Large Language Models (LLMs)

RAG (Retrieval-Augmented Generation)

Zeham Management Technologies BootCamp

by SDAIA

September the 18th, 2024

SDAIA
الهيئة السعودية للبيانات والذكاء الاصطناعي
Saudi Data & AI Authority

Agenda

- Introduction to RAG (Retrieval-Augmented Generation)
- RAG's components
- Practical Activity
- Retriever Workflow
- Generator Component

Introduction to RAG (Retrieval-Augmented Generation)

What is RAG? (Retrieval-Augmented Generation)

RAG stands for **Retrieval-Augmented Generation.** RAG is a method that combines finding information from a large source and using AI to generate responses based on that information.

Why do we need RAG?

Before RAG, traditional AI models could only use the information they were trained on, which meant they couldn't look up new facts or access external data, often leading to incomplete or outdated answers.

Incorrect answer with normal LLM

نظام النقاط في المخالفات المرورية

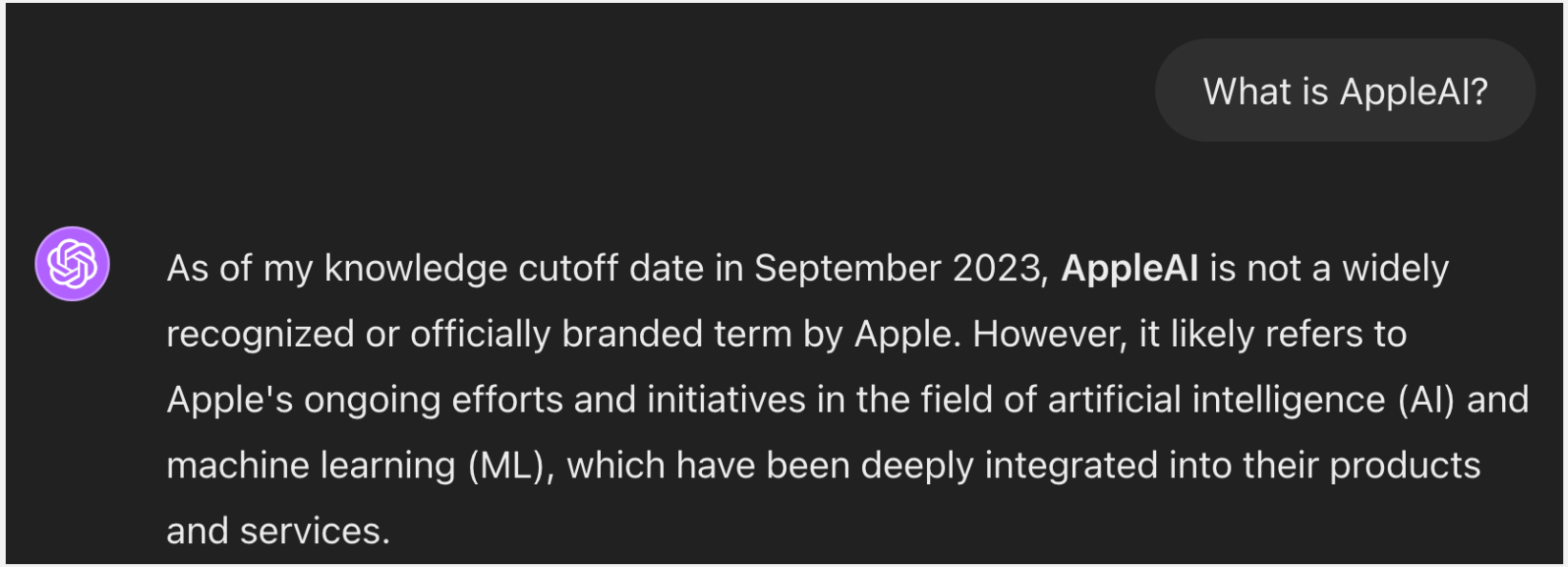
تطبق عقوبة سحب رخصة القيادة إذا بلغ عدد النقاط المسجلة بحق مرتكب المخالفات (24) نقطة على النحو التالي :

1. عند حصول المخالف للمرة الأولى على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة ثلاثة أشهر.
2. عند حصول المخالف للمرة الثانية على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة ستة أشهر.
3. عند حصول المخالف للمرة الثالثة على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة سنة هجرية.
4. عند حصول المخالف للمرة الرابعة على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة

Source

Why do we need RAG? Cont.

