Search

[Write](https://medium.com/new-story?source=---two_column_layout_nav----------------------------------)

Rishabh Sahai

Get unlimited access to the best of Medium for less than $1/week.

[Become a member](https://medium.com/plans?source=upgrade_membership---post_top_nav_upsell----------------------------------)

Top highlight

**Construct Knowledge Graphs From Unstructured Text**

[[](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)

[[Neo4j Developer Blog](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)

[Noah Mayerhofer](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)

·

Follow

Published in

[Neo4j Developer Blog](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)

·

8 min read

·

Sep 21, 2023

165

4

*This is the fifth blog post of Neo4j’s NaLLM project. We started this project to explore, develop, and showcase practical uses of these LLMs in conjunction with Neo4j. As part of this project, we have constructed and publicly displayed demonstrations in a*[*GitHub repository*](https://github.com/neo4j/NaLLM)*, providing an open space for our community to observe, learn, and contribute. Additionally, we have been writing about our findings in a blog post. You can read the previous four blog posts here:*

* [*Harnessing LLMs with Neo4j*](https://medium.com/neo4j/harnessing-large-language-models-with-neo4j-306ccbdd2867)
* [*Fine-tuning vs. Retrieval-augmented generation*](https://medium.com/neo4j/knowledge-graphs-llms-fine-tuning-vs-retrieval-augmented-generation-30e875d63a35)
* [*Multi-hop Question Answering*](https://medium.com/neo4j/knowledge-graphs-llms-multi-hop-question-answering-322113f53f51)
* [*Knowledge Graphs & LLMs: Real-Time Graph Analytics*](https://medium.com/neo4j/knowledge-graphs-llms-real-time-graph-analytics-89b392eaaa95)

This blog post will explore a use case we investigated during our project: extracting information from unstructured data. Organizations have long faced challenges in extracting meaningful insights from unstructured data. Such data encompasses textual content, images, audio, and other non-tabular formats, holding immense potential yet often remaining difficult to use due to its inherent complexity. Our primary focus in this post will be to extract information from unstructured text by converting it into nodes and relationships.

A head made of wires and wires

Description automatically generated

Illustration from [Imagine.art](http://imagine.art/" \t "_blank) of “extracting information from unstructured data”

Recent years have witnessed significant advancements in natural language processing techniques, revolutionizing the transformation of unstructured data into valuable knowledge. With the emergence of powerful language models like OpenAI’s GPT models and leveraging the power of machine learning, the process of converting unstructured text data into structured representations has become more accessible and efficient.

One such representation is **knowledge graphs,** which offer a robust framework for representing complex relationships and connections among various entities. They provide a structured representation of the data, enabling intuitive querying and exploration of the information contained within. This structured nature allows for advanced semantic analysis, reasoning, and inference, facilitating more accurate and comprehensive decision-making processes.

A screenshot of a diagram

Description automatically generated

Example of Knowledge Extraction Pipeline

We will explore how Large Language Models (**LLMs**) have simplified the conversion of unstructured data into knowledge graphs, using an approach that utilizes the language skills of LLMs to perform nearly all parts of the process. The process can be divided into three steps:

1. Extracting nodes and edges
2. Entity disambiguation
3. Importing into Neo4j

Let’s walk through each of these steps:

**1. Extracting nodes and relationships:** To tackle this problem, we take the simplest possible approach, where we pass the input data to the LLM and let it decide which nodes and relationships to extract. We ask the LLM to return the extracted entities in a specific format, including a name, a type, and properties. This allows us to extract nodes and edges from the input text.

However, LLMs have a limitation known as the ***context window*** (between 4 and 16,000 tokens for most LLMs), which can be easily overwhelmed by larger inputs, hindering the processing of such data. To overcome this limitation, we employ a strategy of dividing the input text into smaller, more manageable chunks that fit within the context window.

Determining the optimal splitting points for the text is a challenge of its own. To keep things simple, we have chosen to divide the text into chunks of maximum size, maximizing the utilization of the context window per chunk. Additionally, we introduce some overlap from the previous chunk to account for cases where a sentence or description spans across multiple chunks. This approach allows us to extract nodes and edges from each chunk, representing the information contained within it.

To maintain consistency in the labeling of different types of entities across chunks, we provide the LLM with a list of node types that were extracted in the previous chunks. Those start forming the extracted “schema.” We have observed that this approach enhances the uniformity of the final labels. For example, instead of the LLM generating separate types for “Company” and “Gaming Company,” it consolidates all types of companies under a “Company” label.

