

```
!pip install langchain langgraph langchain-huggingface langchain-ollama langchain-openai langchain-community
!pip install pinecone sentence-transformers pymupdf serpapi google-search-results openapi pyarrow fastparquet
```

```
Requirement already satisfied: pyarrow in /usr/local/lib/python3.11/dist-packages (18.1.0)
Requirement already satisfied: fastparquet in /usr/local/lib/python3.11/dist-packages (2024.11.0)
Requirement already satisfied: certifi≥2019.11.17 in /usr/local/lib/python3.11/dist-packages (from pinecone) (2025.1.3)
Requirement already satisfied: pinecone-plugin-interface<0.0.8, ≥0.0.7 in /usr/local/lib/python3.11/dist-packages (from pinecone) (0.0.7)
Requirement already satisfied: python-dateutil≥2.5.3 in /usr/local/lib/python3.11/dist-packages (from pinecone) (2.8.2)
Requirement already satisfied: typing-extensions≥3.7.4 in /usr/local/lib/python3.11/dist-packages (from pinecone) (4.12.0)
Requirement already satisfied: urllib3≥1.26.0 in /usr/local/lib/python3.11/dist-packages (from pinecone) (2.3.0)
Requirement already satisfied: transformers<5.0.0, ≥4.41.0 in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (4.41.0)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (4.67.1)
Requirement already satisfied: torch≥1.11.0 in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (2.5.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (1.6.0)
Requirement already satisfied: scipy in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (1.14.1)
Requirement already satisfied: huggingface-hub≥0.20.0 in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (0.27.0)
Requirement already satisfied: Pillow in /usr/local/lib/python3.11/dist-packages (from sentence-transformers) (11.1.0)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from serpapi) (2.32.3)
Requirement already satisfied: inflection≥0.5.1 in /usr/local/lib/python3.11/dist-packages (from openapi) (0.5.1)
Requirement already satisfied: jsonschema≥3.2.0 in /usr/local/lib/python3.11/dist-packages (from openapi) (4.23.0)
Requirement already satisfied: pandas≥1.5.0 in /usr/local/lib/python3.11/dist-packages (from fastparquet) (2.2.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (from fastparquet) (2.0.2)
Requirement already satisfied: cramjam≥2.3 in /usr/local/lib/python3.11/dist-packages (from fastparquet) (2.9.1)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from fastparquet) (2025.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from fastparquet) (24.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from huggingface-hub≥0.20.0→sentence-transformers) (3.16.1)
Requirement already satisfied: pyyaml≥5.1 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub≥0.20.0→sentence-transformers) (6.0.2)
Requirement already satisfied: attrs≥22.2.0 in /usr/local/lib/python3.11/dist-packages (from jsonschema≥3.2.0→openapi) (25.1.0)
Requirement already satisfied: jsonschema-specifications≥2023.03.6 in /usr/local/lib/python3.11/dist-packages (from jsonschema≥3.2.0→openapi) (2025.9.1)
Requirement already satisfied: referencing≥0.28.4 in /usr/local/lib/python3.11/dist-packages (from jsonschema≥3.2.0→openapi) (0.36.0)
Requirement already satisfied: rpds-py≥0.7.1 in /usr/local/lib/python3.11/dist-packages (from jsonschema≥3.2.0→openapi) (0.22.0)
Requirement already satisfied: pytz≥2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas≥1.5.0→fastparquet) (2025.2)
Requirement already satisfied: tzdata≥2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas≥1.5.0→fastparquet) (2025.2)
Requirement already satisfied: six≥1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil≥2.5.3→pinecone) (1.17.0)
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0→sentence-transformers) (3.4.2)
Requirement already satisfied: Jinja2 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0→sentence-transformers) (3.1.4)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12=12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.127)
Requirement already satisfied: nvidia-cuda-runtime-cu12=12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.127)
Requirement already satisfied: nvidia-cuda-cupti-cu12=12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.127)
Requirement already satisfied: nvidia-cudnn-cu12=9.1.0.70 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (9.1.0.70)
Requirement already satisfied: nvidia-cublas-cu12=12.4.5.8 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.5.8)
Requirement already satisfied: nvidia-cufft-cu12=11.2.1.3 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (11.2.1.3)
Requirement already satisfied: nvidia-curand-cu12=10.3.5.147 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (10.3.5.147)
Requirement already satisfied: nvidia-cusolver-cu12=11.6.1.9 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (11.6.1.9)
Requirement already satisfied: nvidia-cusparse-cu12=12.3.1.170 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.3.1.170)
Requirement already satisfied: nvidia-cusparselt-cu12=0.6.2 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (0.6.2)
Requirement already satisfied: nvidia-nccl-cu12=2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (2.21.5)
Requirement already satisfied: nvidia-nvtx-cu12=12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.127)
Requirement already satisfied: nvidia-nvjitlink-cu12=12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0) (12.4.127)
Requirement already satisfied: triton=3.2.0 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0→sentence-transformers) (3.2.0)
Requirement already satisfied: sympy=1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch≥1.11.0→sentence-transformers) (1.13.1)
Requirement already satisfied: mpmath<1.4, ≥1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy=1.13.1→torch) (1.3.0)
Requirement already satisfied: regex≠2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers<5.0.0, ≥4.41.0) (2025.11.3)
Requirement already satisfied: tokenizers<0.22, ≥0.21 in /usr/local/lib/python3.11/dist-packages (from transformers<5.0.0, ≥4.41.0) (0.20.3)
Requirement already satisfied: safetensors≥0.4.3 in /usr/local/lib/python3.11/dist-packages (from transformers<5.0.0, ≥4.41.0) (0.5.2)
Requirement already satisfied: charset-normalizer<4, ≥2 in /usr/local/lib/python3.11/dist-packages (from requests→serpapi) (3.4.0)
Requirement already satisfied: idna<4, ≥2.5 in /usr/local/lib/python3.11/dist-packages (from requests→serpapi) (3.10)
Requirement already satisfied: joblib≥1.2.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn→sentence-transformers) (1.4.2)
Requirement already satisfied: threadpoolctl≥3.1.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn→sentence-transformers) (3.5.0)
Requirement already satisfied: MarkupSafe≥2.0 in /usr/local/lib/python3.11/dist-packages (from Jinja2→torch≥1.11.0→sentence-transformers) (3.0.2)
```

```
pip install gradio
```

```
Requirement already satisfied: gradio in /usr/local/lib/python3.11/dist-packages (5.23.1)
Requirement already satisfied: aiofiles<24.0, ≥22.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (23.2.1)
Requirement already satisfied: anyio<5.0, ≥3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (4.9.0)
Requirement already satisfied: fastapi<1.0, ≥0.115.2 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.115.12)
Requirement already satisfied: fmpy in /usr/local/lib/python3.11/dist-packages (from gradio) (0.5.0)
Requirement already satisfied: gradio-client=1.8.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (1.8.0)
Requirement already satisfied: groovy~=0.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.1.2)
Requirement already satisfied: httpx≥0.24.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.28.1)
Requirement already satisfied: huggingface-hub≥0.28.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.29.3)
Requirement already satisfied: Jinja2<4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.1.6)
Requirement already satisfied: markupsafe<4.0, ≥2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.0.2)
Requirement already satisfied: numpy<3.0, ≥1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.0.2)
Requirement already satisfied: orjson~=3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.10.15)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from gradio) (24.2)
Requirement already satisfied: pandas<3.0, ≥1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.2.2)
Requirement already satisfied: pillow<12.0, ≥8.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (11.1.0)
Requirement already satisfied: pydantic≥2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.10.6)
Requirement already satisfied: pydub in /usr/local/lib/python3.11/dist-packages (from gradio) (0.25.1)
```