Without the ability to retrieve new information, AI models were limited to their training data, making it hard to handle **specific or up-to-date questions**, resulting in less accurate and less reliable answers.



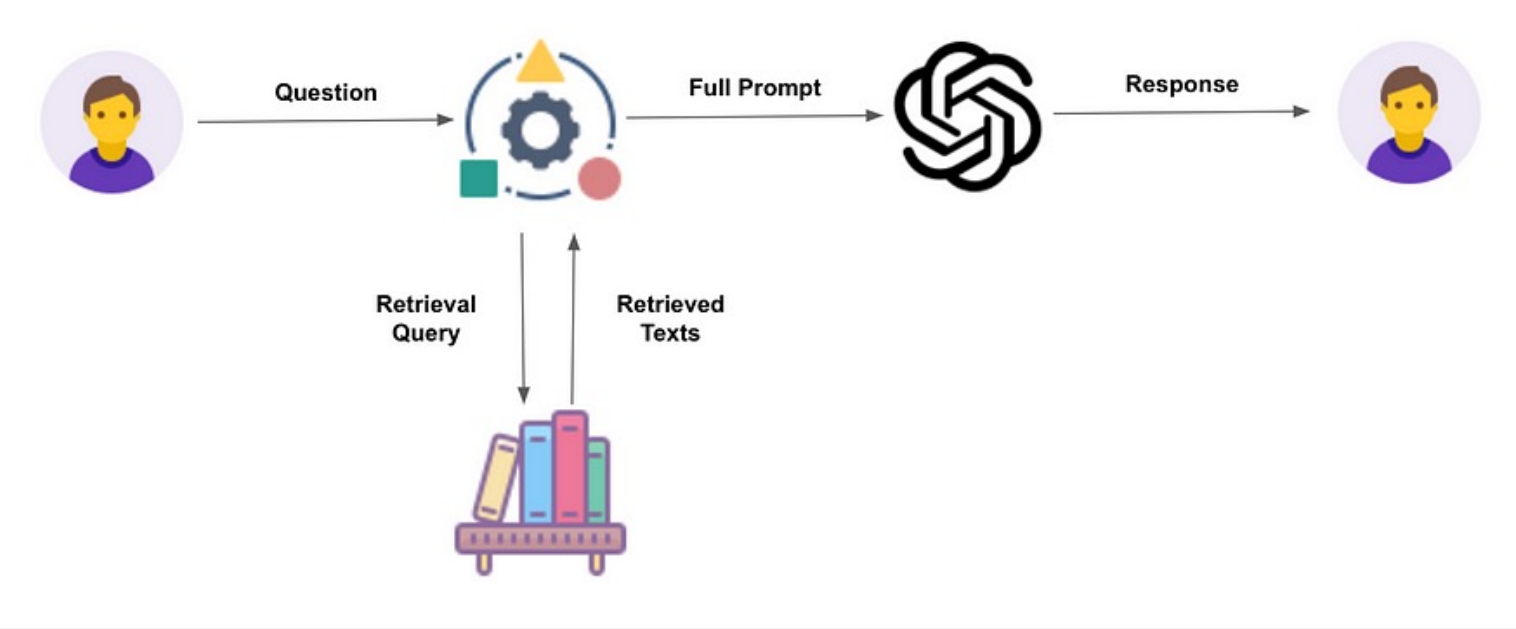
When inquiring about a recent event from a typical LLM, it may respond with "as of my last knowledge," indicating that the model's information is outdated.

How RAG Solves the Problem

Imagine that an AI is trying to answer a question, but instead of just relying on what it already knows, it can now search through huge databases, like looking things up on the internet, and then use that information to create a more accurate and up-to-date response. Or even search through documents to provide you with accurate response.

RAG and Its Components

RAG has two main parts: one that **retrieves information** from a large source, and one that **generates answers** using AI. These two work together to create more accurate and detailed responses.



Source

RAG’s components

Retriever Component in RAG

The **retriever** is responsible for finding relevant information from large sources of data. In RAG, the retriever scans external databases or knowledge banks to pull the most relevant facts, which will later be used to help generate accurate answers.

