Un ChatGPT sur vos données, comment ça marche ? 🤔

Hugo Vassard

Touraine Tech 2024
Tours, Jeudi 8 février 2024

WHO AM I ?

Hugo VASSARD

#DataScience #DataEngineering

#IpponTechnologies

#France #Nantes

#LindyHop #Charleston

#RollerCoaster #Skydiving #BungeeJumping

WHAT IS THIS PRESENTATION ABOUT ?

How to make a ChatGPT on our documents ?

⇒ *This is called **RAG** : "Retrieval-Augmented Generation"*

WHAT IS THIS PRESENTATION ABOUT ?

I recently worked on several projects around RAG, including :

#1 : SalesGPT

Goal : Help sales team to process calls for tender

#2 : CoproGPT

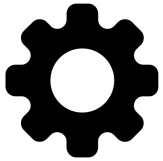
Goal : Find information in condominium general meeting minutes

#3 : Q&A chatbot 100% AWS (AWS GenAI contest)

Goal : Make a GenAI app with Bedrock (the AWS GenAI service)

WHAT IS THIS PRESENTATION ABOUT ?

How such an app works ?



How to build it ?

Figures, tips and +

BUT BEFORE THAT ...

DEMO !

NOW :
SOME EXPLANATIONS ON HOW THIS WORKS

THE CORE TOOL OF THIS DEMO

LangChain

Framework to easily build LLM applications

Creation : Oct. 2022

Open Source

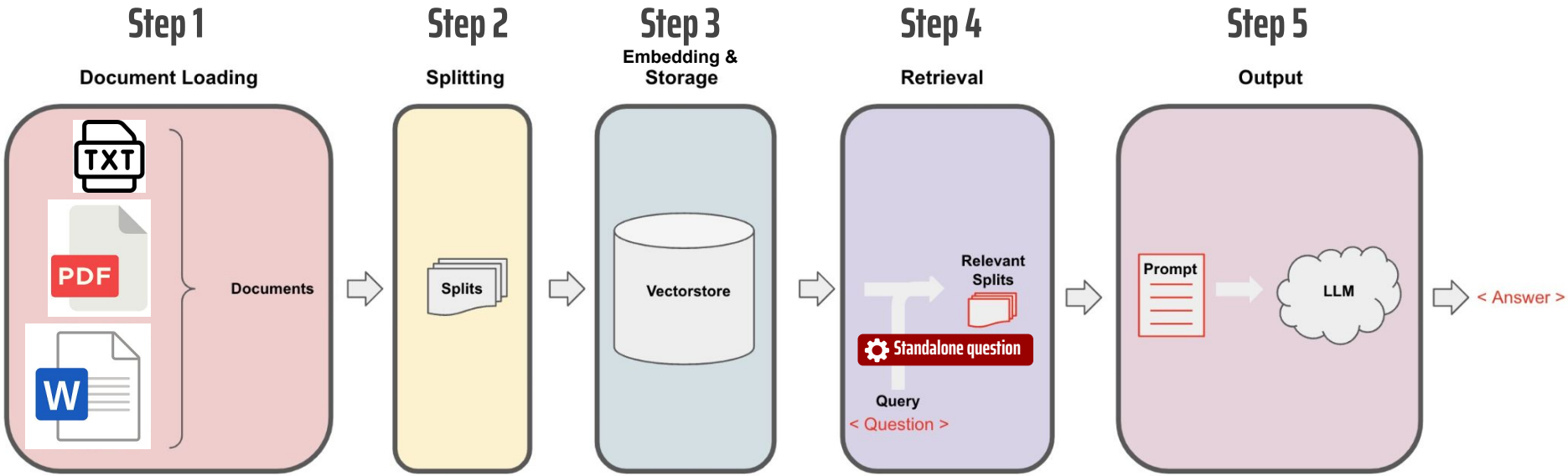
Integration with

Common API to easily replace a component

Ex : ~~GPT~~ ⇒ Llama2

Summarization
Chatbot
Document question answering
Structured data analysis

THE WHOLE WORKFLOW FOR DOCUMENT Q&A



Source : https://python.langchain.com/docs/use_cases/question_answering/

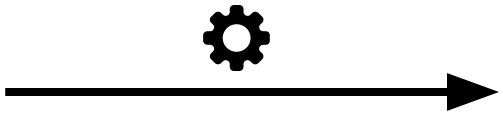
BUT LET'S DISCOVER IT STEP BY STEP

EXPLANATION #1

What happens during the upload of the files ?

STEP 1 : DOCUMENT LOADING

Your file
(.pdf, .txt, .docx ...)



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod teapor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod teapor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae

Extracted text

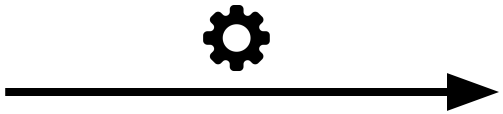
+

```
{
  "source_file" : "file1.txt",
  …
}
```

Metadata

STEP 1 : DOCUMENT LOADING

Your file
(.pdf, .txt, .docx ...)



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae

```
Document(
        page_content = "Lorem ipsum ...",
        metadata = {
                "source_file" : "file1.txt",
                ...
        }
)
```

🦜🔗 LangChain **Document object**

1 Document = extracted text + metadata

STEP 1 : DOCUMENT LOADING

file1.txt

Loading

```
Document(
  page_content = "Lorem ipsum ...",
  metadata = {
    "source_file" : "file1.txt",
    ...
  }
)
```

Document 1

file2.pdf

Loading

```
Document(
  page_content = "Lorem ipsum ...",
  metadata = {
    "source_file" : "file2.pdf",
    ...
  }
)
```

Document 2

file3.docx

Loading

```
Document(
  page_content = "Lorem ipsum ...",
  metadata = {
    "source_file" : "file3.docx",
    ...
  }
)
```

Document 3

How to perform the loading ?

