# Chapter 8. Semantic Search and RAG

the fast adoption of text generation models led many users to ask the models questions and expect factual answers. And while the models were able to answer fluently and confidently, their answers were not always correct or up-to-date. This problem grew to be known as model “hallucinations,” and one of the leading ways to reduce it is to build systems that can retrieve relevant information and provide it to the LLM to aid it in generating more factual answers. This method, called RAG, is one of the most popular applications of LLMs.

Retrieval-Augmented Generation (RAG)
The mass adoption of LLMs quickly led to people asking them questions and expecting factual answers. While the models can answer some questions correctly, they also confidently answer lots of questions incorrectly. The leading method the industry turned to remedy this behavior is RAG

RAG systems incorporate search capabilities in addition to generation capabilities. They can be seen as an improvement to generation systems because they reduce their hallucinations and improve their factuality. They also enable use cases of “chat with my data” that consumers and companies can use to ground an LLM on internal company data, or a specific data source of interest (e.g., chatting with a book).

This also extends to search systems. More search engines are incorporating an LLM to summarize results or answer questions submitted to the search engine. Examples include Perplexity, Microsoft Bing AI, and Google Gemini.

Advanced RAG techniques:

- Query Re-writing : sometimes user might give a lengthy prompt(which can be simplified and reduced in size), this might make context remembering and giving correct responses, so we can re-write the query entirely to simplify the process.
- Multi Query : sometimes it is better to rewrite a single query into multiple queries, so that we can fetch better results overall, eg .

> User Question: “Compare the financial results of Nvidia in 2020 vs. 2023”

We may find one document that contains the results for both years, but more likely, we’re better off making two search queries:

> Query 1: “Nvidia 2020 financial results”Query 2: “Nvidia 2023 financial results”

We then present the top results of both queries to the model for grounded generation. An additional small improvement here is to also give the query rewriter the option to determine if no search is required and if it can directly generate a confident answer without searching.

- **Multi-hop RAG**

  A more advanced question may require a series of sequential queries. Take for example a question like:

  > User Question: “Who are the largest car manufacturers in 2023? Do they each make EVs or not?”

  To answer this, the system must first search for:

  > Step 1, Query 1: “largest car manufacturers 2023”

  Then after it gets this information (the result being Toyota, Volkswagen, and Hyundai), it should ask follow-up questions:

  > Step 2, Query 1: “Toyota Motor Corporation electric vehicles”Step 2, Query 2: “Volkswagen AG electric vehicles”Step 2, Query 3: “Hyundai Motor Company electric vehicles”

- **Query routing**

  An additional enhancement is to give the model the ability to search multiple data sources. We can, for example, specify for the model that if it gets a question about HR, it should search the company’s HR information

system (e.g., Notion) but if the question is about customer data, that it should search the customer relationship management (CRM) (e.g., Salesforce).

RAG Evaluation

There are still ongoing developments in how to evaluate RAG models. A good paper to read on this topic is “Evaluating verifiability in generative search engines” (2023), which runs human evaluations on different generative search systems.2

It evaluates results along four axes:

Fluency

Whether the generated text is fluent and cohesive.

Perceived utility

Whether the generated answer is helpful and informative.

Citation recall

The proportion of generated statements about the external world that are fully supported by their citations.

Citation precision

The proportion of generated citations that support their associated statements.

While human evaluation is always preferred, there are approaches that attempt to automate these evaluations by having a capable LLM act as a judge (called LLM-as-a-judge) and score the different generations along the different axes. Ragas is a software library that does exactly this. It also scores some additional useful metrics like:

Faithfulness

Whether the answer is consistent with the provided context

Answer relevance

How relevant the answer is to the question

The Ragas documentation site provides more details about the formulas to

actually calculate these metrics.