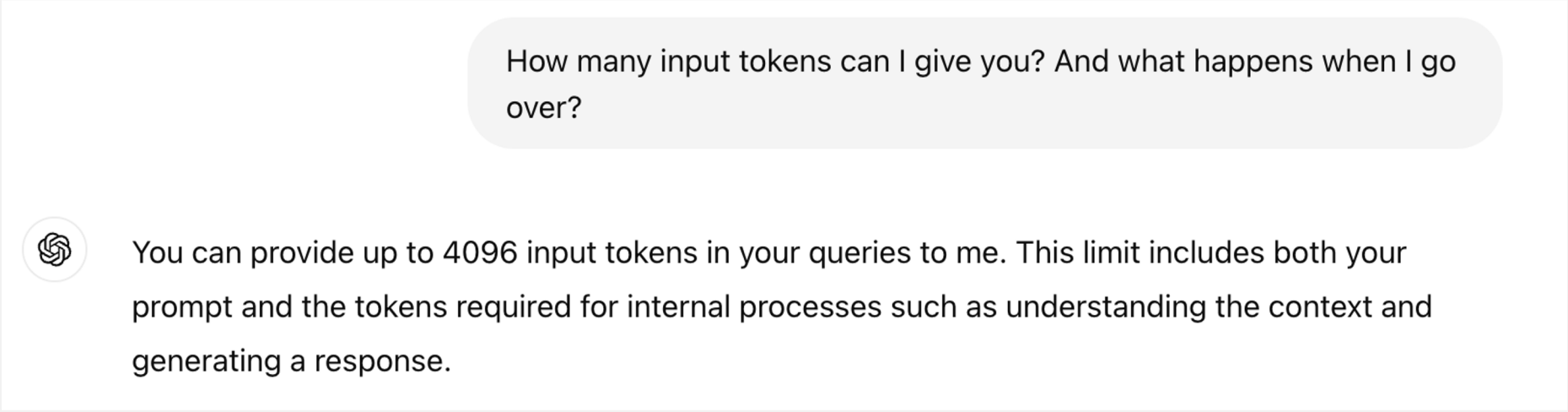
Retriever Component in RAG

Let's say I brought the relevant text to the LLM, how would the response look like?

As you can see the LLM responded with more accurate and precise answer

Retriever Component in RAG

Now let's imagine I am building a model that answers traffic violation questions. If the traffic laws are large, then the query might exceed the maximum number of tokes. This will lead to refusing to take the query due to the query size.



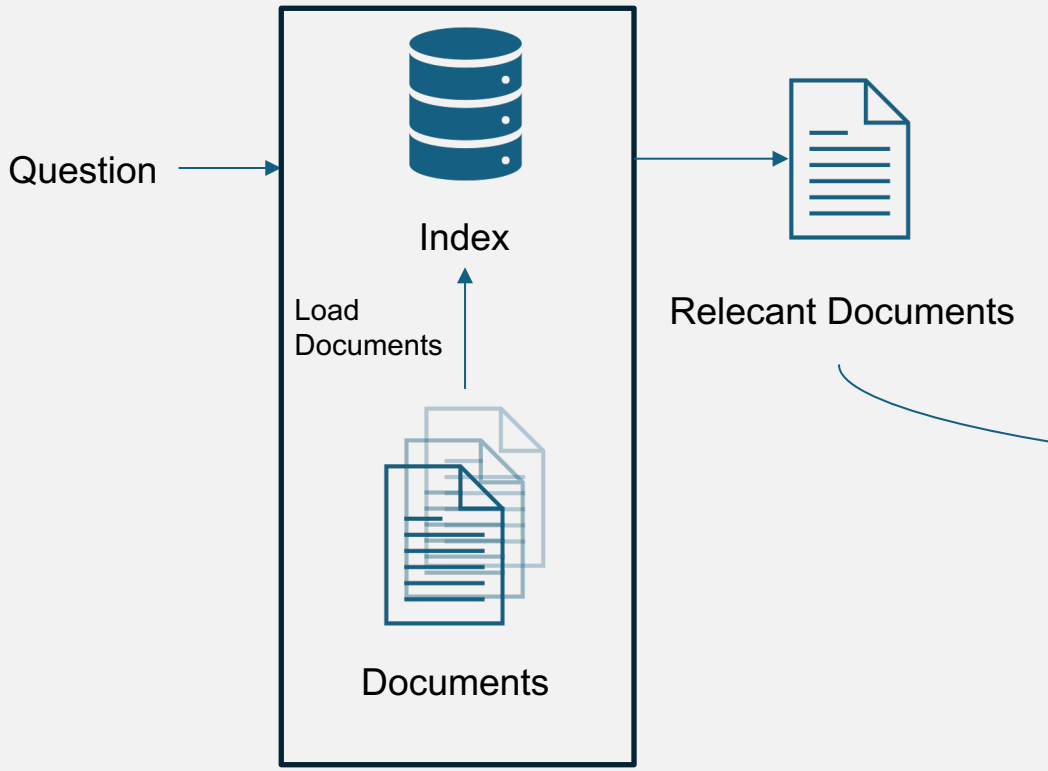
Building a Vector Database for Retrieval

