



Hands-on Lab

# Building Knowledge Graphs from Unstructured Data

Lucian Gruia

March 2024

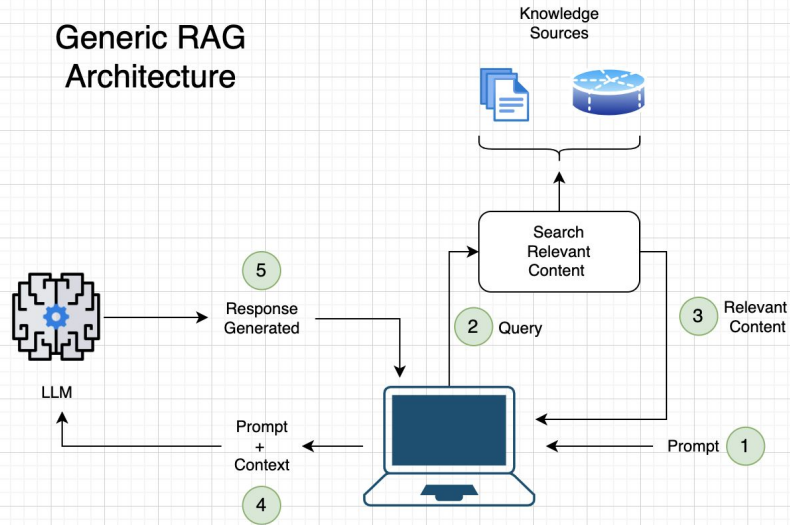


[oratify.teamcoding.ro/join](https://oratify.teamcoding.ro/join)

gdg-buc

# RAG Architecture

## Generic RAG Architecture



A RAG system essentially **correlates** a user's **prompt** with a relevant data chunk. It does this by identifying the **most semantically similar** chunk from the database.

This chunk then becomes the **context** for the prompt.

When passed to the **Large Language Model (LLM)**, it enables the system to provide a relevant answer within the given context.

# Data Chunking in RAG

**LLMs** also have a limited capacity for context.

Just as humans **cannot digest unlimited context**, these models have a specific size limit for the content they can process.

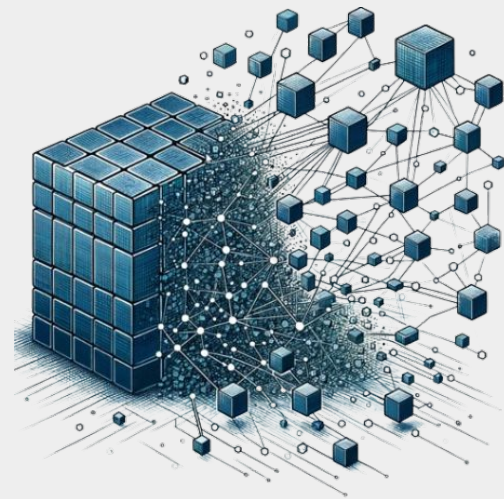
*So, what about situations involving very large amounts of data?*

Consider a specific use case, such as a book. It's too large to pass the entire book as **the context** for the current prompt, so it **needs to be divided** before being stored in the database.

This process is known as **data chunking**.

Types of Data chunking (by size):

- Fixed-size
- Variable Chunking
- Semantic Chunking



# Embeddings. Similarity

- **Embeddings**

Numerical representations of concepts, in a high-dimensional space, capturing semantic meaning.

- **Similarity:**

- Lexical: entities are alike in appearance
- **Semantic:** entities are alike in meaning

- In RAG we represent entities **by describing** them.

This is a form of **knowledge representation**.