🦜🔗 LangChain

Document Loaders

- .txt ⇒ TextLoader()
- .pdf ⇒ UnstructuredPDFLoader()
- .docx ⇒ Docx2txtLoader()

STEP 1 : DOCUMENT LOADING

LangChain

160+ Document loaders

Including :

Microsoft Azure
Blob Storage

Confluence

Airbyte

git

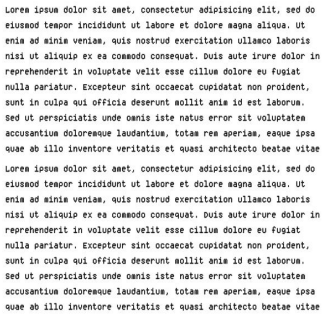
arXiv

Google
BigQuery

Figma

STEP 2 : DOCUMENT SPLITTING

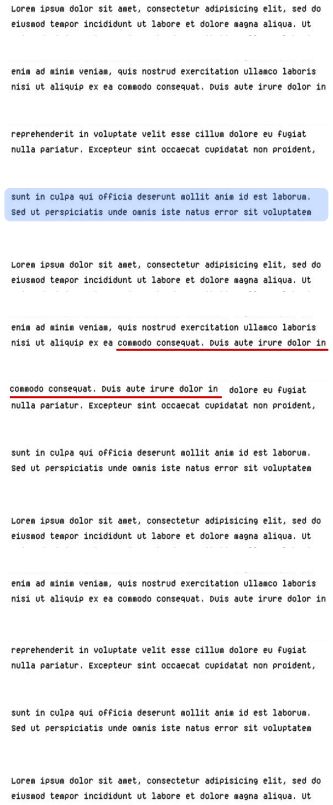
page_content of the Document object



Split into chunks

Split config

```
chunk_size = 50
chunk_overlap = 20
```



1 chunk of size 50 char.

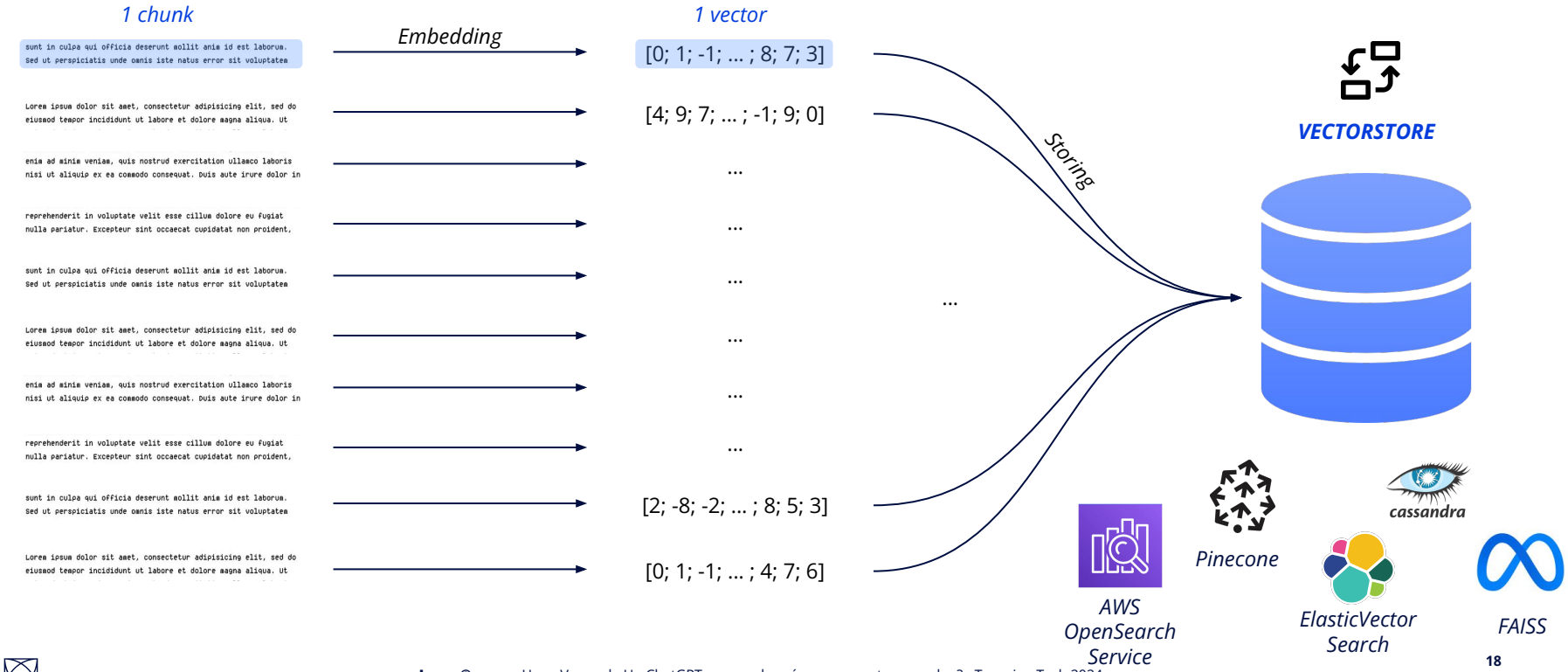
Overlap of 20 char. between 2 consecutive chunks