Requirement already satisfied: python-multipart \geq 0.0.18 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.0.20)

Requirement already satisfied: pyyaml $<$ 7.0, \geq 5.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (6.0.2)

Requirement already satisfied: ruff \geq 0.9.3 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.11.2)

Requirement already satisfied: safehttpx $<$ 0.2.0, \geq 0.1.6 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.1.6)

Requirement already satisfied: semantic-version \sim =2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.10.0)

Requirement already satisfied: starlette $<$ 1.0, \geq 0.40.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.46.1)

Requirement already satisfied: tomlkit $<$ 0.14.0, \geq 0.12.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.13.2)

Requirement already satisfied: typer $<$ 1.0, \geq 0.12 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.15.2)

Requirement already satisfied: typing-extensions \sim =4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (4.12.2)

Requirement already satisfied: uvicorn \geq 0.14.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.34.0)

Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from gradio-client=1.8.0 \rightarrow gradio) (202

Requirement already satisfied: websockets $<$ 16.0, \geq 10.0 in /usr/local/lib/python3.11/dist-packages (from gradio-client=1.8

Requirement already satisfied: idna \geq 2.8 in /usr/local/lib/python3.11/dist-packages (from anyio $<$ 5.0, \geq 3.0 \rightarrow gradio) (3.10)

Requirement already satisfied: sniffio \geq 1.1 in /usr/local/lib/python3.11/dist-packages (from anyio $<$ 5.0, \geq 3.0 \rightarrow gradio) (1.

