A background network graph with nodes of various colors (blue, cyan, purple, green) and sizes, connected by thin grey lines. The nodes are scattered across the slide, with some larger nodes and some smaller ones.

# Deriving Maximum Insight: Open-Source Graph-Enhanced RAG for Complex Question Answering

Mykyta Kemarskyi  
CTO 12new.ai



What is our refund policy?

### Knowledge Base

Unstructured documents



### Retriever

(Top-K similar chunks)



### LLM

(Context + question)



### Generated answer



We offer a 30-day money-back guarantee for all purchases. To request a refund, please contact our support team and provide your order number.

# Vanilla RAG struggles with:

- Multi-hop reasoning
- Modeling relationships between entities
- Global queries (e.g. «What are the main themes in this data?») )
- Contradiction & inconsistency detection (e.g. “Which sources disagree about who owns Company X?”)
- Temporal reasoning & evolution over time (e.g. “How did the ownership of Company X change between 2018–2023?”)

# Promising solution: RAG + knowledge graphs

- GraphRAG Pattern Catalog - <https://graphrag.com/reference/>

Solution	License	Automatic knowledge graph building from unstructured text	Context retrieval	Generation of the answer using the retrieved context
Microsoft GraphRAG <a href="https://github.com/microsoft/graphrag">https://github.com/microsoft/graphrag</a>	MIT	✓	✓	✓
LlamaIndex (PropertyGraphIndex) <a href="https://github.com/run-llama/llama_index">https://github.com/run-llama/llama_index</a>	MIT	✓	✓	✓
Graphiti <a href="https://github.com/getzep/graphiti">https://github.com/getzep/graphiti</a>	Apache-2.0	✓	✓	✗

# Text corpora for testing

Text corpus	Length	Size (UTF-8 encoding)
«A Christmas Carol in Prose; Being a Ghost Story of Christmas» by Charles Dickens	28,541 words	162,261 bytes
Synthetic events dataset (toy dataset)	1798 chars JSON	1798 bytes

# Microsoft GraphRAG

<b>Custom ontology support</b>	<b>×</b>
<b>Approach</b>	<ul style="list-style-type: none"><li>- Clusters related entities into hierarchical communities</li><li>- Generates summary of each community</li><li>- Uses these summaries to provide global context to LLM</li></ul>
<b>Interface</b>	<ul style="list-style-type: none"><li>- CLI</li><li>- Python API</li></ul>
<b>Assumed use case</b>	<b>Query-Focused Summarization</b> (answering global queries) on the static knowledge base
<b>Caveats</b>	<ul style="list-style-type: none"><li>- Costs (indexing, query) - &gt;100x expensive than vanilla RAG</li><li>- Slow (10s of seconds per query)</li><li>- No incremental indexing</li><li>- Not quite production-ready</li><li>- Difficult to trace answer provenance (original sources contributing to the answer)</li></ul>

<b>Performance on the «Christmas Carol» text corpus (~29K English words)</b>	<b>Microsoft GraphRAG</b>	<b>Vanilla RAG</b>
<b>RAG configuration</b>	<b>GraphRAG defaults (chunk size - 1200 chars, overlap - 100 chars)</b>	<b>chunk size - 1200 chars, overlap - 100 chars Top 3 similar chunks (no rerank)</b>
<b>Indexing cost (tokens)</b>	- Inference (input): ~335K tokens - Inference (output): ~727K tokens - Embedding: ~76K tokens	Embedding: ~44K tokens
<b>Indexing time (seconds)</b>	390 seconds	3 seconds
<b>Query cost (tokens)</b>	Global query - inference (input): ~28K tokens - inference (output): ~5K tokens Local query: - inference (input): ~11K tokens - inference (output): ~2K tokens	- inference (input): ~3500 tokens - inference (output): ~700 tokens
<b>Query time (seconds)</b>	Global query: ~78 seconds Local query: ~50 seconds	~6.5 seconds

# Microsoft GraphRAG Query types

Query type	Focus
Local	Retrieval of information about specific entity
Global	Global queries
Drift	Local, but expanded with summaries
Basic	Vanilla RAG (designed to compare results depending on the type of query)

**You are supposed to choose a query type explicitly!  
No auto-selection**

```
graphrag query \  
> --root . \  
> --method local --verbose \  
> --query "Who participated in the family dinner?"
```

There are two dinners described in the data, each with different participants.