chunks created

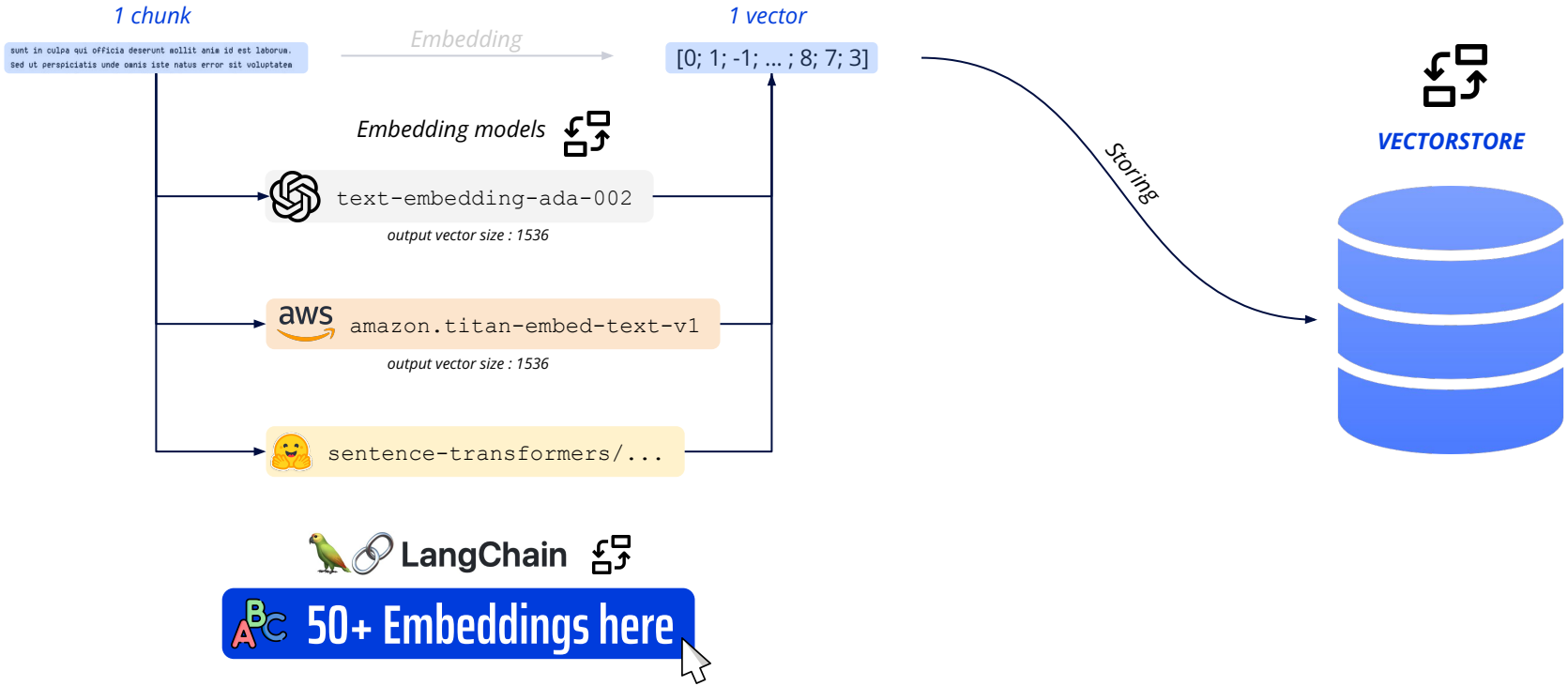
STEP 3 : EMBEDDING AND STORING

LangChain

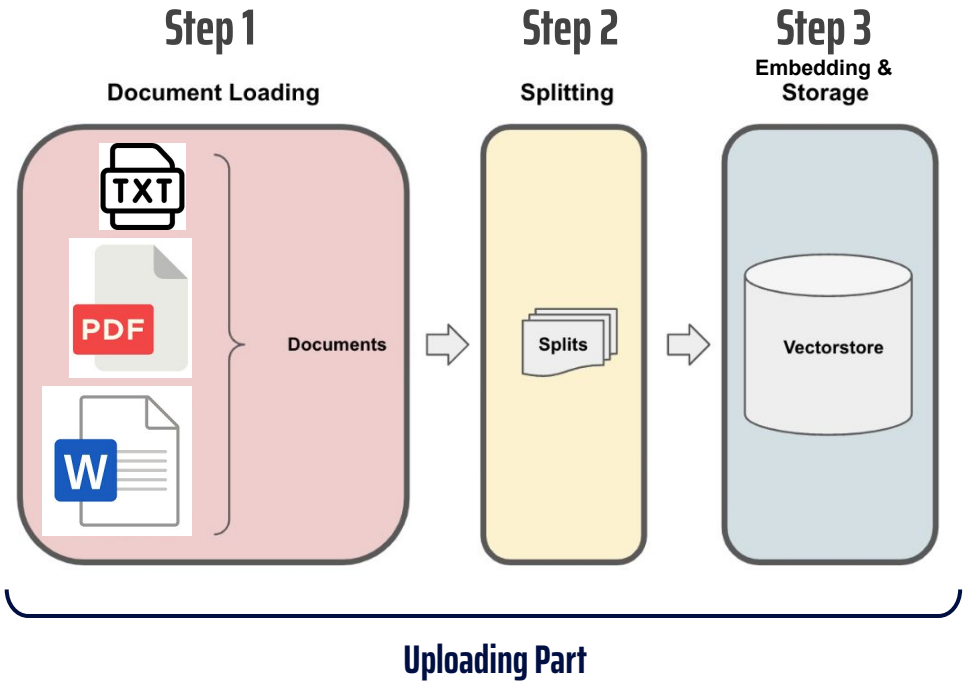
65+ Vectorstores here



STEP 3 : EMBEDDING AND STORING



WHAT WE HAVE SEEN SO FAR



EXPLANATION #2

How to get an answer ?