Requirement already satisfied: certifi in /usr/local/lib/python3.11/dist-packages (from httpx \geq 0.24.1 \rightarrow gradio) (2025.1.31

Requirement already satisfied: httpcore=1.* in /usr/local/lib/python3.11/dist-packages (from httpx \geq 0.24.1 \rightarrow gradio) (1.0

Requirement already satisfied: h11 $<$ 0.15, \geq 0.13 in /usr/local/lib/python3.11/dist-packages (from httpcore=1.* \rightarrow httpx \geq 0.2

Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from huggingface-hub \geq 0.28.1 \rightarrow gradio)

Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from huggingface-hub \geq 0.28.1 \rightarrow gradio)

Requirement already satisfied: tqdm \geq 4.42.1 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub \geq 0.28.1 \rightarrow gra

Requirement already satisfied: python-dateutil \geq 2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas $<$ 3.0, \geq 1.0 \rightarrow

Requirement already satisfied: pytz \geq 2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas $<$ 3.0, \geq 1.0 \rightarrow gradio) (2

Requirement already satisfied: tzdata \geq 2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas $<$ 3.0, \geq 1.0 \rightarrow gradio)

Requirement already satisfied: annotated-types \geq 0.6.0 in /usr/local/lib/python3.11/dist-packages (from pydantic \geq 2.0 \rightarrow gra

Requirement already satisfied: pydantic-core=2.27.2 in /usr/local/lib/python3.11/dist-packages (from pydantic \geq 2.0 \rightarrow grad

Requirement already satisfied: click \geq 8.0.0 in /usr/local/lib/python3.11/dist-packages (from typer $<$ 1.0, \geq 0.12 \rightarrow gradio) (8

Requirement already satisfied: shellingham \geq 1.3.0 in /usr/local/lib/python3.11/dist-packages (from typer $<$ 1.0, \geq 0.12 \rightarrow grad

Requirement already satisfied: rich \geq 10.11.0 in /usr/local/lib/python3.11/dist-packages (from typer $<$ 1.0, \geq 0.12 \rightarrow gradio) (

Requirement already satisfied: six \geq 1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil \geq 2.8.2 \rightarrow pandas $<$ 3

Requirement already satisfied: markdown-it-py \geq 2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich \geq 10.11.0 \rightarrow type

Requirement already satisfied: pygments $<$ 3.0.0, \geq 2.13.0 in /usr/local/lib/python3.11/dist-packages (from rich \geq 10.11.0 \rightarrow ty

Requirement already satisfied: charset-normalizer $<$ 4, \geq 2 in /usr/local/lib/python3.11/dist-packages (from requests \rightarrow huggin

Requirement already satisfied: urllib3 $<$ 3, \geq 1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests \rightarrow huggingface-

Requirement already satisfied: mdurl \sim =0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py \geq 2.2.0 \rightarrow rich \geq 1

```
import fitz # PyMuPDF for PDF processing
from datetime import datetime
from langchain.text_splitter import RecursiveCharacterTextSplitter
from langchain_huggingface import HuggingFaceEmbeddings
from langchain.memory import ConversationBufferMemory
from langchain_core.output_parsers import StrOutputParser
from langchain_core.prompts import ChatPromptTemplate
from pinecone import Pinecone
from langgraph.graph import StateGraph, MessagesState, START, END
from langgraph.prebuilt import ToolNode, tools_condition
from langchain.tools import Tool
from sentence_transformers import CrossEncoder, SentenceTransformer
from langchain.output_parsers import PydanticOutputParser
from pydantic import BaseModel, Field
from langchain.utilities import SerpAPIWrapper
from langchain_core.messages import SystemMessage, HumanMessage
from typing import TypedDict
import pandas as pd
import os
```

Double-click (or enter) to edit

```
# Fetch folders from google drive into the current session
# !pip install gdown
# !gdown --folder https://drive.google.com/drive/folders/1E4LtM210hm0wTNm7sYJUCZsk9l0TTytV?usp=sharing
```

```
# folder_path = "/content/data/"
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
folder_path = "/content/drive/My Drive/GenAI/data/"
```

🔗 Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount

```
os.environ["SERPAPI_API_KEY"] = "650421ed0daa29addfe17bed6601a809a70d0aacea2a8c804397b4c61020ff29"
os.environ["PINECONE_API_KEY"] = "psck_5qAp4G_QyLWLWYcjNsy23CDhghkH4cAvesEhYHMKWTFkC32i8SyzjrDLShwnLPcWS97PUM"
os.environ["OPENAI_API_KEY"] = 'sk-proj-KP0_jRkwjRJVGf5cm4cvIzysL-6sdf0d3QxoN2pEZYqk6SeUMM3VA-R0sZTnKwF-vjxmvPgZTzt3BlbkfJhoI'
```

```
# Setup vector database
pinecone_api_key = os.getenv("PINECONE_API_KEY")
```

```

pc = Pinecone(api_key=pinecone_api_key)
index_name = "profilestore"
index = pc.Index(index_name)

embedder = HuggingFaceEmbeddings(model_name = 'BAAI/bge-base-en-v1.5')

# Conversation history
memory = ConversationBufferMemory(memory_key="chat_history", return_messages=True)

# SerpAPI
serpapi_tools = SerpAPIWrapper(serpapi_api_key=os.getenv("SERPAPI_API_KEY"))

