

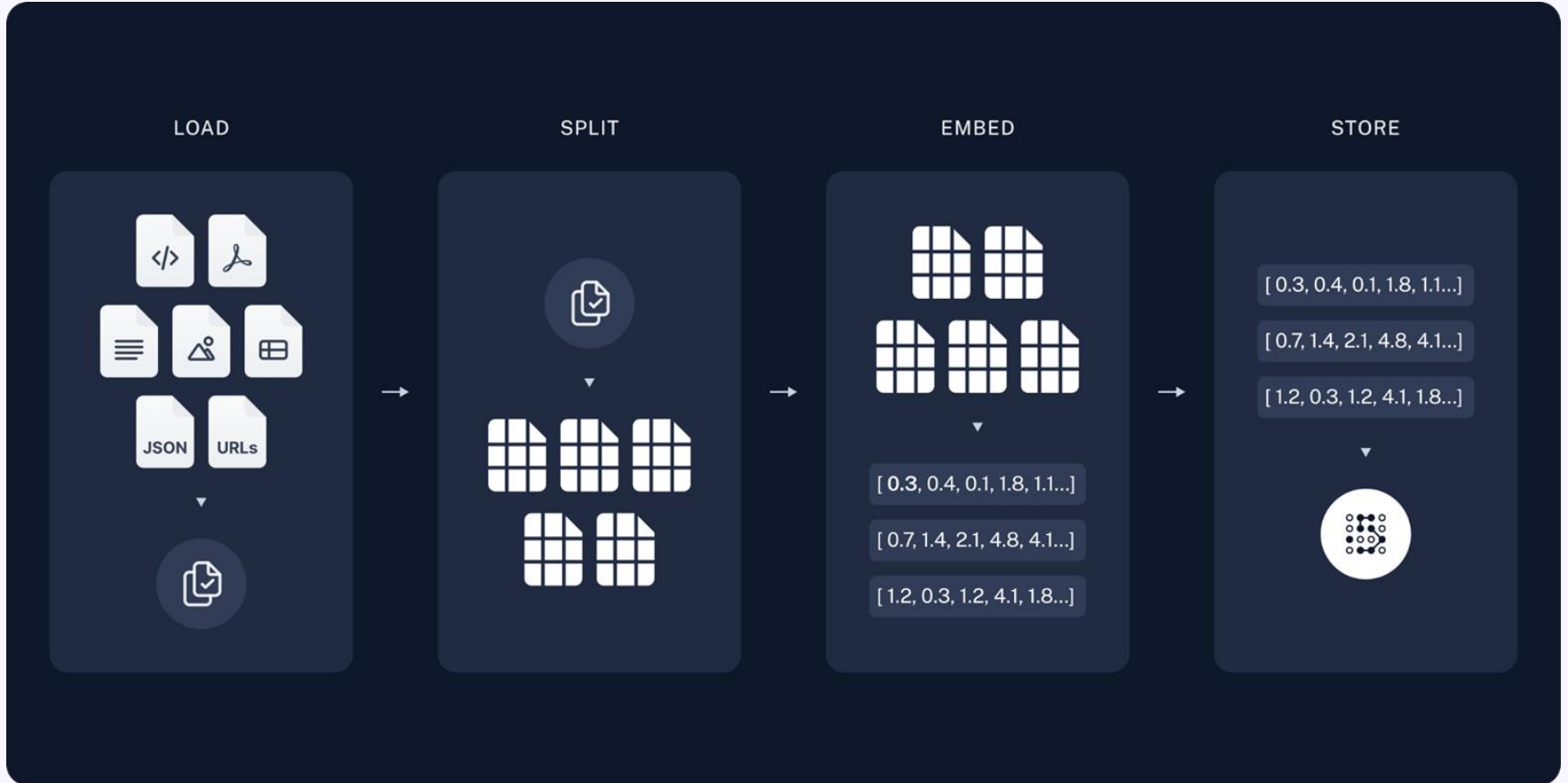
When Simple RAG Fails (and how to fix it)



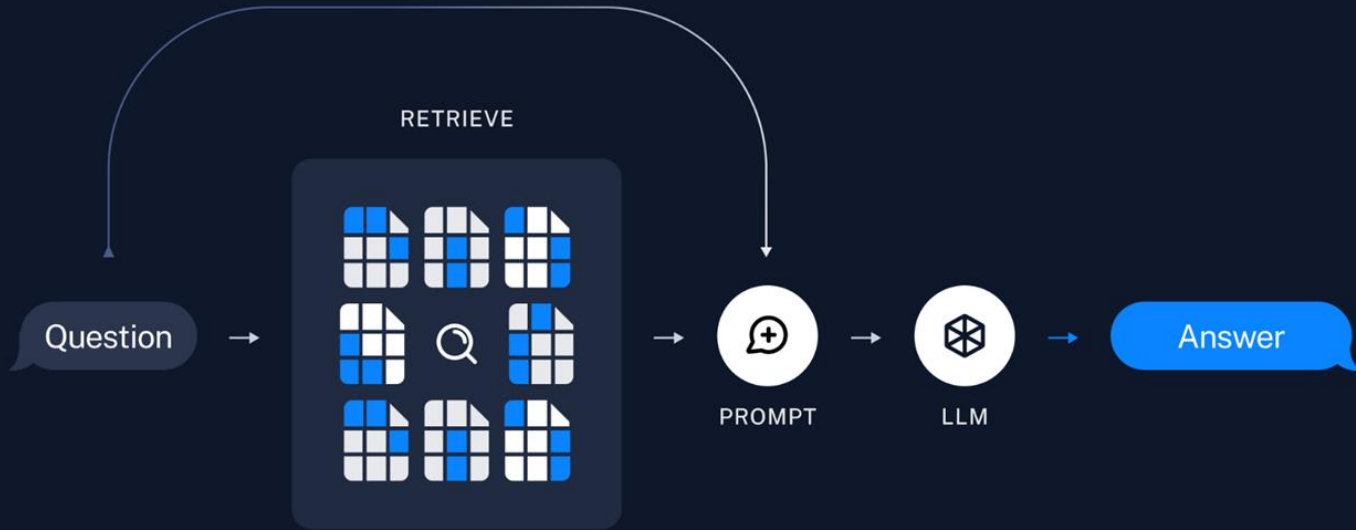
LangChain

The platform developers and enterprises choose to build AI apps from prototype to production.

