

SQL with Gen AI

Tables, Transformed with AI

Agenda

Application Overview

3

Data Overview

4

- Tabular Database
- Document Database

4

6

Agents Overview

10

- SQL Agent (Scenario 1)
- Analyst Agent (Scenario 2)
- SQL Agent (Scenario 2)

10

13

16

Application Overview

Objectives

Automate SQL query generation for efficient data retrieval, manipulation, and visualization

Core Concepts

- Leverage LLMs to translate natural language into database queries
- Retrieve relevant documents to provide context for query generation
- Implement an agentic communication pipeline for interactions

Tabular Database



Source Data: External CSV files

Use cases

- The generated queries are intended for use on the tabular database
- Agents retrieve data points from embeddings of rows for use in queries
 - Why embeddings?

Data Manipulation


The CSV is loaded into database. Row-level embeddings are generated and saved within the same database for similarity search.

Chunking Strategy

- Rows are converted into sentences using column names to provide semantic clarity
- Embeddings are stored in the structured database

Access database			Create sentence per row	Create and store embedding per row
Name	Address	Rating	<ol style="list-style-type: none">1. The Name is Pasta Planet. The Address is 521 Napoleon Rd, 11225. The Rating is 4.2. The Name is Griddle Spot. The Address is 184 Kindle Rd, 11223. The Rating is 5.3. The Name is Pizza House. The Address is 777 Nostrant, 11226. The Rating is 5	{row_1: [embeddings...]}
Pasta Planet	521 Nep.. 11225	4		
Griddle Spot	184 Kin... 11223	5		{row_2:[embeddings...]}
Pizza House	777 No... 11226	5		{row_3:[embeddings...]}

Document Database



Document
Database
(vectors)

Source materials: User manuals and textbooks

Data Manipulation

PDF is parsed into markdown text and then inserted into a Document Database

Use case

The document database contains supporting information that can provide added context that aids in query generation