HERE IT COMES !

LLM

Most known : GPT, Llama 2, Claude 2, ...

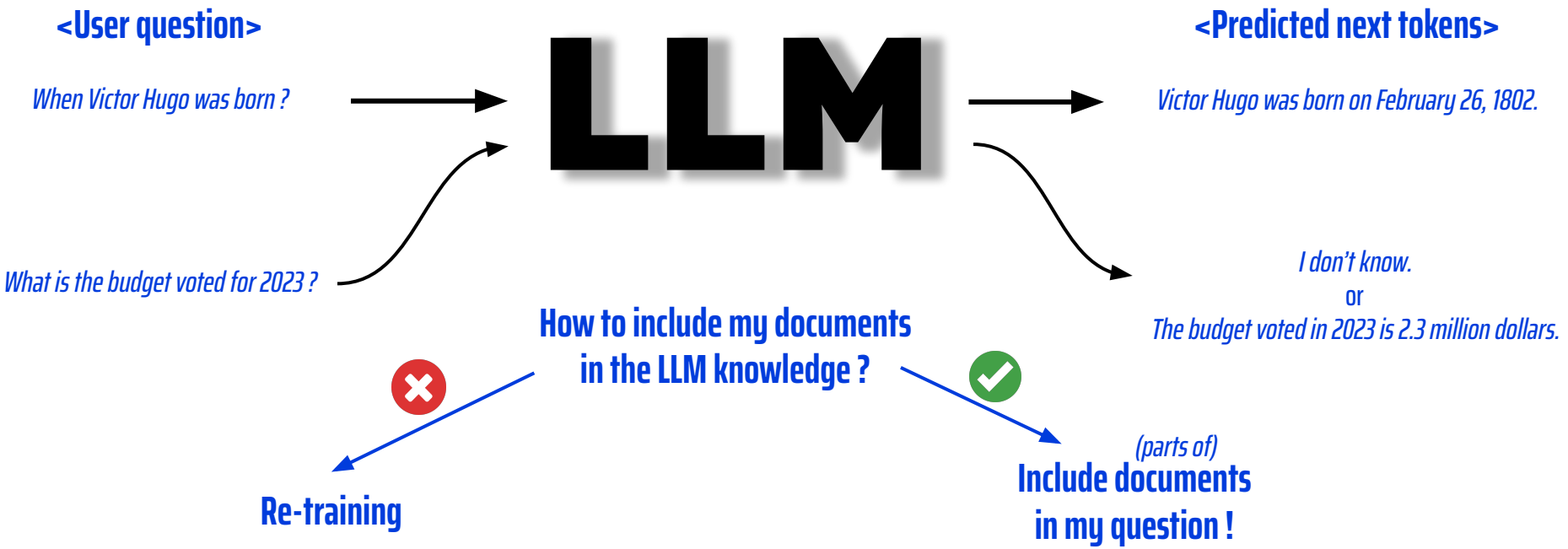
Belongs to GenAI tools

Predict next tokens

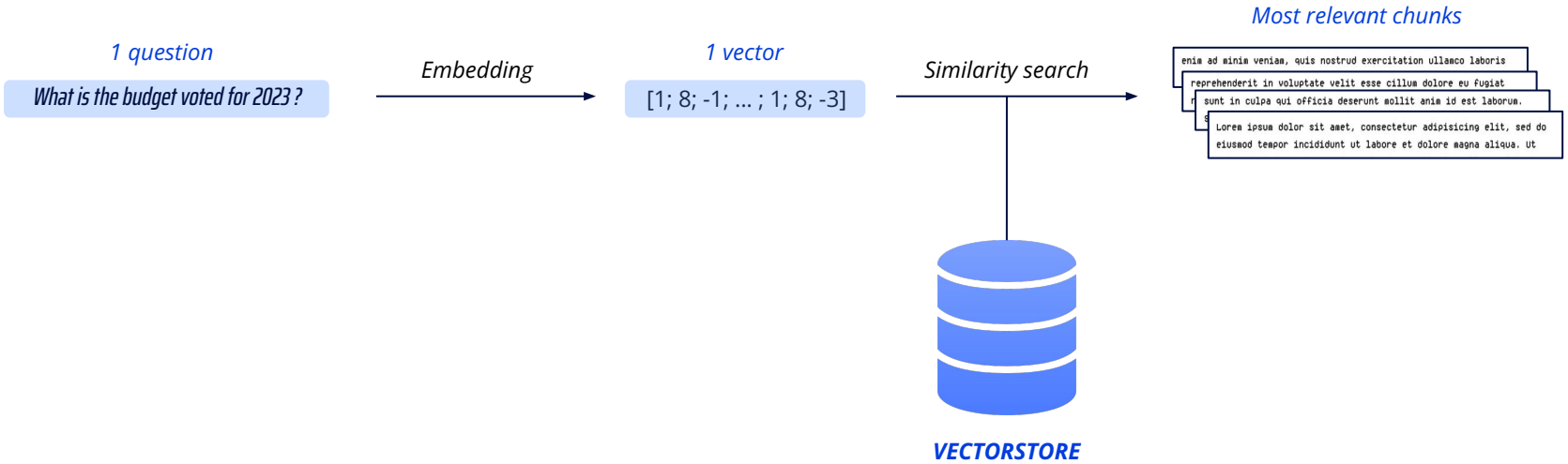
Large Language Model

Trained on a VAST AMOUNT of text

A FAMOUS LLM USAGE : ChatGPT

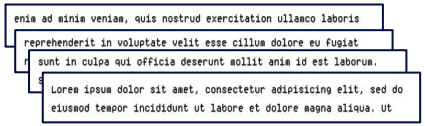


STEP 4 : RETRIEVE RELEVANT CHUNKS



STEP 5 : GET AN ANSWER

Most relevant chunks (context documents)



User question

What is the budget voted for 2023 ?

