Build RAG pipeline

```
pip install langchain langchain-community faiss-cpu sentence-transformers transformers
```

```
Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.whl (207.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 207.5/207.5 MB 6.2 MB/s eta 0:00:00
Downloading nvidia_nccl_cu12-2.21.5-py3-none-manylinux2014_x86_64.whl (188.7 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 188.7/188.7 MB 6.0 MB/s eta 0:00:00
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 21.1/21.1 MB 64.5 MB/s eta 0:00:00
Downloading marshmallow-3.26.1-py3-none-any.whl (50 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 50.9/50.9 kB 4.3 MB/s eta 0:00:00
Downloading python_dotenv-1.1.1-py3-none-any.whl (20 kB)
Downloading typing_inspect-0.9.0-py3-none-any.whl (8.8 kB)
Downloading mypy_extensions-1.1.0-py3-none-any.whl (5.0 kB)
Installing collected packages: python-dotenv, nvidia-nvjitlink-cu12, nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-c
  Attempting uninstall: nvidia-nvjitlink-cu12
    Found existing installation: nvidia-nvjitlink-cu12 12.5.82
    Uninstalling nvidia-nvjitlink-cu12-12.5.82:
      Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82
  Attempting uninstall: nvidia-nccl-cu12
    Found existing installation: nvidia-nccl-cu12 2.23.4
    Uninstalling nvidia-nccl-cu12-2.23.4:
      Successfully uninstalled nvidia-nccl-cu12-2.23.4
  Attempting uninstall: nvidia-curand-cu12
    Found existing installation: nvidia-curand-cu12 10.3.6.82
    Uninstalling nvidia-curand-cu12-10.3.6.82:
      Successfully uninstalled nvidia-curand-cu12-10.3.6.82
  Attempting uninstall: nvidia-cufft-cu12
    Found existing installation: nvidia-cufft-cu12 11.2.3.61
    Uninstalling nvidia-cufft-cu12-11.2.3.61:
      Successfully uninstalled nvidia-cufft-cu12-11.2.3.61
  Attempting uninstall: nvidia-cuda-runtime-cu12
    Found existing installation: nvidia-cuda-runtime-cu12 12.5.82
    Uninstalling nvidia-cuda-runtime-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-runtime-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-nvrtc-cu12
    Found existing installation: nvidia-cuda-nvrtc-cu12 12.5.82
    Uninstalling nvidia-cuda-nvrtc-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-nvrtc-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-cupti-cu12
    Found existing installation: nvidia-cuda-cupti-cu12 12.5.82
    Uninstalling nvidia-cuda-cupti-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-cupti-cu12-12.5.82
  Attempting uninstall: nvidia-cublas-cu12
    Found existing installation: nvidia-cublas-cu12 12.5.3.2
    Uninstalling nvidia-cublas-cu12-12.5.3.2:
      Successfully uninstalled nvidia-cublas-cu12-12.5.3.2
  Attempting uninstall: nvidia-cusparse-cu12
    Found existing installation: nvidia-cusparse-cu12 12.5.1.3
    Uninstalling nvidia-cusparse-cu12-12.5.1.3:
      Successfully uninstalled nvidia-cusparse-cu12-12.5.1.3
  Attempting uninstall: nvidia-cudnn-cu12
    Found existing installation: nvidia-cudnn-cu12 9.3.0.75
    Uninstalling nvidia-cudnn-cu12-9.3.0.75:
      Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
  Attempting uninstall: nvidia-cusolver-cu12
    Found existing installation: nvidia-cusolver-cu12 11.6.3.83
    Uninstalling nvidia-cusolver-cu12-11.6.3.83:
      Successfully uninstalled nvidia-cusolver-cu12-11.6.3.83
Successfully installed dataclasses-json-0.6.7 faiss-cpu-1.11.0.post1 httpx-sse-0.4.1 langchain-community-0.3.27 marshmallow-3.26.1 myp
```

```
from langchain_community.document_loaders import TextLoader
from langchain.text_splitter import RecursiveCharacterTextSplitter
from langchain_community.embeddings import HuggingFaceEmbeddings
from langchain_community.vectorstores import FAISS
from langchain_community.llms import HuggingFacePipeline
from transformers import AutoTokenizer, AutoModelForQuestionAnswering, pipeline

# 1️⃣ Create planets.txt
content = """Mercury is the smallest planet in the Solar System and closest to the Sun.