- The first step is to create a vector database, first we convert data (like text or images) into vectors using embeddings.
- An embedding model, such as BERT or Word2Vec, transforms information into high-dimensional vectors, where similar items are located closer together.
- This forms the foundation of the vector database, allowing for efficient and accurate similarity searches later.

Querying a Vector Database

- The input (e.g., a question) is **converted into a vector** using an embedding model.
- The **query vector** is compared to the stored vectors in the database.
- A **similarity search** is conducted to find the closest matching vectors.
- **Algorithms** like **cosine similarity** or **Euclidean distance** measure how similar the vectors are.
- The most **relevant information** is returned based on similarity scores.

Retriever Workflow



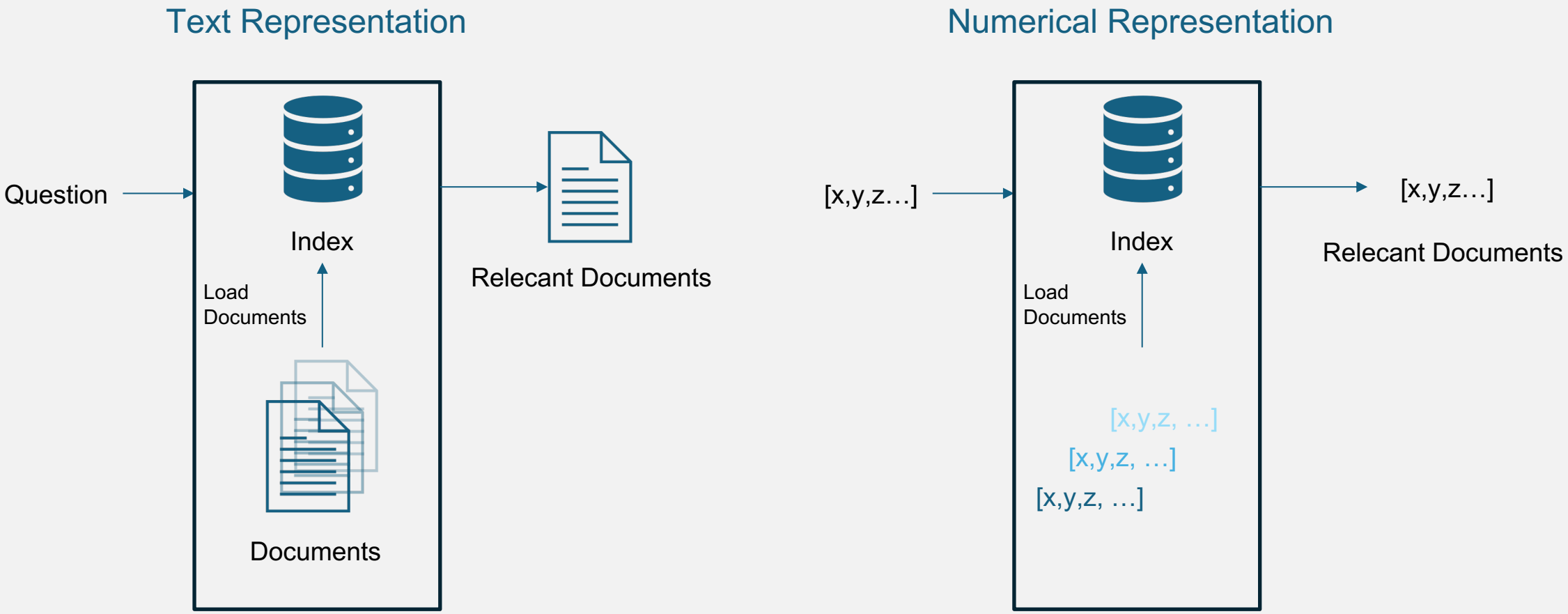
Relecant Documents

نظام النقاط في المخالفات المرورية تطبق عقوبة سحب رخصة القيادة إذا بلغ عدد النقاط المسجلة بحق مرتكب المخالفات (24) نقطة على النحو التالي :

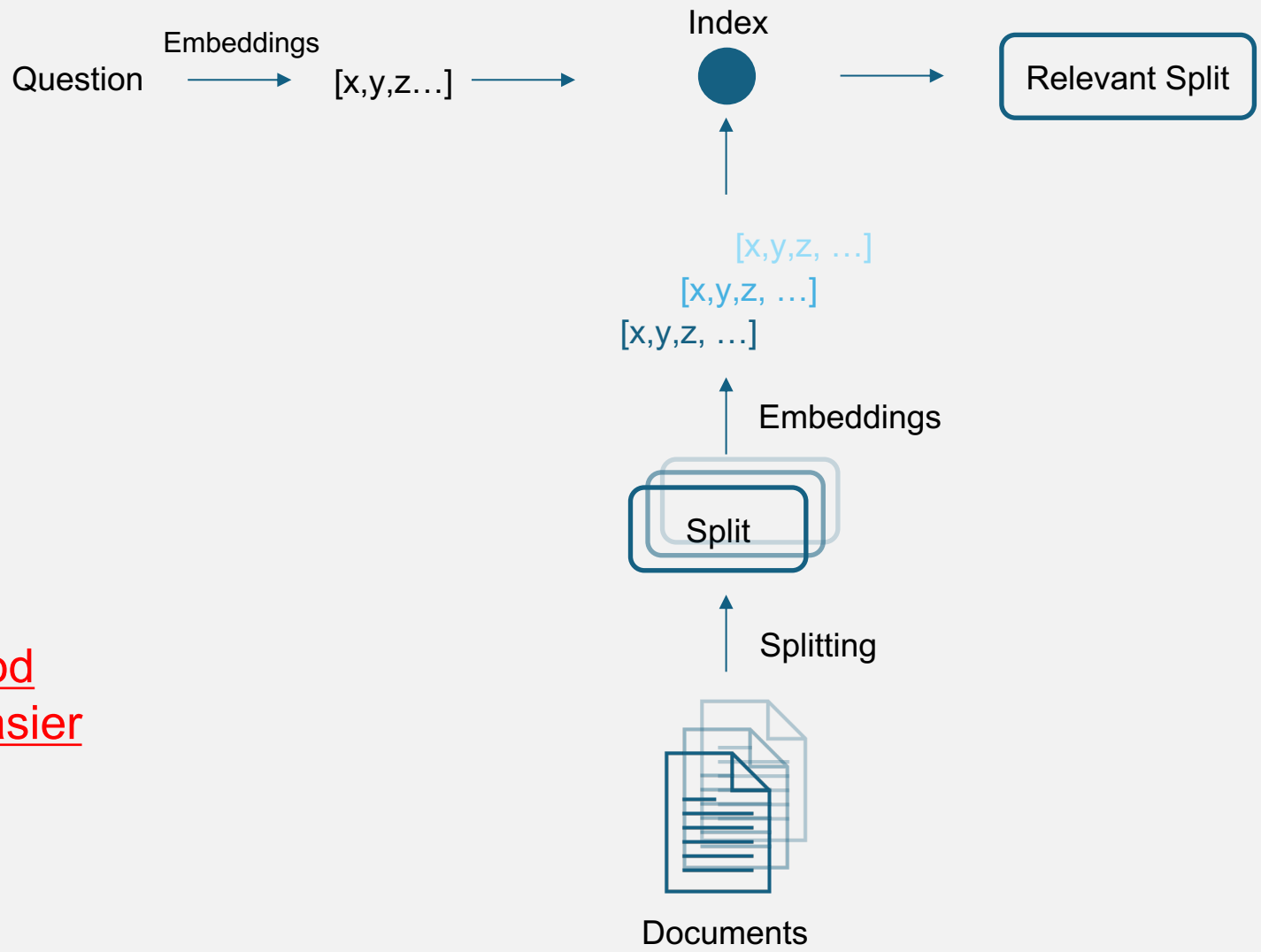
1. عند حصول المخالف للمرة الأولى على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة ثلاثة أشهر.
2. عند حصول المخالف للمرة الثانية على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة ستة أشهر.
3. عند حصول المخالف للمرة الثالثة على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة لمدة سنة هجرية.
4. عند حصول المخالف للمرة الرابعة على (24) نقطة خلال سنة هجرية تسحب رخصة القيادة