Template of question

TASK:
You are a chatbot that answers questions about files that a user has given you. The most relevant parts of these files will be provided to you as context documents. You MUST NOT provide any information unless it is written in the context documents

CONTEXT DOCUMENTS:
{context}

INSTRUCTION:
Generate the response, written in the user language, based on the context documents.

QUESTION:
{question}

ANSWER:

Prompt

Final (big) question to send to the LLM

TASK:
You are a chatbot that answers questions about files that a user has given you. The most relevant parts of these files will be provided to you as context documents. You MUST NOT provide any information unless it is written in the context documents

CONTEXT DOCUMENTS:

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut

sunt in culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem

reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,

enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in

INSTRUCTION:
Generate the response, written in the user language, based on the context documents.

QUESTION:

What is the budget voted for 2023 ?

ANSWER:

STEP 5 : GET AN ANSWER

Final (big) question to send to the LLM

TASK:
You are a chatbot that answers questions about files that a user has given you. The most relevant parts of these files will be provided to you as context documents. You MUST NOT provide any information unless it is written in the context documents

CONTEXT DOCUMENTS:

```
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do
eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut
```

```
sunt in culpa qui officia deserunt mollit anim id est laborum.
Sed ut perspiciatis unde omnis iste natus error sit voluptatem
```

```
reprehenderit in voluptate velit esse cillum dolore eu fugiat
nulla pariatur. Excepteur sint occaecat cupidatat non proident,
```

```
enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in
```

INSTRUCTION:
Generate the response, written in the user language, based on the context documents.

QUESTION:

What is the budget voted for 2023 ?

ANSWER:

LangChain

70+ LLM providers here

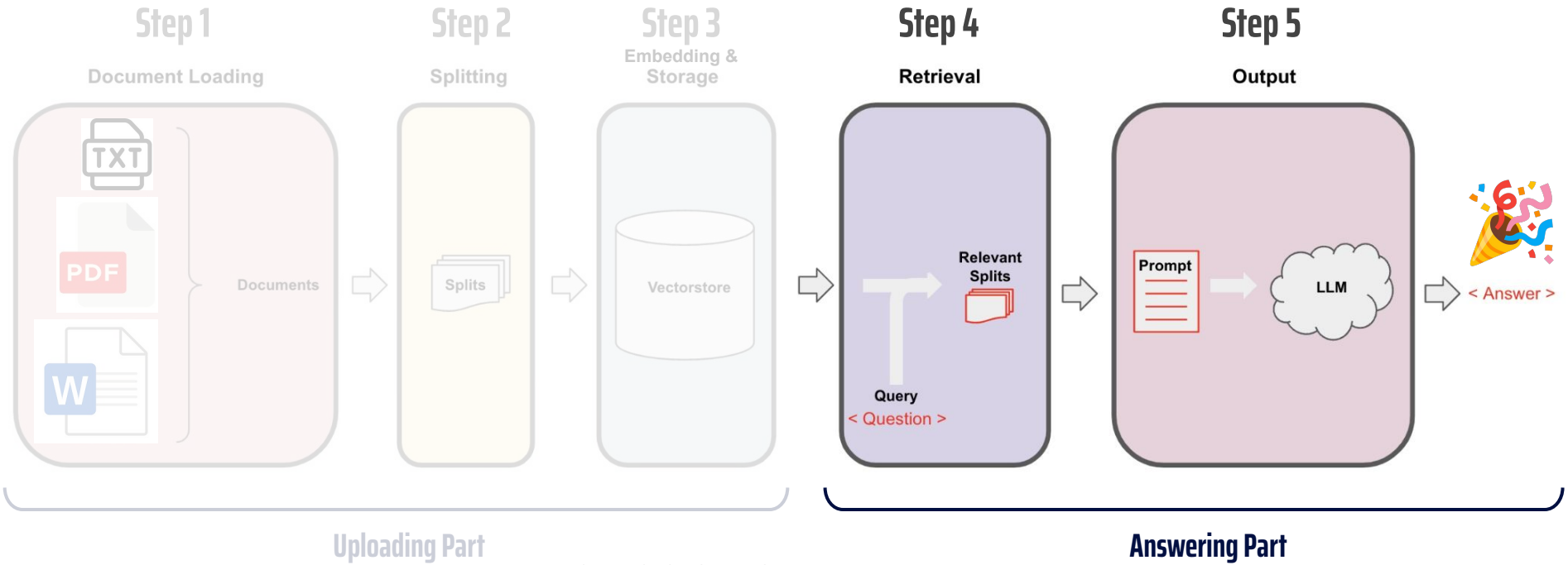
LLM

<Predicted next tokens>

The budget voted for 2023 is 18,500€.

