Stanford CS224W

GraphRAG

Retrieval-augmented generation (RAG) with graphs

CS224W: Machine Learning with Graphs
Costas Mavromatis
http://cs224w.stanford.edu



Three parts of the lecture

Part I: Why knowledge graphs?

Organize knowledge for questions and updates.

Part II: GNN-RAG (customization)

Learn retrieval for a known graph.

Part III: Bring Your Own Graph (generalization)

Retrieve across new graphs and schemas.

Part I · GraphRAG
Why knowledge graphs?

CS224W: Machine Learning with Graphs
Costas Mavromatis
http://cs224w.stanford.edu



What is a knowledge graph (KG)?

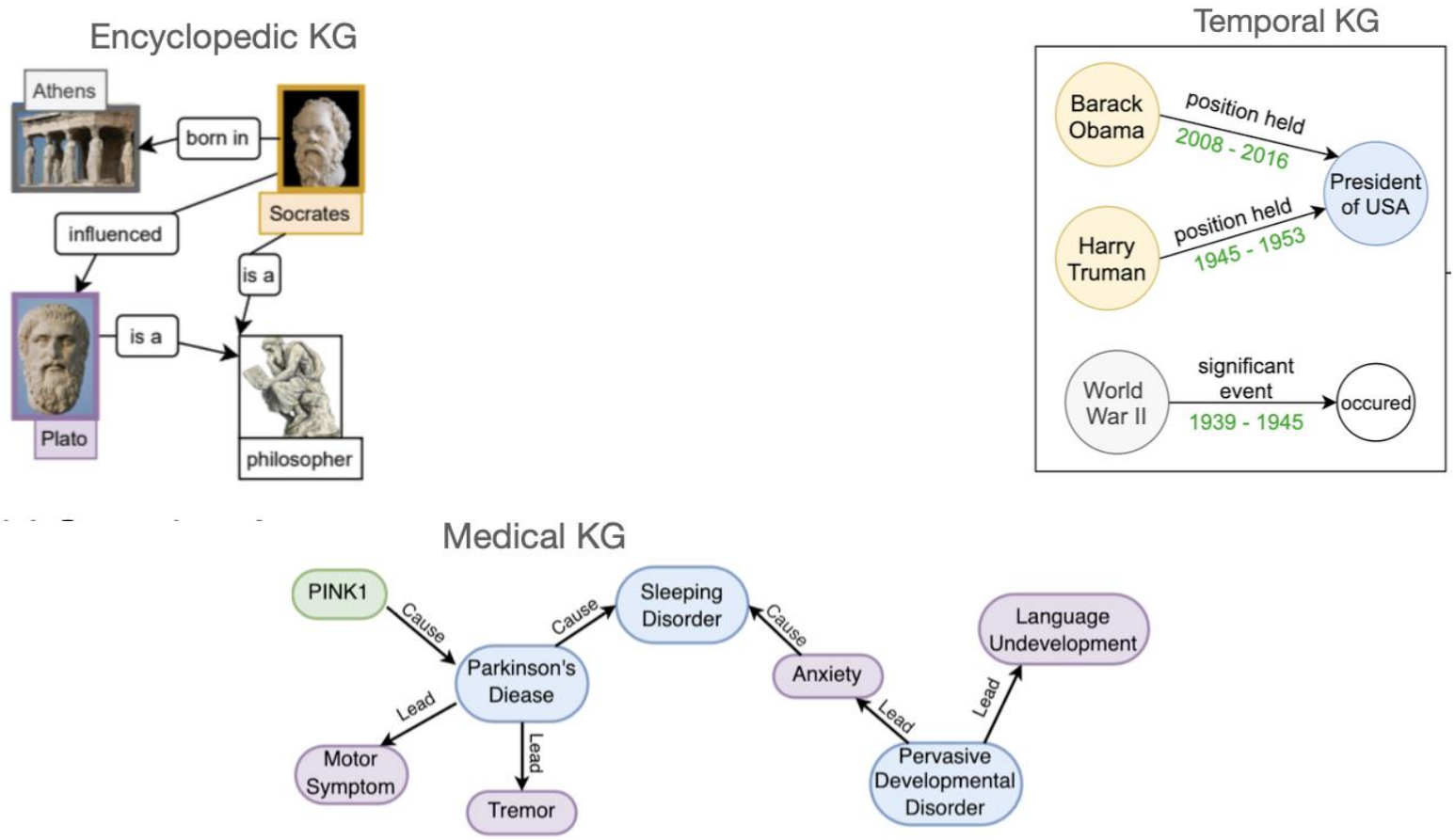
A graph contains nodes and edges.

A KG represents entities and named relations.



(Tarantino, directed, Pulp Fiction)

What is a knowledge graph (KG)?



The same representation organizes many kinds of knowledge.

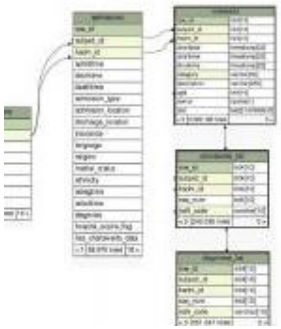
Where do graphs come from?

Extract from documents

Text

Tarantino directed
Pulp Fiction.

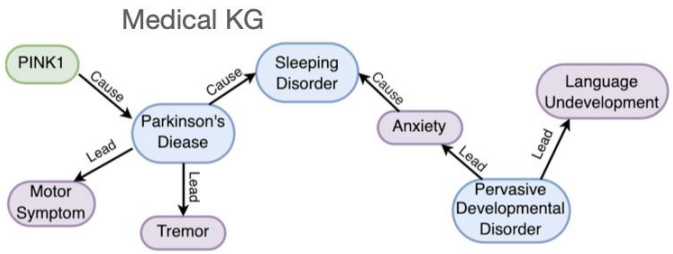
Map tables and joins



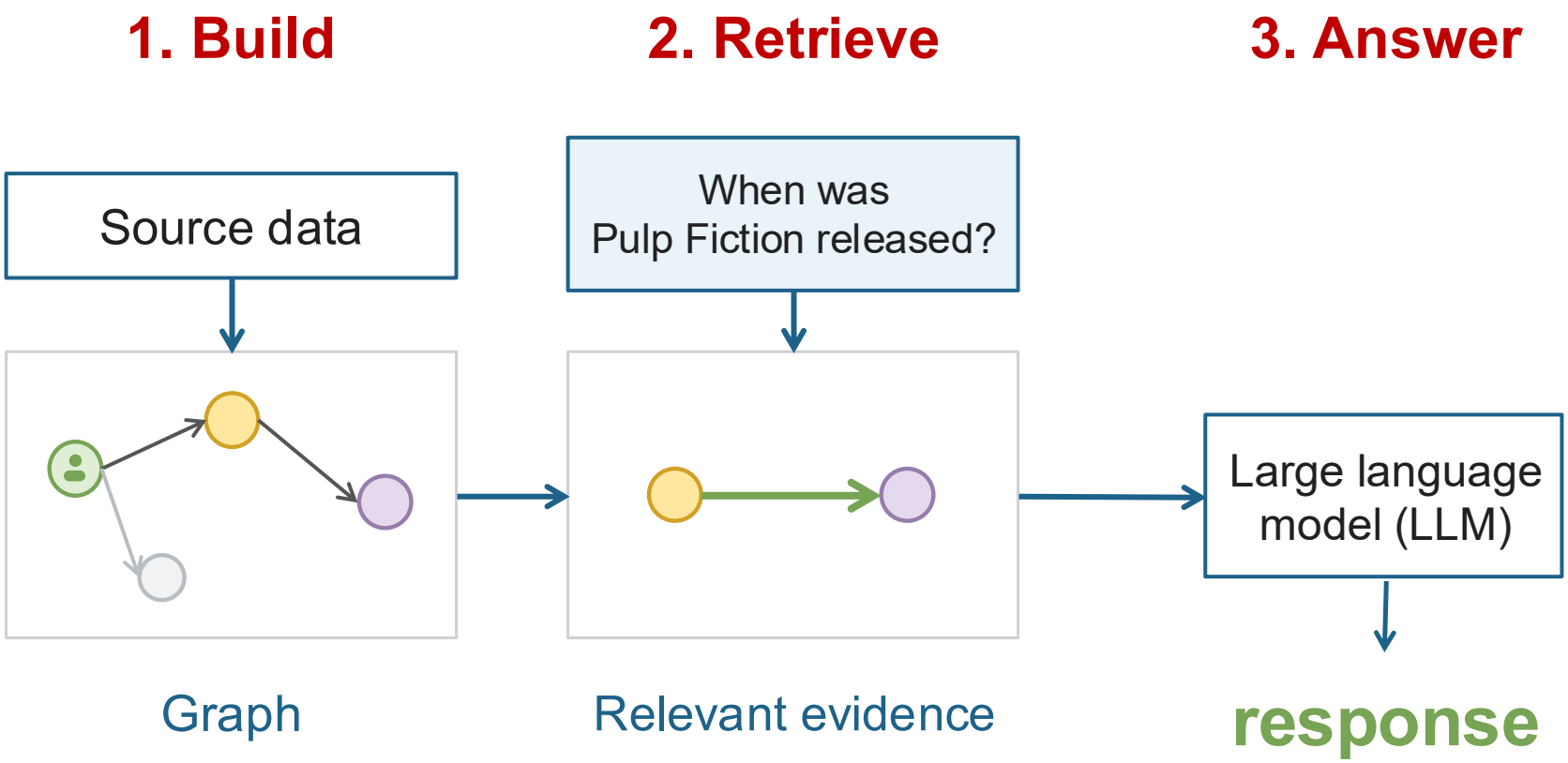
Observe interactions



Use an existing KG



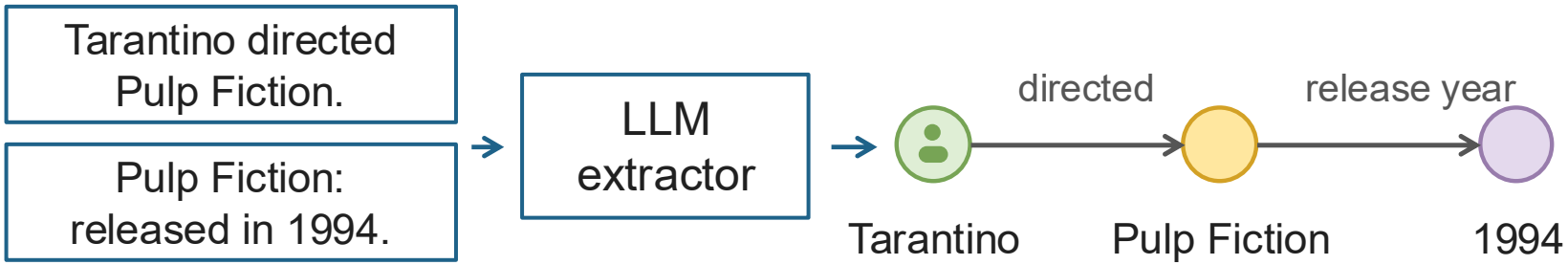
GraphRAG: build, retrieve, answer



Retrieval-augmented generation (RAG) grounds answers in evidence.