One ***notable hurdle*** in our approach is the problem of duplicate entities. Since each chunk is processed semi-independently, information about the same entity found in different chunks will create duplicates when we combine the results. Naturally, this issue brings us to our next step.

**2. Entity disambiguation:** We now have a set of entities. To address the issue of duplication, we employ LLMs once again. First, we organize the entities into sets based on their type. Subsequently, we provide each set to the LLM, enabling it to merge duplicate entities while simultaneously consolidating their properties. We use LLMs for this since we don’t know what name each entity has been given. For example, the initial extraction could have ended up with two nodes: (Alice {name: “Alice Henderson”}) and (Alice Henderson {age: 25}). These reference the same entity and should be merged to a single node with both the name and age property. We use LLMs to accomplish this since it’s great at quickly understanding which nodes actually reference the same entity.

By iteratively performing this procedure for all entity groups, we obtain a structured data set that is ready for further processing.

**3. Importing the data into Neo4j:** In the final step of the process, we focus on importing the results we got from the LLM into a Neo4j database. This requires a format that Neo4j can understand. To accomplish this, we parse the generated text from the LLM and transform it into separate CSV files, corresponding to the various node and relationship types. These CSV files are subsequently mapped to a format compatible with the [Neo4j Data Importer tool](https://data-importer.neo4j.io/connection/connect). Through this conversion, we gain the advantage of previewing the data before initiating the import process into a Neo4j database, harnessing the capabilities offered by the Neo4j Importer tool.

A diagram of a computer program

Description automatically generated

Overview of the application

Putting this all together, we have created an application consisting of three parts: a UI to input a file, a controller that executes the previously explained process, and an LLM that the controller talks to. This demo application can be found [here](https://nallm-experiments.ew.r.appspot.com/), and the source code can be found on [GitHub](https://github.com/neo4j/NaLLM).

We also created a version of this pipeline that works essentially in the same way but with the option to include a schema. This schema works like a filter where the user can restrict which types of nodes and relationships and which properties the LLM should include in its result.

**[NaLLM Graph Construction Demo](https://nallm-experiments.ew.r.appspot.com/?source=post_page-----877be33300a2--------------------------------" \t "_blank)**

[nallm-experiments.ew.r.appspot.com](https://nallm-experiments.ew.r.appspot.com/?source=post_page-----877be33300a2--------------------------------" \t "_blank)

If you are interested in learning more about generative AI and knowledge graphs, I would suggest taking a look at [Neo4j’s page about generative AI](https://neo4j.com/generativeai/).

**Demonstration**

I tested the application by giving it the Wikipedia page for the [James Bond franchise](https://en.wikipedia.org/wiki/James_Bond) and inspected the generated knowledge graph.

A network of colorful dots

Description automatically generated with medium confidence

Example of the resulting graph

The provided graph subset showcases the generated graph, which, in my opinion, provides a reasonably accurate depiction of the Wikipedia article. The graph primarily consists of nodes representing books and individuals associated with those books, such as authors and publishers.

However, there are a few issues with the graph. For instance, Ian Fleming is labeled as a publisher rather than an author for most of the books he wrote. This discrepancy may be attributed to the difficulty the language model had in comprehending that particular aspect of the Wikipedia article.

Another problem is the inclusion of relationships between book nodes and the directors of films with the same titles, instead of creating separate nodes for the movies.

Finally, It’s worth noting that the LLM appears to be quite literal in its interpretation of relationships, as evidenced by using the relationship type “used” to connect the James Bond character with the cars he drove. This literal approach may stem from the article’s usage of the verb “used” rather than “drove.”

A full video of the demonstration can be found here:

Demo of KG Construction

**Problems**

For a demonstration, this approach worked fairly well, and we think it shows that it’s possible to use LLMs to create knowledge graphs. However, we acknowledge certain issues need to be addressed within this approach:

* **Unpredictable output:** This is inherent to the nature of LLMs. We do not know how an LLM will format its results. Even if we ask it to output in a specific format, it might not obey. This might cause problems when trying to parse what it generates. We saw one instance of this while chunking the data: Most of the time, the LLM generated a simple list of nodes and edges, but sometimes the LLM would number the list. Tools to work around this are starting to be released, such as [Guardrails](https://shreyar.github.io/guardrails/) and [OpenAIs Function API](https://openai.com/blog/function-calling-and-other-api-updates" \t "_blank). It’s still early in the world of LLM tooling, so we anticipate that this will not be a problem for long.
* **Speed**: This approach is slow and often takes several minutes for just a single reasonably large web page. There might be a fundamentally different approach that can make the extraction go faster.
* **Lack of accountability**: There is no way of knowing why the LLM decided to extract some information from the source documents or if the information even exists in the source. The data quality of the resulting knowledge graph is, therefore, much lower than the graph created by processes not leveraging LLMs.