OpenAI *AWS Bedrock* *Azure OpenAI* *Cohere* *Hugging Face Hub* *Anthropic*

WHAT WE HAVE SEEN SO FAR



Source : https://python.langchain.com/docs/use_cases/question_answering/

4

A bit of tech, figures and advices

1 - COSTS

There are 2 main costs : IA & Infrastructure

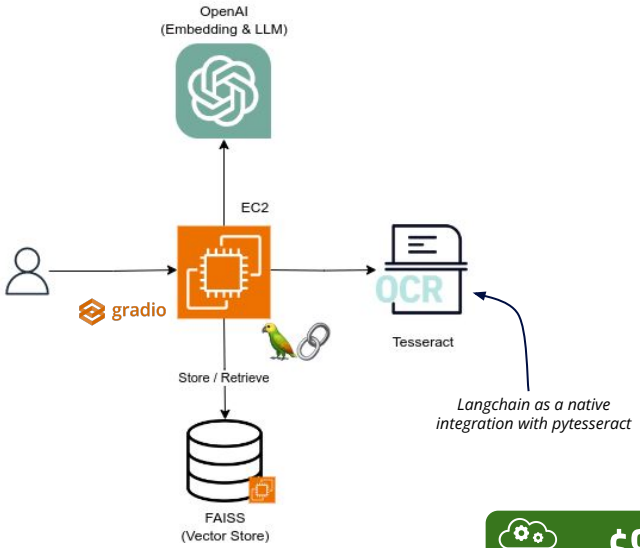
- **IA :** Exact amount is almost impossible to predict

  ⇒ I made a calculator to have an estimate

- **Infrastructure :** Costs depend a lot on the size of the project, nb of users, volumes ...

Note : We could also host our own open source LLM and embedding model to avoid third party ML API costs

2 - ARCHITECTURES



Architecture #1 - Simple Architecture

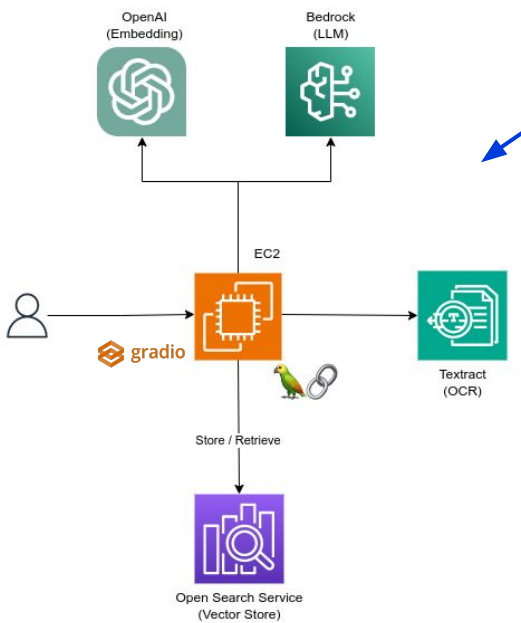
~ $800/year

with EC2 t3.large up 24h/24 7j/7
(2vCPUs, 8GiB Memory, $0.096/hr)

~ $2500/year

with the same EC2 +
OpenSearch Cluster ($1700/yr)
(r6gd.large.search, $0.1910/hr)

The one used in this demo



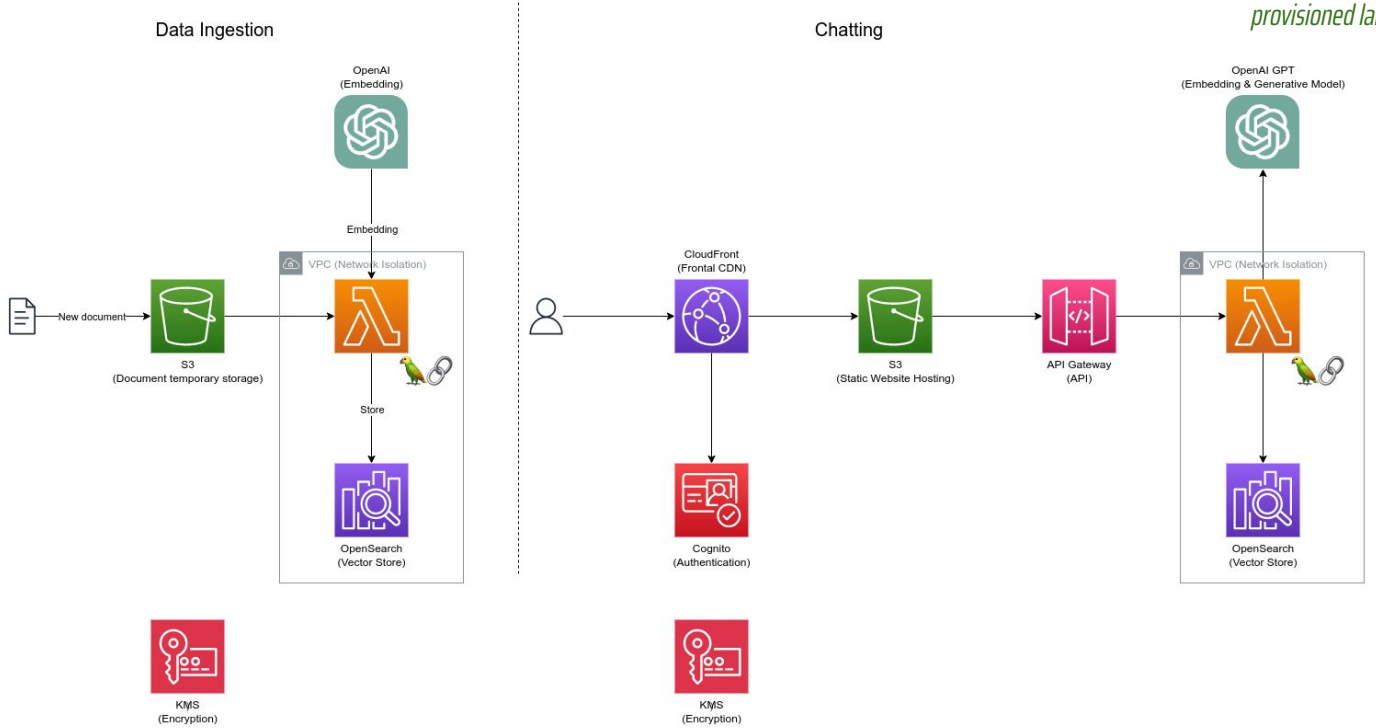
Architecture #2 - With more AWS Services