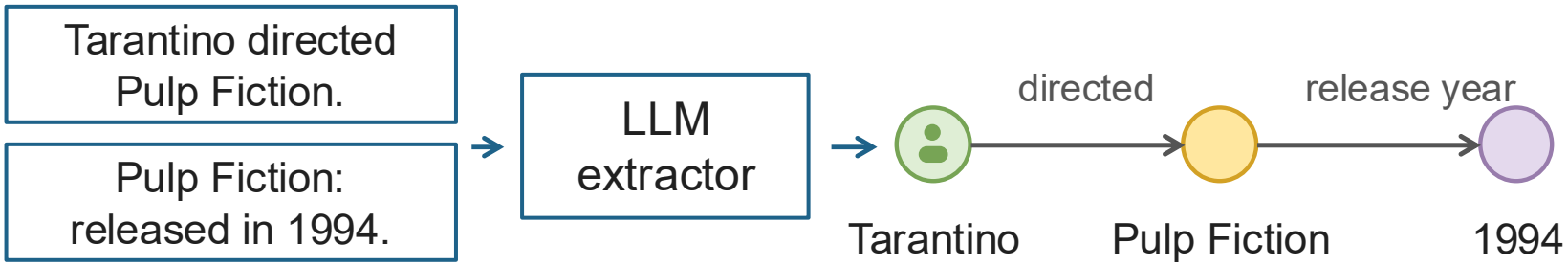
GraphRAG: build, retrieve, answer

1. Build: the LLM extracts facts

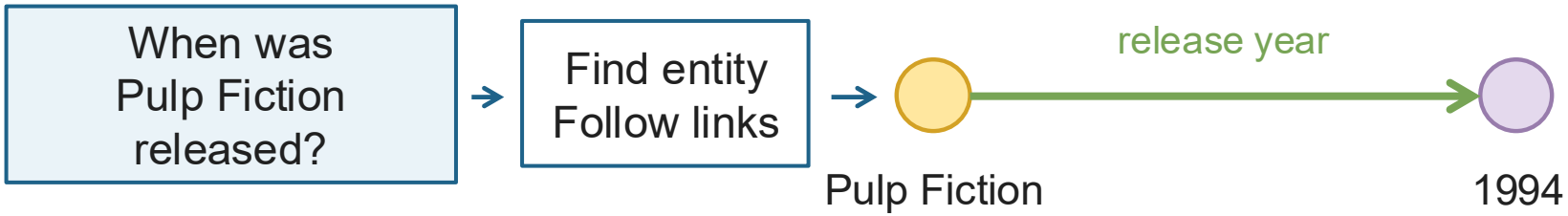


GraphRAG: build, retrieve, answer

1. Build: the LLM extracts facts

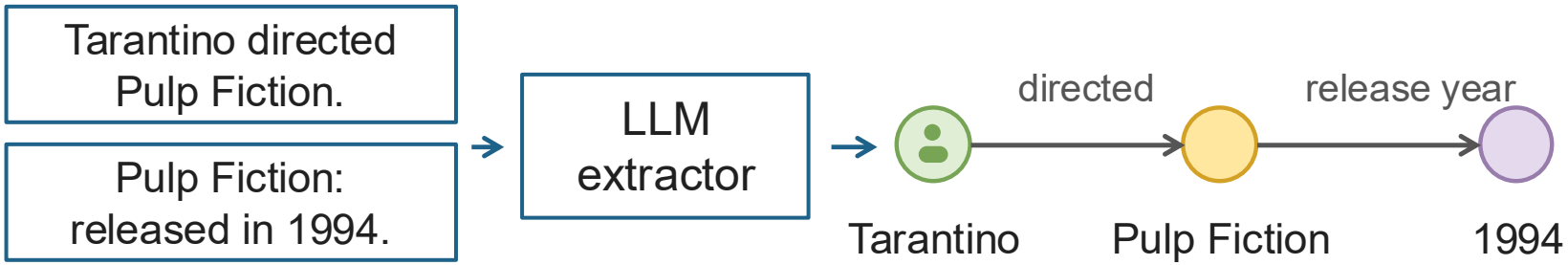


2. Retrieve: find and follow links

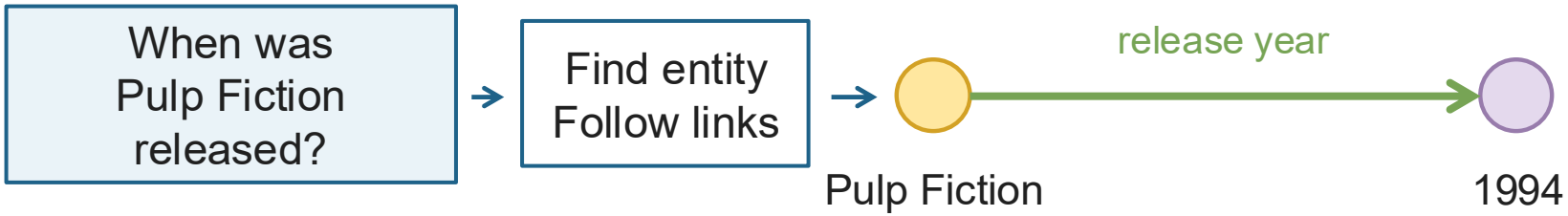


GraphRAG: build, retrieve, answer

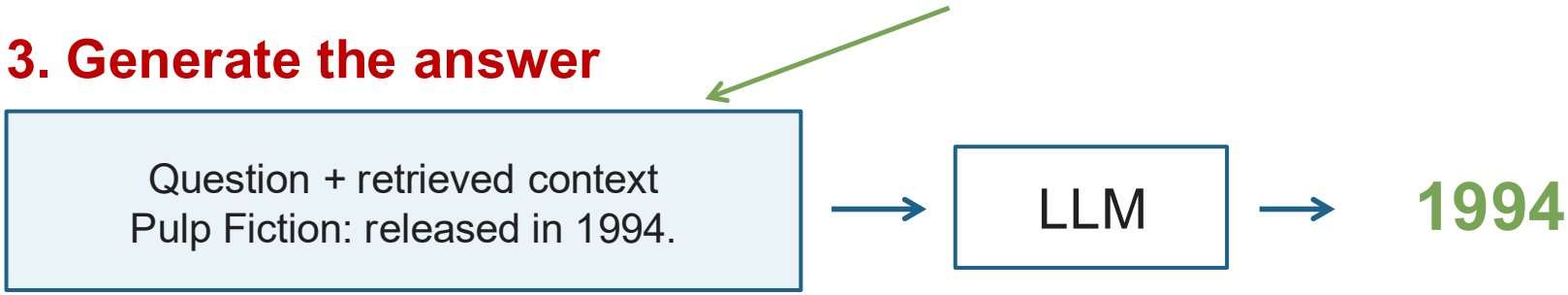
1. Build: the LLM extracts facts



2. Retrieve: find and follow links



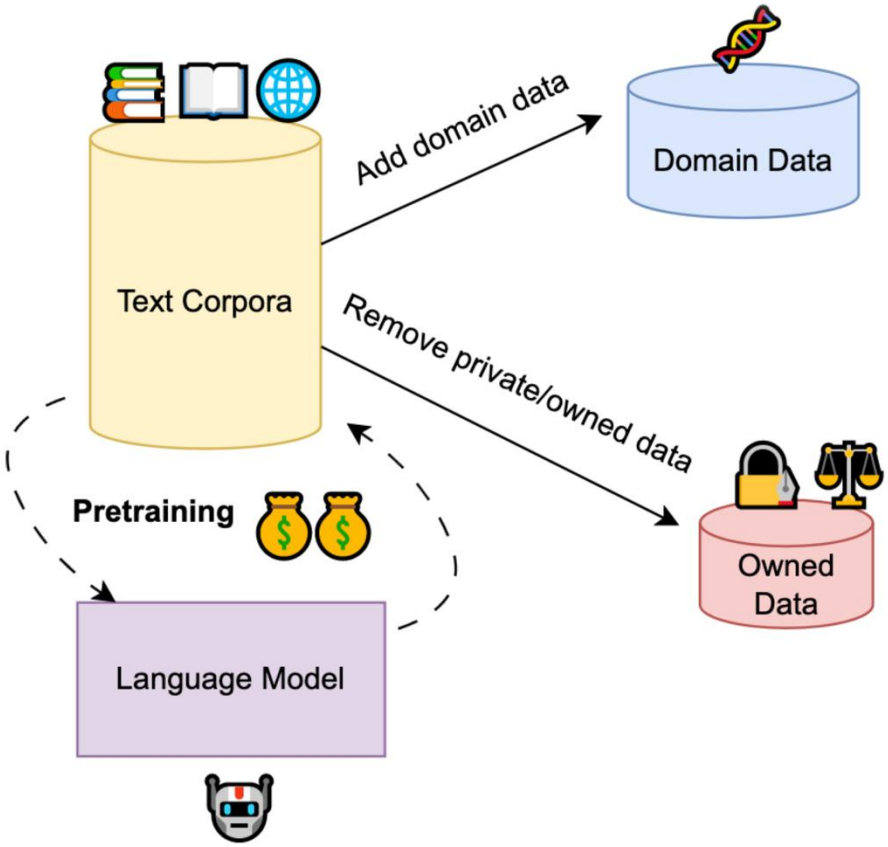
3. Generate the answer



Why Retrieve?

KGs encode information:

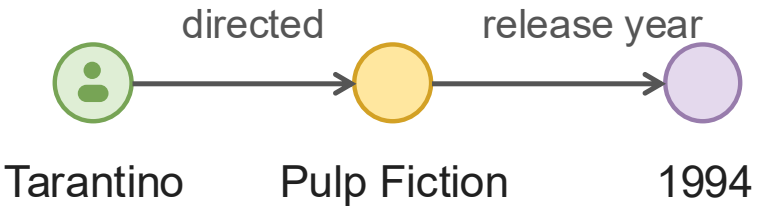
- Models may lack domain knowledge or recent updates.
- Updating knowledge can be…**costly.**



But… why a graph?

Tarantino directed Pulp Fiction.

Pulp Fiction: released in 1994.



But… why a graph?

Information management

Organize and connect dispersed knowledge.



Answer performance

Aggregation
Multi-hop questions

Maintain knowledge

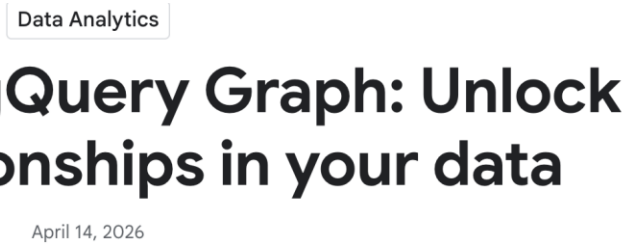
Knowledge updates
Heterogeneous sources

Graphs and ontologies in industry

Databricks



Google Cloud



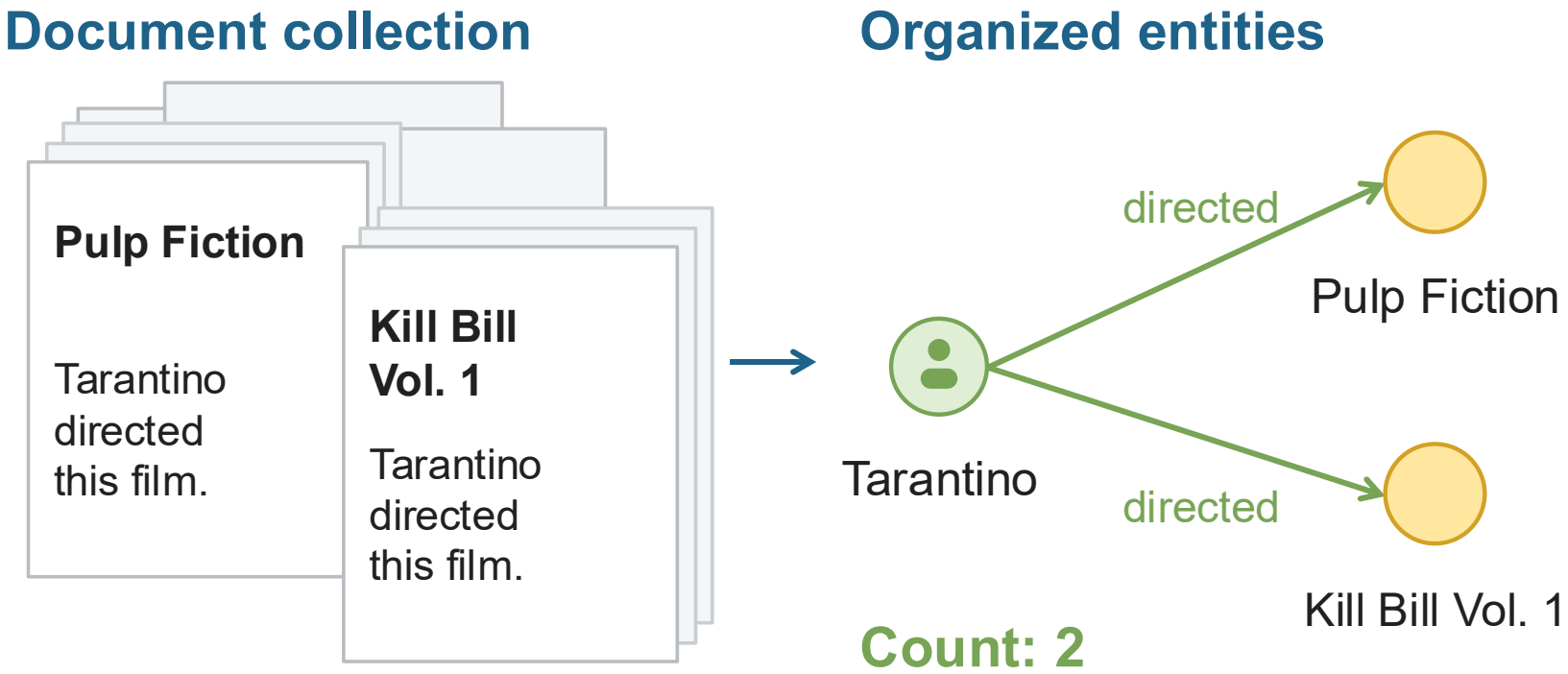
Microsoft

Moving to GraphRAG 1.0 – Streamlining ergonomics for developers and users

Published December 16, 2024

Organizing information

Q: How many Tarantino films are in this collection?

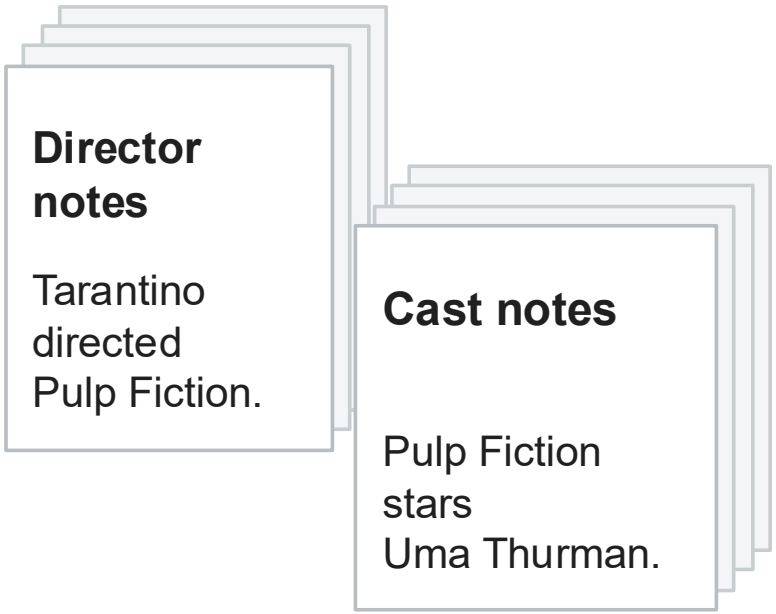


Organize mentions into entities, then aggregate.

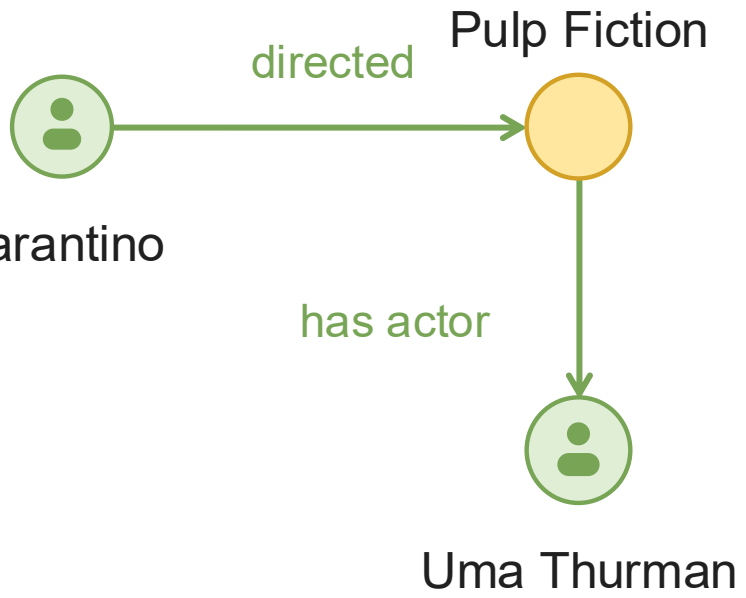
Connecting information

Q: Who acted in a Tarantino film?

Separate collections



Connected evidence

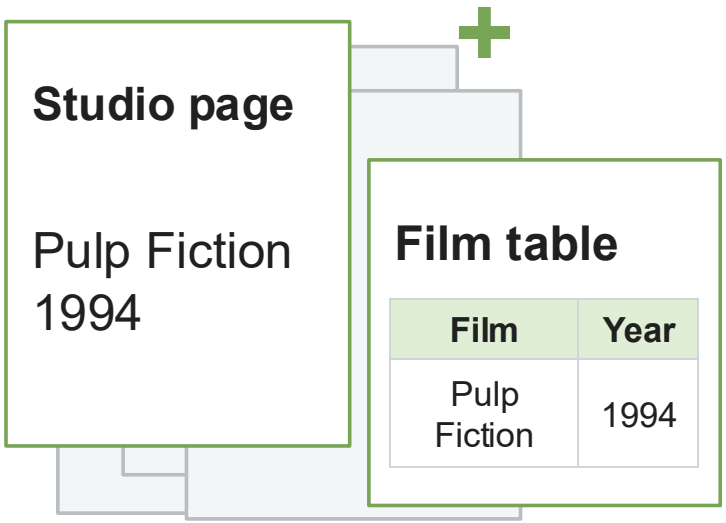


Shared entities connect evidence across document collections.

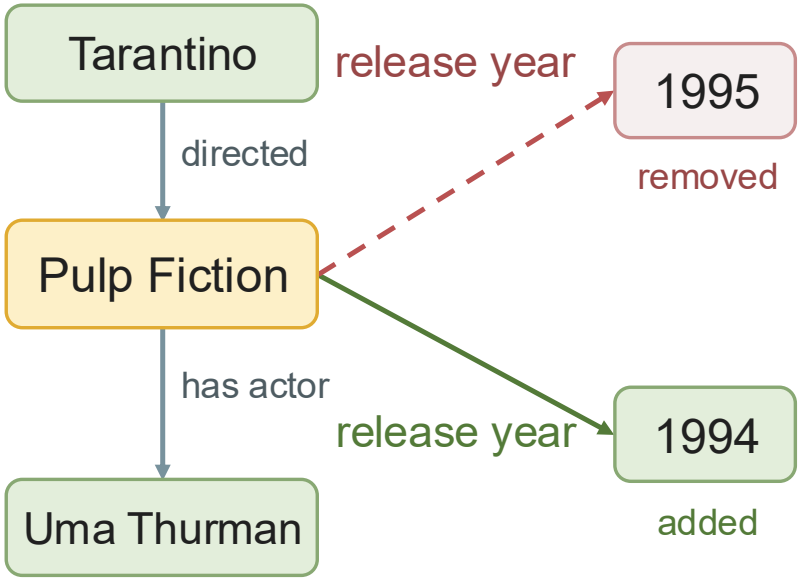
Updating information

New sources revise the same graph.

Added documents + table

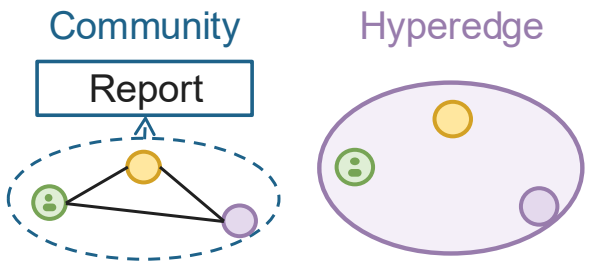
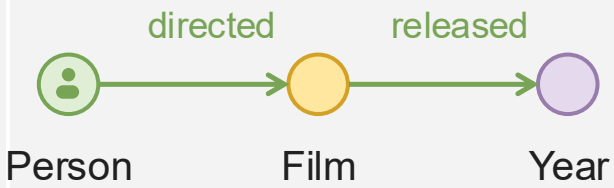
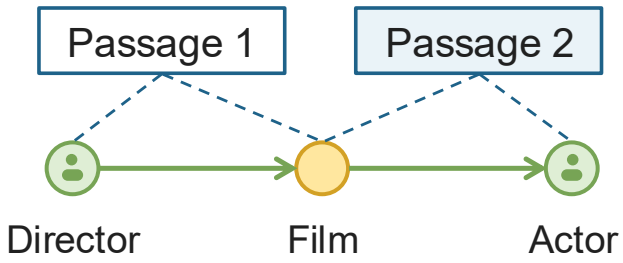


One shared graph

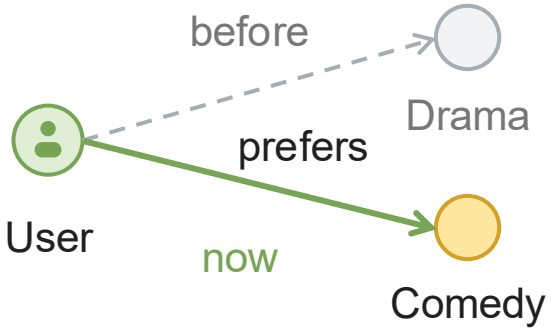
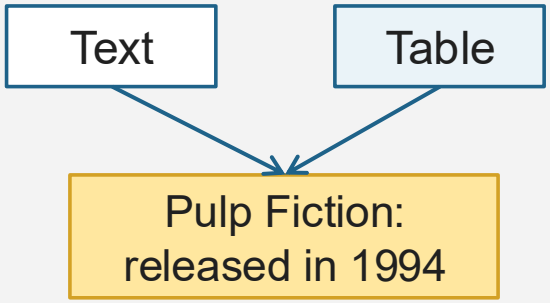


Integrate new sources and update facts without retraining the LLM.