Retriever Workflow (Document Loading)

Now to query relevant document, we need to first get the numerical representation of the documents. The reason for that, is that it very easy to copare vectores rather than words.

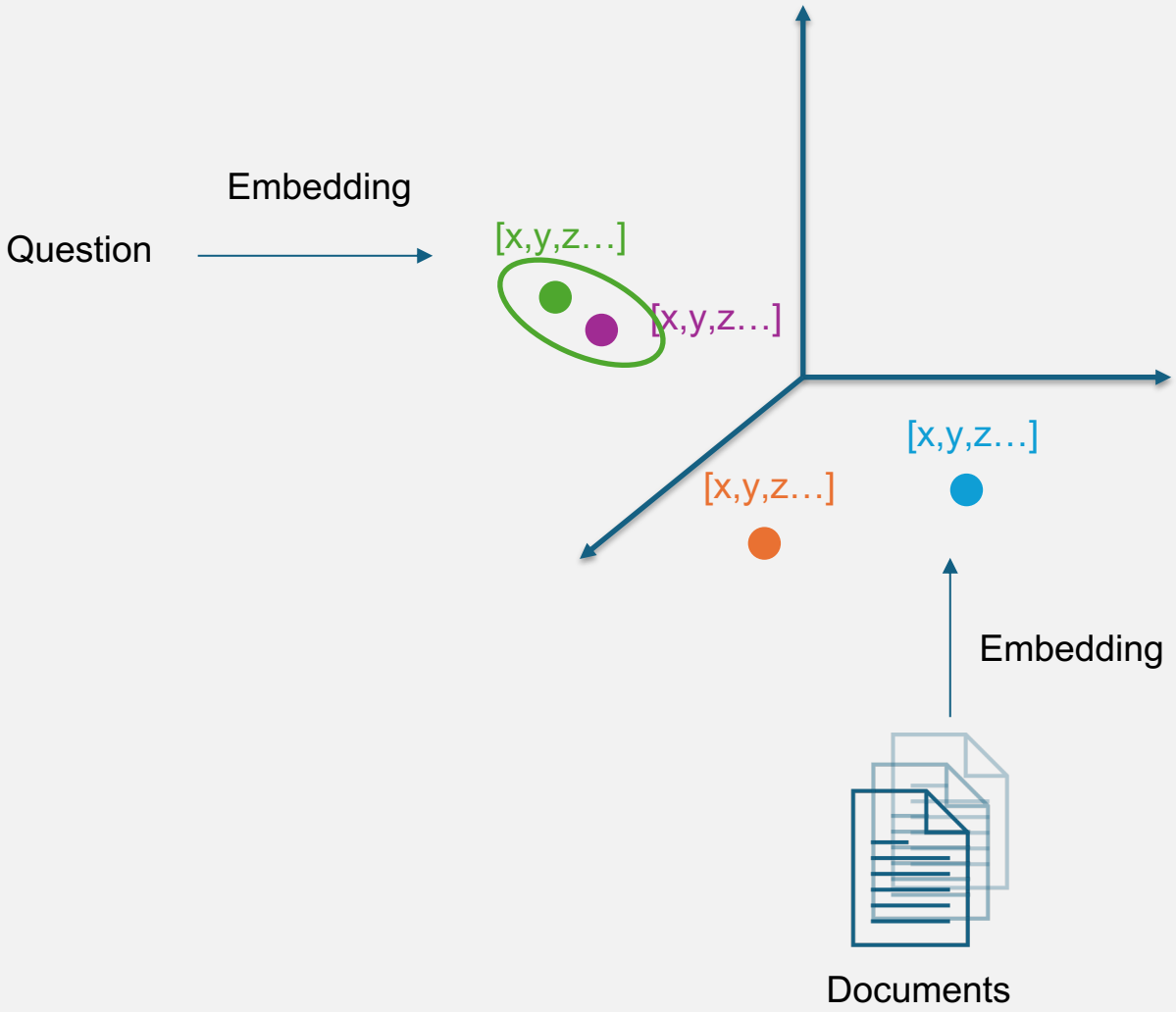


Retriever Workflow (Loading, Splitting, and Embedding)

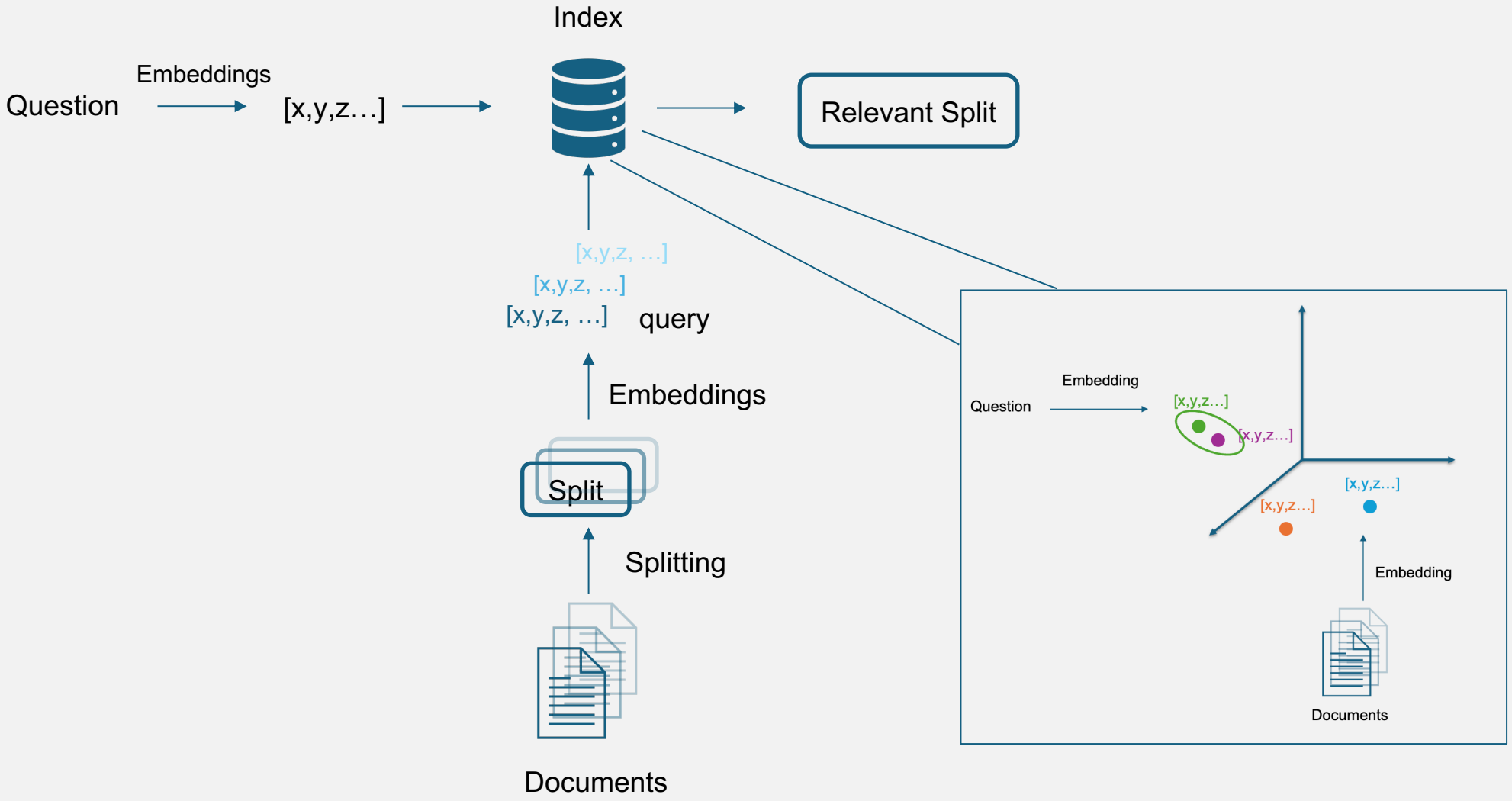


This method enables easier retrival!