2 - ARCHITECTURES

~ $3000/year

30 users, 1000 new documents/month, 1000 questions/month

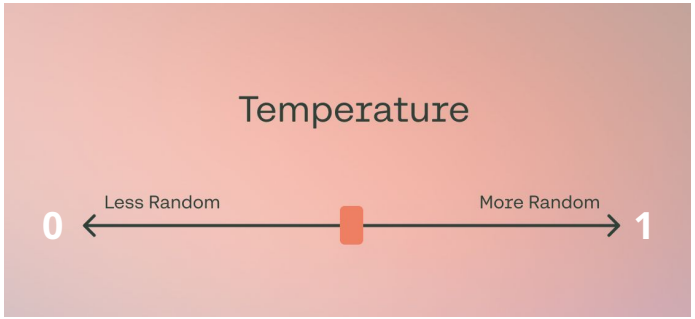
With the same OpenSearch Cluster ($1700/yr) + provisioned lambda ($800/yr) + other services



Architecture #3 - A more evolved architecture

3 - TEMPERATURE OF THE LLM

Deterministic.



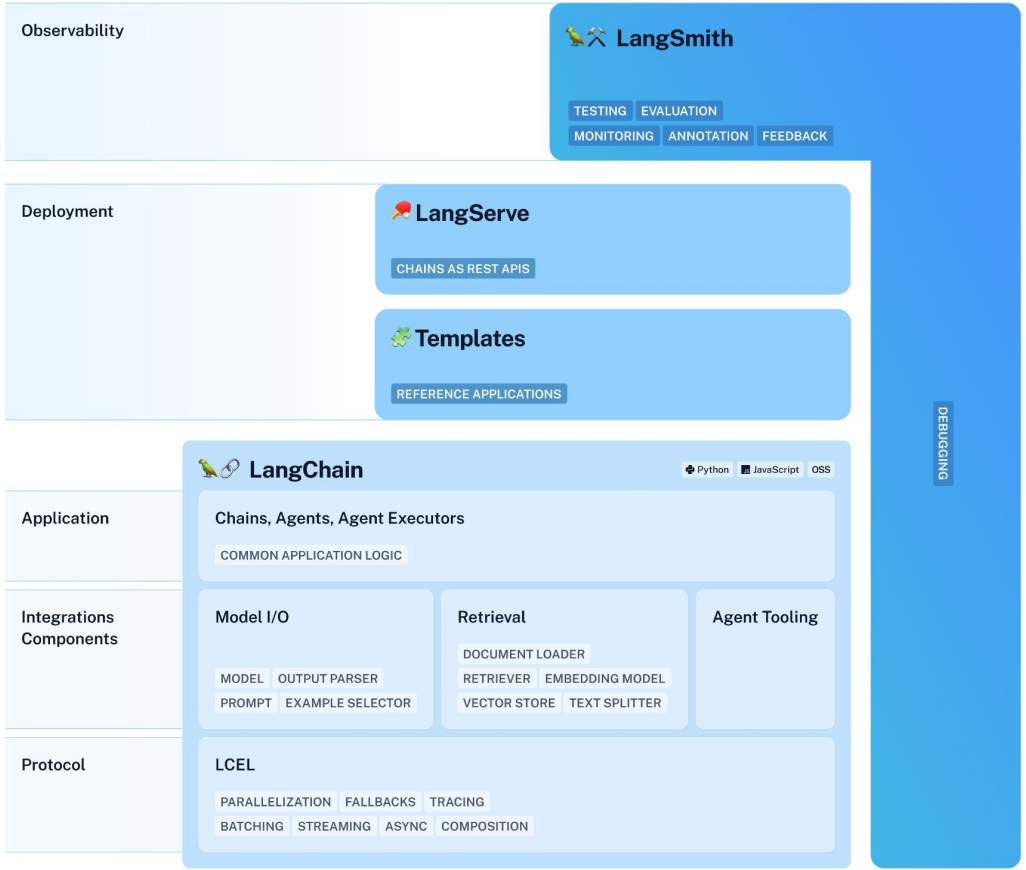
Source : https://txt.cohere.com/llm-parameters-best-outputs-language-ai/

Tip :

T=0.7 is commonly used and works well

4 - USEFUL TOOLS FOR PRODUCTION

LangServe / LangSmith :
Released in Oct. 2023



https://python.langchain.com/docs/get_started/introduction

04 — CONCLUSION / FEEDBACK

CONCLUSION / FEEDBACK

- A LOT OF **GOOD POINTS**
  - → 💡 REALLY **INTERESTING PROJECT & TECHNOLOGIES**
  - → 💼 GREAT **ADAPTABILITY OF USE CASES**
  - → 💯 WORKS **VERY WELL** (DEPENDS A LOT ON THE MODELS YOU CHOOSE)
  - → 💪 LANGCHAIN IS A GREAT TOOL
    - → **VARIOUS EXCHANGEABLE** COMPONENTS
    - → **VARIOUS TASKS** : Q&A OVER (UN)STRUCTURED DATA, SUMMARIZATION, TAGGING, ...