Venus is similar in size to Earth but has a thick, toxic atmosphere.
Earth is the only planet known to support life.
Mars is called the Red Planet due to its iron oxide-rich soil.
Jupiter is the largest planet and has a giant storm called the Great Red Spot.
Saturn is famous for its prominent ring system.
Uranus rotates on its side, making its seasons very extreme.
```

```
Neptune is the farthest planet and has strong winds.
"""
with open("planets.txt", "w", encoding="utf-8") as f:
    f.write(content)

# 2️⃣ Load document
loader = TextLoader("planets.txt", encoding="utf-8")
documents = loader.load()

# 3️⃣ Split into chunks
splitter = RecursiveCharacterTextSplitter(chunk_size=200, chunk_overlap=20)
chunks = splitter.split_documents(documents)

# 4️⃣ Create embeddings (Hugging Face)
embeddings = HuggingFaceEmbeddings(model_name="sentence-transformers/all-MiniLM-L6-v2")

# 5️⃣ Store in FAISS
vectorstore = FAISS.from_documents(chunks, embeddings)
retriever = vectorstore.as_retriever()

# 6️⃣ Load Hugging Face QA model
model_name = "deepset/roberta-base-squad2"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForQuestionAnswering.from_pretrained(model_name)

qa_pipeline = pipeline("question-answering", model=model, tokenizer=tokenizer)

# 7️⃣ RAG query function
def rag_query(question):
    docs = retriever.get_relevant_documents(question)
    context = " ".join([d.page_content for d in docs])
    result = qa_pipeline(question=question, context=context)
    return result["answer"]

# 8️⃣ Test queries
questions = [
    "Which planet is known for its rings?",
    "Which planet has the Great Red Spot?",
    "Why is Mars called the Red Planet?"
]

for q in questions:
    answer = rag_query(q)
    print(f"❓ {q}\n💡 {answer}\n")
```

```
/tmp/ipython-input-4079589468.py:30: LangChainDeprecationWarning: The class `HuggingFaceEmbeddings` was deprecated in LangChain 0.2.2 an
  embeddings = HuggingFaceEmbeddings(model_name="sentence-transformers/all-MiniLM-L6-v2")
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secre
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
modules.json: 100%                                                  349/349 [00:00<00:00, 20.5kB/s]
config_sentence_transformers.json: 100%                                         116/116 [00:00<00:00, 6.36kB/s]
README.md:         10.5k/? [00:00<00:00, 369kB/s]
sentence_bert_config.json: 100%                                          53.0/53.0 [00:00<00:00, 1.41kB/s]
config.json: 100%                                                   612/612 [00:00<00:00, 22.3kB/s]
model.safetensors: 100%                                             90.9M/90.9M [00:01<00:00, 79.2MB/s]
tokenizer_config.json: 100%                                           350/350 [00:00<00:00, 26.6kB/s]
vocab.txt:         232k/? [00:00<00:00, 7.07MB/s]
tokenizer.json:         466k/? [00:00<00:00, 8.61MB/s]
special_tokens_map.json: 100%                                            112/112 [00:00<00:00, 5.06kB/s]
config.json: 100%                                                   190/190 [00:00<00:00, 3.32kB/s]
tokenizer_config.json: 100%                                              79.0/79.0 [00:00<00:00, 2.05kB/s]
config.json: 100%                                                   571/571 [00:00<00:00, 20.0kB/s]
vocab.json:         899k/? [00:00<00:00, 13.1MB/s]
merges.txt:         456k/? [00:00<00:00, 10.8MB/s]
special_tokens_map.json: 100%                                            772/772 [00:00<00:00, 28.2kB/s]
model.safetensors: 100%                                             496M/496M [00:02<00:00, 224MB/s]
Device set to use cuda:0
/tmp/ipython-input-4079589468.py:45: LangChainDeprecationWarning: The method `BaseRetriever.get_relevant_documents` was deprecated in la
  docs = retriever.get_relevant_documents(question)
❓ Which planet is known for its rings?
💡 Saturn
```