Retriever Workflow (similarity search)



Retriever Workflow (similarity search)



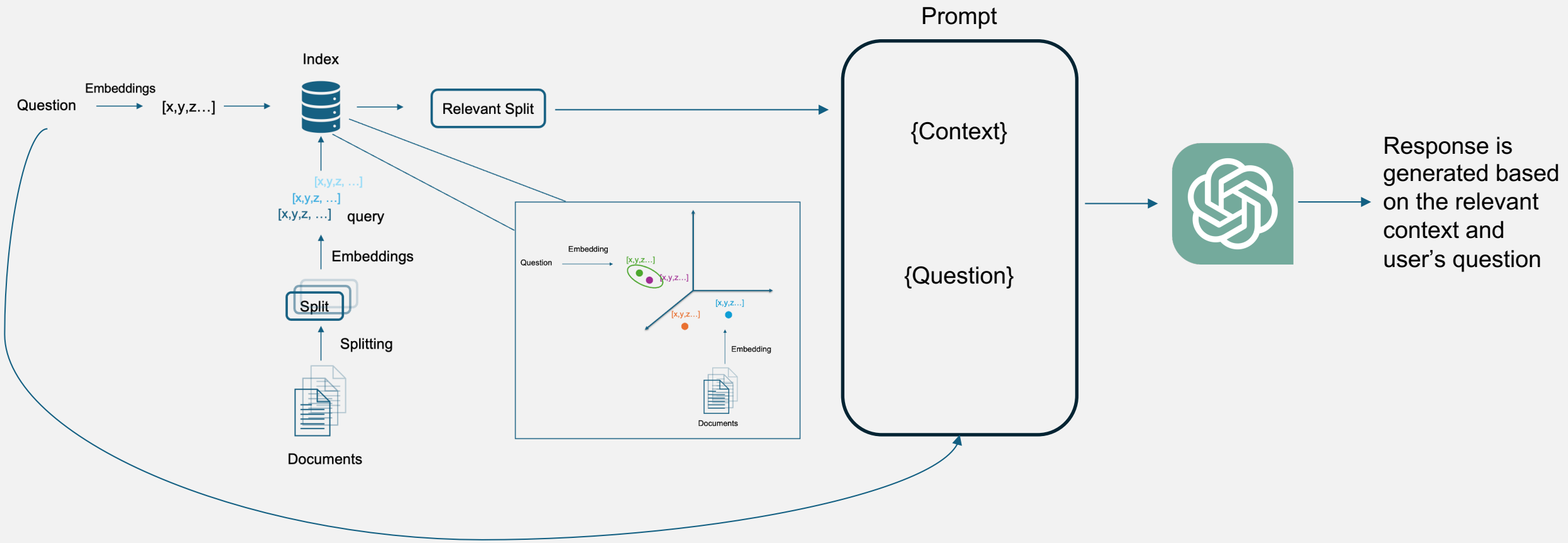
Generator Component in RAG

Once the retriever finds relevant information, the **generator** uses this data to **create a response**. The generator is powered by a **language model**, such as GPT, which takes the retrieved information and **generates a coherent answer** based on it. This allows the AI to provide detailed and contextually accurate responses.

Generator Component in RAG

It first takes the relevant split data as a **context;** this gives the model the necessary **background information** to generate an accurate and context-aware response. The context guides the model to focus on the most relevant facts and **craft a coherent, human-like answer** based on both the query and the retrieved data.

Generator's Part in the Process



Practical Activity

Let's create a Multi-Agent!

Tutorial:

RAG_Tutorial.ipynb

Exercise:

RAG_Exercise.ipynb

Thank you!

SDAIA
الهيئة السعودية للبيانات والذكاء الاصطناعي
Saudi Data & AI Authority