```

```

🔗 /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),
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(
<ipython-input-5-6b17b6ba41e4>:15: LangChainDeprecationWarning: Please see the migration guide at: https://python.langcha
memory = ConversationBufferMemory(memory_key="chat_history", return_messages=True)

```

```

# OpenAI
import openai
from langchain_openai import ChatOpenAI, OpenAI

openai.api_key = os.getenv("OPENAI_API_KEY")
openaillm = ChatOpenAI(model = "gpt-4-turbo", temperature=0.5)

```

```

pip install langchain_experimental

```

```

🔗 Requirement already satisfied: langchain_experimental in /usr/local/lib/python3.11/dist-packages (0.3.4)
Requirement already satisfied: langchain-community<0.4.0, ≥0.3.0 in /usr/local/lib/python3.11/dist-packages (from langcha
Requirement already satisfied: langchain-core<0.4.0, ≥0.3.28 in /usr/local/lib/python3.11/dist-packages (from langchain_e
Requirement already satisfied: langchain<1.0.0, ≥0.3.21 in /usr/local/lib/python3.11/dist-packages (from langchain-commun
Requirement already satisfied: SQLAlchemy<3, ≥1.4 in /usr/local/lib/python3.11/dist-packages (from langchain-community<0.
Requirement already satisfied: requests<3, ≥2 in /usr/local/lib/python3.11/dist-packages (from langchain-community<0.4.0,
Requirement already satisfied: PyYAML ≥5.3 in /usr/local/lib/python3.11/dist-packages (from langchain-community<0.4.0, ≥0
Requirement already satisfied: aiohttp<4.0.0, ≥3.8.3 in /usr/local/lib/python3.11/dist-packages (from langchain-community
Requirement already satisfied: tenacity≠8.4.0, <10, ≥8.1.0 in /usr/local/lib/python3.11/dist-packages (from langchain-com
Requirement already satisfied: dataclasses-json<0.7, ≥0.5.7 in /usr/local/lib/python3.11/dist-packages (from langchain-co
Requirement already satisfied: pydantic-settings<3.0.0, ≥2.4.0 in /usr/local/lib/python3.11/dist-packages (from langchain
Requirement already satisfied: langsmith<0.4, ≥0.1.125 in /usr/local/lib/python3.11/dist-packages (from langchain-communi
Requirement already satisfied: httpx-sse<1.0.0, ≥0.4.0 in /usr/local/lib/python3.11/dist-packages (from langchain-communi
Requirement already satisfied: numpy<3, ≥1.26.2 in /usr/local/lib/python3.11/dist-packages (from langchain-community<0.4.
Requirement already satisfied: jsonpatch<2.0, ≥1.33 in /usr/local/lib/python3.11/dist-packages (from langchain-core<0.4.0
Requirement already satisfied: packaging<25, ≥23.2 in /usr/local/lib/python3.11/dist-packages (from langchain-core<0.4.0,
Requirement already satisfied: typing-extensions ≥4.7 in /usr/local/lib/python3.11/dist-packages (from langchain-core<0.4
Requirement already satisfied: pydantic<3.0.0, ≥2.5.2 in /usr/local/lib/python3.11/dist-packages (from langchain-core<0.4
Requirement already satisfied: aiohappyeyeballs ≥2.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3
Requirement already satisfied: aiosignal ≥1.1.2 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3→l
Requirement already satisfied: attrs ≥17.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3→lang
Requirement already satisfied: frozenlist ≥1.1.1 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3→
Requirement already satisfied: multidict<7.0, ≥4.5 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3
Requirement already satisfied: propcache ≥0.2.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3→l
Requirement already satisfied: yarl<2.0, ≥1.17.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp<4.0.0, ≥3.8.3→
Requirement already satisfied: marshmallow<4.0.0, ≥3.18.0 in /usr/local/lib/python3.11/dist-packages (from dataclasses-js
Requirement already satisfied: typing-inspect<1, ≥0.4.0 in /usr/local/lib/python3.11/dist-packages (from dataclasses-json
Requirement already satisfied: jsonpointer ≥1.9 in /usr/local/lib/python3.11/dist-packages (from jsonpatch<2.0, ≥1.33→la
Requirement already satisfied: langchain-text-splitters<1.0.0, ≥0.3.7 in /usr/local/lib/python3.11/dist-packages (from la
Requirement already satisfied: httpx<1, ≥0.23.0 in /usr/local/lib/python3.11/dist-packages (from langsmith<0.4, ≥0.1.125→
Requirement already satisfied: orjson<4.0.0, ≥3.9.14 in /usr/local/lib/python3.11/dist-packages (from langsmith<0.4, ≥0.1
Requirement already satisfied: requests-toolbelt<2.0.0, ≥1.0.0 in /usr/local/lib/python3.11/dist-packages (from langsmith
Requirement already satisfied: zstandard<0.24.0, ≥0.23.0 in /usr/local/lib/python3.11/dist-packages (from langsmith<0.4, ≥
Requirement already satisfied: annotated-types ≥0.6.0 in /usr/local/lib/python3.11/dist-packages (from pydantic<3.0.0, ≥2
Requirement already satisfied: pydantic-core=2.27.2 in /usr/local/lib/python3.11/dist-packages (from pydantic<3.0.0, ≥2.
Requirement already satisfied: python-dotenv ≥0.21.0 in /usr/local/lib/python3.11/dist-packages (from pydantic-settings<3
Requirement already satisfied: charset-normalizer<4, ≥2 in /usr/local/lib/python3.11/dist-packages (from requests<3, ≥2→
Requirement already satisfied: idna<4, ≥2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3, ≥2→langchain-co
Requirement already satisfied: urllib3<3, ≥1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3, ≥2→langch
Requirement already satisfied: certifi ≥2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3, ≥2→langch
Requirement already satisfied: greenlet≠0.4.17 in /usr/local/lib/python3.11/dist-packages (from SQLAlchemy<3, ≥1.4→lang
Requirement already satisfied: anyio in /usr/local/lib/python3.11/dist-packages (from httpx<1, ≥0.23.0→langsmith<0.4, ≥0
Requirement already satisfied: httpcore=1.* in /usr/local/lib/python3.11/dist-packages (from httpx<1, ≥0.23.0→langsmith
Requirement already satisfied: h11<0.15, ≥0.13 in /usr/local/lib/python3.11/dist-packages (from httpcore=1.*→httpx<1, ≥
Requirement already satisfied: mypy-extensions ≥0.3.0 in /usr/local/lib/python3.11/dist-packages (from typing-inspect<1, ≥
Requirement already satisfied: sniffio ≥1.1 in /usr/local/lib/python3.11/dist-packages (from anyio→httpx<1, ≥0.23.0→lan

