```
!pip install gradio langchain langchain_community langchain_milvus transformers pymilvus
```

```
Successfully installed dataclasses-json-0.6.7 httpx-sse-0.4.1 langchain_comm
```

```
!pip install replicate
```

```
b/python3.11/dist-packages (from replicate) (0.28.1)
n3.11/dist-packages (from replicate) (25.0)
/python3.11/dist-packages (from replicate) (2.11.7)
local/lib/python3.11/dist-packages (from replicate) (4.14.1)
1/dist-packages (from httpx<1,>=0.21.0->replicate) (4.9.0)
.11/dist-packages (from httpx<1,>=0.21.0->replicate) (2025.7.14)
ython3.11/dist-packages (from httpx<1,>=0.21.0->replicate) (1.0.9)
/dist-packages (from httpx<1,>=0.21.0->replicate) (3.10)
n3.11/dist-packages (from httpcore==1.*->httpx<1,>=0.21.0->replicate) (0.16.0)
cal/lib/python3.11/dist-packages (from pydantic>1.10.7->replicate) (0.7.0)
al/lib/python3.11/dist-packages (from pydantic>1.10.7->replicate) (2.33.2)
local/lib/python3.11/dist-packages (from pydantic>1.10.7->replicate) (0.4.1)
thon3.11/dist-packages (from anyio->httpx<1,>=0.21.0->replicate) (1.3.1)

━━━━━━ 48.6/48.6 kB 3.3 MB/s eta 0:00:00
```

```
!pip install sentence-transformers
```

```
    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
```

```
import os
import gradio as gr
import tempfile
from langchain_milvus import Milvus
from langchain_community.llms import Replicate
from langchain_community.document_loaders import TextLoader
from langchain.text_splitter import CharacterTextSplitter
from transformers import AutoTokenizer
from langchain.prompts import PromptTemplate
from langchain.chains import RetrievalQA, LLMChain
from langchain.chains.combine_documents.stuff import StuffDocumentsChain
from langchain_community.embeddings import HuggingFaceEmbeddings

# Replicate token and model setup
os.environ['REPLICATE_API_TOKEN'] = "r8_JHp9bIjwl9PVLop6MbYpQkgsWnGhVxJ1fudMy"
model_path = "ibm-granite/granite-3.3-8b-instruct"
tokenizer = AutoTokenizer.from_pretrained(model_path)
model = Replicate(model=model_path, replicate_api_token=os.environ['REPLICATE_API_TOKEN']

# 📦 Temporary Milvus DB
db_file = tempfile.NamedTemporaryFile(prefix="milvus_", suffix=".db", delete=False).name
embeddings = HuggingFaceEmbeddings(model_name="all-MiniLM-L6-v2")

vector_db = Milvus(
    embedding_function=embeddings,
    connection_args={"uri": db_file},
    auto_id=True,
    index_params={"index_type": "AUTOINDEX"},
)

# 📄 Financial Literacy Content
filename = "finance_literacy.txt"
with open(filename, "w") as f:
    f.write("""
```

```
UPI: Unified Payments Interface (UPI) is a real-time payment system developed by NPCI. It

Avoiding Online Scams: Never share your OTP, UPI PIN, or passwords. Beware of links prom

Interest Rates: Interest is the cost of borrowing money. Loans usually have an interest r

Budgeting: Create a monthly budget by listing your income and expenses. Prioritize needs

Digital Wallets: Apps like PhonePe, Google Pay, Paytm allow you to pay bills, recharge, a

Safe Loan Practices: Always read terms before taking a loan. Choose loans with lower inte

How to use BHIM app: Download, register with your mobile number linked to your bank, set

Fraud Protection: Use two-factor authentication. Don't save card details on unknown websi
""")

# Load & Split Content
loader = TextLoader(filename)
documents = loader.load()
splitter = CharacterTextSplitter.from_huggingface_tokenizer(
    tokenizer=tokenizer,
    chunk_size=tokenizer.model_max_length // 2,
    chunk_overlap=0,
)
texts = splitter.split_documents(documents)
for i, doc in enumerate(texts):
    doc.metadata["doc_id"] = i + 1
vector_db.add_documents(texts)

#  Financial Literacy Prompt
template = """
You are a helpful Digital Financial Literacy Assistant.