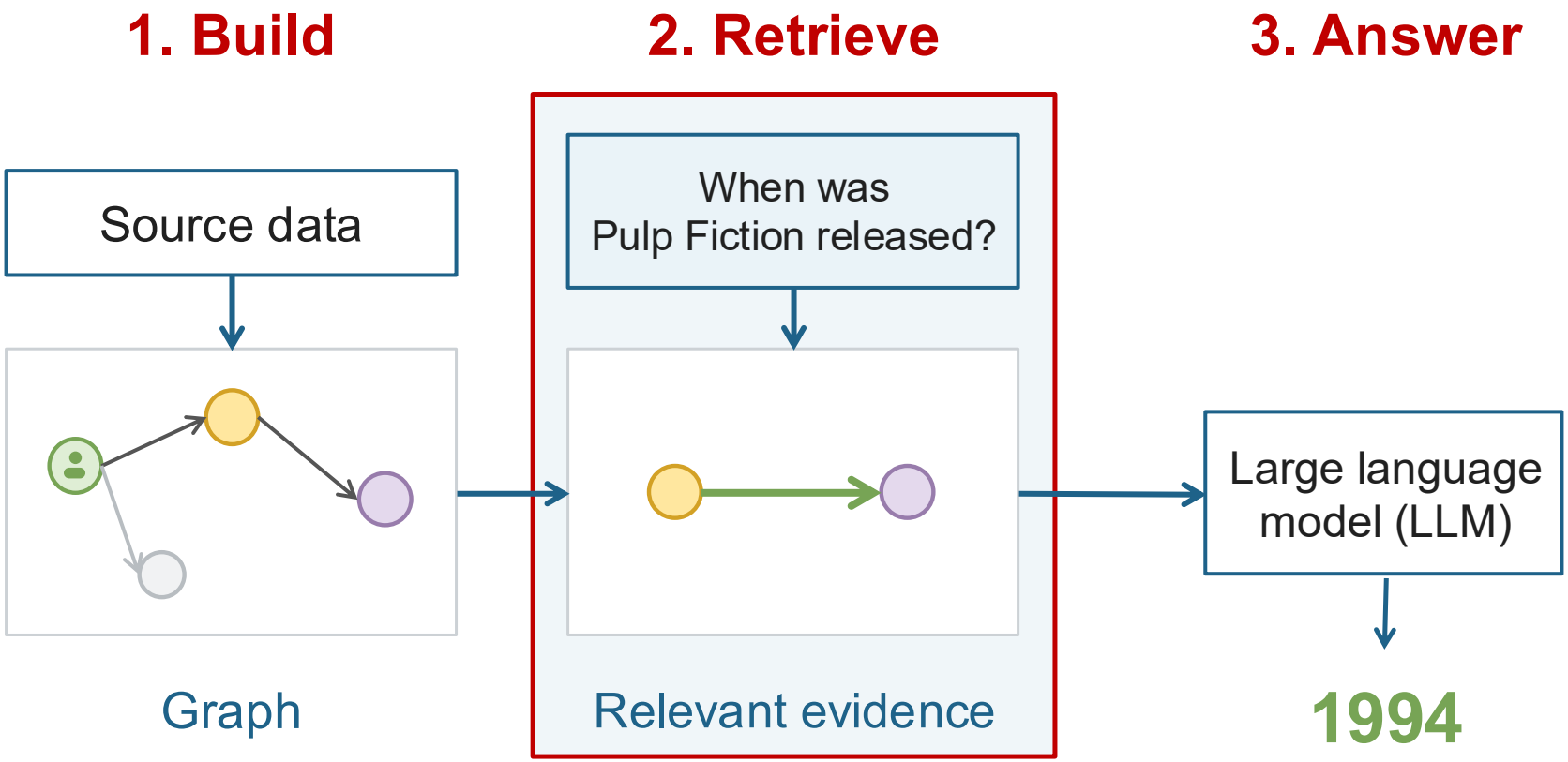
Graph construction approaches

Approach / examples	Graph structure	Example question
Microsoft GraphRAG (2024) HyperGraphRAG (2025)		What themes recur? Who joined this event?
HippoRAG (2024) LightRAG (2024) KGGen (2025)		Which years were Tarantino's films released?
HippoRAG 2 (2025) AWS lexical graph (2025)		Which passages connect Tarantino to this actor?

Graph construction approaches

Approach / examples	Graph structure	Example question
Graphiti (Zep) 32k Github stars		What does this user prefer now?
Semantica 14k Github stars		Which sources support this film fact?

GraphRAG: our focus is retrieval



Part II · GNN-RAG
Retrieval on a known graph

CS224W: Machine Learning with Graphs
Costas Mavromatis
http://cs224w.stanford.edu



GraphRAG (a bit more formally)

1. Graph

Nodes represent entities. Labeled edges represent relations between them.

$$G = (V, \mathcal{R}, E), \quad E \subseteq V \times \mathcal{R} \times V$$

Model input q

In what year was Quentin Tarantino's Palme d'Or–winning film released?

GraphRAG (a bit more formally)

1. Graph

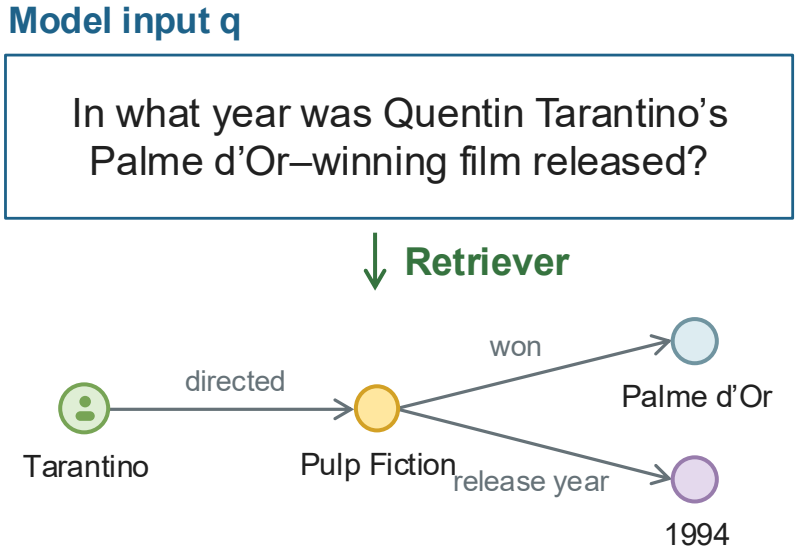
Nodes represent entities. Labeled edges represent relations between them.

$$G = (V, \mathcal{R}, E), \quad E \subseteq V \times \mathcal{R} \times V$$

2. Retrieve

The retriever selects a subgraph of G for the input question q.

$$G_q = R(G, q) \qquad |G_q| \ll |G|$$



GraphRAG (a bit more formally)

1. Graph

Nodes represent entities. Labeled edges represent relations between them.

$$G = (V, \mathcal{R}, E), \quad E \subseteq V \times \mathcal{R} \times V$$

2. Retrieve

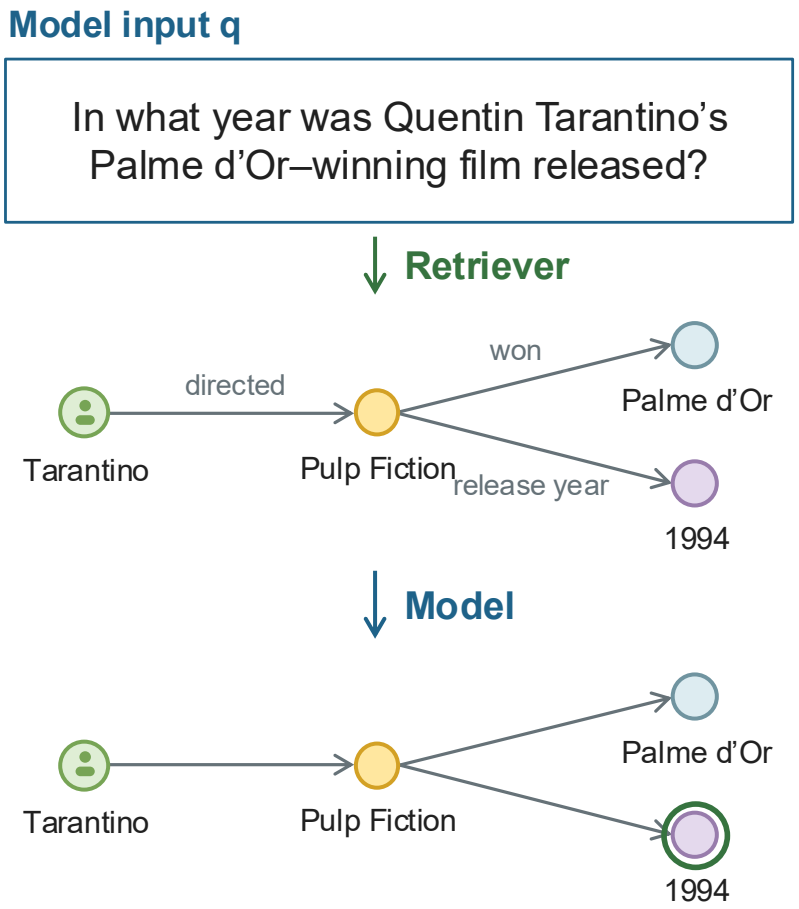
The retriever selects a subgraph of G for the input question q.

$$G_q = R(G, q) \qquad |G_q| \ll |G|$$

3. Answer

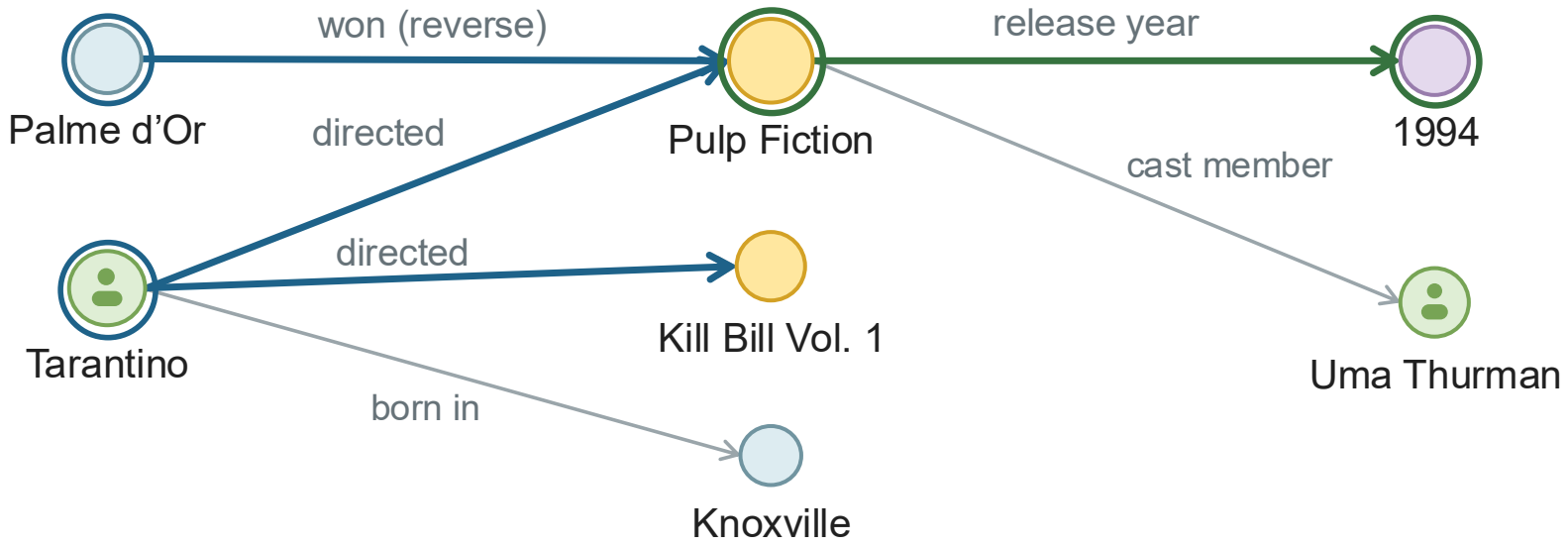
The model predicts the answer from q and the retrieved subgraph.

$$a = M(G_q, q)$$



Retrieval as graph traversal

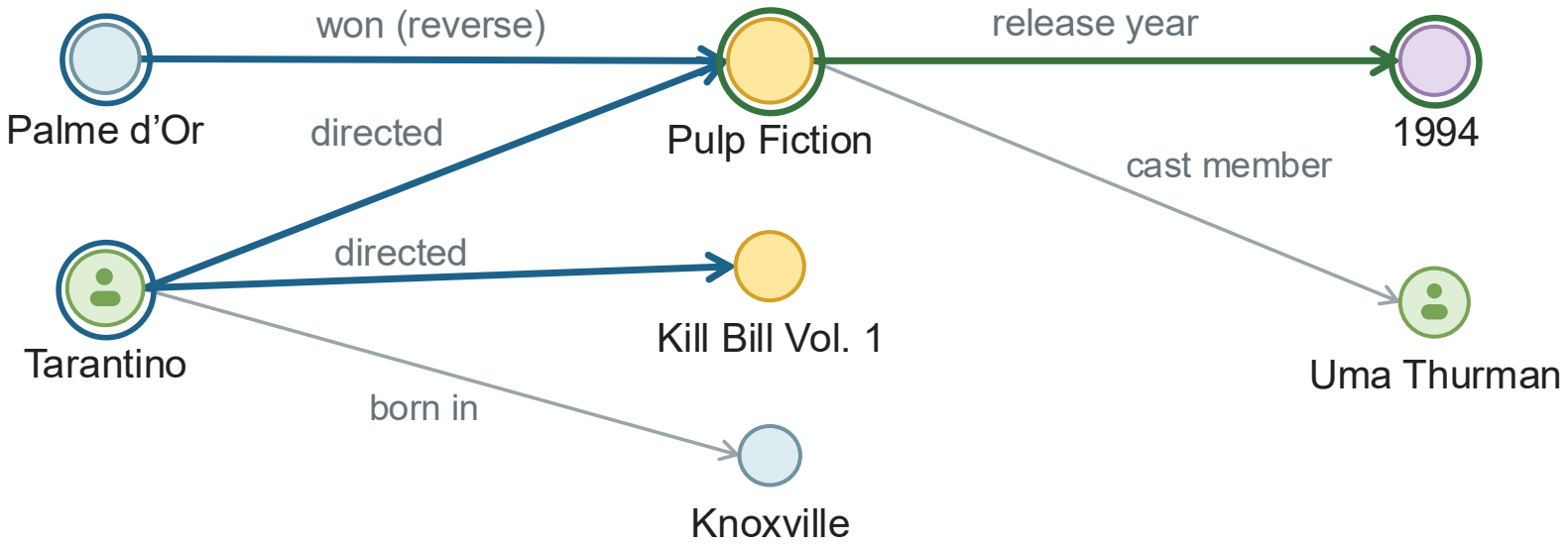
In what year was Quentin Tarantino's Palme d'Or–winning film released?



0. Start **1. Select relations** **2. Select neighbors** **3. Repeat**

Retrieval as graph traversal

In what year was Quentin Tarantino's Palme d'Or–winning film released?

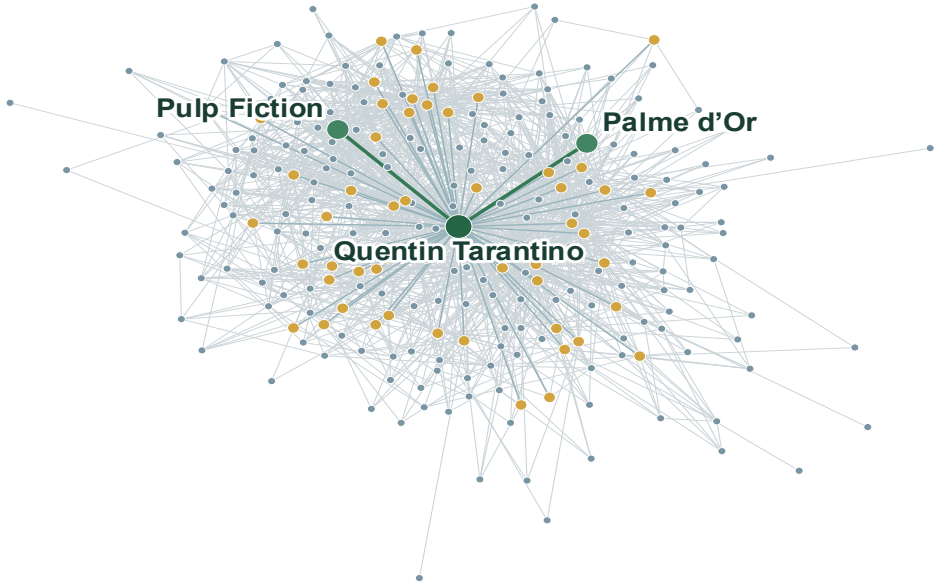


0. Start **1. Select relations** **2. Select neighbors** **3. Repeat**

Let's use an LLM agent!

Wait…

Tarantino's neighborhood…



An agent's context can grow exponentially with search depth.

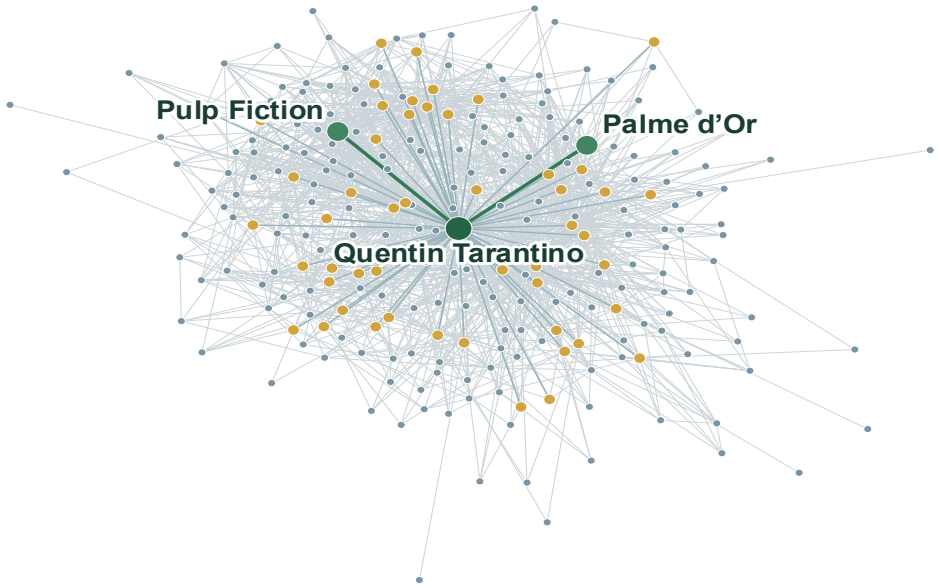
Assume a mean out-degree of 10 and 10 tokens per fact.

≈110,000 tokens after 4 hops



Google Knowledge Graph (2020) **>500B facts**

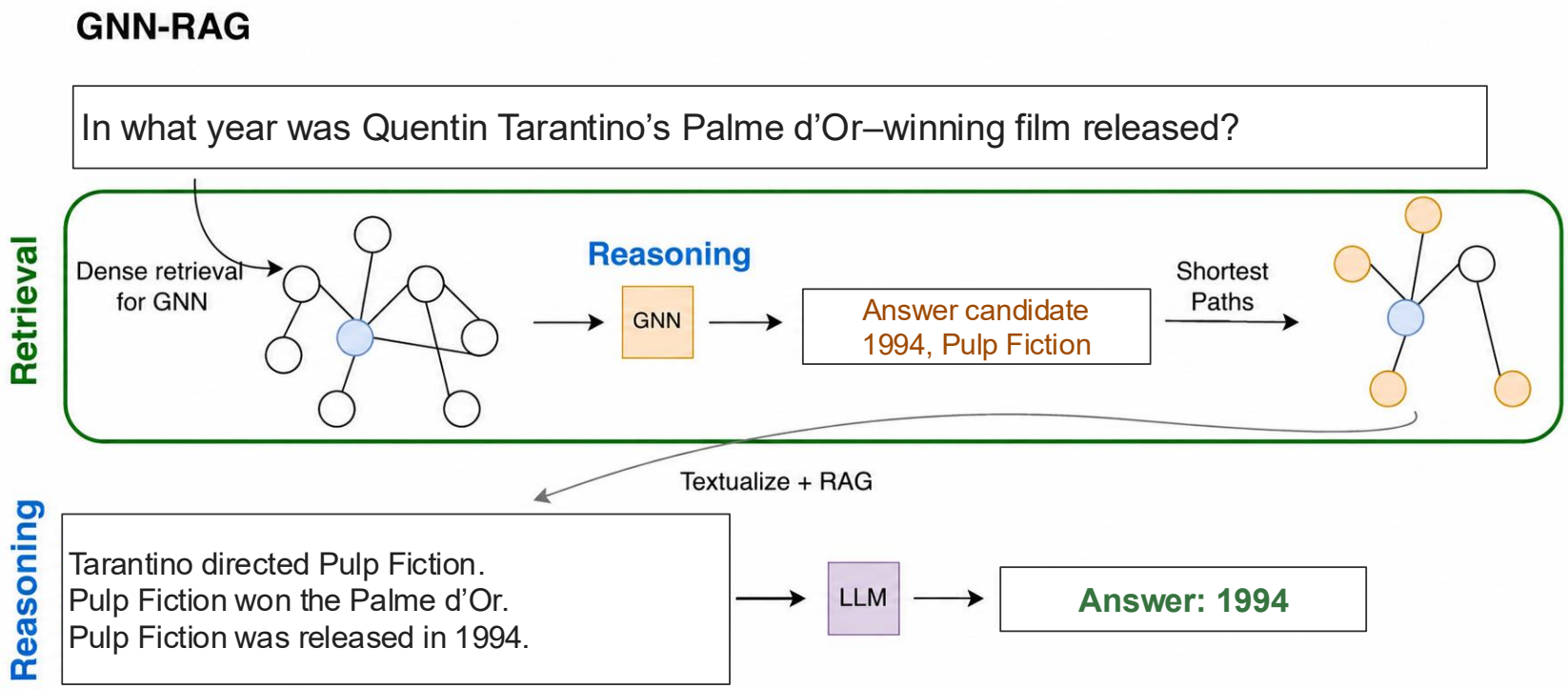
GNNs!



Solution:
Graph Neural Network

<0.2M parameters
no fixed token window

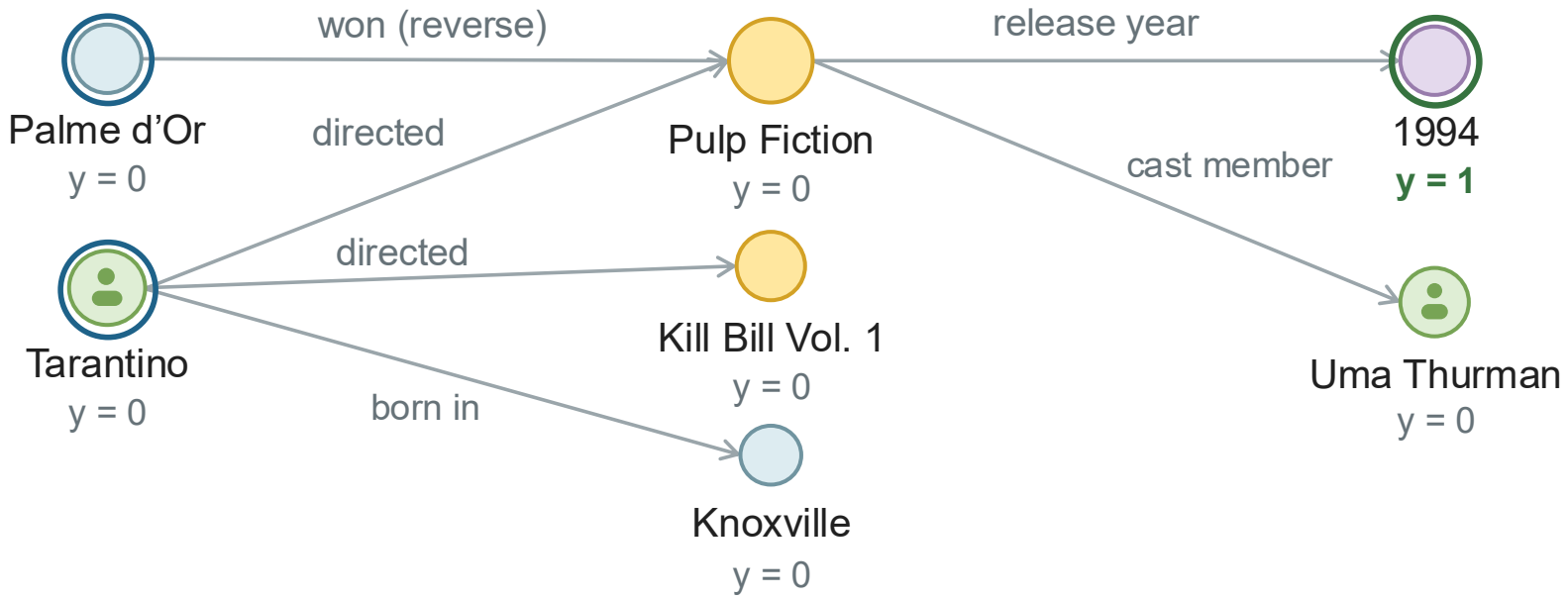
GNN-RAG



The GNN ranks nodes. Shortest paths supply the facts the model uses to answer.

QA as node classification

In what year was Quentin Tarantino's Palme d'Or–winning film released?



$$\hat{\mathbf{y}} = \mathrm{GNN}(G, q)$$

The GNN predicts an answer score for each node in the graph.