Typical RAG Pipeline: Indexing



Typical RAG Pipeline: Querying





When does this fail?

Failure Modes

- Questions are not relevant to corpus
- Questions are vague
- Questions are not about fact retrieval
- Questions contain multiple sub questions
- Questions require multi-hop logic
- Questions include some non-semantic components
- Conflicting information

Questions are not relevant to corpus

Example:

- Asking ChatLangChain about classical music

Potential Fixes

- Prompting to make sure LLM stays on topic
- Some classification step/check before doing retrieval

Questions are vague

Example:

- “How do I build this?”

Potential Fixes

- Prompting to ask follow up questions
- Some classification step/check before doing retrieval

Questions are not about fact retrieval

Example:

- Asking for a summary of a PDF

Potential Fixes

- Precalculate summaries during indexing and route queries there
- “Just-in-time” summaries

Questions contain multiple sub questions

Example:

- “How do I use Pinecone to build a RAG Agent” is really “How do I use Pinecone” and “How do I build a RAG Agent”

Potential Fixes

- Generate multiple queries and look up all

Questions require multi-hop logic

Example:

- “Who won the 2023 super bowl and where was their head coach from?”

Potential Fixes

- Query planning
- Iterative retrieval

Questions include some non-semantic components

Example:

- “What are movies about aliens in 1980” - should filter by **year=1980**

Potential Fixes

- Index metadata along with the documents, and then use LLM to generate metadata filters

Conflicting information

Examples:

- Same information in two different Notion pages
- Things changing over time

Potential Fixes

- Assign an importance score to sources, retrieve based on importance, and include in the prompt
- Assign a timestamp to sources, retrieve based on timestamp, and include in the prompt
- Filter based on time



Use cases

Q&A with RAG

SQL

Tool use

CSV

Query analysis

Quickstart

Techniques

How-To Guides

Graphs

Interacting with APIs

Extraction

Chatbots

Code understanding

Synthetic data generation

Summarization

Tagging

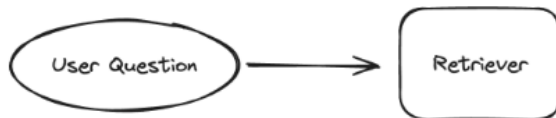
Web scraping

[Home](#) > [Use cases](#) > [Query analysis](#)

Query analysis

"Search" powers many use cases - including the "retrieval" part of Retrieval Augmented Generation. The simplest way to do this involves passing the user question directly to a retriever. In order to improve performance, you can also "optimize" the query in some way using *query analysis*. This is traditionally done by rule-based techniques, but with the rise of LLMs it is becoming more popular and more feasible to use an LLM for this. Specifically, this involves passing the raw question (or list of messages) into an LLM and returning one or more optimized queries, which typically contain a string and optionally other structured information.

Without Query Analysis



With Query Analysis



Problems Solved

Quickstart

Techniques

How to



Which technique should I use?



Look at your data!

How to decide what data points to look for?

Explicit Feedback

- Thumbs up/down from users

Implicit Feedback

- Did they click on a link?

LLM Eval

- Did the user say anything in the conversation to suggest if it was a good or bad response?



Thoughts on web3	Thoughts on web3
<p>AI: I'm sorry, but I cannot provide specific information on web3 based on the available source and chat history. If you would like, I can provide a general comment on advancements in technology and user experiences. Let me know if you would like me to provide that instead.</p> <p>1:18:53 PM</p>	<p>AI: Paul Graham's essay "Do things that don't scale" emphasizes the importance of focusing on delivering an excellent user experience, even if it means initially operating in ways that may seem inconsequential or laborious. This mindset is relevant when considering the concept of web3, as it highlights the significance of prioritizing user needs and satisfaction. In the context of web3 technologies, founders and developers should aim to create platforms that offer users an "insanely great experience" from the beginning, even if the product is still in its early stages. By concentrating on providing exceptional user experiences and being attentive to user feedback, web3 projects can optimize their path towards success and potentially dominate the market.</p> <p>1:18:53 PM</p>
<p>AI: I'm sorry, but I cannot provide specific information on web3 based on the available source and chat history. If you would like, I can provide a general comment on advancements in technology and user experiences. Let me know if you would like me to provide that instead.</p> <p>1:18:55 PM</p>	<p>AI: Paul Graham's essay "Do things that don't scale" emphasizes the importance of focusing on delivering an excellent user experience, even if it means initially operating in ways that may seem inconsequential or laborious. This mindset is relevant when considering the concept of web3, as it highlights the significance of prioritizing user needs and satisfaction. In the context of web3 technologies, founders and developers should aim to create platforms that offer users an "insanely great experience" from the beginning, even if the product is still in its early stages. By concentrating on providing exceptional user experiences and being attentive to user feedback, web3 projects can optimize their path towards success and potentially dominate the market.</p> <p>1:18:54 PM</p>

Left is better Tie Right is better

New Round

<https://www.ragarena.com/>



Thank you!