

[Open in app](#)[≡ Medium](#)

E

Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Data Science in You... · [Follow publication](#)

GraphRAG using LangChain

codes explained with example, Generative AI

4 min read · Jul 17, 2024



Mehul Gupta

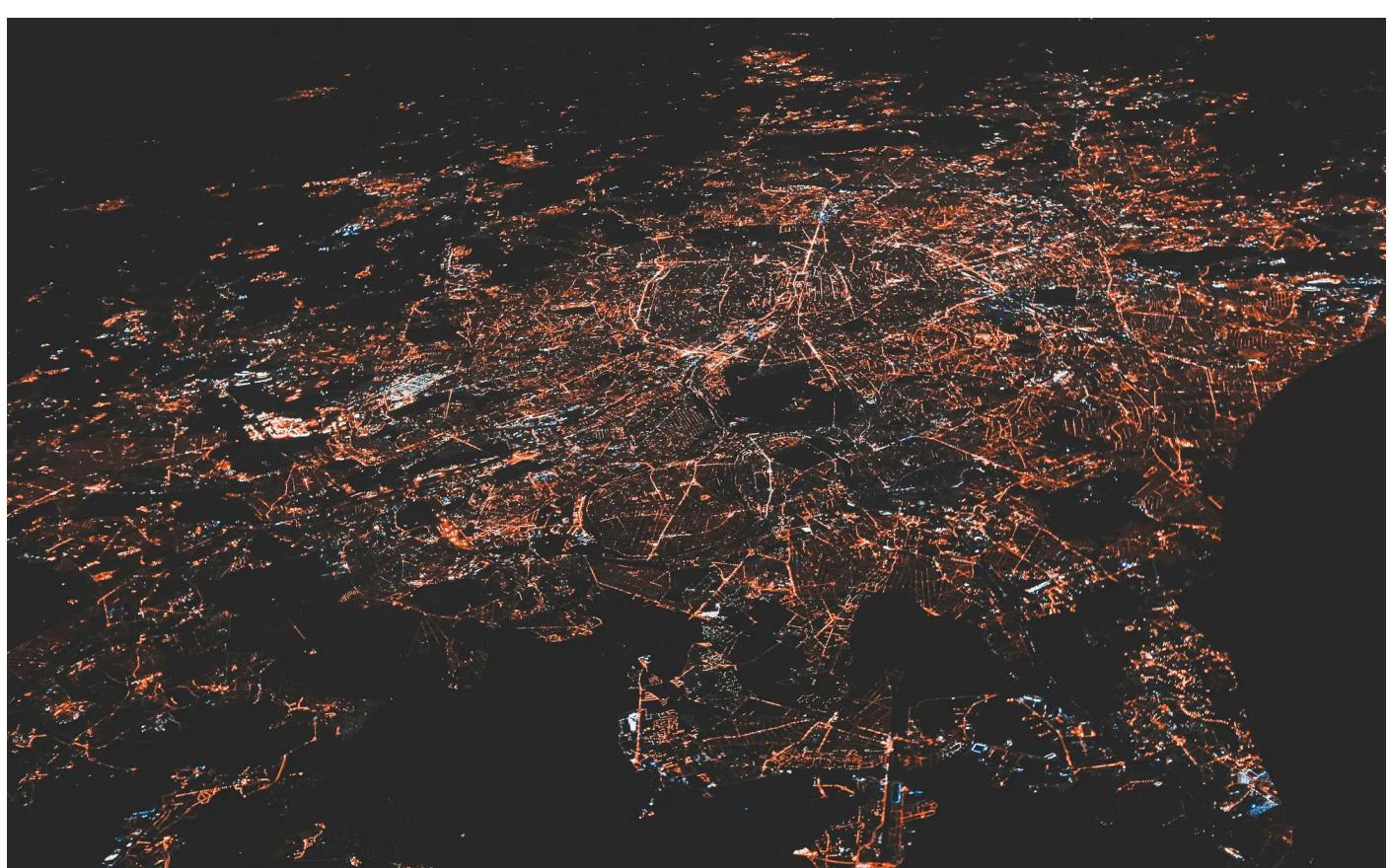
[Follow](#)[Listen](#)[Share](#)[More](#)

Photo by [Nastya Dulhiier](#) on [Unsplash](#)

GraphRAG has been the talk of the town since Microsoft release their GraphRAG git repo which became an instant hit on git.