GNNs for knowledge graph QA

General GNN layer

Message $\mathbf{m}_u^{(\ell)} = \mathrm{MSG}^{(\ell)}\left(\mathbf{h}_u^{(\ell-1)}\right)$

Aggregation $\mathbf{h}_v^{(\ell)} = \mathrm{AGG}^{(\ell)}\left(\{\mathbf{m}_u^{(\ell)} : u \in N(v)\}, \mathbf{m}_v^{(\ell)}\right)$

GNNs for knowledge graph QA

General GNN layer

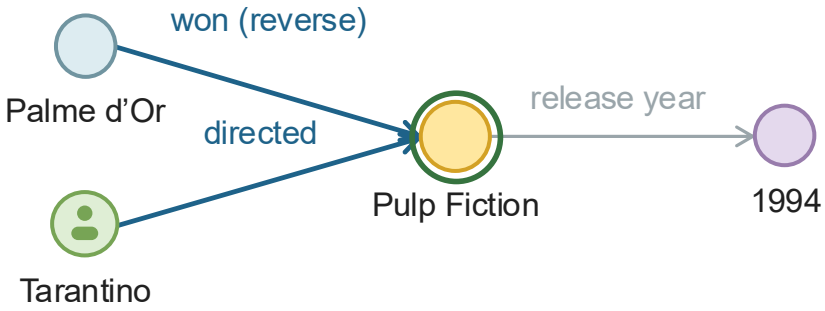
Message $\mathbf{m}_u^{(\ell)} = \mathrm{MSG}^{(\ell)}\left(\mathbf{h}_u^{(\ell-1)}\right)$

Aggregation $\mathbf{h}_v^{(\ell)} = \mathrm{AGG}^{(\ell)}\left(\{\mathbf{m}_u^{(\ell)} : u \in N(v)\}, \mathbf{m}_v^{(\ell)}\right)$

GNN-RAG layer

Edge message $\mathbf{m}_{u\to v,r}^{(\ell)} = p_u^{(\ell-1)}\, \sigma(\mathbf{q} \odot \mathbf{r})$

Node update $\mathbf{h}_v^{(\ell)} = \psi\left(\mathbf{h}_v^{(\ell-1)}, \sum_{(u,r)\in\mathcal{N}_{\text{in}}(v)} \mathbf{m}_{u\to v,r}^{(\ell)}\right)$



Node importance $p_u^{(\ell-1)}$

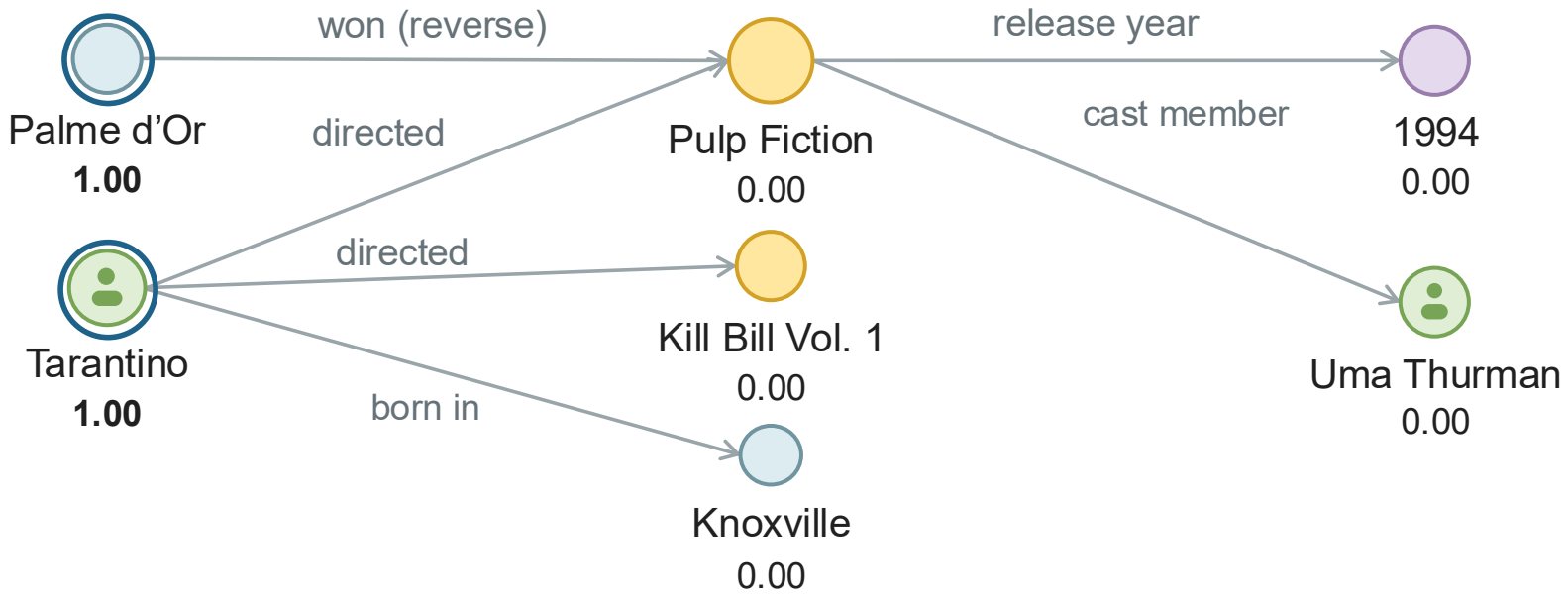
The score of sender **u** at the previous layer weights its message.

Question–relation similarity

Compare the question embedding **q** with the relation embedding **r**.

Walkthrough Example

In what year was Quentin Tarantino's Palme d'Or–winning film released?



Initialization

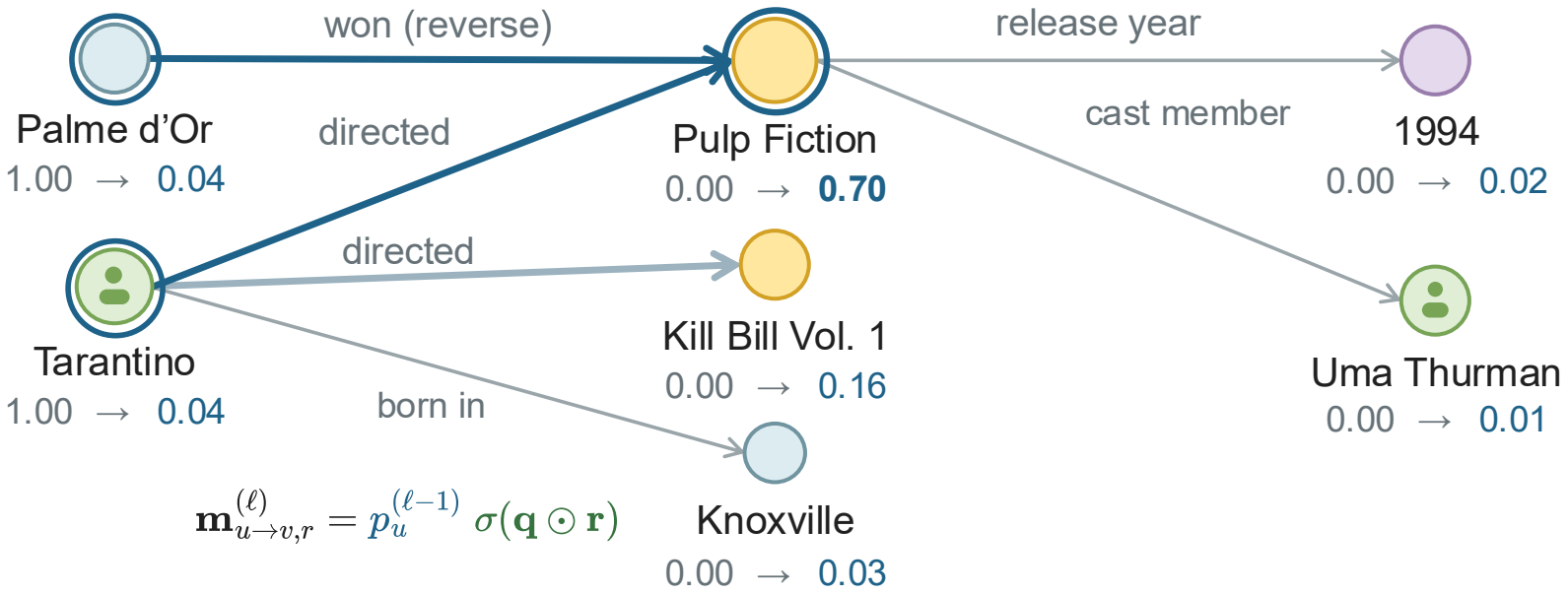
Start at the two entities named in the question. Tarantino and the Palme d'Or each receive a score of 1. Every other node starts at 0.

Initial node scores

$$p_u^{(0)} = \mathbf{1}[u \in S_q]$$

Walkthrough Example

In what year was Quentin Tarantino's Palme d'Or–winning film released?



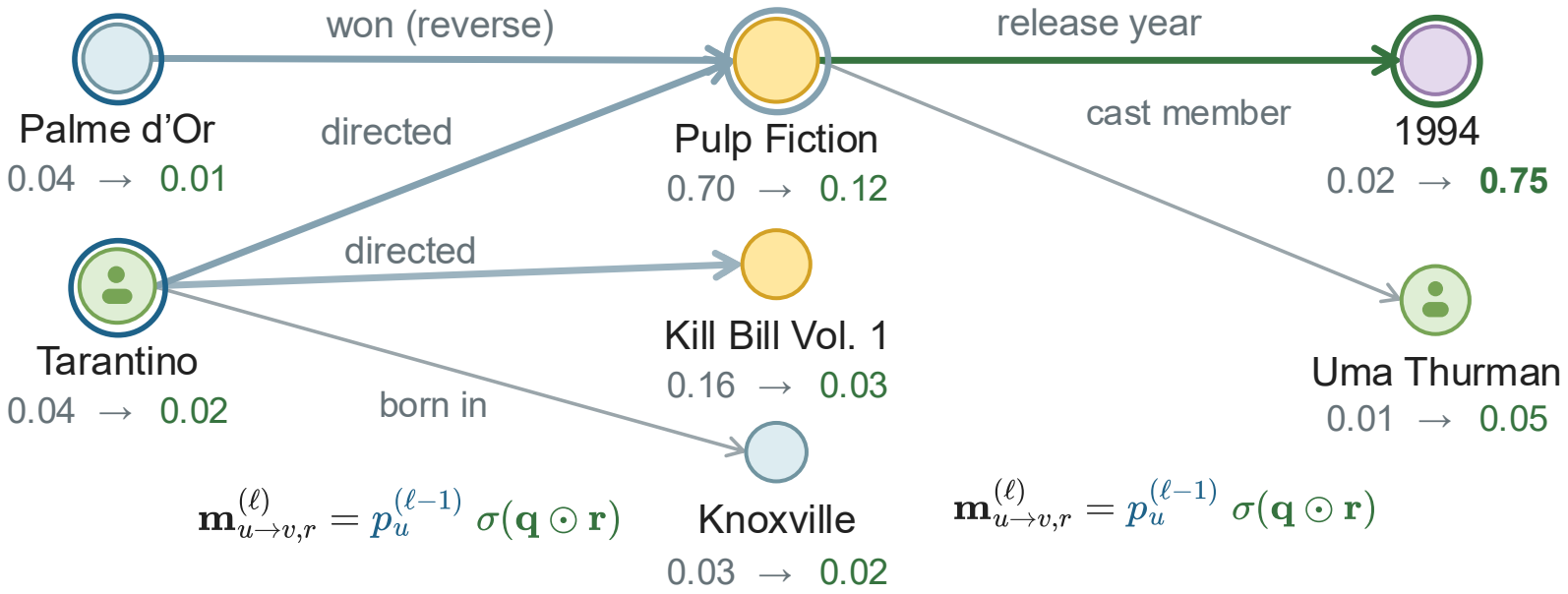
Hop 1. Two messages reach Pulp Fiction

Tarantino reaches both films through "directed". The Palme d'Or also reaches Pulp Fiction through "won" in reverse.

$$\mathbf{p} = \mathrm{softmax}(\mathbf{Hw})$$

Walkthrough Example

In what year was Quentin Tarantino's Palme d'Or–winning film released?