CONCLUSION / FEEDBACK

- ON THE **NEGATIVE SIDE...**
  - → ☐ VERY **POOR RESULTS** WITH SOME MODELS
  - → 👶 TOOLS ARE VERY **YOUNG** (LANGCHAIN JUST RELEASED 0.1)
  - → 🤦 RAG IS **NOT DESIGNED** FOR SUMMARIZATION (BUT IT CAN GET AWAY WITH IT FOR SMALL DOCUMENTS)
  - → 💸 EXACT ML COSTS IS **UNPREDICTABLE** (ONLY AN ESTIMATE)
  - → 📉 SOME MODEL PROVIDERS **WITHOUT SLA** (FOR THE MOMENT)
  - → 📑 HOW MY DATA ARE USED ? ⇒ **READ THE EULA** OF THE MODELS
  - → ⚠️ WEAKNESS : **PROMPT INJECTION** See Gandalf Game

Hugo Vassard

Un ChatGPT sur vos données, comment ça marche ? 🥵 ⇒ 😎

Open Feedback

⇒ Maintenant vous sachez !

Aujourd'hui / demain ?

ippon
POSITIVE TECHNOLOGY

STAND IPPON

On échange ?

Plus tard ?

APPENDIX

More information and figures

EXPLANATION #3

How the chatbot can “remember” previous messages ?

HOW TO REMEMBER PREVIOUS MESSAGES

Prompt sent to the LLM

TASK:
You are a chatbot that answers questions about files that a user has given you. The most relevant parts of these files will be provided to you as context documents. You MUST NOT provide any information unless it is written in the context documents

CONTEXT DOCUMENTS:

```
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do
eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut
```

```
sunt in culpa qui officia deserunt mollit anim id est laborum.
Sed ut perspiciatis unde omnis iste natus error sit voluptatem
```

```
reprehenderit in voluptate velit esse cillum dolore eu fugiat
nulla pariatur. Excepteur sint occaecat cupidatat non proident,
```

```
enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in
```

INSTRUCTION:
Generate the response, written the user language, based on the context documents.

QUESTION:

What is the budget voted for 2023 ?

ANSWER:

May work (I didn't try)

Add chat history in the prompt

LangChain implementation choice

Rephrase the question into a standalone question

Here's a 1st question and its answer :

What is the budget voted for 2023 ?

The budget voted for 2023 is $12,500.

Now, you could ask ...

How many people voted against it ? ← This **is not** a standalone question (we don't know what "it" means)

How many people voted against the 2023 budget ? ← This **is** a standalone question (we don't need previous messages)

~~HOW TO REMEMBER PREVIOUS MESSAGES~~

HOW TO REPHRASE AUTOMATICALLY ?

LLM

HOW TO REPHRASE AUTOMATICALLY ?

Tip :
You are not obliged to use the same LLM for rephrasing and answering

Prompt for rephrasing the user question

TASK :
Given the following conversation and a follow-up question, rephrase the follow-up question to be a standalone question, in its original language.

Chat history :

What is the budget voted for 2023 ?

The budget voted for 2023 is $12,500.

Follow-Up Input:

How many people voted against it ?

Standalone question :

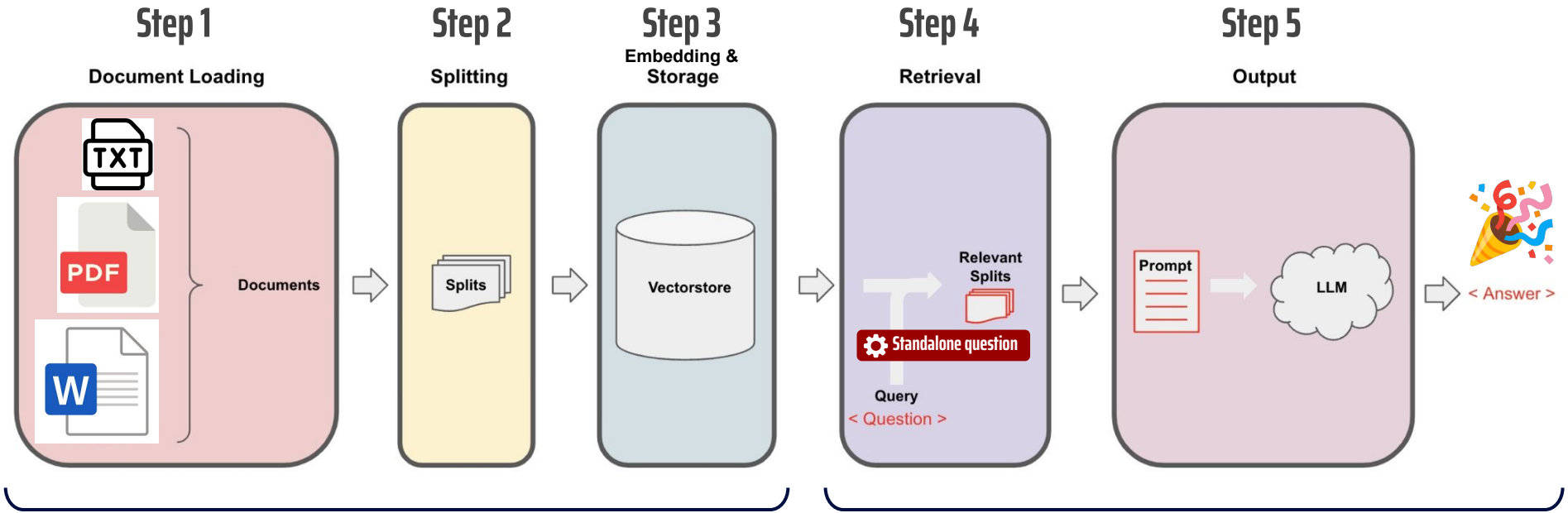
LLM

<Predicted next tokens>

How many people voted against the 2023 budget ?

Now use this standalone question in the previous LLM prompt !

WHAT WE HAVE SEEN SO FAR



Source : https://python.langchain.com/docs/use_cases/question_answering/