Cratchit Christmas Day dinner (the Cratchit family)


- Participants include Bob Cratchit and Mrs Cratchit, along with their children Martha Cratchit, Peter Cratchit (often called Master Peter), Belinda Cratchit, and Tiny Tim. The scene also notes that there are other children in the family dining together.
- This is supported by:
  - Relationships indicating the parental units and children (e.g., Bob Cratchit with Christmas Day; Mrs Cratchit with Christmas Day; Bob Cratchit with Martha, Peter, Tiny Tim, Belinda, etc.) [Data: Relationships (107, 115, 102, 104, 108, 110, 111, 112); Entities (74, 75, 78, 77, 76)].

Christmas Dinner at Fred's House

- Participants include Scrooge and Fred, with "his relatives and friends" noted as attendees. Topper is described in related materials as a guest in Scrooge's family-centered gatherings, suggesting additional familiar faces at related Christmas gatherings.
- This is supported by:
  - The Christmas Dinner at Fred's House entry (Scrooge and Fred's relatives and friends) [Data: Entities (128)].
  - A related note that Topper is present in Scrooge's Christmas context [Data: Relationships (133); Entities (120, 122)].


If you'd like, I can list the exact named individuals from the Cratchit family dinner (Bob, Mrs. Cratchit, Martha, Peter, Belinda, Tiny Tim) with precise data references, or summarize any particular dinner in more detail.

# LlamaIndex (PropertyGraphIndex)

<b>Custom ontology support</b>	
<b>Approach</b>	Customizable scaffolding for RAG pipeline (unstructured data ingestion, indexing, retrieval), including GraphRAG
<b>Interface</b>	<ul style="list-style-type: none"><li>- Python API</li><li>- TypeScript API (limited, e.g. has no PropertyGraphIndex)</li></ul>
<b>Assumed use case</b>	<ul style="list-style-type: none"><li>- Customizable graph building</li><li>- Customizable graph querying (both keyword-based node search and semantics-based)</li></ul>

<b>Performance on the «Christmas Carol» text corpus (~29K English words)</b>	<b>LlamaIndex (PropertyGraphIndex)</b>	<b>Vanilla RAG</b>
<b>RAG configuration</b>	<b>chunk size - 1200 chars, overlap - 100 chars include source text chunks</b>	<b>chunk size - 1200 chars, overlap - 100 chars Top 3 similar chunks (no rerank)</b>
<b>Indexing cost (tokens)</b>	- Inference (input): ~48K tokens - Inference (output): ~195K tokens - Embedding: ~120K tokens	Embedding: ~44K tokens
<b>Indexing time (seconds)</b>	105 seconds	3 seconds
<b>Query cost (tokens)</b>	- inference (input): ~3500 tokens - inference (output): ~1600 tokens	- inference (input): ~3500 tokens - inference (output): ~700 tokens
<b>Query time (seconds)</b>	~16 seconds	~6.5 seconds

