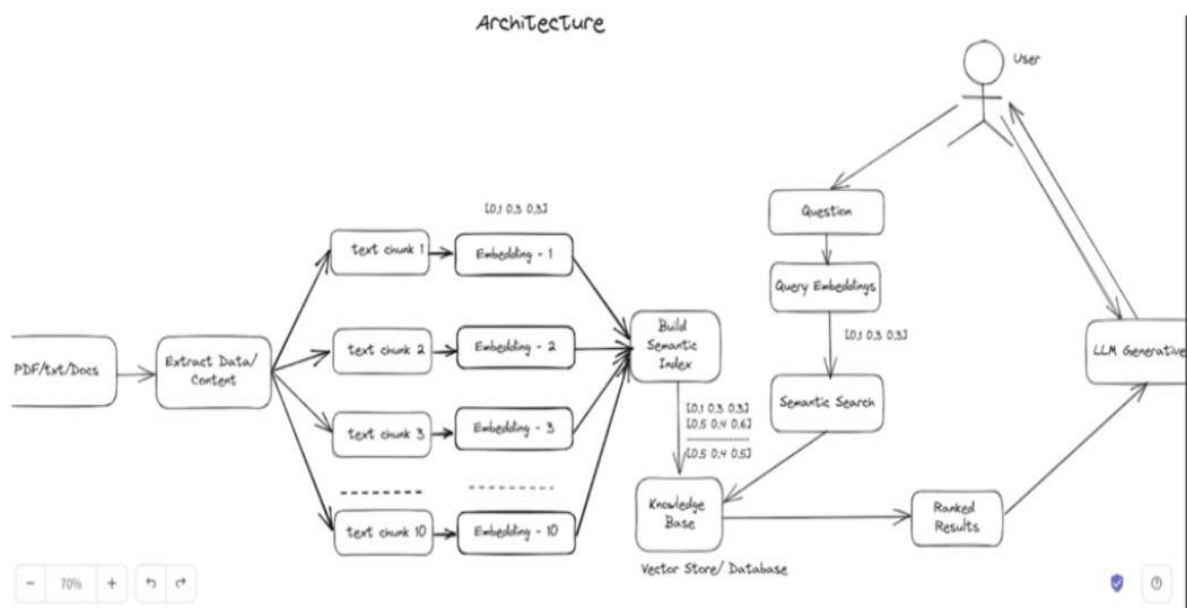


##RAG(Reterival Augemented Generation)

0.1 Architecture:

1. Data Ingestion
2. Data Reterival
3. Data Generation



1

1.1 Installing the necessary libraries

```
pip install langchain openai tiktoken rapidocr-onnxruntime
```

1.2 Fetching OpenAI API key

```
[2]: from google.colab import userdata
OPENAI_API_KEY=userdata.get("OPENAI_API_KEY")
```

1.3 Setting Enviornment Variable

```
[3]: import os
os.environ["OPENAI_API_KEY"] = OPENAI_API_KEY
```

2 Data Ingestion

```
[4]: import requests
      from langchain.document_loaders import TextLoader
      from langchain.vectorstores import FAISS

[5]: url = "https://raw.githubusercontent.com/langchain-ai/langchain/master/docs/
      ↪docs/modules/state_of_the_union.txt"

[6]: response=requests.get(url)

[7]: response

[7]: <Response [200]>

[10]: rawdata=response.text

[11]: with open("state_of_the_union.txt","w") as f:
      f.write(rawdata)

[13]: loader=TextLoader("/content/state_of_the_union.txt")

[14]: document=loader.load()

[ ]: print(document[0].page_content)
```

3 Chunking of the Data

4 Here is all the text splitter which is available in Langchain

https://python.langchain.com/docs/modules/data_connection/document_transformers/

4.1 CharacterTextSplitter v/s RecursiveCharacterTextSplitter

4.2 you can visualise the chunking also

<https://chunkviz.up.railway.app/>

```
[18]: from langchain.text_splitter import RecursiveCharacterTextSplitter

[19]: text_splitter=RecursiveCharacterTextSplitter(chunk_size=500,chunk_overlap=50)

[20]: text_chunks=text_splitter.split_documents(document)

[ ]: text_chunks

[24]: print(text_chunks[3].page_content)
```

Please rise if you are able and show that, Yes, we the United States of America stand with the Ukrainian people.

