McKinsey & Company

What is retrieval-augmented generation (RAG)?

October 30, 2024 | Article

Retrieval-augmented generation, or RAG, is a process applied to large language models to make their outputs more relevant for the end user.

DOWNLOADS

Article (5 pages)

In recent years, large language models (LLMs) have made tremendous progress in their ability to generate content. But some leaders who hoped these models would increase business efficiency and productivity have been disappointed. Off-the-shelf generative AI (gen AI) tools have yet to live up to the considerable hype surrounding them. Why is that? For one thing, LLMs are trained on only the information that's available to the providers that build them. This can limit their utility in environments where a wider range of more nuanced, enterprise-specific knowledge is needed.

Get to know and directly engage with senior McKinsey experts on RAG.

Lareina Yee is a senior partner in McKinsey's Bay Area office, where **Michael Chui** is a senior fellow and **Roger Roberts** is a partner; **Mara Pometti** is a consultant in the London office; **Patrick Wollner** is a consultant in the Vienna office; and **Stephen Xu** is a senior director of product management in the Toronto office.

Retrieval-augmented generation, or RAG, is a process applied to LLMs to make their outputs more relevant in specific contexts. RAG allows LLMs to access and reference information outside the LLMs own training data, such as an organization's specific knowledge base, before generating a response—and, crucially, with citations included. This capability enables LLMs to produce highly specific outputs without extensive fine-tuning or training, delivering some of the benefits of a custom LLM at considerably less expense.

Consider a typical gen AI chatbot that's deployed in a customer service context. While it may offer some general guidance, because the chatbot is working from an LLM that was trained on only a specific amount of information, it's therefore not accessing the enterprise's unique policies, procedures, data, or knowledge base. As a result, its answers will lack specificity and relevance to a user's inquiry. For example, when a customer asks about the status of their account or payment options, the chatbot might respond with only generic information; because the chatbot isn't accessing the company's specific data, the response it gives doesn't consider that customer's specific situation.

Because RAG deployments have access to vast amounts of information that is more up to date and enterprise-specific, they can provide much more accurate, relevant, and coherent outputs. This is particularly helpful in applications and use cases that require highly accurate outputs, such as enterprise knowledge management and copilots that are specific to a given domain (for example, a workflow or process, journey, or function within the company).

Learn more about QuantumBlack, AI by McKinsey.

How does RAG work?

RAG involves two phases: ingestion and retrieval. To understand these concepts, it helps to imagine a large library with millions of books.

The initial “ingestion” phase is akin to stocking the shelves and creating an index of their contents, which allows a librarian to quickly locate any book in the library’s collection. As part of this process, a set of dense vector representations—numerical representations of data, also known as “embeddings” (for more, see sidebar, “What are embeddings?”)—is generated for each book, chapter, or even selected paragraphs.

Once the library is stocked and indexed, the “retrieval” phase begins. Whenever a user asks a question on a specific topic, the librarian uses the index to locate the most relevant books. The selected books are then scanned for relevant content, which is carefully extracted and synthesized into a concise output. The original question informs the initial research and selection process, guiding the librarian to present only the most pertinent and accurate information in response. This process might involve summarizing key points from multiple sources, quoting authoritative texts, or even generating new content based on the insights that can be gleaned from the library’s resources.

Through these ingestion and retrieval phases, RAG can generate highly specific outputs that would be impossible for traditional LLMs to produce on their own. The stocked library and index provide a foundation for the librarian to select and synthesize information in response to a query, leading to a more relevant and thus more helpful answer.

In addition to accessing a company's internal "library," many RAG implementations can query external systems and sources in real time. Examples of such searches include the following:

- *Database queries.* RAG can retrieve relevant data that are stored in structured formats, such as databases or tables, making it easy to search and analyze this information.
- *Application programming interface (API) calls.* RAG can use APIs to access specific information from other services or platforms.
- *Web search/scraping.* In some cases, RAG implementations can scrape web pages for relevant information, although this method is more prone to errors than others, due to the underlying data quality.

Which areas of the business stand to benefit from RAG systems?