Graph RAG is an advanced version of standard RAG that uses Knowledge Bases instead of vector similarity and vector DBs for retrieval from external documents, making the retrieval more comprehensive and wholesome.

I've already covered GraphRAG in detail in the below posts:

[What is GraphRAG?](#)

[How GraphRAG works?](#)

Graph RAG Crash course is live now:

GraphRAG Crash Course

RAG is amongst the most important concepts in Generative AI that help you to talk to your external files like CSV...

datasciencepocket.gumroad.com

In this post, I will run through a basic example of how to set GraphRAG using LangChain and use it to improve your RAG systems (using any LLM model or API)

My debut book: LangChain in your Pocket is out now !!

LangChain in your Pocket: Beginner's Guide to Building Generative AI Applications using LLMs

Amazon.com: LangChain in your Pocket: Beginner's Guide to Building Generative AI Applications using LLMs eBook : Gupta...

www.amazon.com

We will be discussing two approaches

1. LLMGraphTransformer

1. You 1st need to pip install a few essential libraries

```
pip install --upgrade --quiet json-repair networkx langchain-core langchain

#versions used
langchain==0.2.8
langchain-community==0.2.7
langchain-core==0.2.19
langchain-experimental==0.0.62
langchain-google-vertexai==1.0.3
```

Note: You can skip Google VertexAI and use any other LLM as well

2. Import required functions. Initialize your LLM object & reference text. Use any SOTA LLM for best results as Knowledge Graph creation is a complicated task.

```
import os
from langchain_experimental.graph_transformers import LLMGraphTransformer
```

```

from langchain_google_vertexai import VertexAI
import networkx as nx
from langchain.chains import GraphQACChain
from langchain_core.documents import Document
from langchain_community.graphs.networkx_graph import NetworkxEntityGraph

llm = VertexAI(max_output_tokens=4000, model_name='text-bison-32k')

text = """
Marie Curie, born in 1867, was a Polish and naturalised-French physicist and
She was the first woman to win a Nobel Prize, the first person to win a Nobel
Her husband, Pierre Curie, was a co-winner of her first Nobel Prize, making
She was, in 1906, the first woman to become a professor at the University of
"""

```

3. Next, we need to load this text as **GraphDocuments** and create a **GraphTransformer** object using the LLM-loaded

```

documents = [Document(page_content=text)]
llm_transformer = LLMGraphTransformer(llm=llm)
graph_documents = llm_transformer.convert_to_graph_documents(documents)