```

```

from langchain_experimental.tools import PythonREPLTool

current_year = datetime.now().year

python_repl = PythonREPLTool()
math_tool = Tool(name="Python REPL", func=python_repl.run, description="Useful for executing math operations")

def serapi_search(query):
    return serpapi_tools.run(query)

fraud_tool = Tool(
    name = "SerpAPI_Search",
    func = serapi_search,
    description = "latest banking fraud detection site:fincen.gov OR site:fbi.gov OR site:forbes.com OR site:bloomberg.com"
)

financial_tool = Tool(
    name = "SerpAPI_Search",
    func = serapi_search,
    description = f"latest financial fraud trends {current_year} OR money laundering techniques {current_year}"
)

tools = [fraud_tool, financial_tool, math_tool]
openai.bind_tools(tools)

🔄 RunnableBinding(bound=ChatOpenAI(client=<openai.resources.chat.completions.completions.Completions object at 0x7d9111364ed0>, async_client=<openai.resources.chat.completions.AsyncCompletions object at 0x7d91117a4250>, root_client=<openai.OpenAI object at 0x7d911179ff10>, root_async_client=<openai.AsyncOpenAI object at 0x7d91117a6d50>, model_name='gpt-4-turbo', temperature=0.5, model_kwargs={}, openai_api_key=SecretStr('*****')), kwargs={'tools': [{'type': 'function', 'function': {'name': 'SerpAPI_Search', 'description': 'latest banking fraud detection site:fincen.gov OR site:fbi.gov OR site:forbes.com OR site:bloomberg.com', 'parameters': {'properties': {'__arg1': {'title': '__arg1', 'type': 'string'}}}, 'required': ['__arg1'], 'type': 'object'}}}, {'type': 'function', 'function': {'name': 'SerpAPI_Search', 'description': 'latest financial fraud trends 2025 OR money laundering techniques 2025', 'parameters': {'properties': {'__arg1': {'title': '__arg1', 'type': 'string'}}}, 'required': ['__arg1'], 'type': 'object'}}}, {'type': 'function', 'function': {'name': 'Python REPL', 'description': 'Useful for executing math operations', 'parameters': {'properties': {'__arg1': {'title': '__arg1', 'type': 'string'}}}, 'required': ['__arg1'], 'type': 'object'}}}], config={}, config_factories=[]))

from sklearn.ensemble import IsolationForest
import joblib

class ModelParams(BaseModel):
    gender: str = Field()
    amt: int = Field()
    unix_time: int = Field()
    category: str = Field()
    merchant: str = Field()
    city_pop: int = Field()

model_parser = PydanticOutputParser(pydantic_object=ModelParams)

def fraud_detection_check(query):
    print("Query output " + str(query))
    # Load model
    model = joblib.load(folder_path + 'unsupervised learning/isolation_forest_model.joblib')
    features = ["gender", "amt", "unix_time", "category", "merchant", "city_pop"]

    # ✅ Check model type
    if not isinstance(model, IsolationForest):
        raise TypeError(f"Expected IsolationForest model, but got {type(model)}")

    print("Query is " + str(query))
    query["unix_time"] = datetime.strptime(query["unix_time"], "%Y-%m-%d %H:%M:%S").timestamp()

    new_transaction = pd.DataFrame([query])
    new_transaction_encoded = pd.get_dummies(new_transaction)

    df = pd.read_parquet(folder_path + "unsupervised learning/dataset.parquet")

    # Ensure all columns match the training dataset
    df_selected = pd.get_dummies(df[features])
    missing_cols = set(df_selected.columns) - set(new_transaction_encoded.columns)

    missing_df = pd.DataFrame(0, index=new_transaction_encoded.index, columns=list(missing_cols))
    new_transaction_encoded = pd.concat([new_transaction_encoded, missing_df], axis=1)