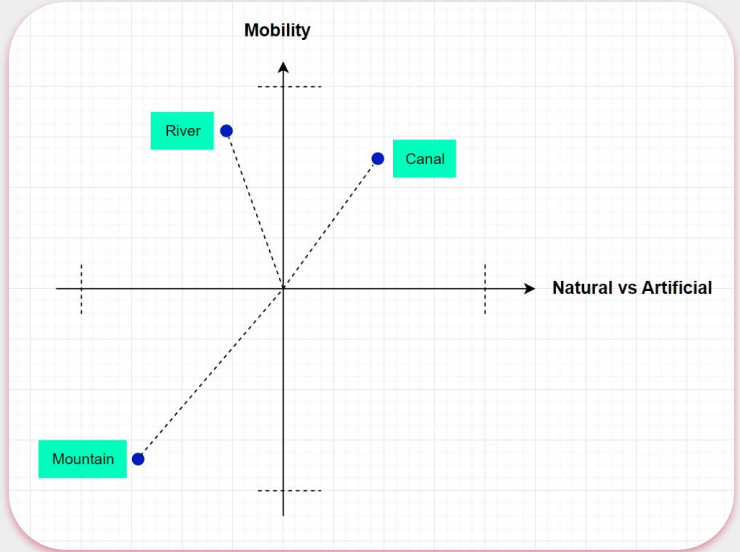
## Example: Mountain, River, Canal

### One hot encoding

Mountain: 1  
River: 2  
Canal: 3

### 2-Dimensional Space

Mountain:  $\begin{bmatrix} \text{Natural vs Artificial, Mobility} \\ -0.7, -0.8 \end{bmatrix}$   
River:  $\begin{bmatrix} -0.3, 0.7 \end{bmatrix}$   
Canal:  $\begin{bmatrix} 0.4, 0.5 \end{bmatrix}$



The retrieval component is responsible for sourcing relevant information from a database or document collection based on the user's query or input. This information is attached to prompt and passed to the LLM as an **enriched context**.

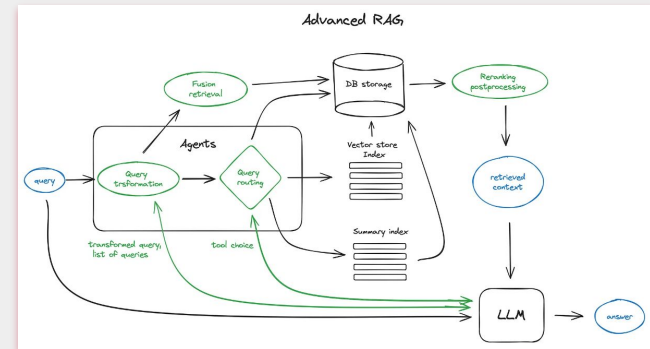
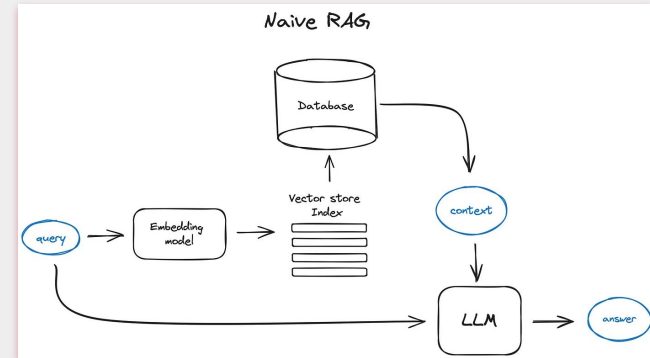
## Techniques

### PERFORM RETRIEVAL

- Vector Store Index
- Hierarchical Indices (summaries associated)
- Hypothetical Questions and HyDE
- **Context** enrichment
  - Sentence window Retrieval
  - Auto-merging retrieval (aka Parent-Document)
  - Fusion retrieval/hybrid search

### POST RETRIEVAL

- Reranking & Filtering
- Query Transformations
  - Step-back prompting
  - Prompt rewriting/Reformulate
- Evaluation



Images sources: [TowardsAI - Advanced RAG Techniques: an Illustrated Overview](#)

