1. [All Collections](https://help.openai.com/en)
2. [ChatGPT](https://help.openai.com/en/collections/3742473-chatgpt)
3. [GPTs](https://help.openai.com/en/collections/8475420-gpts)
4. [Building GPTs](https://help.openai.com/en/collections/8475422-building-gpts)
5. Retrieval Augmented Generation (RAG) and Semantic Search for GPTs

**Retrieval Augmented Generation (RAG) and Semantic Search for GPTs**

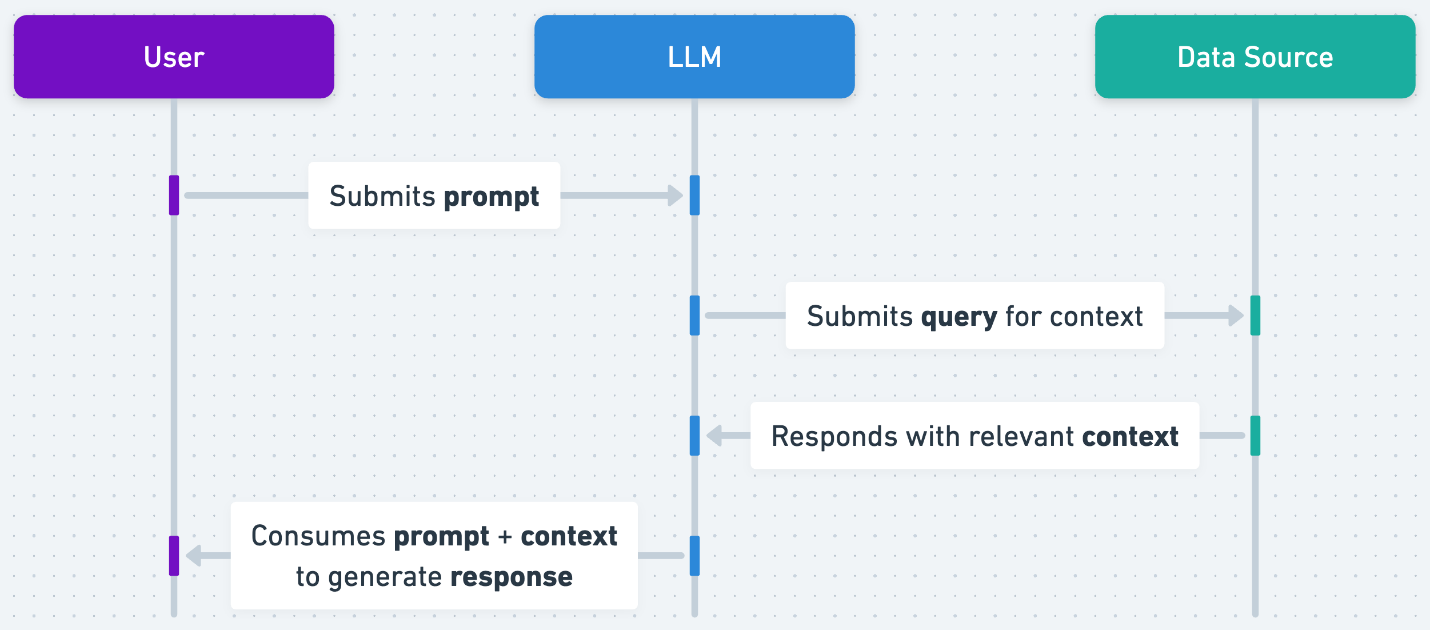
Learn about RAG and how it is useful to GPT builders

Updated: 17 days ago

**What is Retrieval Augmented Generation (RAG), and why is it valuable for GPT builders?**

Retrieval Augmented Generation (RAG) is a technique that improves a model’s responses by injecting external context into its prompt at runtime. Instead of relying solely on the model’s pre-trained knowledge, RAG retrieves relevant information from connected data sources and uses it to generate a more accurate and context-aware response.

In GPTs, RAG is performed **automatically** when **knowledge retrieval** is enabled and files have been uploaded. The model dynamically retrieves relevant information from those files to supplement the user’s prompt.



*Basic RAG workflow*

**Why is RAG valuable?**

RAG is especially helpful when your GPT needs to answer questions about content that isn’t part of its training data — such as company-specific documentation, internal processes, or recent events.

**Example:**

Imagine you're building a GPT to help your support team answer product questions. The base GPT model has broad general knowledge, but it doesn’t know your product’s latest update logs or help center content.

With RAG, your GPT can retrieve and use relevant internal support tickets or FAQs from uploaded files and respond using that custom knowledge — without you needing to hard-code every answer.

**What is Semantic Search?**

Semantic search is the method GPTs use to find relevant information across uploaded files. Unlike keyword search, which looks for exact word matches, semantic search finds conceptually similar content — even if the exact terms don’t match.

This is done using a **vector database**, where text is stored as embeddings (numerical representations of meaning). When a user asks a question, the GPT converts that question into a vector and compares it to the stored vectors, retrieving the most relevant text chunks.

|  |  |
| --- | --- |
| Data source | Search method |
| Document management systems (Google Drive, Sharepoint, etc.) | Keyword search, custom query string |
| Relational databases (Postgres, MySQL, etc.) | SQL query |
| Vector databases | Semantic search query |

**How does GPT knowledge retrieval work?**

When you upload files to a custom GPT and enable **knowledge retrieval**, the following happens behind the scenes:

1. **Chunking**: Files are automatically broken into smaller sections (e.g., paragraphs or logical blocks).
2. **Embedding**: Each chunk is converted into an embedding using OpenAI’s embedding models.
3. **Storage**: The embeddings are stored in OpenAI’s internal vector store.
4. **Querying**: When a user asks a question, the GPT creates a vector for the prompt and retrieves semantically similar chunks.
5. **Response generation**: The retrieved chunks are included as context in the GPT's prompt to generate a more informed answer.

You don’t need to manage a vector database manually — this all happens seamlessly within the GPT builder.

**Example Use Case**

If you’re building a customer support GPT, you might:

* Upload your knowledge base PDFs or internal wiki content
* Enable knowledge retrieval
* Let the GPT semantically search those documents and return helpful, accurate answers based on them

The GPT can now answer user questions like:

*“How can I reset my password?”*  
→ Using context from your own documentation — even if that exact question wasn’t trained into the base model.

**Summary**

* **RAG** boosts response quality by incorporating real-time knowledge from your files.
* **Semantic search** allows GPTs to retrieve conceptually relevant content, not just keywords.
* **GPTs with knowledge retrieval** automatically use these methods — no extra setup required beyond uploading your files.