```

```

# Reorder columns to match training data
new_transaction_encoded = new_transaction_encoded[df_selected.columns]

# Predict fraud score (-1 = fraud, 1 = normal)
fraud_score = model.predict(new_transaction_encoded)[0]

# Convert to readable format
fraud_detected = 1 if fraud_score == -1 else 0

return fraud_detected

# Test Fraud
date_str = "2024-03-23 00:30:00"
unix = datetime.strptime(date_str, "%Y-%m-%d %H:%M:%S").timestamp()

fraud_detection_check({
    "gender": "F",
    "amt": 300,
    "unix_time": "2024-03-23 14:30:00",
    "category": "Transaction",
    "merchant": "Small Buisness",
    "city_pop": 500000000000
})

🔗 Query output {'gender': 'F', 'amt': 300, 'unix_time': '2024-03-23 14:30:00', 'category': 'Transaction', 'merchant': 'Small Buisness', 'city_pop': 500000000000}
Query is {'gender': 'F', 'amt': 300, 'unix_time': '2024-03-23 14:30:00', 'category': 'Transaction', 'merchant': 'Small Buisness', 'city_pop': 500000000000}
0

# RISK ANALYSIS GRAPH

class RiskAnalysisState(TypedDict):
    messages: list[dict[str, str]]
    query: dict
    rules: str
    risk_score: float

def reasoner(state: RiskAnalysisState):
    print("Reasoner " + str(state))
    user_query = state.get("query", "")
    rules = state.get("rules", "")

    # Chain-of-Thought Reasoning Prompt
    context_prompt = f"""
You are a compliance risk evaluator. Follow this step-by-step approach to validate the financial profile:
Keep your output a bit small

1. **Extract Key Details**: Identify major financial transactions, loan history, and unusual activity.
2. **Compare with Policy Rules**: Use the provided policy guidelines to check for violations.
3. **Assess Risk Score**: Assign a risk score (0-1).
4. **Explain the Decision**: Provide justification for the assigned risk score.
User Request:
{user_query}

Regulatory Rules:
{rules}
"""

    message = [{"role": "system", "content": context_prompt}]
    response = openai.LLM.invoke(message + state["messages"])

    print("Reasoner response", response)

    risk_score = extract_risk_score(response.content)
    updated_risk_score = risk_score if risk_score is not None else state.get("risk_score", 0)

    return {"messages": state["messages"] + [response], "risk_score": updated_risk_score }

import re

def extract_risk_score(response_text):
    match = re.search(r"Risk Score\s*([0-1])", response_text)
    return float(match.group(1)) if match else None