**Summary**

This blog post explored a use case of Large Language Models with Neo4j to extract insights from unstructured data by converting it into a structured representation in the form of a knowledge graph.

We discussed a three-step approach focusing on extracting nodes and relationships, entity disambiguation, and importing the data into Neo4j. By utilizing LLMs, anyone can automate the extraction process and efficiently process large amounts of unstructured data.

*However, there are challenges* to address, including unpredictable output formatting, speed limitations, and the lack of accountability. Despite these issues, the combined power of LLMs and Neo4j offers a promising solution for unlocking the hidden value in unstructured data, even for non-technical users.

We hope that this blog post has provided you with valuable insights and practical knowledge to leverage LLMs and Neo4j in extracting knowledge from unstructured data. The code for this project can be found on [GitHub](https://github.com/neo4j/NaLLM), and if you want to know more about AI and Neo4j, I suggest taking a look at [Neo4j’s page about generative AI](https://neo4j.com/generativeai/).

**[Generative AI](https://neo4j.com/generativeai/?source=post_page-----877be33300a2--------------------------------" \t "_blank)**

[POWER LLMS WITH ENTERPRISE KNOWLEDGE By capturing the context of your enterprise data, knowledge graphs enable AI…](https://neo4j.com/generativeai/?source=post_page-----877be33300a2--------------------------------" \t "_blank)

[neo4j.com](https://neo4j.com/generativeai/?source=post_page-----877be33300a2--------------------------------" \t "_blank)

[AI](https://medium.com/tag/ai?source=post_page-----877be33300a2---------------ai-----------------)

[Generative Ai Use Cases](https://medium.com/tag/generative-ai-use-cases?source=post_page-----877be33300a2---------------generative_ai_use_cases-----------------)

[Neo4j](https://medium.com/tag/neo4j?source=post_page-----877be33300a2---------------neo4j-----------------)

[Knowledge Graph](https://medium.com/tag/knowledge-graph?source=post_page-----877be33300a2---------------knowledge_graph-----------------)

[Unstructured Data](https://medium.com/tag/unstructured-data?source=post_page-----877be33300a2---------------unstructured_data-----------------)

165

4

[[](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)

[[](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)

[Written by Noah Mayerhofer](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)

[51 Followers](https://medium.com/@noah.mayerhofer/followers?source=post_page-----877be33300a2--------------------------------)

·Writer for

[Neo4j Developer Blog](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)

Follow

More from Noah Mayerhofer and Neo4j Developer Blog

[[](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

in

[Neo4j Developer Blog](https://medium.com/neo4j?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

**[Building A Graph+LLM Powered RAG Application from PDF Documents](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)**

[A Step-by-step Walkthrough with GenAI-Stack and OpenAI](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[6 min read·Jan 15](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

--

[2](https://medium.com/neo4j/building-a-graph-llm-powered-rag-application-from-pdf-documents-24225a5baf01?responsesOpen=true&sortBy=REVERSE_CHRON&source=author_recirc-----877be33300a2----0---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

in

[Neo4j Developer Blog](https://medium.com/neo4j?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

**[Why Vector Search Didn’t Work for Your RAG Solution?](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)**

[Exploring The Shortcomings of Text Embedding Retrieval for LLM Generation](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[11 min read·Dec 19, 2023](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

--

[1](https://medium.com/neo4j/why-vector-search-didnt-work-for-your-rag-solution-d95119a8c42e?responsesOpen=true&sortBy=REVERSE_CHRON&source=author_recirc-----877be33300a2----1---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[Alex Gilmore](https://medium.com/@a-gilmore?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/@a-gilmore?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[Alex Gilmore](https://medium.com/@a-gilmore?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

in

[Neo4j Developer Blog](https://medium.com/neo4j?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

**[YouTube Transcripts Into Knowledge Graphs for RAG Applications](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)**

[A walkthrough covering how to scrape YouTube transcripts into a Neo4j graph to be used in RAG applications — includes Github repo](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[8 min read·Jan 5](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