# Graphiti

<b>Custom ontology support</b>	
<b>Approach</b>	Temporally-aware (native support of evolving data, allowing point-in-time queries)
<b>Interface</b>	<ul style="list-style-type: none"><li>- Python API</li><li>- MCP server</li></ul>
<b>Assumed use case</b>	<ul style="list-style-type: none"><li>- Evolving data</li><li>- Preserving history of changes</li></ul>



```
Query: Who is Alice Smith current manager?  
Expected: Eva Martinez (CTO at MegaSoft since 2024)  
09:00:45 - INFO - HTTP Request: POST https://api.openai.com/v1/embeddings "HTTP/1.1 200 OK"  
09:00:46 - INFO - HTTP Request: POST https://api.openai.com/v1/embeddings "HTTP/1.1 200 OK"  
  
ALL FACTS (10 edges, including superseded) [1148ms]:  
[SUPERSEDED] Alice Smith's manager is Bob Wilson.  
Valid: 2020-01-15 -> 2020-02-01  
[SUPERSEDED] Alice Smith is reporting to Bob Wilson.  
Valid: 2021-03-01 -> 2021-09-15  
[SUPERSEDED] Alice Smith's new manager at MegaSoft is David Chen.  
Valid: 2023-02-01 -> 2023-03-15  
[SUPERSEDED] Alice Smith is a Staff Engineer at MegaSoft.  
Valid: 2023-02-01 -> 2023-03-15  
[CURRENT] Alice Smith leads a team of 5 developers  
Valid: 2022-06-01 -> current  
[SUPERSEDED] Alice Smith's starting salary is 70000 dollars per year.  
Valid: 2020-01-15 -> 2021-09-15  
[SUPERSEDED] Alice Smith is working on the mobile team.  
Valid: 2020-02-01 -> 2021-09-15  
[CURRENT] Alice Smith is leading the migration to Kubernetes.  
Valid: 2023-03-15 -> current  
[SUPERSEDED] Alice Smith now reports to Carol Johnson.  
Valid: 2021-09-15 -> 2022-06-01  
[CURRENT] Alice Smith is working on the cloud infrastructure team at MegaSoft.  
Valid: 2023-03-15 -> current  
  
CURRENT FACTS ONLY (10 edges, filtered) [424ms]:  
[CURRENT] Alice Smith leads a team of 5 developers  
Valid since: 2022-06-01  
[CURRENT] Alice Smith is leading the migration to Kubernetes.  
Valid since: 2023-03-15  
[CURRENT] Alice Smith is working on the cloud infrastructure team at MegaSoft.  
Valid since: 2023-03-15  
[CURRENT] Alice Smith left TechCorp.  
Valid since: 2023-02-01  
[CURRENT] Alice Smith has been promoted to Tech Lead at TechCorp  
Valid since: 2022-06-01  
[CURRENT] Alice Smith's salary at MegaSoft is 180000 dollars after promotion.  
Valid since: 2024-01-10  
[CURRENT] Alice Smith holds the position of Principal Engineer at MegaSoft.  
Valid since: 2024-06-01  
[CURRENT] Alice Smith was promoted to Principal Engineer at MegaSoft.  
Valid since: 2024-01-10  
[CURRENT] Alice Smith reports to David Chen at MegaSoft.  
Valid since: 2023-03-15  
[CURRENT] Alice Smith started advising the startup Cloudvine on weekends.  
Valid since: 2024-06-01
```


<b>Performance on the synthetic HR events toy dataset.</b>	<b>Graphiti</b>
<b>Episode ingestion time (seconds)</b>	<b>~15 seconds</b> (180 character JSON)
<b>Query cost (tokens)</b>	Low (does not involve inference, only entity/relationship search, except for cross-encoder rerank configuration)
<b>Query time (seconds)</b>	P95 300ms (but no generation involved!)

RAG Benchmark	License	Approach	Covers types of queries
<p>GraphRAG-Bench  <a href="https://github.com/GraphRAG-Bench/GraphRAG-Benchmark">https://github.com/GraphRAG-Bench/GraphRAG-Benchmark</a></p>	MIT	Quantitative metrics (+ they use the aggregate <i>accuracy</i> metric)	<ul style="list-style-type: none"> <li>- Fact Retrieval (Local query)</li> <li>- Complex Reasoning (Multi-hop reasoning)</li> <li>- Contextual Summarize (Global query)</li> <li>- Creative Generation</li> </ul>
<p>Microsoft Benchmark-QED  <a href="https://github.com/microsoft/benchmark-qed">https://github.com/microsoft/benchmark-qed</a></p>	MIT	Pairwise comparison	<ul style="list-style-type: none"> <li>- Local queries</li> <li>- Global queries</li> </ul>