User Question: {question}
Give a simple, culturally-inclusive explanation in easy language.
Include tips to stay safe, explain tools like UPI, wallets, loans, interest, scams, budge
"""
prompt = PromptTemplate(template=template, input_variables=["question"])

# Chains
llm_chain = LLMChain(llm=model, prompt=prompt)
combine_chain = StuffDocumentsChain(llm_chain=llm_chain)

rag_chain = RetrievalQA(
    retriever=vector_db.as_retriever(),
    combine_documents_chain=combine_chain,
    return_source_documents=False
)

#  Gradio Interface
def ask_finance_agent(query):
    try:
        response = rag_chain.run(query)
        return response
    except Exception as e:
        return f"❌ Error: {str(e)}"

iface = gr.Interface(
    fn=ask_finance_agent,
    inputs=gr.Textbox(label="Ask a question about digital finance", placeholder="e.g. How
    outputs=gr.Textbox(label="Answer"),
    title="AI Agent for Digital Financial Literacy",
    description="Ask about UPI, loans, scams, budgeting, and digital tools. This assistan
    theme="default"
```

```
)

iface.launch(share=True)
```

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: U
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings t
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access
  warnings.warn(
tokenizer_config.json:         9.93k/?  [00:00<00:00,  205kB/s]
vocab.json:          777k/?  [00:00<00:00,  11.4MB/s]
merges.txt:          442k/?  [00:00<00:00,  11.5MB/s]
tokenizer.json:          3.48M/?  [00:00<00:00,  22.2MB/s]
added_tokens.json:  100%                                                207/207  [00:00<00:00,  6.56kB/s]
special_tokens_map.json:  100%                                          801/801  [00:00<00:00,  27.8kB/s]
/tmp/ipython-input-3965163926.py:22: LangChainDeprecationWarning: The class
  embeddings = HuggingFaceEmbeddings(model_name="all-MiniLM-L6-v2")
modules.json:  100%                                                     349/349  [00:00<00:00,  35.0kB/s]
config_sentence_transformers.json:  100%                                116/116  [00:00<00:00,  13.7kB/s]
README.md:          10.5k/?  [00:00<00:00,  733kB/s]
sentence_bert_config.json:  100%                                        53.0/53.0  [00:00<00:00,  6.21kB/s]
config.json:  100%                                                      612/612  [00:00<00:00,  58.7kB/s]
model.safetensors:  100%                                              90.9M/90.9M  [00:01<00:00,  101MB/s]
tokenizer_config.json:  100%                                            350/350  [00:00<00:00,  37.4kB/s]
vocab.txt:          232k/?  [00:00<00:00,  16.7MB/s]
tokenizer.json:          466k/?  [00:00<00:00,  22.6MB/s]
special_tokens_map.json:  100%                                          112/112  [00:00<00:00,  14.3kB/s]
config.json:  100%                                                      190/190  [00:00<00:00,  12.3kB/s]
/usr/local/lib/python3.11/dist-packages/torch/nn/modules/module.py:1750: Fut
  return forward_call(*args, **kwargs)
/tmp/ipython-input-3965163926.py:75: LangChainDeprecationWarning: The class
  llm_chain = LLMChain(llm=model, prompt=prompt)
/tmp/ipython-input-3965163926.py:76: LangChainDeprecationWarning: This class
  combine_chain = StuffDocumentsChain(llm_chain=llm_chain)
/tmp/ipython-input-3965163926.py:78: LangChainDeprecationWarning: This class
  rag_chain = RetrievalQA(
Colab notebook detected. To show errors in colab notebook, set debug=True in
* Running on public URL: https://e2a4446a784f892a64.gradio.live

This share link expires in 1 week. For free permanent hosting and GPU upgrad
```

AI Agent for Digital Financial Literacy

Ask about UPI, loans, scams, budgeting, and digital tools. This assistant helps you stay safe and informed.

Ask a question about digital finance