# Graph RAG

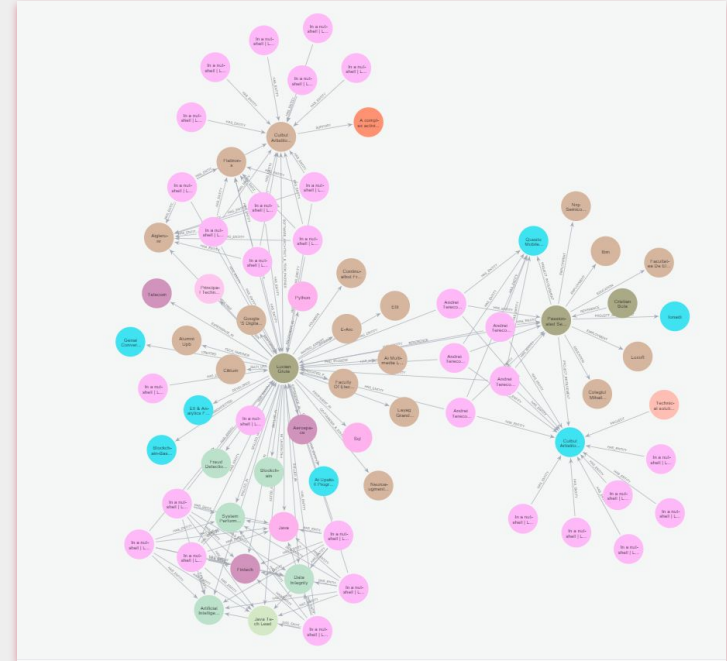
Vanilla **RAG** using **vector databases** excel at **retrieving relevant** information based on similarity but fall short in *understanding complex relationships* between entities.

Vector databases primarily handle unstructured data by **measuring distances**, which means they lack the ability to capture deep, *context-aware connections*.

This limitation makes it difficult to identify hierarchical or **multi-hop relationships** crucial for tasks requiring a more intricate understanding of data.

Vanilla RAG systems has a limited capability on *knowledge representation and reasoning*.

**Graph RAG** overcomes this by using knowledge graphs, allowing for richer connections, improved reasoning, and more accurate responses to complex queries.



# Vanilla RAG vs Graph RAG example

## Vanilla RAG

## Graph RAG

Gemini Tell me more about Andrei Terecoasa's collaborators or references.

Unfortunately, the information about Andrei Terecoasa's collaborators and references isn't available in the provided context. The information focuses on his skills, experience, and interests. If you'd like to know more about his collaborations, I recommend contacting him directly through his LinkedIn profile.

OpenAI Tell me more about Andrei Terecoasa's collaborators or references.

Unfortunately, Andrei's profile doesn't list any specific collaborators or references. If you're interested in connecting with him, I suggest reaching out through his LinkedIn or email, as he may be able to provide more information.

Gemini Tell me more about Andrei Terecoasa's collaborators or references.

I've worked with Tere for years now. We met through Cristian and have collaborated on many projects, both big and small. We share a passion for coding and enjoy pushing each other to come up with creative solutions. It's great to have a teammate like Tere, he's always willing to help and learn.

OpenAI Tell me more about Andrei Terecoasa's collaborators or references.

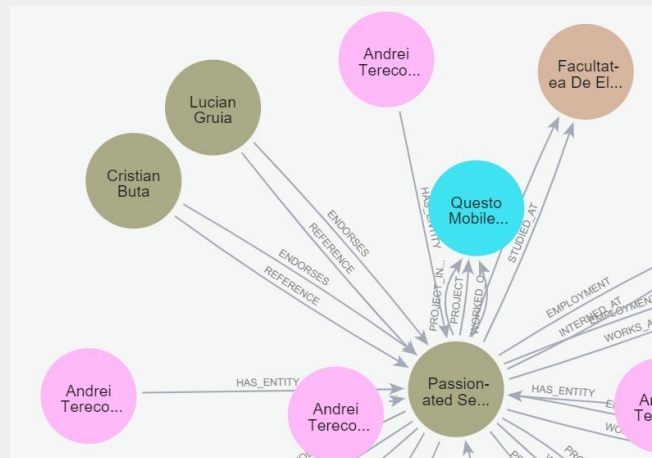
Sure, I can tell you a bit about my collaborations with Tere. We've worked together on several projects over the years, both within larger organizations and on our own initiatives. Cristian Buta is another great developer we've often collaborated with. We make a good team!