--

[2](https://medium.com/neo4j/youtube-transcripts-knowledge-graphs-for-rag-applications-2cc790543d4b?responsesOpen=true&sortBy=REVERSE_CHRON&source=author_recirc-----877be33300a2----2---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[[Tomaz Bratanic](https://medium.com/@bratanic-tomaz?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)](https://medium.com/@bratanic-tomaz?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[Tomaz Bratanic](https://medium.com/@bratanic-tomaz?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

in

[Neo4j Developer Blog](https://medium.com/neo4j?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

**[Implementing Advanced Retrieval RAG Strategies With Neo4j](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)**

[Go beyond typical RAG strategies](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[6 min read·Nov 17, 2023](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

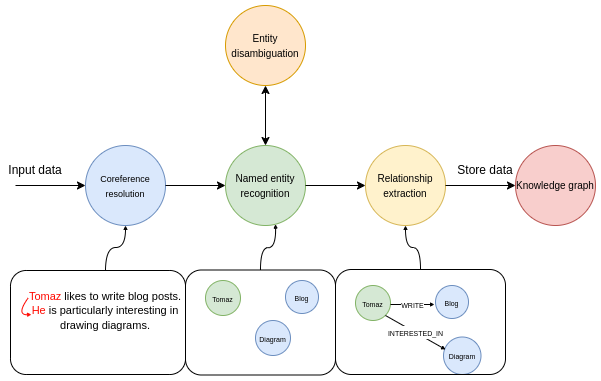
--

[1](https://medium.com/neo4j/implementing-advanced-retrieval-rag-strategies-with-neo4j-c968a002c513?responsesOpen=true&sortBy=REVERSE_CHRON&source=author_recirc-----877be33300a2----3---------------------2b2b704e_af32_4759_ba4c_b142bb307fa1-------)

[See all from Noah Mayerhofer](https://medium.com/@noah.mayerhofer?source=post_page-----877be33300a2--------------------------------)

[See all from Neo4j Developer Blog](https://medium.com/neo4j?source=post_page-----877be33300a2--------------------------------)

Recommended from Medium

[[](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[Tomaz Bratanic](https://medium.com/@bratanic-tomaz?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@bratanic-tomaz?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[Tomaz Bratanic](https://medium.com/@bratanic-tomaz?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

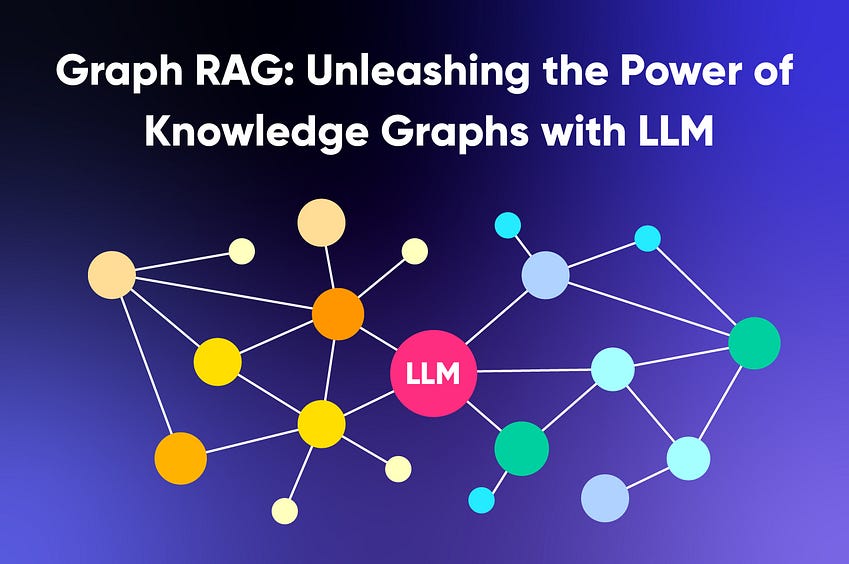
**[Constructing knowledge graphs from text using OpenAI functions](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[Seamlessy implement information extraction pipeline with LangChain and Neo4j](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[11 min read·Oct 20, 2023](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[10](https://medium.com/@bratanic-tomaz/constructing-knowledge-graphs-from-text-using-openai-functions-096a6d010c17?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[NebulaGraph Database](https://medium.com/@nebulagraph?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@nebulagraph?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[NebulaGraph Database](https://medium.com/@nebulagraph?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

**[Graph RAG: Unleashing the Power of Knowledge Graphs with LLM](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[In the era of information overload, sifting through vast amounts of data to provide accurate search results in an engaging and…](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[6 min read·Sep 8, 2023](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[1](https://medium.com/@nebulagraph/graph-rag-the-new-llm-stack-with-knowledge-graphs-e1e902c504ed?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