```



```

def decision_node(state: RiskAnalysisState) → str:
    human_review_required = state.get("risk_score", 0) > 0.7
    return "human_review" if human_review_required else "low_risk_message"

def human_review(state: RiskAnalysisState):
    last_message = state["messages"][-1]["content"]
    risk_score = extract_risk_score(last_message)

    # 🚨 If the risk score is high, pause for human review
    if risk_score ≥ 0.7:
        msg = "🚨 High-Risk Alert! Manual review required."
        return {"messages": [{"role": "assistant", "content": msg}]}

    return {"messages": state["messages"]}

def evaluate_risk(state: RiskAnalysisState):
    print("risk " + str(state))
    user_query = state.get("query", "")
    risk_score = state.get("risk_score", "")

    print("Risk score before evaluation", risk_score)

    is_fraud = fraud_detection_check(user_query)

    print("Is fraudulent?", is_fraud)

    final_risk_score = is_fraud * 0.5 + risk_score * 0.5

    print("Risk score after evaluation", final_risk_score)

    response = openaillm.invoke(f"""
        The Unsupervised Isolated Model has given a risk score of {str(is_fraud)}
        The risk score is either 0 or 1 only
        """)

    return {"messages": [response] + state["messages"], "risk_score": final_risk_score}

import json

def low_risk_message(state: RiskAnalysisState):
    assistant_response = [msg.content for msg in state["messages"]]
    rules = state.get("rules", "")
    risk = state.get("risk_score", "")

    # ✅ Structured reasoning for final decision
    final_low_risk_prompt = ChatPromptTemplate([
        ("system", "You are a compliance analyst reviewing financial transactions for compliance assessment."),
        ("system", "This is an analytical tool for risk assessment insights. It does not make final financial decisions."),
        ("user", f"***Policy Context Retrieved**:\n{rules}"),
        ("user", f"***Previous Analysis**:\n{assistant_response}"),
        ("user", f"***Generate a structured response including a risk score, key analysis points, and recommendations. The updated"),
        ("system", f"*** Generate an analysis or remedial action that can take place to reduce the profile risk. Do not include an
    ])

    # ✅ Let LLM generate the final structured response
    final_response = openaillm.invoke(final_low_risk_prompt.format_prompt().to_messages() + state["messages"])

    return {"messages": [{"role": "assistant", "content": final_response.content}]}

builder = StateGraph(RiskAnalysisState)

builder.add_node("reasoner", reasoner)
builder.add_node("tools", ToolNode(tools))
builder.add_node("evaluate", evaluate_risk)
builder.add_node("human_review", human_review)
builder.add_node("low_risk_message", low_risk_message)

builder.add_edge(START, "reasoner")
builder.add_conditional_edges("reasoner", tools_condition)
builder.add_edge("tools", "reasoner")
builder.add_edge("reasoner", "evaluate")
builder.add_conditional_edges("evaluate", decision_node)

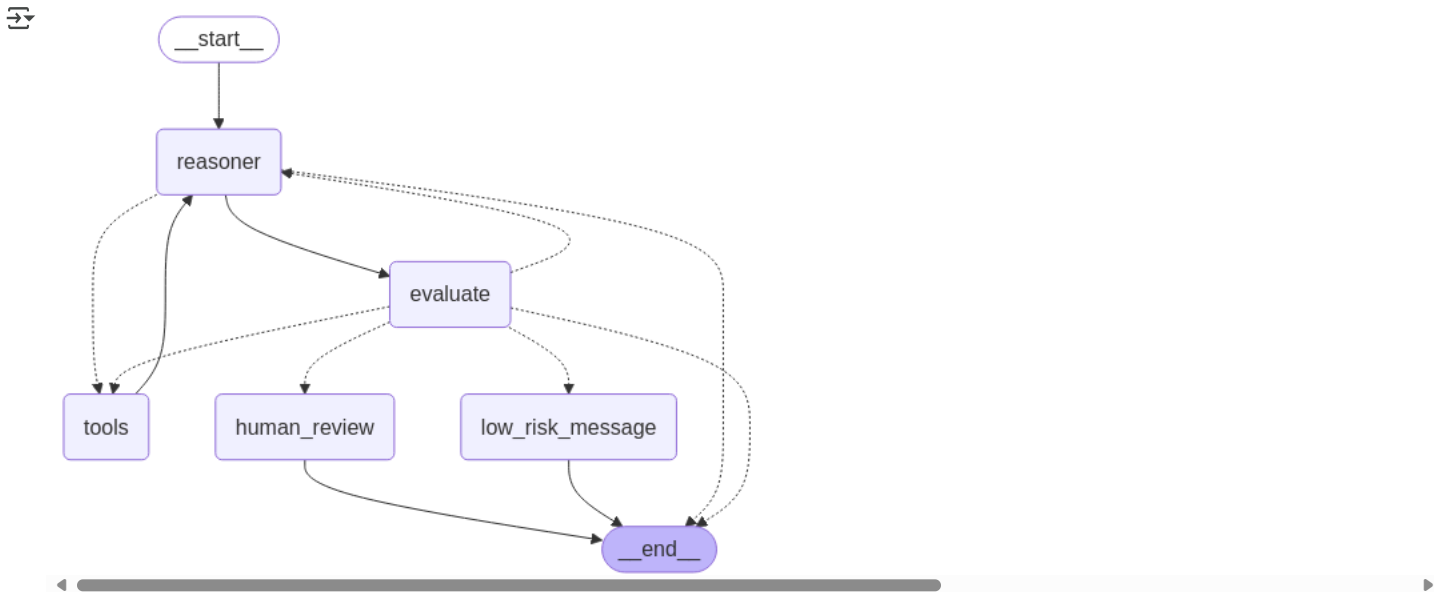
builder.add_edge("human_review", END)

```

```
builder.add_edge("low_risk_message", END)

validate_graph = builder.compile()
```

validate_graph



```
date_str = "2024-03-23 00:30:00"
unix = datetime.strptime(date_str, "%Y-%m-%d %H:%M:%S").timestamp()
```

```
query = {
    "gender": "M",
    "amt": 500,
    "unix_time": "2024-03-23 00:30:00",
    "category": "From Software",
    "merchant": "Amazon",
    "city_pop": 5000000000
}
```

```
rules = ""
*Schedule A - Retail Determination Process*
```