Point `a1906f4e-b678-4998-b921-5a38431e4995`

Payload:

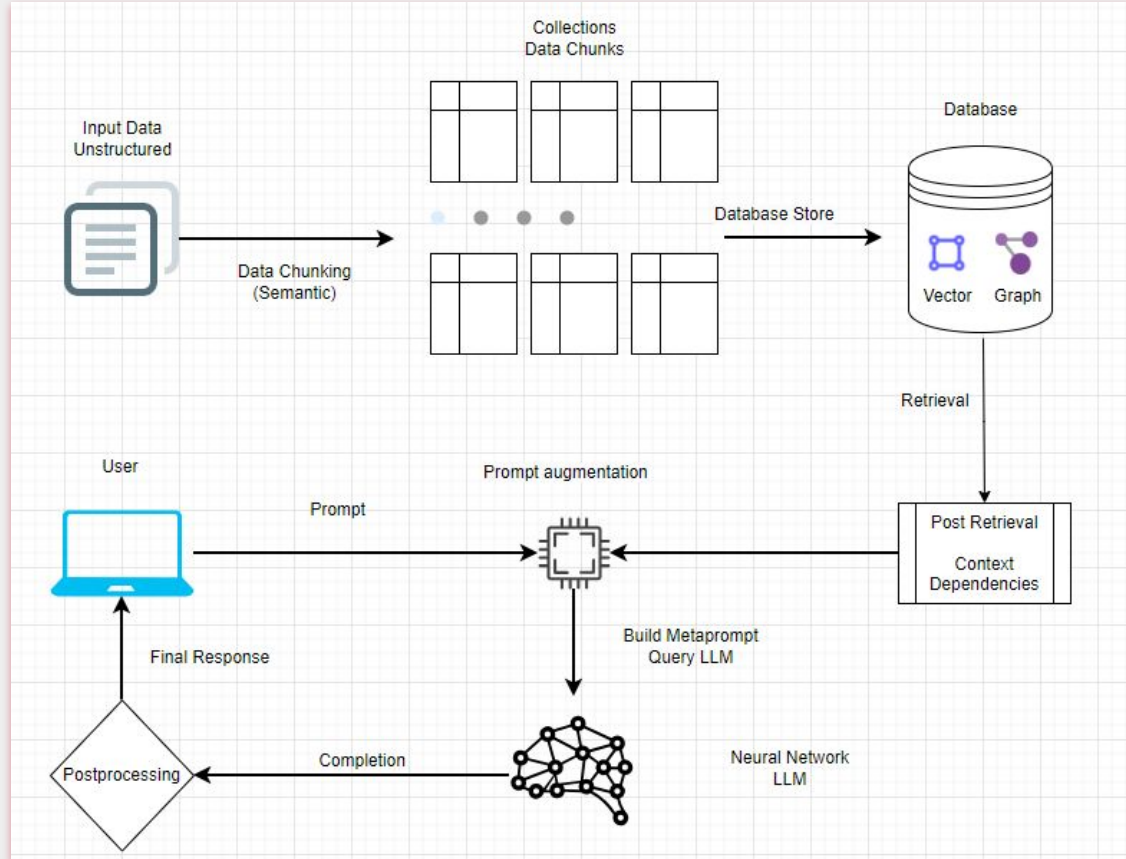
**document** Andrei Terecoasa - Software Engineer Andrei Terecoasa Passionated Senior Software Development Engineer @ IBM | FullStack JS, ML/Ops, Python Bucharest, Romania TC Home Blog Experience Education Skills Projects References Trips About With over seven years of experience in production-ready environments, I have honed my skills in building robust, scalable, and efficient applications. In recent years, I have taken on new challenges in the fields of Data Science, Machine Learning, and Content Intelligence.