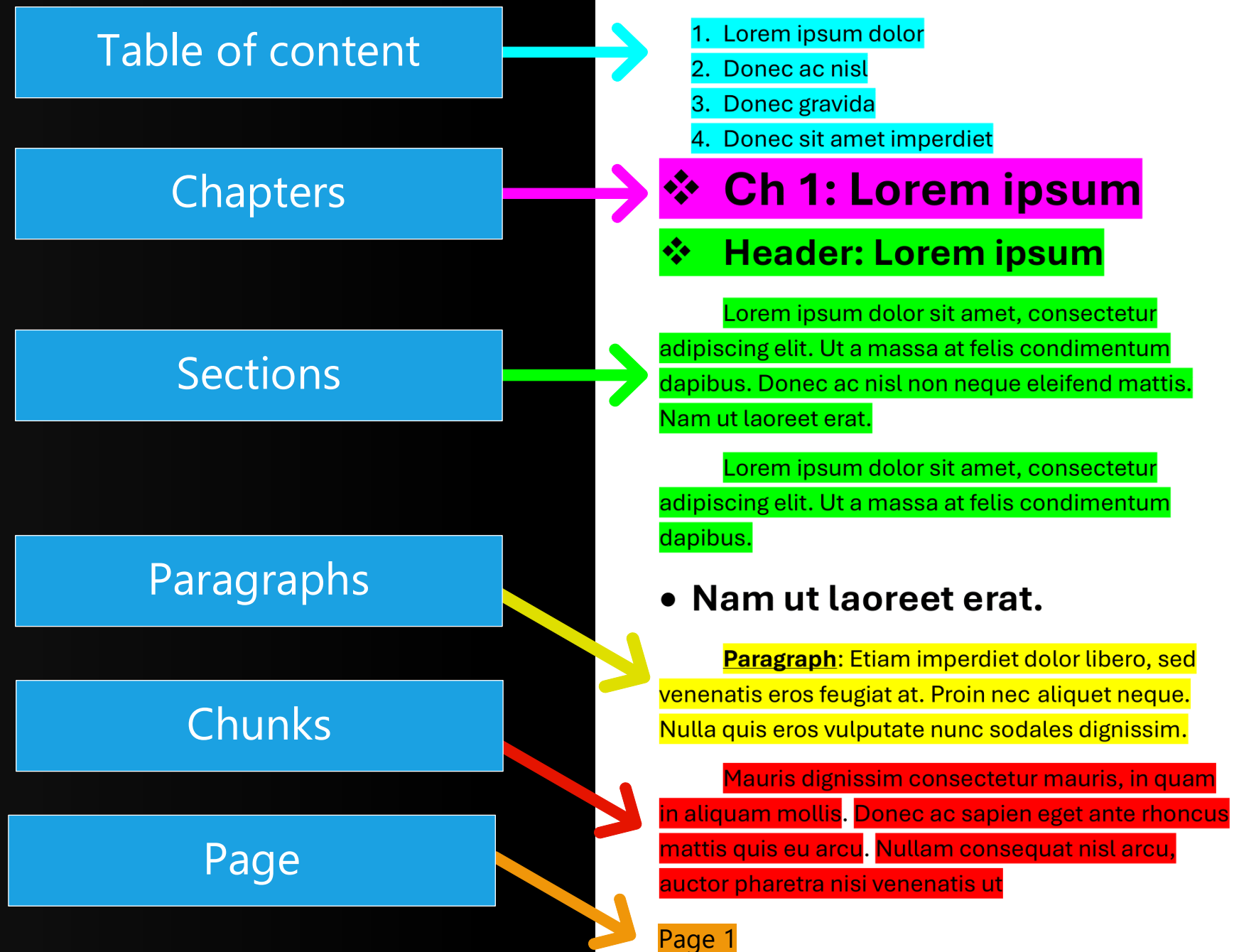
Chunking Strategy

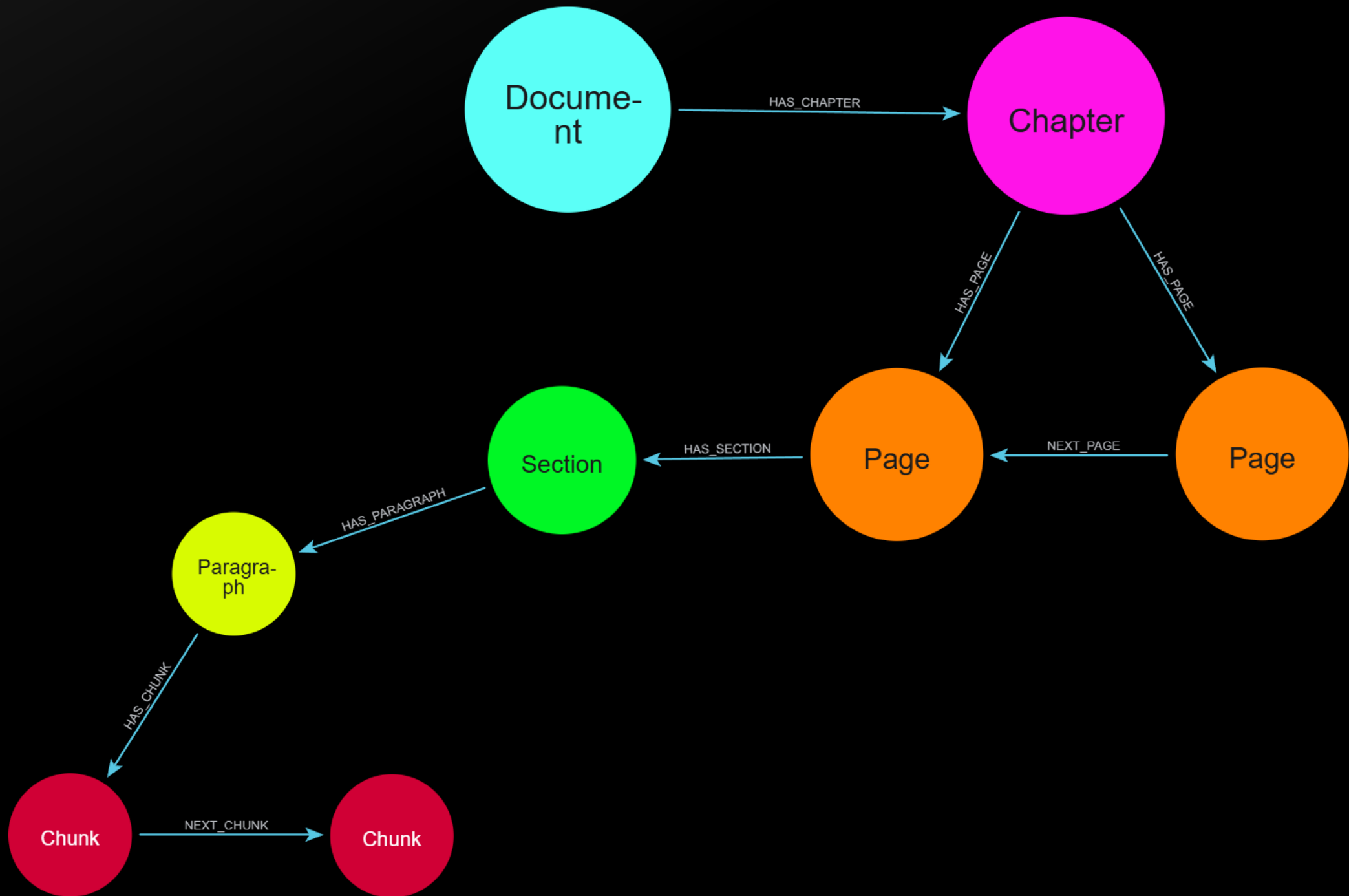
Problem with Traditional Vector Search

- Returns only the closest chunk
- No awareness of boundaries like Page, Section, or Chapter

Parent-Child Retriever Solution

Leverages graph relationships to define context-aware retrieval windows





SQL Agent

The SQL agent interprets the user's question and responds to it with SQL queries

Scenario 1: User asks a question without selecting a document database

- **Input:** User's question and selected table name
- **Output:** Database Query
- **Database:** Reads database schema and reviews results of generated queries

Input	What Country has the highest happiness index in 2024?
Output	SELECT Country, GDP_2024 FROM "1990_2025_global_gdp" ORDER BY GDP_2024 DESC LIMIT 1;

Identify Query type

Input Question: "Which country had the highest GDP in 2024?"

- **Invoke OpenAI GPT model**
- **Prompt:** "Does the users question, relate to a retrieval or manipulation query?"