```

4. Its time to create the **Knowledge Graph**. For this, you better provide a list of entities and relationships you wish to extract else LLM might identify everything as an entity or relationship

```

llm_transformer_filtered = LLMGraphTransformer(
    llm=llm,
    allowed_nodes=["Person", "Country", "Organization"],
    allowed_relationships=["NATIONALITY", "LOCATED_IN", "WORKED_AT", "SPOUSE"]
)
graph_documents_filtered = llm_transformer_filtered.convert_to_graph_documents()

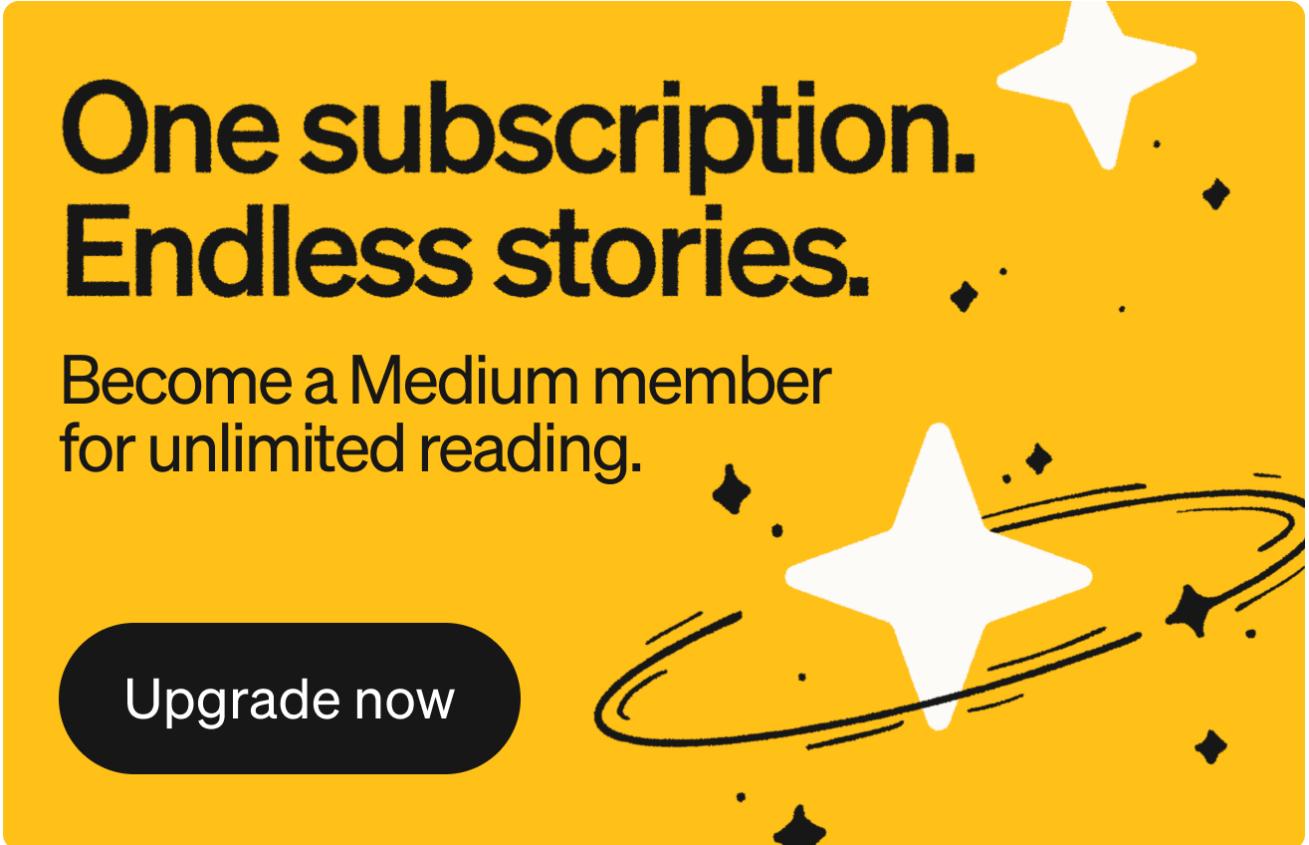
```

As you must have guessed, the above snippet creates

Node = “Person”, “Country”, “Organization”

Relation = [“NATIONALITY”, “LOCATED_IN”, “WORKED_AT”, “SPOUSE”]

Note: Any other potential node or relation would be discarded. If you aren't sure, you can just pass the LLM object and let the LLM decide



5. We now need to create a **Networkx graph** and add the above-identified nodes and edges to this graph

```
graph = NetworkxEntityGraph()

# Add nodes to the graph
for node in graph_documents_filtered[0].nodes:
    graph.add_node(node.id)

# Add edges to the graph
for edge in graph_documents_filtered[0].relationships:
    graph._graph.add_edge(
        edge.source.id,
        edge.target.id,
        relation=edge.type,
    )
```

6. Let's create a **GraphQACChain** now that will help us to interact with the

Knowledge Base

```
chain = GraphQACChain.from_llm(  
    llm=llm,  
    graph=graph,  
    verbose=True  
)
```

7. Call the chain object with your query

```
question = """Who is Marie Curie?"""  
chain.run(question)
```

Output

```
> Entering new GraphQACChain chain...  
Entities Extracted:  
Marie Curie  
Full Context:  
Marie Curie NATIONALITY Poland  
Marie Curie NATIONALITY France  
Marie Curie WORKED_AT University of Paris  
Marie Curie SPOUSE Pierre Curie  
  
> Finished chain.  
'Marie Curie was a Polish and French physicist and chemist who conducted pioneering research on radioactivity. She was the first woman to win a Nobel Prize, the first person and only woman to win the Nobel Prize twice, and the only person to win the Nobel Prize in two different scientific fields.'
```