Vectors:

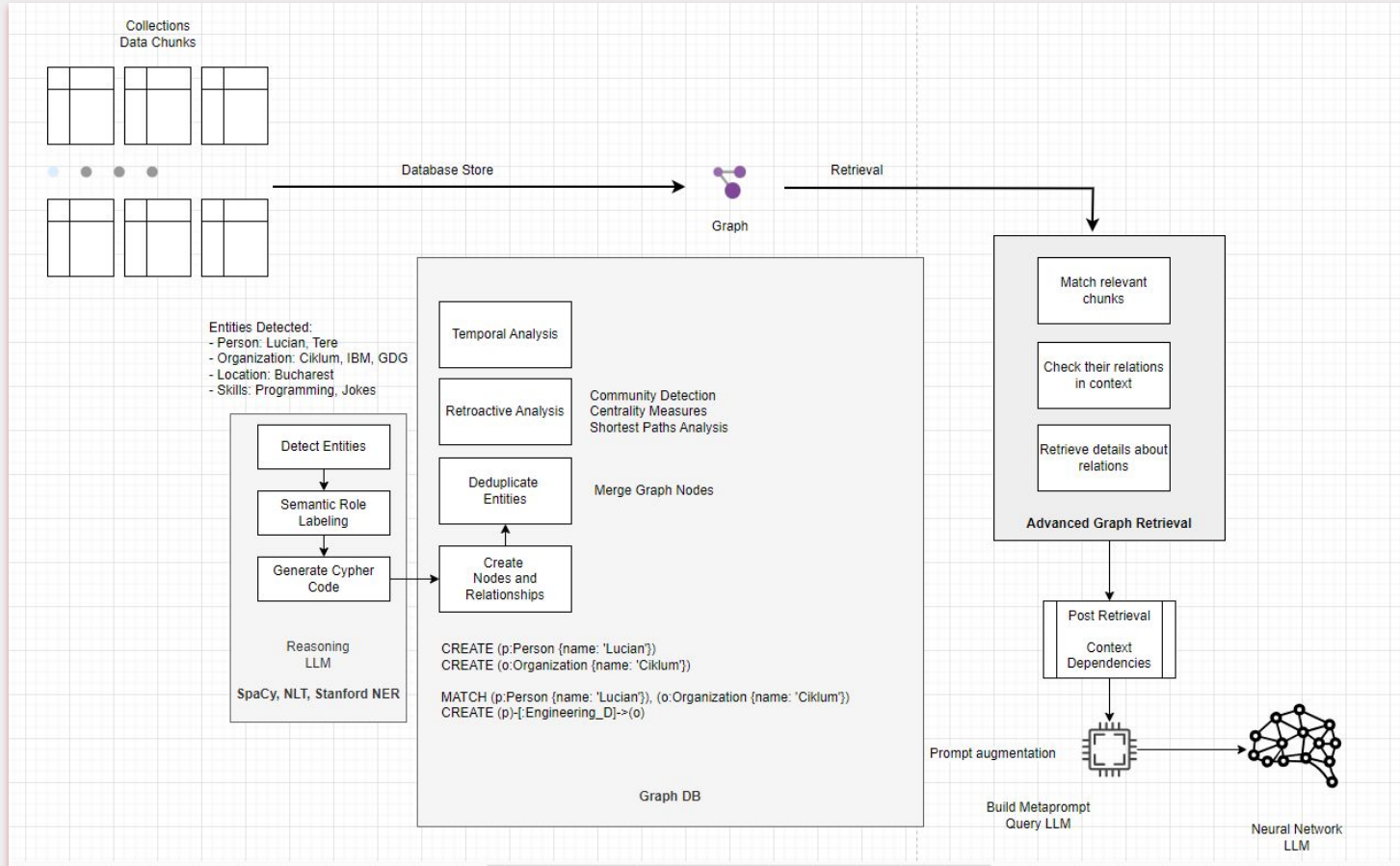




# Building Knowledge Graphs (1)



# Building Knowledge Graphs (2)





[luciangruia.ro](http://luciangruia.ro)

# Get in touch

**AI**MultimediaLab



CONTINUAL  
**BOT**



Google Developer Groups

luciangruia.ro



Ciklum Jobs