Hop 2. From Pulp Fiction to 1994

The question asks for a year, so the "release year" message carries more weight than "cast member".

$$\mathbf{p} = \text{softmax}(\mathbf{Hw})$$

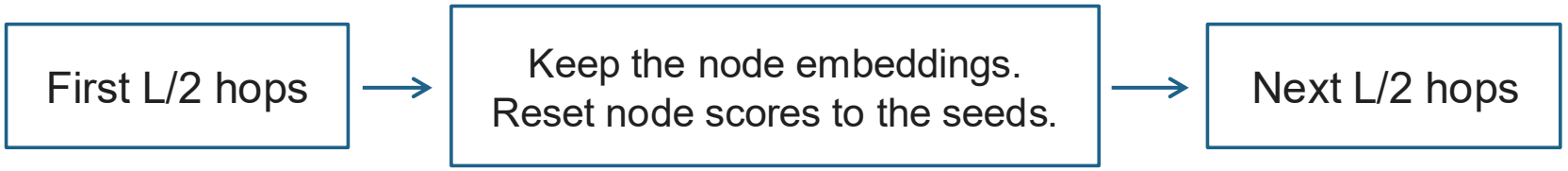
GNN-RAG (a few more details)

1. Multi-head message passing

The walkthrough used one head. Each of the K heads learns where to attend.

$$\mathbf{q}^{(k)} = \gamma_k(\mathrm{LM}(q)), \quad k = 1, \ldots, K$$

2. Warm-start



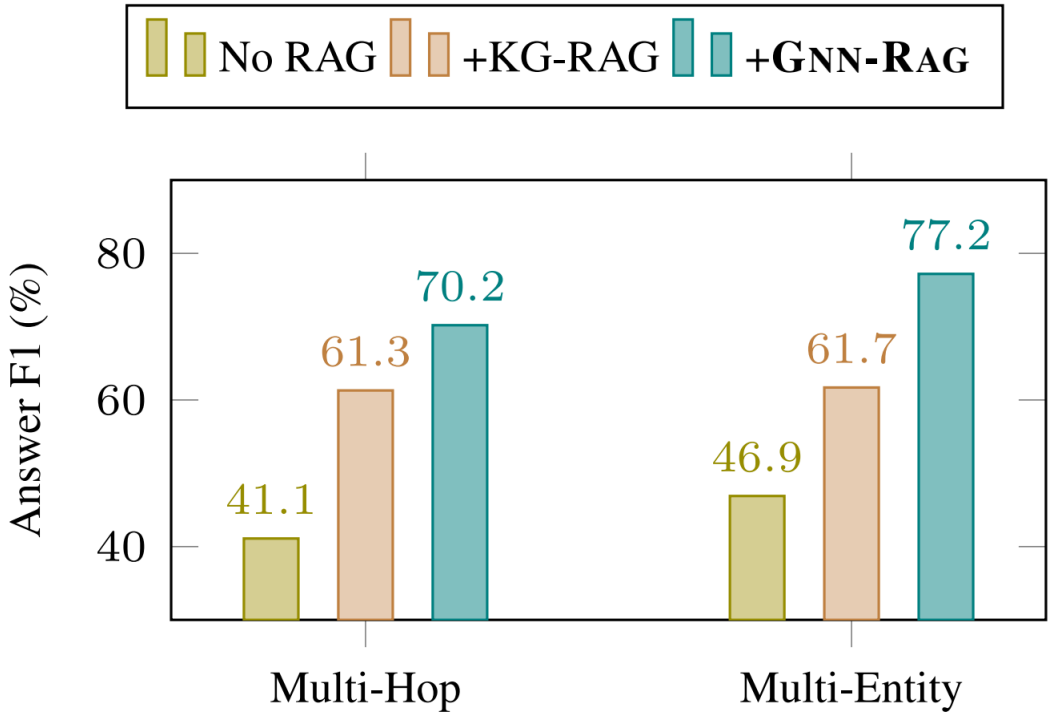
3. Train on the answer nodes

$$\mathcal{L} = \mathrm{loss}\left(\mathbf{p}^{(L)}, \tilde{\mathbf{y}}\right)$$

4. Retrieve paths

Retrieve paths from linked question entities to predicted answers. Give these paths to the LLM as context.

Results on complex questions



+8.9

F1 points on multi-hop questions

+15.5

F1 points on multi-entity questions

GNN-RAG accuracy and efficiency

Answer accuracy (Hit %)

Method	WebQSP	CWQ
ToG (70B)	68.9	57.6
ToG (ChatGPT)	76.2	58.9
RoG (fine-tuned 7B)	85.7	62.6
G-Retriever	70.1	—
GNN-RAG	**85.7**	**66.8**

Retrieval efficiency on CWQ

Retriever	Recall@10 (%)	Median KG tokens
RoG	54.5	201
SubgraphRAG	58.7	1,442
GNN-RAG	**64.1**	**114**

On CWQ, GNN-RAG improves Hit by 4.2 percentage points over RoG while reducing the median graph context by 43%.

11/19/2026 Stanford CS224W: Machine Learning with Graphs

Part III · BYOKG-RAG
Retrieval on unseen graphs

CS224W: Machine Learning with Graphs
Costas Mavromatis
http://cs224w.stanford.edu



What if the graph is new?

1. Graph

Part II assumed a known graph. Now the client supplies a new graph and its schema.

$$G = (V, \mathcal{R}, E), \quad E \subseteq V \times \mathcal{R} \times V$$

2. Retrieve

Select evidence for the question.

$$G_q = R(G, q)$$

3. Answer

Answer from the retrieved evidence.

$$a = M(G_q, q)$$

In more detail… graph schemas

The **schema** tells which node types, relations and properties it can query.

`(:Film)-[:RELEASE_YEAR]->(:Year)`

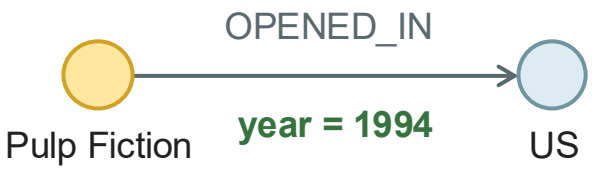
Film.title (string) Year.value (integer)



`(:Film)-[:OPENED_IN]->(:Market)`

Film.title (string) Market.code (string)

OPENED_IN.year (integer)



In more detail… graph queries

Graph stores such as Amazon Neptune and Neo4j can return paths that match a query.

```
MATCH p = (:Film {title: 'Pulp Fiction'})
  -[:RELEASE_YEAR]->(y:Year)
RETURN p, y.value AS year
```



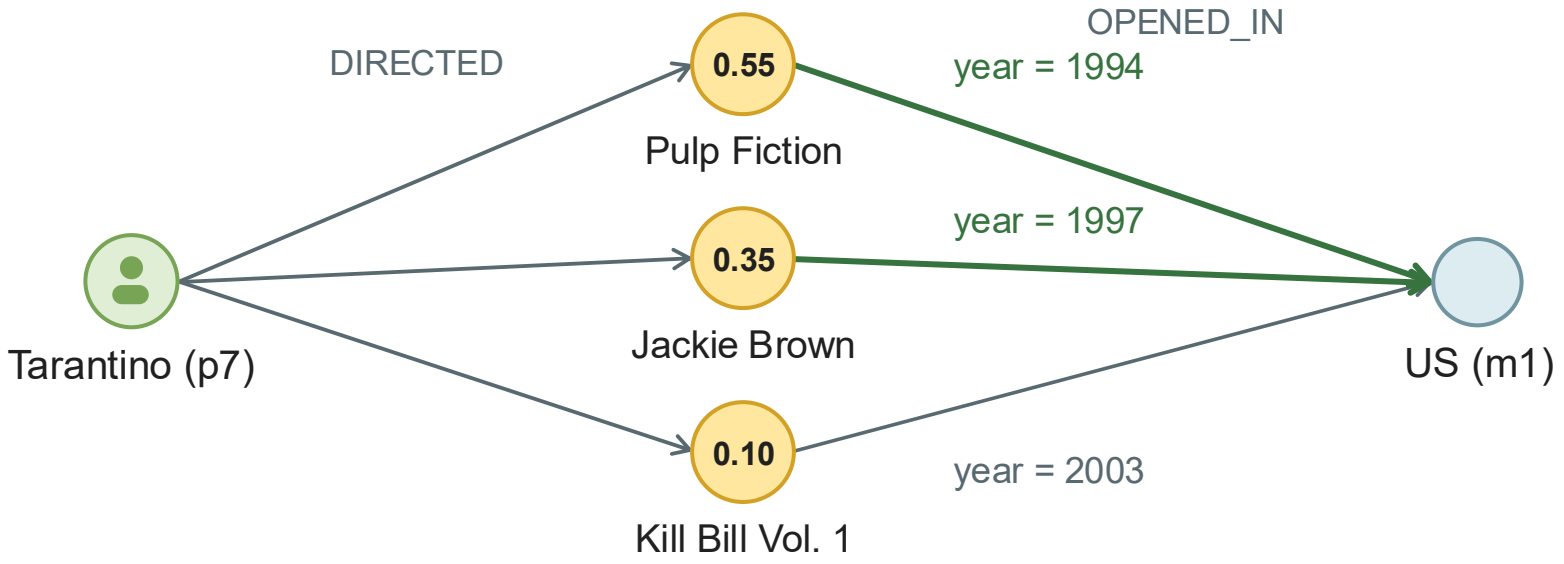

```
MATCH p = (:Film {title: 'Pulp Fiction'})
  -[r:OPENED_IN]->(:Market {code: 'US'})
RETURN p, r.year AS year
```



The question changes retrieval

Model input q

How many films directed by Tarantino opened in the US before 2000?



GNN-RAG-style ranking

probabilities

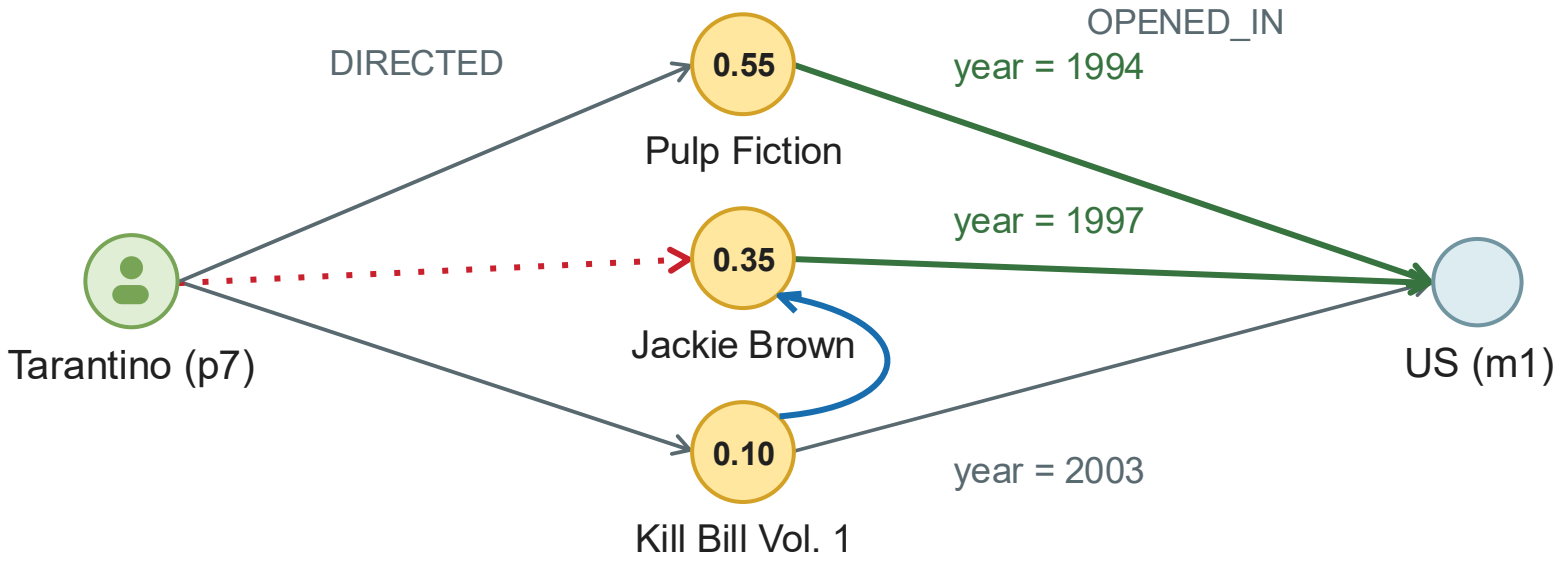
Graph query

```
MATCH (:Person {id: 'p7'})-[:DIRECTED]->(f:Film)
MATCH (f)-[r:OPENED_IN]->(:Market {id: 'm1'})
WHERE r.year < 2000
RETURN count(DISTINCT f) AS films
```

..and the graph itself

Model input q

How many films directed by Tarantino opened in the US before 2000?