We can even use GraphIndexCreator for implementing GraphRAG

2. GraphIndexCreator

Another approach is to use GraphIndexCreator in LangChain which is very similar to the above approach

```
from langchain.indexes import GraphIndexCreator  
from langchain.chains import GraphQACChain  
  
index_creator = GraphIndexCreator(llm=llm)  
  
with open("/home/cdsu/sample.txt") as f:  
    all_text = f.read()  
  
text = "\n".join(all_text.split("\n\n"))
```

```
graph = index_creator.from_text(text)

chain = GraphQACChain.from_llm(llm, graph=graph, verbose=True)
chain.run("What did Pierre Curie won?")
```

As must have understood

It first creates a GraphIndexCreator using an LLM

Reads text from a .txt file

Creates graph using the index creator

Runs the GraphQA chain on the graph similar to above approach

Output

```
> Entering new GraphQACChain chain...

Entities Extracted:
Pierre Curie
Full Context:
Pierre Curie was a co-winner of Marie Curie's first Nobel Prize

> Finished chain.
' Pierre Curie won the Nobel Prize in Physics in 1903, together with his wife'
```

As I experimented, the LLMGraphTransformer approach looked better compared to GraphIndexCreator in terms of response but yes, both are quite easy to implement. Do remember, that these examples use very small datasets. If you are using a big dataset with paid APIs, be cautious as Knowledge Graph creation can lead a number of hits, costing you dearly.

With this, it's a wrap. See you soon!

