Wang. 2025. TigerVector: Supporting vector search in graph databases for advanced RAGs. arXiv:2501.11216 (2025).

References

[89] Peng, B., Zhu, Y., Liu, et al. Graph retrieval-augmented generation: A survey. *arXiv preprint arXiv:2408.08921* (2024).

[90] Es, S., James, J., Anke, L. E, et al. Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations,*150-158 (2024).

[91] Yu, H., Gan, A., Zhang, K, et al. Evaluation of retrieval-augmented generation: A survey. In *CCF Conference on Big Data,* 102-120(2024).

[92] Gu, H., Zhou, K., Han, X, et al. PokeMQA: Programmable knowledge editing for Multi-hop Question Answering. ACL, 8069-8083, 2024.

Q&A Session