- 1. *Data Collection*
 - Sources: Utilizes credit reports, consumer panels, internal systems for retail banking data accuracy.
 - Importance: Ensures reliable data for comprehensive analysis.
- 2. *Credit Quality Analysis*
 - Metrics: Calculates delinquency rates at 30, 60, and 90 days to assess risk levels in different loan segments (credit car
- 3. *Trend Analysis*
 - Comparison: Analyzes performance across various consumer segments to track changes over time.
- 4. *Delinquency Rate Calculation*
 - Percentage Determination: Identifies the proportion of delinquent borrowers contributing to credit risk assessment.
- 5. *Credit Utilization Metrics*
 - Ratio Assessment: Evaluates average credit utilization, indicating consumer leverage and future default risk potential.
- 6. *Risk-Weighted Asset (RWA) Calculation*
 - Risk Assessment: Measures the riskiness of assets against regulatory requirements for capital adequacy.
- 7. *Stress Testing*
 - Scenario Simulation: Assesses potential impacts on credit performance under adverse economic conditions to ensure solvenc
- 8. *Liquidity Assessment*
 - Liquidity Monitoring: Evaluates the bank's ability to meet short-term obligations using retail deposits and liquid assets
- 9. *Regulatory Compliance*
 - Standards Adherence: Ensures data consistency with GAAP and Basel III regulations for accurate reporting.
- 10. *Reporting & Analysis*
 - Comprehensive Reporting: Presents detailed insights into credit performance, trends, risk metrics, and regulatory compli

```
def chatbot(query,rules):
    requestObj = {"query": query, "rules": rules, "messages": []}
    response = validate_graph.invoke(requestObj)
    chatbot(query,rules)
```

```
Reasoner {'messages': [], 'query': {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Software', 'merchant': 'Amazon'}, 'rules': [{'id': 1, 'content': 'If the transaction amount is greater than $500, the transaction is fraudulent.'}], 'messages': []}
Reasoner response content="1. **Extract Key Details**:\n - **Transaction Amount**: $500\n - **Transaction Date**: 2024-03-23 00:30:00\n Risk score before evaluation 0\n Query output {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Software', 'merchant': 'Amazon'}\n Query is {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Software', 'merchant': 'Amazon'}\n Is fraudulent? 1\n Risk score after evaluation 0.5"
```

```
import gradio as gr
import json

def process_input1(user_input):
    """Processes JSON input and stores it."""
    try:
        parsed_dict = json.loads(user_input) # Convert JSON string to dictionary
        processed_data = parsed_dict

        return [("User", user_input), ("Bot", f"✅ Successfully processed JSON: {processed_data}")], processed_data
    except json.JSONDecodeError:
        return [("Bot", "❌ Invalid JSON format. Please enter a valid JSON.)"], None

def process_final(processed_input1, input2):
    """Combines stored JSON data with second input."""
    if processed_input1 is None:
        return [("Bot", "❌ No valid JSON processed. Please start again.)"]

    requestObj = {"query": processed_input1, "rules": input2, "messages": []}
    response = validate_graph.invoke(requestObj)

    return [("User", input2), ("Bot", response["messages"][-1]["content"])]

with gr.Blocks() as demo:
    gr.Markdown("# 🤖 AI Chatbot\nWelcome! Enter your inputs below.")

    chatbot = gr.Chatbot(label="Chat") # Chat-style component
    state = gr.State() # Store processed JSON

    with gr.Row():
        input1 = gr.Textbox(label="Enter a transaction profile as a JSON please")
        btn1 = gr.Button("Submit Transaction ")

    btn1.click(process_input1, inputs=input1, outputs=[chatbot, state])

    with gr.Row(visible=False) as second_input_row:
        input2 = gr.Textbox(label="Add the Regulatory Rules to validate the Profile")
        btn2 = gr.Button("Generate Analysis")

    btn1.click(lambda: gr.update(visible=True), outputs=second_input_row)
    btn2.click(process_final, inputs=[state, input2], outputs=chatbot)

demo.launch(debug=True)
```



```
*** <ipython-input-33-c6add50967f3>:27: UserWarning: You have not specified a value for the `type` parameter. Defaulting to t
chatbot = gr.Chatbot(label="Chat") # Chat-style component
Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by

Colab notebook detected. This cell will run indefinitely so that you can see errors and logs. To turn off, set debug=False
* Running on public URL: https://ce7141da9d3a616e2e.gradio.live

This share link expires in 72 hours. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal i
```



```
Reasoner {'messages': [], 'query': {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Soft
Reasoner response content='1. **Extract Key Details**:\n - **Transaction Amount**:$500\n - **Date**: 2024-03-23\n
risk {'messages': [AIMessage(content='1. **Extract Key Details**:\n - **Transaction Amount**:$500\n - **Date**: 202
Submit Transaction
Risk score before evaluation 0.1
Query output {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Software', 'merchant': 'Am
Query is {'gender': 'M', 'amt': 500, 'unix_time': '2024-03-23 00:30:00', 'category': 'From Software', 'merchant': 'Amazon
Is fraudulent? 1
Risk score after evaluation 0.55
```

```
openai.llm.invoke("hello")
```

Start coding or [generate](#) with AI.