Artificial Intelligence

Machine Learning

Data Science

Programming

ChatGPT





Follow

Published in Data Science in Your Pocket

8.1K followers · Last published 6 hours ago

YouTube : <https://www.youtube.com/@datascienceninjapocket>



Follow



Written by Mehul Gupta 

19.8K followers · 38 following

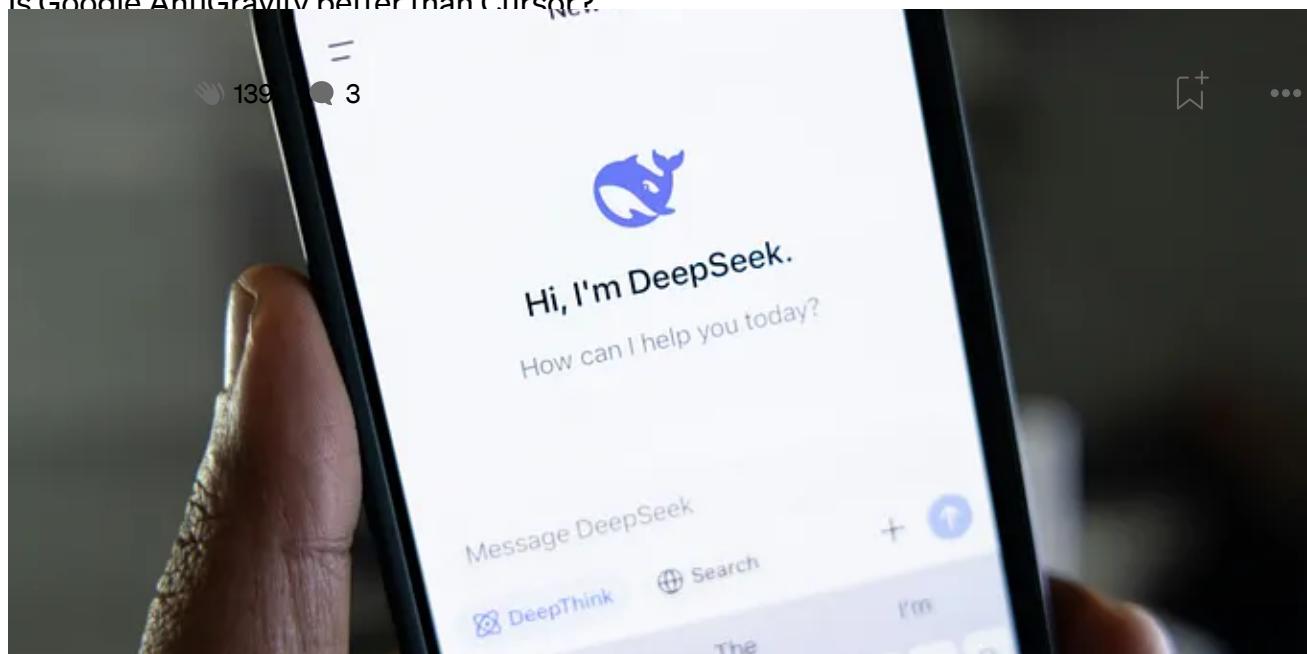
Email : datasciencepocket@gmail.com

More from Mehul Gupta and Data Science in Your Pocket

[AI] In Data Science in Your Pocket by Mehul Gupta 

Cursor vs Google AntiGravity : Best AI IDE for Vibe Coders?

Is Google AntiGravity better than Cursor?