Output Query Type: retrieval

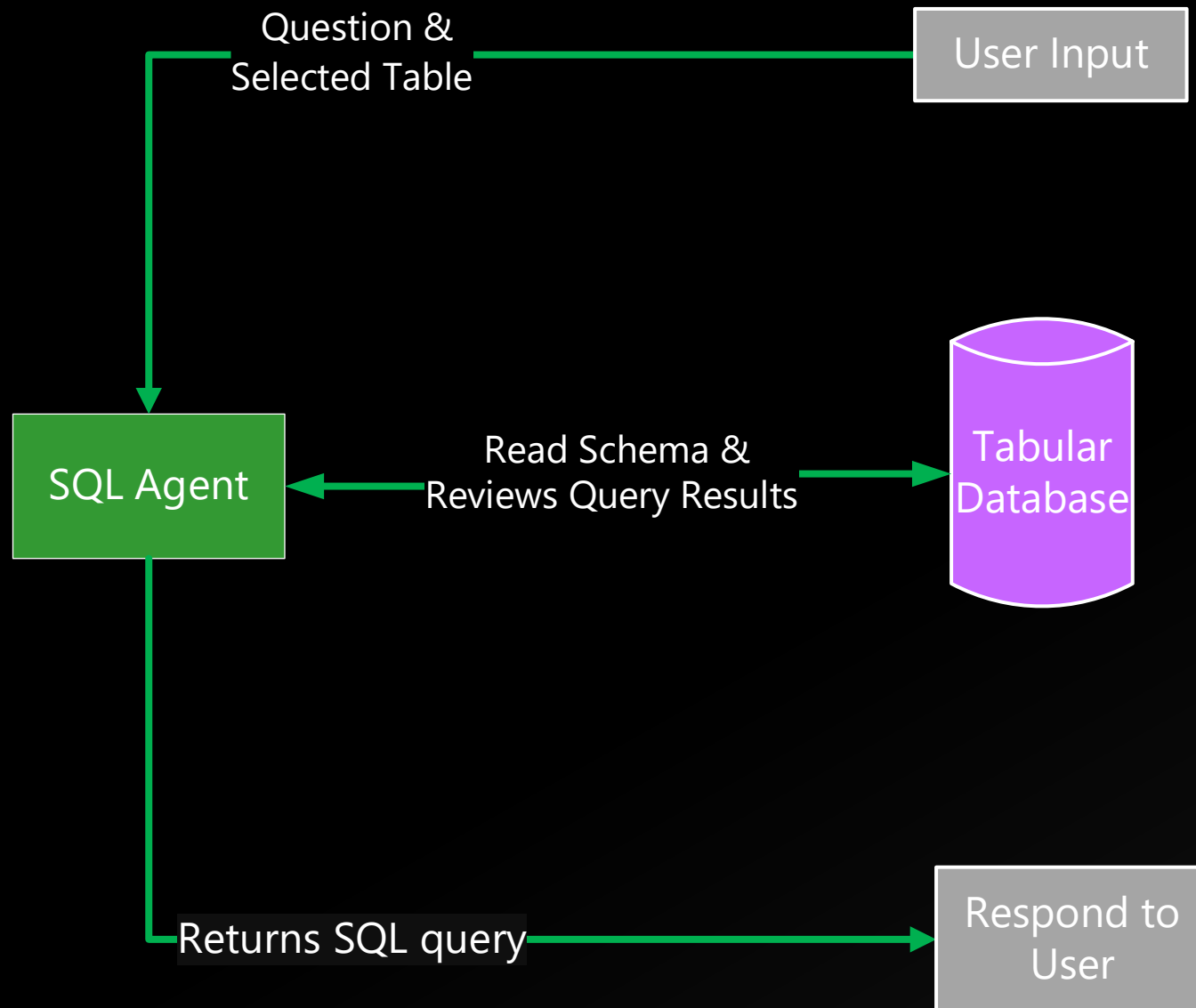
Create SQL Query

Input Question & Query Type

- **Invoke OpenAI GPT model**
- **Prompt:** "Write a **retrieval** SQL query that answers the users **question** using the data in **table**. Test the query by observing the results for accuracy."
- **Use tool calling for tabular database schema access and query execution**
- Tool generates queries and tests each one by reviewing results.
- Returns best query

Output Generated Query:

```
``SELECT Country, GDP_2024 FROM  
"1990_2025_global_gdp" ORDER BY GDP_2024  
DESC LIMIT 1;``
```



Analyst Agent

Retrieves relevant context from documents, enabling the SQL Agent to use the information from retrieved data in query.

Scenario 2: User asks question with document selected. Table does not have address of NY Aquarium, but document database does.

- **Input:** User's question
- **Output:** Retrieval from document database and summarization of retrieval. Identifies relevant columns from selected table
- **Database:** Reads columns(names & types) from tabular database and relevant chapters/sections from document database

Input	What are some good restaurants near NY Aquarium?
Output	The address is 602 Surf Ave, Brooklyn, NY 11224. The relevant columns of table are address, name and rating

Retrieve related documents

Input Question: What are some good restaurants near NY Aquarium?