GNN-RAG-style ranking

probabilities

Graph query

```
MATCH (:Person {id: 'p7'})-[:DIRECTED]->(f:Film)
MATCH (f)-[r:OPENED_IN]->(:Market {id: 'm1'})
WHERE r.year < 2000
RETURN count(DISTINCT f) AS films
```

1 answer missed

How do we retrieve from a new graph?

1. Graph

Part II assumed a known graph. Now the client supplies a new graph and its schema.

$$G = (V, \mathcal{R}, E), \quad E \subseteq V \times \mathcal{R} \times V$$

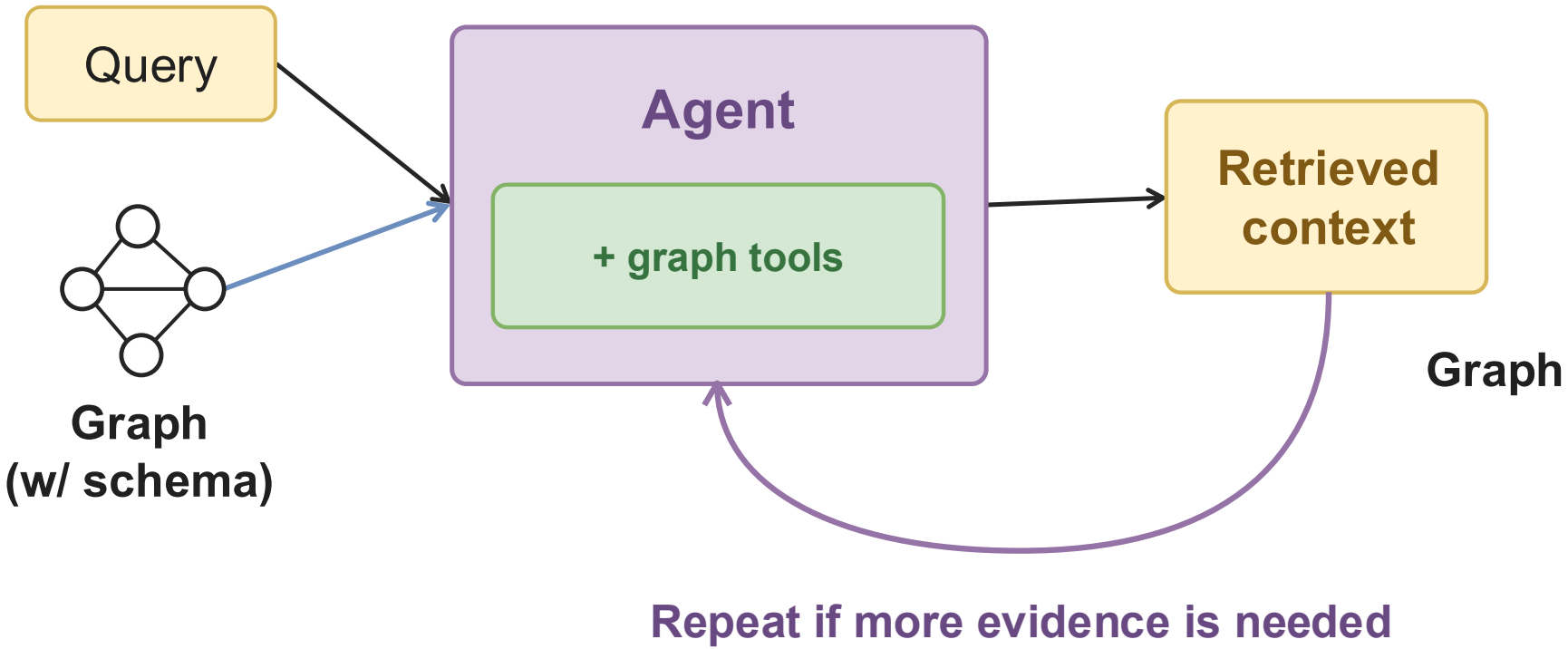
2. Retrieve

$$G_q = R(G, q)$$

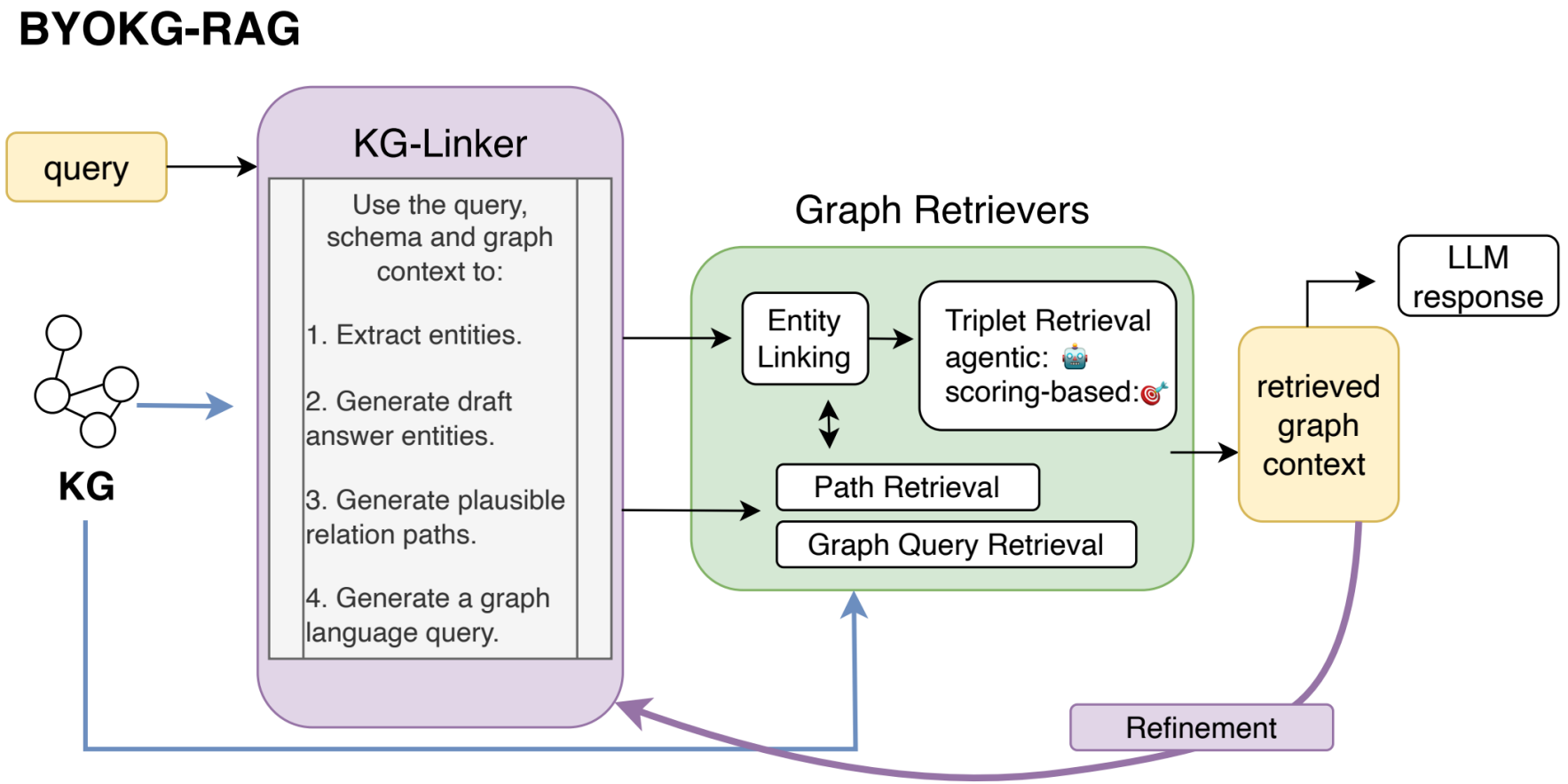
Let's use an LLM agent! (but with more tools)

```
MATCH (:Person {id: 'p7'})-[:DIRECTED]->(f:Film)
MATCH (f)-[r:OPENED_IN]->(:Market {id: 'm1'})
WHERE r.year < 2000
RETURN count(DISTINCT f) AS films
```

Graph retrieval with an LLM agent



The BYOKG-RAG framework



Bring your graph in practice

Initialize the engine and retrieve context for a question.

```
from graphrag_toolkit.byokg_rag import graphstore

graph_store = graphstore.NeptuneAnalyticsGraphStore(
   graph_identifier=graph_id)

byokg_query_engine = ByoKGQueryEngine(
   graph_store=graph_store, llm_generator=llm)



question = input("Question: ")
context = byokg_query_engine.query(question)
```

https://github.com/awslabs/graphrag-toolkit/tree/main/byokg-rag

Configuring the graph tools

Connect the engine to the graph through four tools.

```
byokg_query_engine = ByoKGQueryEngine(
    graph_store=graph_store, llm_generator=llm,

    entity_linker=EntityLinker(entity_matcher),

    path_retriever=PathRetriever(
        path_verbalizer=PathVerbalizer()),

    triplet_retriever=AgenticRetriever(
        llm_generator=llm, graph_traversal=traversal,
        graph_verbalizer=TripletGVerbalizer()),

    graph_query_executor=GraphQueryRetriever(graph_store),
)
```

Results on diverse benchmarks

Zero-shot graph question answering

	Freebase-KG		Temp-Wikidata	DiseaseDB-KG	Cypher-KG
	WebQSP-IH	CWQ-IH	CronQuestions	MedQA	Northwind
graph-query$^{\alpha}$	75.2	54.3	19.2	<u>62.5</u>	<u>55.3</u>
Text-based Retrieval$^{\beta}$	70.8	56.8	<u>59.8</u>	58.1	0.7
Graph Reranker$^{\gamma}$	76.0	63.0	41.8	58.2	3.4
Agentic Traversal$^{\delta}$	<u>86.2</u>	<u>69.3</u>	57.3	59.2	3.4
BYOKG-RAG	**86.6**	**73.6**	**65.5**	**65.0**	**64.9**

Better generalization with multiple tools!

Case Study

Table 5: Case study on BYOKG-RAG's iterative retrieval.

Question	In what years did Stan Kasten's organization win the World Series?
Answer	1963 World Series | 1988 World Series | 1965 World Series | 1981 World Series | 1959 World Series
BYOKG-RAG (1st iteration)	m.0_yv0g3 -> organization.leadership.person -> Stan Kasten m.0_yv0g3 -> organization.leadership.organization -> Los Angeles Dodgers Stan Kasten -> business.board_member.leader_of > m.0_yv0g3-> organization.leadership.organization -> Los Angeles Dodgers
BYOKG-RAG (2nd iteration)	Los Angeles Dodgers -> sports.sports_team.championships -> **1963 World Series | 1988 World Series | 1965 World Series | 1981 World Series | 1959 World Series**

Summary

Answer the question with evidence from the graph.

I. Why a graph?	Connect and update facts across sources.
II. A known graph	GNN-RAG learns which nodes matter for the question and retrieves paths that connect them.
III. An unseen graph	BYOKG-RAG uses graph tools to retrieve evidence.

References

Thank You!

GNN-RAG: Graph neural retrieval for efficient large language model reasoning on knowledge graphs
C Mavromatis, G Karypis

ReaRev: Adaptive Reasoning for Question Answering over Knowledge Graphs
C Mavromatis, G Karypis

Byokg-rag: Multi-strategy graph retrieval for knowledge graph question answering
C Mavromatis, S Adeshina, VN Ioannidis, Z Han, Q Zhu, I Robinson, B Thompson, H Rangwala, G Karypis