Lists

[[A person in a long dress

Description automatically generated](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)

[[A close up of a window

Description automatically generated](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)

[[A green and purple squares with white arrows

Description automatically generated](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)

[Generative AI Recommended Reading](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)

[52 stories·668 saves](https://tomsmith585.medium.com/list/generative-ai-recommended-reading-508b0743c247?source=read_next_recirc-----877be33300a2--------------------------------)

[[A blurry image of a kitchen

Description automatically generated](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)

[[A close up of a bird

Description automatically generated](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)

[[A cartoon of a child and a robot

Description automatically generated](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)

[What is ChatGPT?](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)

[9 stories·283 saves](https://medium.com/@MediumForTeams/list/what-is-chatgpt-7a5756752f49?source=read_next_recirc-----877be33300a2--------------------------------)

[[Image by vectorjuice on FreePik](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)

[[A toy robot with arms and legs

Description automatically generated](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)

[[A screenshot of a computer

Description automatically generated](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)

[The New Chatbots: ChatGPT, Bard, and Beyond](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)

[12 stories·286 saves](https://medium.com/@MediumStaff/list/the-new-chatbots-chatgpt-bard-and-beyond-5969c7449b7f?source=read_next_recirc-----877be33300a2--------------------------------)

[[A screenshot of a website

Description automatically generated](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)

[[A screenshot of a computer

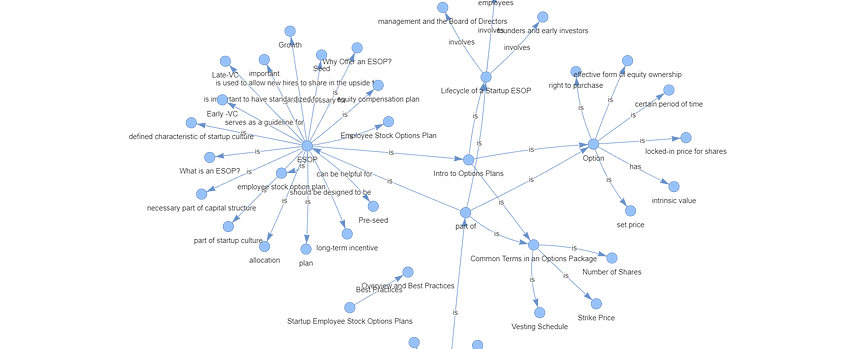
Description automatically generated](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)

[[A blue planet with light coming out of it

Description automatically generated](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)

[Natural Language Processing](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)

[1137 stories·613 saves](https://medium.com/@AMGAS14/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----877be33300a2--------------------------------)

[[](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[Plaban Nayak](https://medium.com/@nayakpplaban?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@nayakpplaban?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[Plaban Nayak](https://medium.com/@nayakpplaban?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

in

[AI Planet](https://medium.com/dphi-tech?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

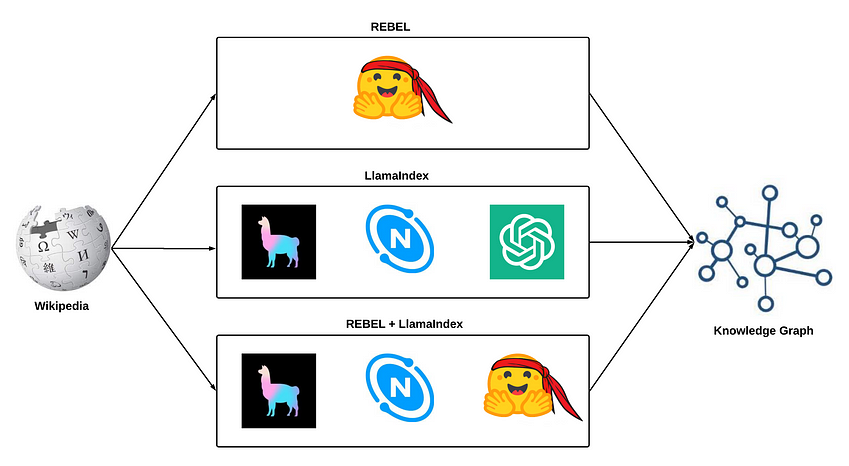
**[Implement RAG with Knowledge Graph and Llama-Index](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[Hallucination is a common problem when working with large language models (LLMs). LLMs generate fluent and coherent text but often generate…](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[25 min read·Dec 3, 2023](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[7](https://medium.com/dphi-tech/implement-rag-with-knowledge-graph-and-llama-index-6a3370e93cdd?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----0---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[Saurav Joshi](https://medium.com/@sauravjoshi23?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@sauravjoshi23?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[Saurav Joshi](https://medium.com/@sauravjoshi23?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