Retrieve documents from document vector database using cosine distance

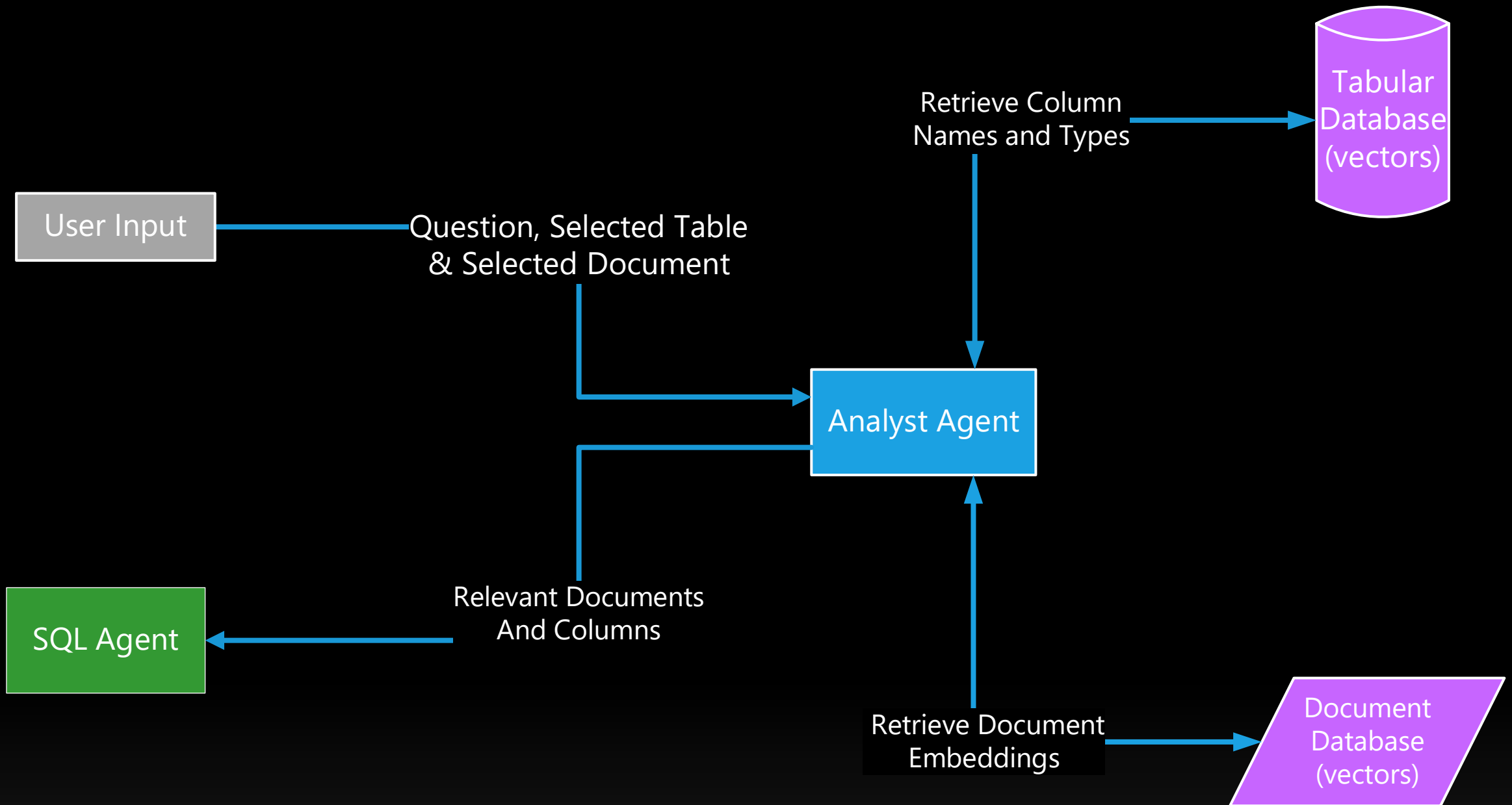
Output Retrieved Document: Proin non urna congue, dignissim tortor et, condimentum nisl. Duis aliquam nunc id risus finibus. The address is 602 Surf Ave, Brooklyn, NY 11224. pretium eros fermentum. Nunc consectetur ipsum risus, at posuere orci fermentum pretium

Summarize retrieval

Input Question, Retrieved Document & Column Names from Selected Table

- **Invoke OpenAI GPT model**
- **Prompt:** "Extract key info from document to answer the question and match relevant table columns."

Output Summary:
The address is **602 Surf Ave, Brooklyn, NY 11224**. The relevant columns are **address, name** and **rating**



SQL Agent

The SQL Agent can use data provided by Analyst Agent to increase specificity resulting in accurate queries.

Scenario 2: User asks question with document selected. Table does not have address of NY Aquarium, but document database does.

- **Input:** User's question, summary of retrieved document and suggested column names
- **Output:** Enhanced question and database Query
- **Database:** Reads schema, row value embeddings, row values and reviews the results of queries from tabular database.

SQL Agent

Input	What are some good restaurants near NY Aquarium? The address is 602 Surf Ave, Brooklyn, NY 11224. The relevant columns are address, name and rating
Enhanced Question	What is the highest rated restaurants near the NY Aquarium? The address is 602 Surf Ave, Brooklyn, NY 11224. Valid zip codes for your query are 11223 & 11225. ¹⁷
Output	<pre>SELECT id, name, rating, rating_count, price_category, address, zipcode FROM google_maps_restaurants WHERE zipcode IN ('11225', '11223') ORDER BY rating DESC</pre>