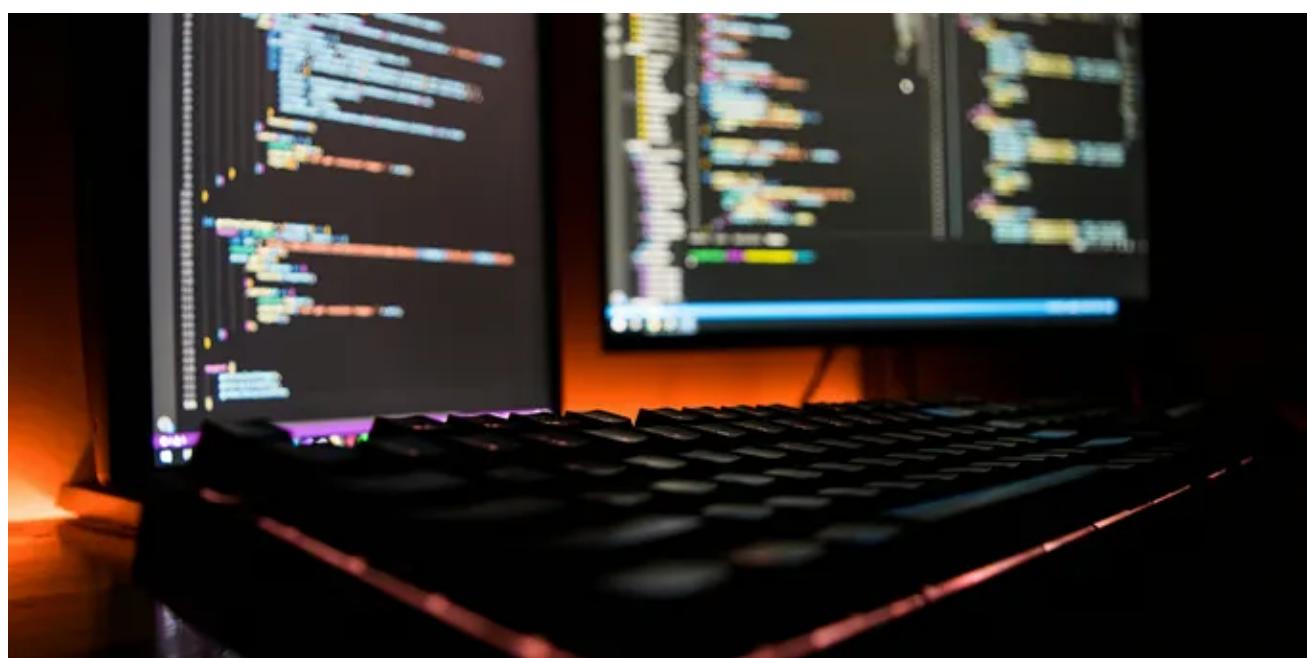


[AI] In Data Science in Your Pocket by Mehul Gupta 

What is DeepSeek mHC ? DeepSeek's new paper changes LLMs forever

mHC: Manifold-Constrained Hyper-Connections

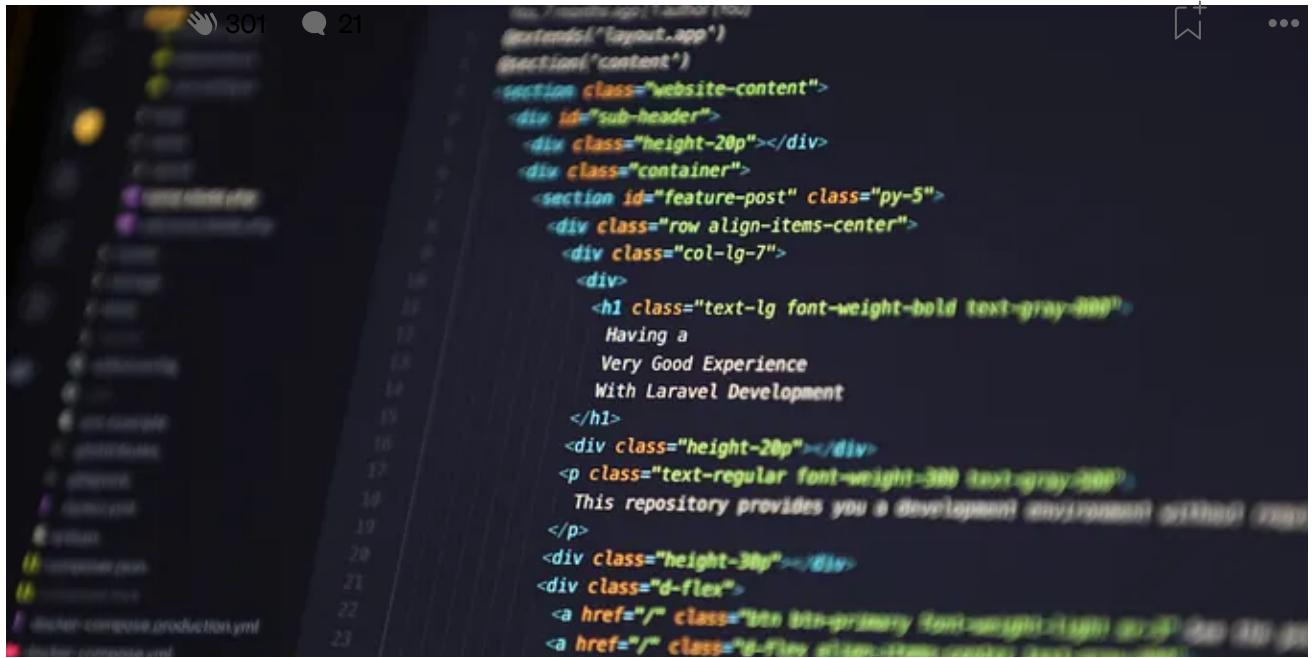
Jan 2 139 292 2



 In Data Science in Your Pocket by Mehul Gupta 

TOON : Bye Bye JSON for LLMs

TOON is a new datatype, more efficient than json



A screenshot of a code editor showing a file named 'layout.app'. The code is written in TOON, a new datatype for LLMs. It includes HTML-like tags such as <section>, <div>, <h1>, <p>, and <a>. The code describes a layout with a sub-header, a container section, and a feature post section. The text within the layout discusses having a very good experience with Laravel development and provides links.

```
<extends('layout.app')>
<section('content')>
  <section class="website-content">
    <div id="sub-header">
      <div class="height-20px"></div>
    <div class="container">
      <section id="feature-post" class="py-5">
        <div class="row align-items-center">
          <div class="col-lg-7">
            <div>
              <h1 class="text-lg font-weight-bold text-gray-800">
                Having a
                Very Good Experience
                With Laravel Development
              </h1>
              <div class="height-20px"></div>
              <p class="text-regular font-weight-300 text-gray-600">
                This repository provides you a development environment without any
              </p>
              <div class="height-30px"></div>
              <div class="d-flex">
                <a href="/" class="btn btn-primary font-weight-light px-3 py-2 w-50">Get Started</a>
                <a href="/" class="bg-gray-300 px-3 py-2 w-50">View Repository</a>
              </div>
            </div>
          </div>
        </div>
      </section>
    </div>
  </section>
</div>
```

 In Data Science in Your Pocket by Mehul Gupta 

GLM 4.7 : Best Open-Sourced LLM is here !!

How to use GLM 4.7 for free?

