

Intro to LangChain

Building Intelligent LLm Apps

What is LangChain

- LangChain is an open-source framework for developing applications powered by large language models (LLMs).
- It helps you go beyond basic prompts.
- Think: “ChatGPT with tools, memory, and reasoning.

Why Use LangChain

- Manage multi-step reasoning
- Integrate tools and APIs
- Add memory to LLMs
- Handle custom workflows
- Build agents that think and act

LangChain Building Blocks

LLMs - The brains (GPT, Claude, Gemini)

Chains - Connect prompts and steps

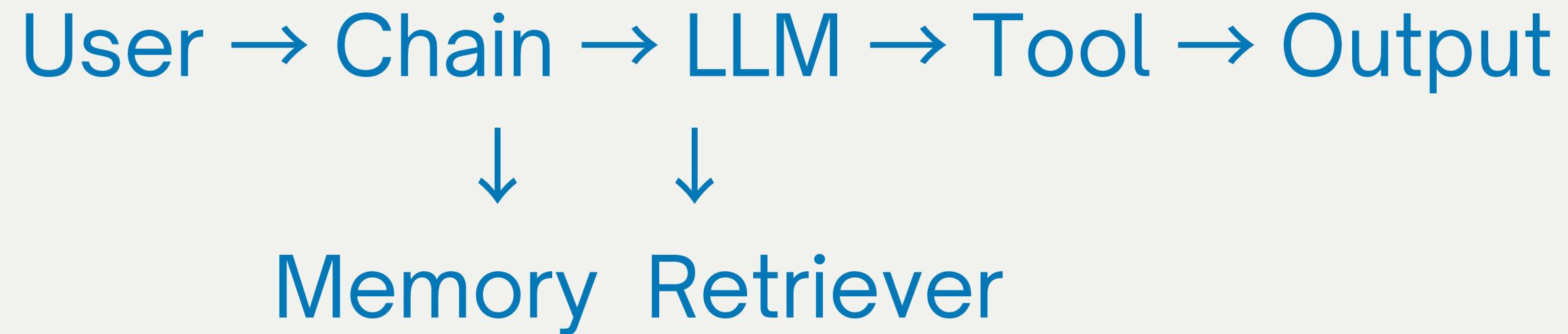
Agents - Let AI choose tools dynamically

Memory - Let AI remember things

Tools - External functions/APIs

Retrievers - Find data from documents or DBs

Flow diagram



A Day in the Life of a LangChain App

- You provide a prompt or user input
- Chain connects the logic (multi-step)
- LLM processes + retrieves documents or uses tools
- Memory stores previous interactions
- Final response is returned

What Can You Build with LangChain

- Chatbots with memory
- Q&A over PDFs, websites, or databases
- Customer support agents
- AI coding assistants
- Research planners

Components Deep Dive

Chains

- Sequential or conditional execution of steps
- Can involve prompt templates + memory + LLMs

Memory

- Tracks past interactions in chat
- Types: BufferMemory, SummaryMemory, VectorStoreMemory
- Helps with context retention across turns

Agents

- LLM decides which tool to use (like calling a calculator, search, or weather API)
- Supports dynamic reasoning

Tools

- Plug-in external logic: search engines, DBs, calculators, APIs
- LangChain gives you prebuilt ones or lets you define your own

Resources

- <https://python.langchain.com/docs/introduction/>
- https://www.youtube.com/watch?v=lG7Uxts9SXs&ab_channel=freeCodeCamp.org

Task

Build an AI assistant that:

- Lets a user upload a document (PDF, TXT, etc.)
- Lets the user ask questions about it in a chat interface
- Uses LangChain + embeddings + LLMs to retrieve relevant chunks
- Keeps a short-term memory of the conversation
- Optionally: Save chat history by session ID
- The work should be pushed to a private git repo

Thank You