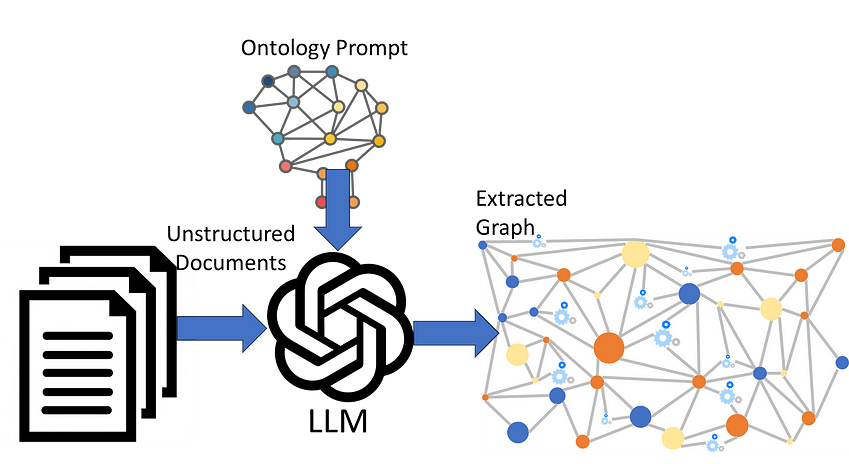
**[Building Knowledge Graphs: REBEL, LlamaIndex, and REBEL + LlamaIndex](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[Exploring building knowledge graphs using LlamaIndex and NebulaGraph](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[10 min read·Oct 4, 2023](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[5](https://medium.com/@sauravjoshi23/building-knowledge-graphs-rebel-llamaindex-and-rebel-llamaindex-8769cf800115?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----1---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[Peter Lawrence, answering users' data questions](https://medium.com/@peter.lawrence_47665?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@peter.lawrence_47665?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[Peter Lawrence, answering users' data questions](https://medium.com/@peter.lawrence_47665?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

in

[GoPenAI](https://medium.com/gopenai?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

**[LLM Ontology-prompting for Knowledge Graph Extraction](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[Prompting an LLM with an ontology to drive Knowledge Graph extraction from unstructured documents](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[5 min read·Jun 27, 2023](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[9](https://medium.com/gopenai/llm-ontology-prompting-for-knowledge-graph-extraction-efdcdd0db3a1?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----2---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)](https://medium.com/@yu-joshua?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[Fanghua (Joshua) Yu](https://medium.com/@yu-joshua?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

**[Why Vectors Should Be Stored Together with Knowledge Graph?](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)**

[An architecture review of Neo4j as both vector and graph store](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[8 min read·Oct 11, 2023](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

--

[1](https://medium.com/@yu-joshua/why-vectors-should-be-stored-together-with-knowledge-graph-41e4280e2f02?responsesOpen=true&sortBy=REVERSE_CHRON&source=read_next_recirc-----877be33300a2----3---------------------3d55a011_dc5e_4dab_852b_524fbeabfe2c-------)

[See more recommendations](https://medium.com/?source=post_page-----877be33300a2--------------------------------)

[Help](https://help.medium.com/hc/en-us?source=post_page-----877be33300a2--------------------------------)

[Status](https://medium.statuspage.io/?source=post_page-----877be33300a2--------------------------------)

[About](https://medium.com/about?autoplay=1&source=post_page-----877be33300a2--------------------------------)

[Careers](https://medium.com/jobs-at-medium/work-at-medium-959d1a85284e?source=post_page-----877be33300a2--------------------------------)

[Blog](https://blog.medium.com/?source=post_page-----877be33300a2--------------------------------)

[Privacy](https://policy.medium.com/medium-privacy-policy-f03bf92035c9?source=post_page-----877be33300a2--------------------------------)

[Terms](https://policy.medium.com/medium-terms-of-service-9db0094a1e0f?source=post_page-----877be33300a2--------------------------------)

[Text to speech](https://speechify.com/medium?source=post_page-----877be33300a2--------------------------------)

[Teams](https://medium.com/business?source=post_page-----877be33300a2--------------------------------)