Dec 23, 2025  170  2



See all from Mehul Gupta

See all from Data Science in Your Pocket

Recommended from Medium

RAG Chunking Techniques for Tabular Data: 10 Powerful Strategies



In Towards AI by Tarun Singh

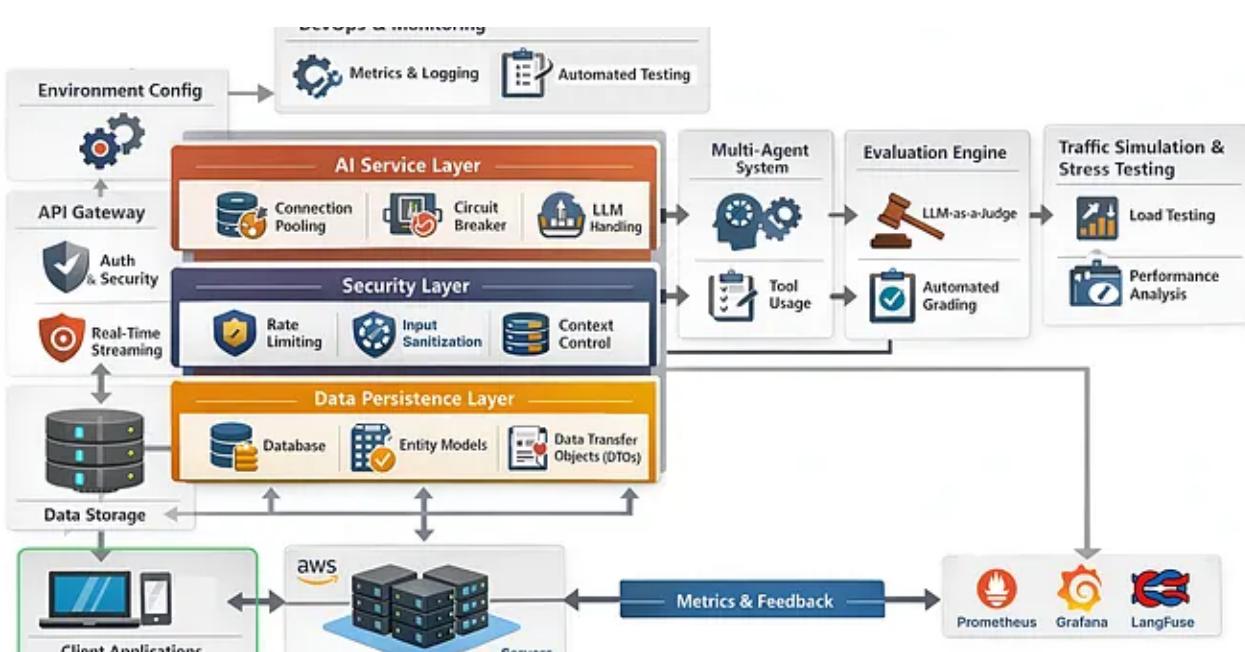
RAG Chunking Techniques for Tabular Data: 10 Powerful Strategies

Level Up Your RAG Apps: Table Edition

Jul 28, 2025 13 2



...



In Level Up Coding by Fareed Khan

Building Your First Ontology

A Hands-On Tutorial

Stop Reading About Ontology. Start Building One.

- A complete, functioning ontology
- Experience with industry-standard tools
- Something tangible for your portfolio
- The confidence to tackle bigger projects

```
graph TD; Ontology[Ontology] --> Restaurant[Restaurant]; Ontology --> Dish[Dish]; Ontology --> Ingredient[Ingredient]
```

Pankaj Kumar

Building Your First Ontology: A Hands-On Tutorial

Stop Reading About Ontology. Start Building One.

Dec 13, 2025 620 6



 In Data Science Collective by Marina Wyss - Gratitude Driven

AI Agents: Complete Course

ABSTRACT

Post-training alignment often reduces LLM diversity, leading to a phenomenon  2.8K own 594 collapse. Unlike prior work that attributes this effect to algorithmic limitations, we identify a fundamental, pervasive data-level driver: *typicality bias* in preference data, whereby annotators systematically favor familiar text as a result of well-established findings in cognitive psychology. We formalize this bias theoretically, verify it on preference datasets empirically, and show that it plays a central role in mode collapse. Motivated by this analysis, we introduce *Verbalized Sampling (VS)*, a simple, training-free prompting strategy to circumvent mode collapse. VS prompts the model to verbalize a probability distribution over a set of responses (e.g., "Generate 5 jokes about coffee and their corresponding probabilities"). Comprehensive experiments show that VS significantly improves performance across creative writing (poems, stories, jokes), dialogue simulation, open-ended QA, and synthetic data generation, without sacrificing factual accuracy and safety. For instance, in creative writing, VS increases diversity by 1.6-2.1 \times over direct prompting. We further observe an emergent trend that more capable models benefit more from VS. In sum, our work provides a new data-centric perspective on mode collapse and a practical inference-time remedy that helps unlock pre-trained generative diversity.