# Other OSS GraphRAG solutions

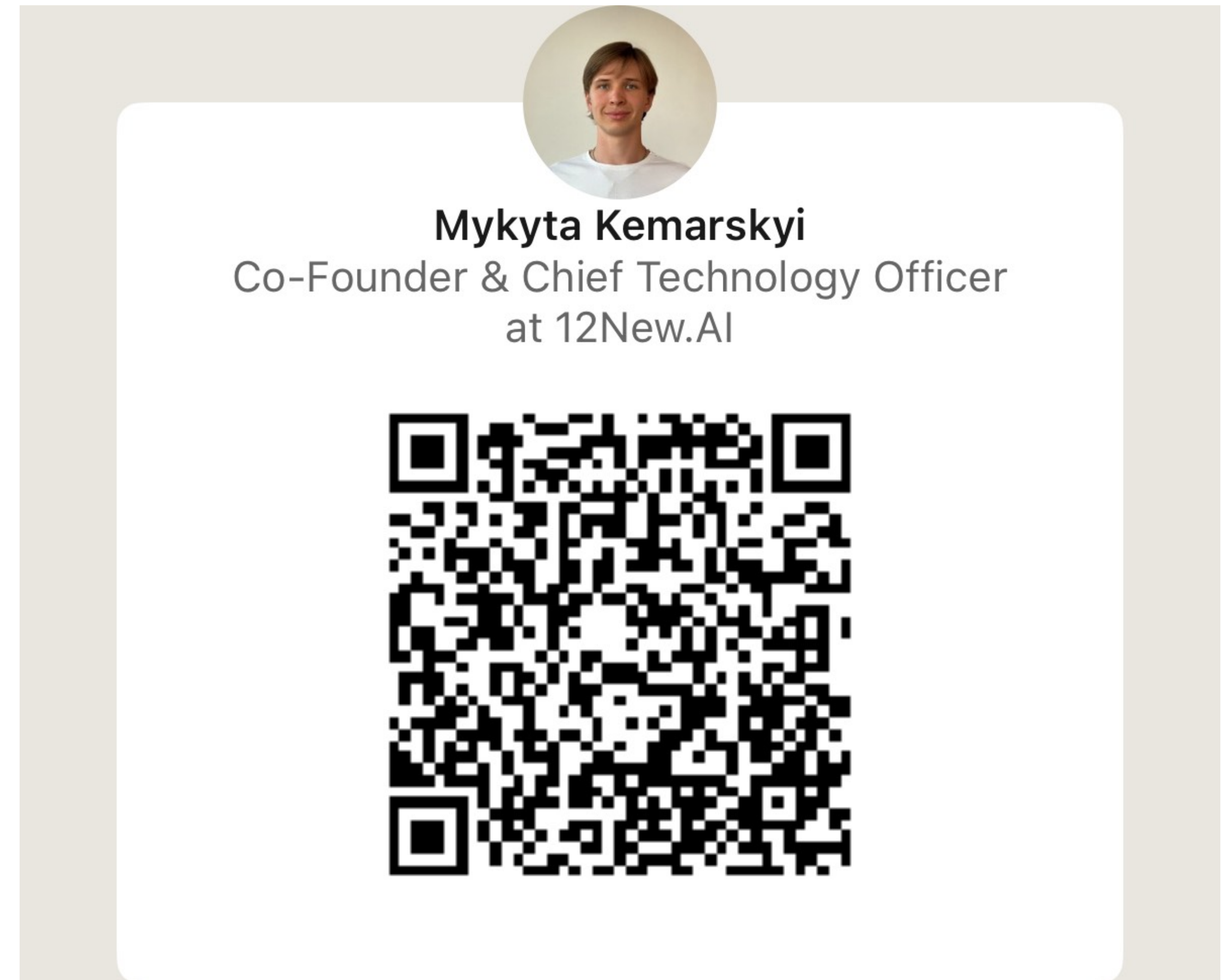
- OpenSPG/KAG (Apache 2.0 <https://github.com/OpenSPG/KAG>)
- LightRAG (MIT <https://github.com/HKUDS/LightRAG>)
- HippoRAG2 (MIT <https://github.com/OSU-NLP-Group/HippoRAG>)

# Conclusions

- Vanilla RAG is often insufficient
- GraphRAG extend what's possible - but comes at cost
-  Lack of widely adopted benchmarks / leaderboards to fairly compare RAG solutions
- **!** Be sure to evaluate each solution on your own data, with your own questions.

# Thank you for attention!

**LinkedIn:** @kemarskyi



A business card for Mykyta Kemarskyi, Co-Founder & Chief Technology Officer at 12New.AI. The card features a circular profile picture of Mykyta at the top center, followed by his name and title in a sans-serif font. Below the text is a large QR code. The card is set against a light beige background with rounded corners.

**Mykyta Kemarskyi**  
Co-Founder & Chief Technology Officer  
at 12New.AI

