## What is RAG?

LLMs, although capable of generating text that is both meaningful and grammatically correct, these LLMs suffer from a problem called hallucination. Hallucination in LLMs is the concept where the LLMs confidently generate wrong answers, that is they make up wrong answers in a way that makes us believe that it is true. This has been a major problem since the introduction of the LLMs. These hallucinations lead to incorrect and factually wrong answers. Hence Retrieval Augmented Generation was introduced.

In RAG, we take a list of documents/chunks of documents and encode these textual documents into a numerical representation called vector embeddings, where a single vector embedding represents a single chunk of document and stores them in a database called vector store. The models required for encoding these chunks into embeddings are called encoding models or bi-encoders. These encoders are trained on a large corpus of data, thus making them powerful enough to encode the chunks of documents in a single vector embedding representation.

## Explain Semantic Chunking

In order to abide by the context window of the LLM , we usually break text into smaller parts / pieces which is called chunking.

Different chunking methods:

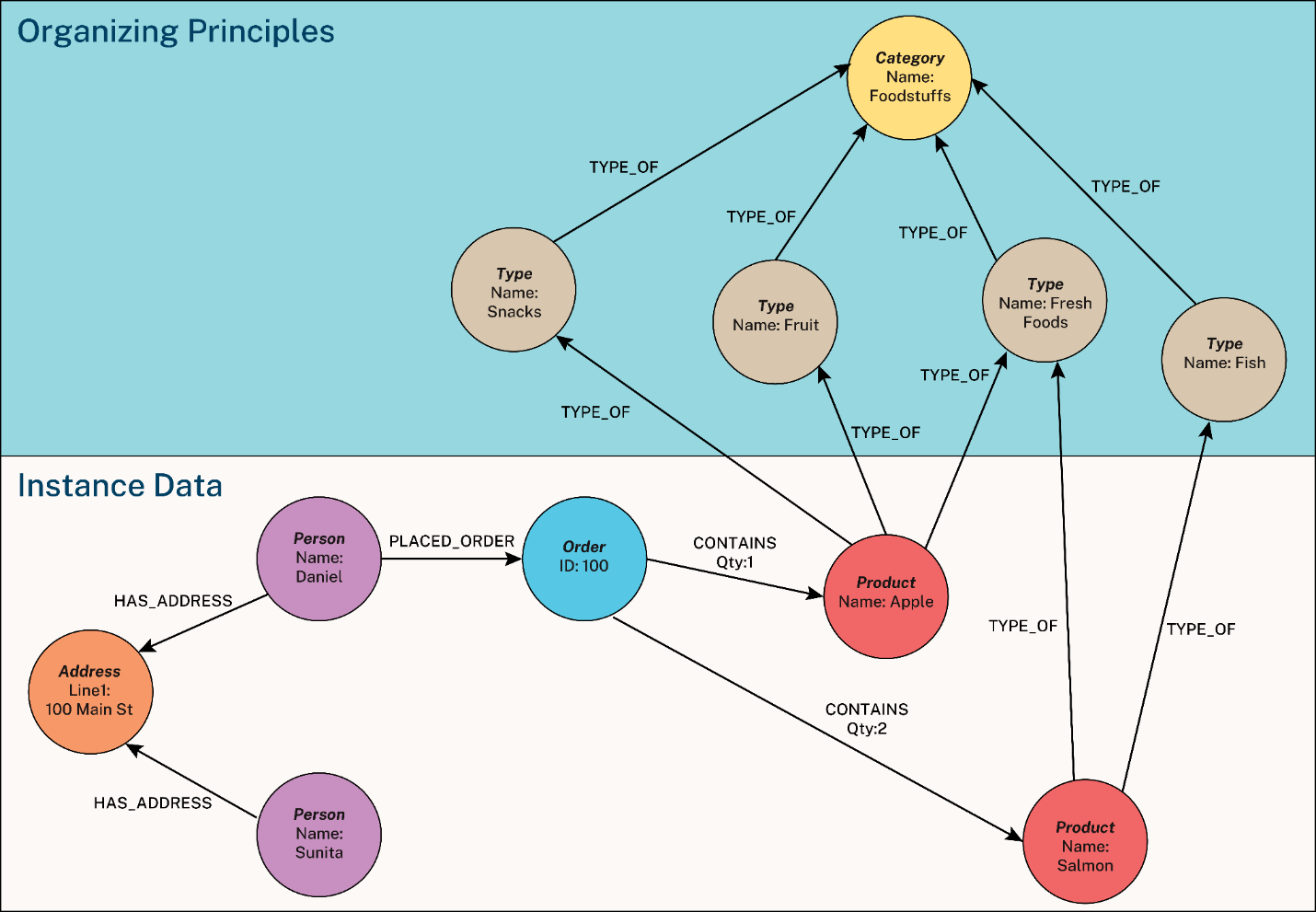
* Fixed size chunking
* Recursive Chunking
* Document Specific Chunking
* Semantic Chunking
* Agentic Chunking

Semantic chunking involves taking the embeddings of every sentence in the document, comparing the similarity of all sentences with each other, and then grouping sentences with the most similar embeddings together. By focusing on the text’s meaning and context, Semantic Chunking significantly enhances the quality of retrieval. It’s a top-notch choice when maintaining the semantic integrity of the text is vital.

The hypothesis here is we can use embeddings of individual sentences to make more meaningful chunks. Basic idea is as follows :-

1. *Split the documents into sentences based on separators(.,?,!)*
2. *Index each sentence based on position.*
3. *Group: Choose how many sentences to be on either side. Add a buffer of sentences on either side of our selected sentence.*
4. *Calculate distance between group of sentences.*
5. *Merge groups based on similarity i.e. keep similar sentences together.*
6. *Split the sentences that are not similar.*

## What Is a Knowledge Graph?



A [knowledge graph](https://neo4j.com/use-cases/knowledge-graph/) is an organized representation of real-world entities and their relationships. It is typically stored in a graph database, which natively stores the relationships between data entities.Entities in a knowledge graph can represent objects, events, situations, or concepts. The relationships between these entities capture the context and meaning of how they are connected.

Now that you understand how knowledge graphs organize and access data with context, let’s look at the building blocks of a knowledge graph data model. The definition of knowledge graphs varies depending on whom you ask, but we can distill the essence into three key components: nodes, relationships, and organizing principles.

***Nodes*** denote and store details about entities, such as people, places, objects, or institutions. Each node has a (or sometimes several) label to identify the node type and may optionally have one or more properties (attributes). Nodes are also sometimes called *vertices*.

***Relationships*** link two nodes together: they show how the entities are related. Like nodes, each relationship has a label identifying the relationship type and may optionally have one or more properties. Relationships are also sometimes called *edges*.

**Organizing Principles** are a framework, or schema, that organizes nodes and relationships according to fundamental concepts essential to the use cases at hand. Unlike many data designs, knowledge graphs easily incorporate multiple organizing principles.

In [**generative AI**](https://neo4j.com/generativeai/)applications**,** knowledge graphs capture and organize key domain-specific or proprietary company information. Knowledge graphs are not limited to structured data; they can handle less organized data as well.

In [**Fraud Detection and Analytics**,](https://neo4j.com/use-cases/fraud-detection/) the knowledge graph represents a network of transactions, their participants, and relevant information about them.