...

:2510.01171v3 [cs.CL] 10 Oct 2025

Problem: Typicality Bias Causes Mode Collapse

Tell me a joke about coffee

Solution: Verbalized Sampling (VS) Mitigates Mode Collapse

Different prompts collapse to different modes:

 In Generative AI by Adham Khaled

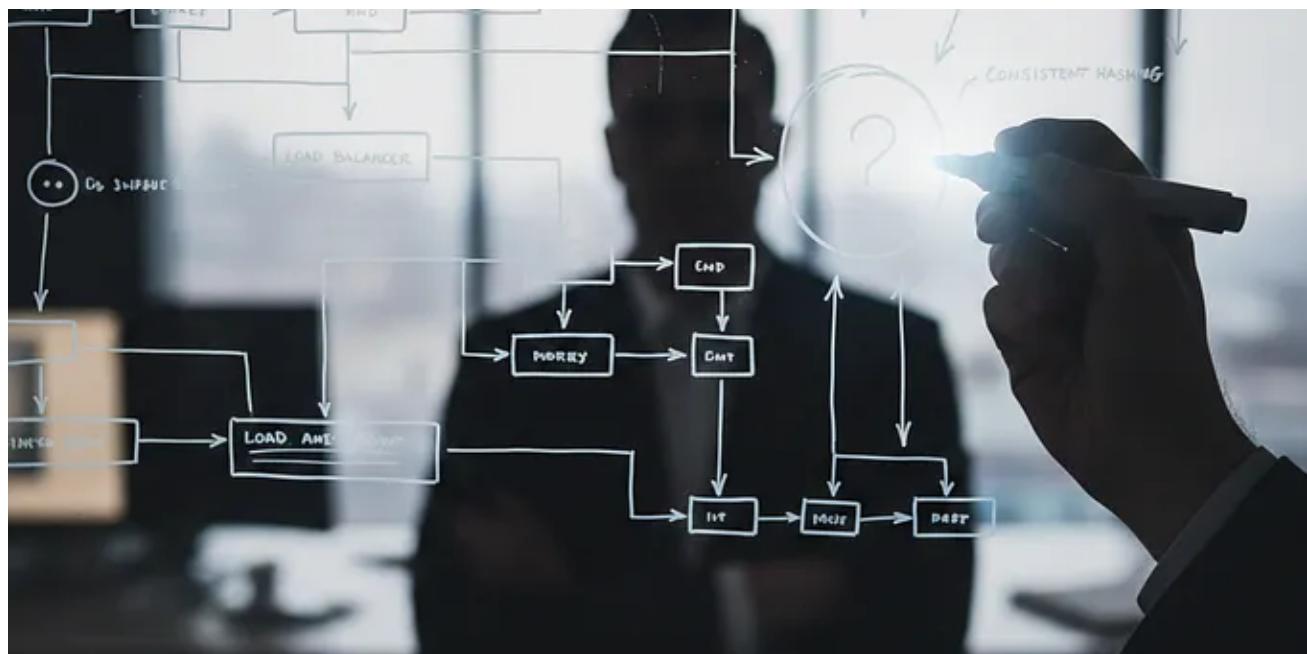
Stanford Just Killed Prompt Engineering With 8 Words (And I Can't Believe It Worked)

ChatGPT keeps giving you the same boring response? This new technique unlocks 2 \times more creativity from ANY AI model—no training required...

 Oct 20, 2025  23K  590



...



 In Beyond Localhost by The Speedcraft Lab

I Thought I Knew System Design Until I Met a Google L7 Interviewer

A single whiteboard question revealed the gap between knowing patterns and actually designing systems that scale.

 Dec 22, 2025  4.4K  82

...

[See more recommendations](#)