RAG has far-reaching applications in various domains, including customer service, marketing, finance, and knowledge management. By integrating RAG into existing systems, businesses can generate outputs that are more accurate than they would be using an off-the-shelf LLM, which can improve customer satisfaction, reduce costs, and enhance overall performance. Here are some specific examples of where and how RAG can be applied:

- *Enterprise-knowledge-management chatbot.* When an employee searches for information within their organization's intranet or other internal knowledge sources, the RAG system can retrieve relevant information from across the organization, synthesize it, and provide the employee with actionable insights.

- *Customer service chatbots.* When a customer interacts with a company's website or mobile app to inquire about a product or service, the RAG system can retrieve relevant information based on corporate policies, customer account data, and other sources, then provide the customer with more accurate and helpful responses.
- *Drafting assistants.* When an employee starts drafting a report or document that requires company-specific data or information, the RAG system retrieves the relevant information from enterprise data sources, such as databases, spreadsheets, and other systems, then provides the employee with prepopulated sections of the document. This output can help the employee develop the document more efficiently and more accurately.

Learn more about QuantumBlack, AI by McKinsey.

What are some challenges associated with RAG?

While RAG is a powerful tool for enhancing an LLM's capabilities, it is not without its limitations. Like LLMs, RAG is only as good as the data it can access. Here are some of its specific challenges:

- *Data quality issues.* If the knowledge that RAG is sourcing is not accurate or up to date, the resulting output may be incorrect.
- *Multimodal data.* RAG may not be able to read certain graphs, images, or complex slides, which can lead to issues in the generated output.

New multimodal LLMs, which can parse complex data formats, can help mitigate this.

- *Bias.* If the underlying data contains biases, the generated output is likely to be biased as well.
- *Data access and licensing concerns.* Intellectual property, licensing, and privacy and security issues related to data access need to be considered throughout the design of a RAG system.

To help address these challenges, enterprises can establish data governance frameworks—or, if they already have them, ramp up those frameworks to help ensure the quality, accessibility, and timeliness of the underlying data used in RAG. Organizations that are implementing RAG systems should also carefully consider any copyright issues with respect to RAG-derived content, biases in the overall data set, and the level of interoperability between data sets that were not previously centrally accessible.

How is RAG evolving?

As RAG's capabilities and potential applications continue to evolve, we expect several emerging trends to shape its future:

- *Standardization.* The increasing standardization of underlying software patterns means that there will be more off-the-shelf solutions and libraries available for RAG implementations, making them progressively easier to build and deploy.
- *Agent-based RAG.* Agents are systems that can reason and interact with each other and require less human intervention than earlier AI systems. These tools can enable RAG systems to flexibly and

efficiently adapt to changing contexts and user needs so they can better respond to more complex and more nuanced prompts.

- *LLMs that are optimized for RAG.* Some LLMs are now being trained specifically for use with RAG. These models are tailored to meet the unique needs of RAG tasks, such as quickly retrieving data from a vast corpus of information, rather than relying solely on the LLM's own parametric knowledge. One example of these optimized LLMs is the AI-powered answer engine Perplexity AI, which has been fine-tuned to perform in various RAG applications (for example, answering complex questions and summarizing text).

LLMs enhanced with retrieval-augmented generation can harness the strengths of both humans and machines, enabling users to tap into vast knowledge sources and generate more accurate and relevant responses. As this technology continues to evolve, we expect significant improvements in its scalability, adaptability, and impact on enterprise applications, with the potential to spur innovation and create value.

Learn more about QuantumBlack, AI by McKinsey. And check out AI-related job opportunities if you're interested in working with McKinsey.

Articles referenced:

- "Why agents are the next frontier of generative AI," *McKinsey Quarterly*, July 24, 2024, Lareina Yee, Michael Chui, and Roger Roberts, with Stephen Xu
- "A data leader's technical guide to scaling gen AI," July 8, 2024, Asin Tavakoli, Carlo Giovine, Joe Caserta, Jorge Machado, and Kayvaun Rowshankish, with Jon Boorstein and Nathan Westby

- “Choose the right transformation ‘bite size’,” March 27, 2024, Eric Lamarre, Kate Smaje, and Rodney Zemmel

How relevant and useful is this article for you?

☆ ☆ ☆ ☆ ☆