

PDF Question Answering system Using langchain

Code and outputs

```
!pip -q install --no-deps langchain langchain-community  
langchain-text-splitters
```

```
!pip -q install faiss-cpu pypdf sentence-transformers transformers  
accelerate
```

2.5/2.5 MB	42.4 MB/s	eta 0:00:00
23.8/23.8 MB	76.0 MB/s	eta 0:00:00
329.1/329.1 kB	18.9 MB/s	eta 0:00:00

```
from google.colab import files  
uploaded = files.upload()
```

generative_ai_lab1.pdf

generative_ai_lab1.pdf(application/pdf) - 4524881 bytes, last modified: 26/1/2026 - 100% done
Saving generative_ai_lab1.pdf to generative_ai_lab1.pdf

```
from langchain_community.document_loaders import PyPDFLoader  
import os
```

```
pdf_file = list(uploaded.keys())[0]    # picks the uploaded PDF  
automatically  
loader = PyPDFLoader(pdf_file)  
docs = loader.load()
```

Total pages loaded: 11

Sample text:

Rashtreeya Sikshana Samithi Trust RV UNIVERSITY School of Computer Science and Engineering Bengaluru - 560059

Lab Record CS3234 - Foundations of Generative AI VI SEMESTER B.Tech (Hons.)

```
from langchain_text_splitters import RecursiveCharacterTextSplitter
```

```
text_splitter = RecursiveCharacterTextSplitter(  
    chunk_size=800,  
    chunk_overlap=150  
)
```

Total chunks created: 21

Sample chunk:

Rashtreeya Sikshana Samithi Trust RV UNIVERSITY School of Computer Science and Engineering Bengaluru - 560059

Lab Record CS3234 - Foundations of Generative AI VI SEMESTER B.Tech (Hons.)

```
from langchain_community.embeddings import HuggingFaceEmbeddings
embedding_model =
HuggingFaceEmbeddings(model_name="sentence-transformers/all-MiniLM-L6-v2")
```

```
/tmp/ipython-input-4292522006.py:3: LangChainDeprecationWarning: The class `HuggingFaceEmbeddings` was deprecated in LangChain 0.2.2 and will be removed in 1.0. An updated v
embedding_model = HuggingFaceEmbeddings(model_name="sentence-transformers/all-MiniLM-L6-v2")
/usr/local/lib/python3.12/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 secret in your Google Colab and restart yo
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, 7.51kB/s]
config_sentence_transformers.json: 100% ██████████ 116/116 [00:00<00:00, 4.22kB/s]
README.md: ██████████ 10.5k/? [00:00<00:00, 827kB/s]
sentence_bert_config.json: 100% ██████████ 53.0/53.0 [00:00<00:00, 2.84kB/s]
config.json: 100% ██████████ 612/612 [00:00<00:00, 25.6kB/s]
model.safetensors: 100% ██████████ 90.9M/90.9M [00:01<00:00, 62.1MB/s]
tokenizer_config.json: 100% ██████████ 350/350 [00:00<00:00, 11.3kB/s]
vocab.txt: ██████████ 232k/? [00:00<00:00, 4.73MB/s]
tokenizer.json: ██████████ 466k/? [00:00<00:00, 11.1MB/s]
special_tokens_map.json: 100% ██████████ 112/112 [00:00<00:00, 2.77kB/s]
config.json: 100% ██████████ 190/190 [00:00<00:00, 5.35kB/s]
```

```
from langchain_community.vectorstores import FAISS
```


```
db = FAISS.from_documents(chunks, embedding_model)
print("FAISS vector store created successfully!")
```

FAISS vector store created successfully!

```
from transformers import pipeline
```

```
qa_pipeline = pipeline(
    "text2text-generation",
    model="google/flan-t5-base",
    max_new_tokens=256
)
```

```
print("LLM loaded successfully!")
```


config.json:  1.40k/? [00:00<00:00, 24.5kB/s]

model.safetensors: 100%  990M/990M [00:08<00:00, 113MB/s]

generation_config.json: 100%  147/147 [00:00<00:00, 12.0kB/s]

tokenizer_config.json:  2.54k/? [00:00<00:00, 53.2kB/s]

spiece.model: 100%  792k/792k [00:00<00:00, 1.77MB/s]

tokenizer.json:  2.42M/? [00:00<00:00, 40.5MB/s]

special_tokens_map.json:  2.20k/? [00:00<00:00, 48.9kB/s]

Device set to use cuda:0

LLM loaded successfully!