Throughout our history we've learned this lesson when dictators do not pay a price for their aggression they cause more chaos.

They keep moving.

And the costs and the threats to America and the world keep rising.

That's why the NATO Alliance was created to secure peace and stability in Europe after World War 2.

```
[25]: from langchain.embeddings import OpenAIEmbeddings
      from langchain.vectorstores import FAISS
```

```
[26]: embeddings=OpenAIEmbeddings(openai_api_key=OPENAI_API_KEY)
```

```
/usr/local/lib/python3.10/dist-packages/langchain_core/_api/deprecation.py:117:
LangChainDeprecationWarning: The class
`langchain_community.embeddings.openai.OpenAIEmbeddings` was deprecated in
langchain-community 0.0.9 and will be removed in 0.2.0. An updated version of
the class exists in the langchain-openai package and should be used instead. To
use it run `pip install -U langchain-openai` and import as `from
langchain_openai import OpenAIEmbeddings`.
warn_deprecated(
```

```
[ ]: !pip install faiss-cpu
```

```
[31]: vectorstore=FAISS.from_documents(text_chunks, embeddings)
```

```
[32]: retriever=vectorstore.as_retriever()
```

```
[33]: from langchain.prompts import ChatPromptTemplate
```

```
[34]: template="""You are an assistant for question-answering tasks.
      Use the following pieces of retrieved context to answer the question.
      If you don't know the answer, just say that you don't know.
      Use ten sentences maximum and keep the answer concise.
      Question: {question}
      Context: {context}
      Answer:
      """
```

```
[35]: prompt=ChatPromptTemplate.from_template(template)
```

```
[36]: from langchain.chat_models import ChatOpenAI
      from langchain.schema.runnable import RunnablePassthrough
      from langchain.schema.output_parser import StrOutputParser
```

```
[39]: output_parser=StrOutputParser()
```

```
[37]: llm_model=ChatOpenAI(openai_api_key=OPENAI_API_KEY,model_name="gpt-3.5-turbo")
```

/usr/local/lib/python3.10/dist-packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning: The class `langchain_community.chat_models.openai.ChatOpenAI` was deprecated in langchain-community 0.0.10 and will be removed in 0.2.0. An updated version of the class exists in the langchain-openai package and should be used instead. To use it run `pip install -U langchain-openai` and import as `from langchain_openai import ChatOpenAI`.

```
warn_deprecated(
```

```
[40]: rag_chain = (
      {"context": retriever, "question": RunnablePassthrough()}
      | prompt
      | llm_model
      | output_parser
      )
```

```
[41]: rag_chain.invoke("How is the United States supporting Ukraine economically and_
      ↪militarily?")
```

[41]: 'The United States is supporting Ukraine economically and militarily by providing military assistance, economic assistance, and humanitarian assistance. They are giving more than \$1 Billion in direct assistance to Ukraine. US forces are not engaged and will not engage in conflict with Russian forces in Ukraine. The world is choosing the side of peace and security in the battle between democracy and autocracy. The NATO Alliance was created to secure peace and stability in Europe after World War 2. Ukrainian citizens are showing incredible resilience and bravery in defending their homeland. President Zelenskyy expressed confidence that "Light will win over darkness" in the struggle. The United States is sending a clear signal of support to Ukraine and the world.'

```
[42]: rag_chain.invoke("What action is the U.S. taking to address rising gas prices?")
```

[42]: "The U.S. is addressing rising gas prices by releasing 30 million barrels from the Strategic Petroleum Reserve and is prepared to do more if needed, in coordination with allies. This action aims to help reduce gas prices domestically. The U.S. is taking steps to ensure that the impact of sanctions on Russia's economy is felt, protecting American businesses and consumers. Additionally, the United States has collaborated with 30 other countries to release 60 million barrels of oil globally. There is a focus on increasing

domestic production, creating more jobs, and improving infrastructure to combat inflation. The plan to fight inflation involves lowering costs, not wages, and increasing the productive capacity of the economy. This approach is supported by economists, business leaders, and the majority of Americans."

[]: