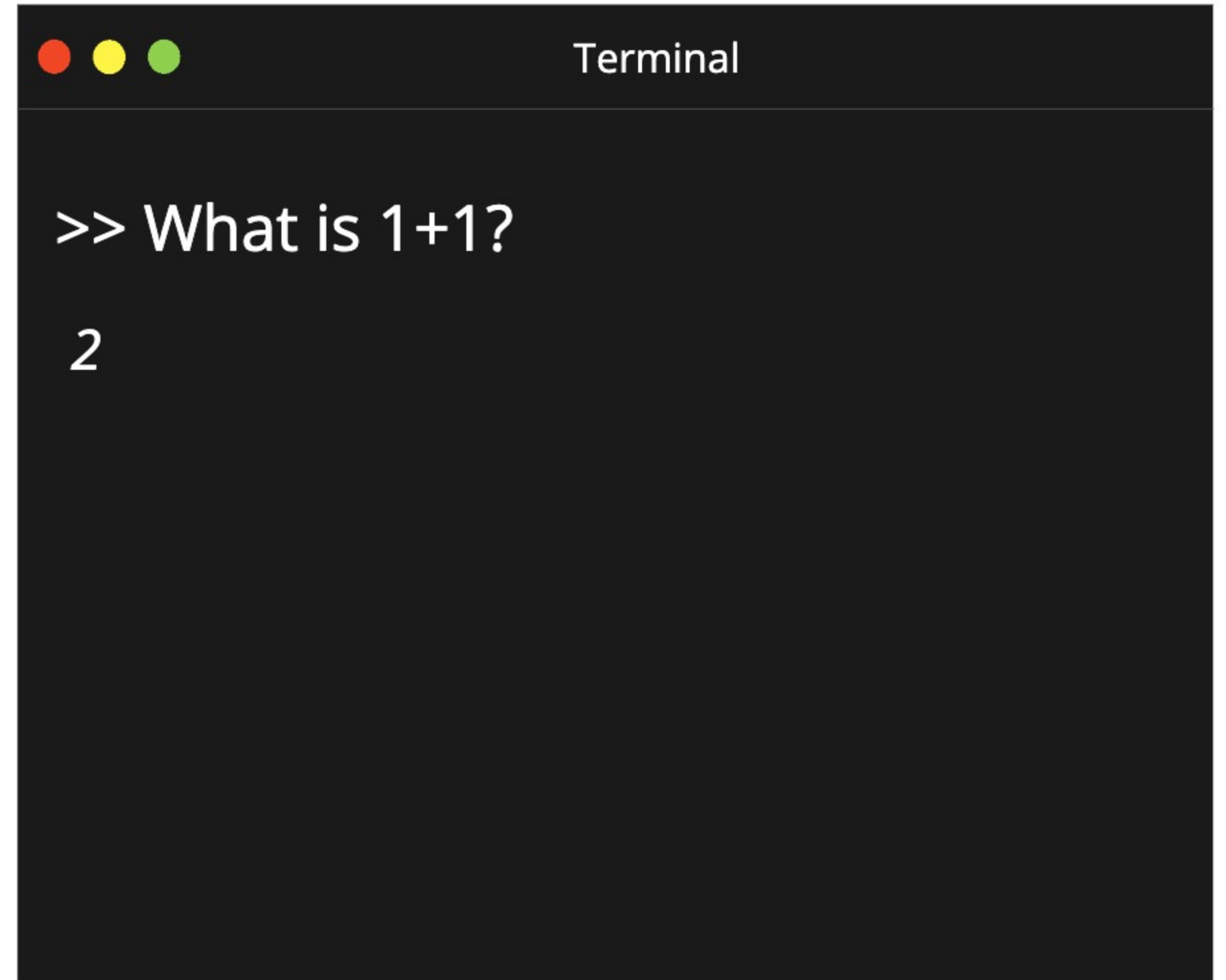


Chatbot!

User types
in a prompt



Gets a
response

A terminal window with a dark background and a title bar with three colored dots (red, yellow, green) on the left. The title bar contains the word 'Terminal'. The terminal shows a prompt '>>' followed by the text 'What is 1+1?' and the response '2' on the next line.

```
Terminal

>> What is 1+1?

2
```



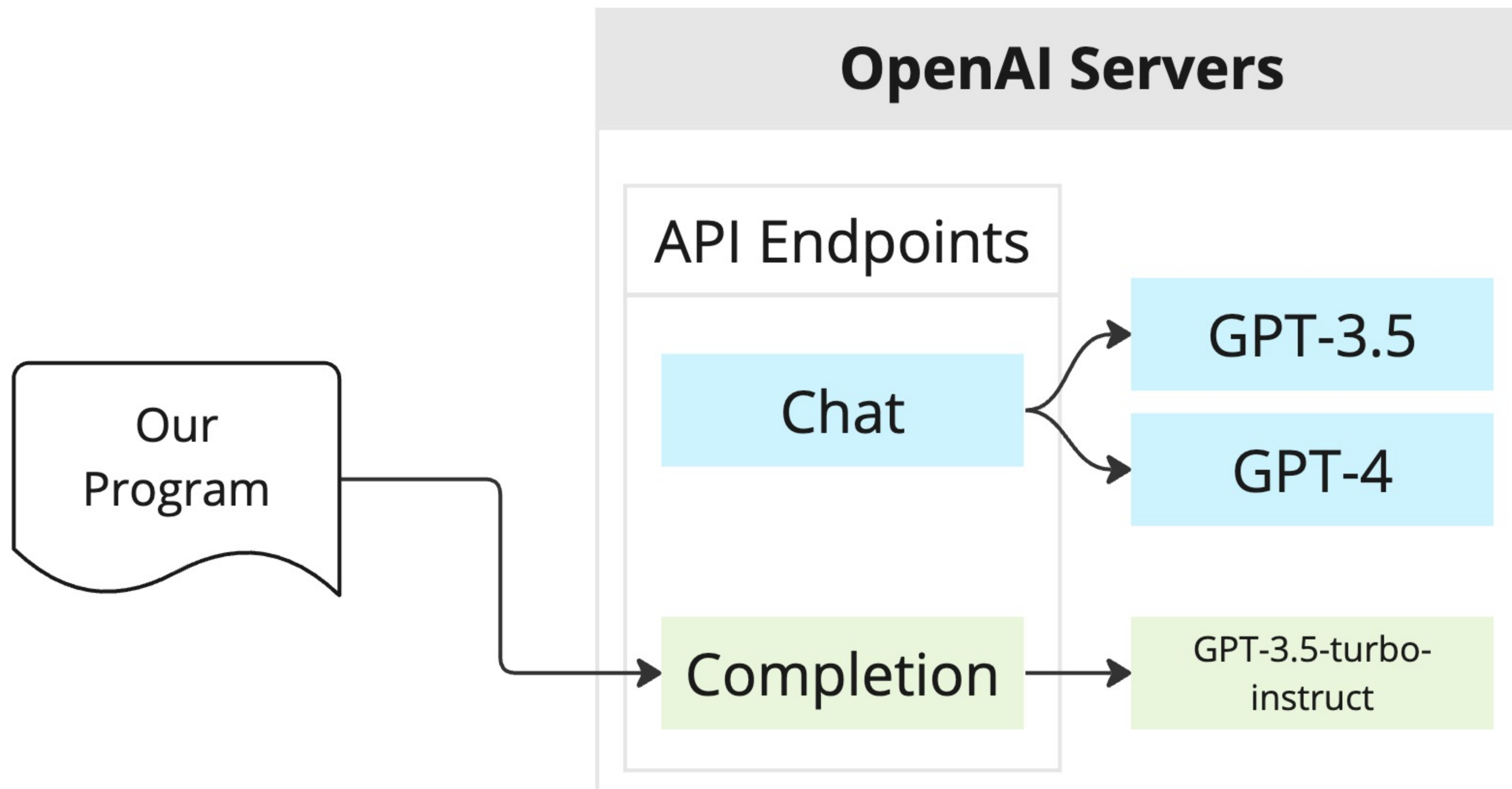
Terminal

>> What is 1+1?

2

>> And 3 more?

5



LLM = Large Language Model

Algorithm that generates text

Many open sourced, many closed
source

Can be hosted by yourself or
others

Examples of LLM's

PaLM

BLOOM

GPT 3.0

LLaMA

GLM

Alpaca

OPT

StableLM

Camel

Most LLM's follow a ***completion*** style of text generation

input

I'm a comedian who jokes about taxes.

Have you ever noticed how taxes are not
fun to pay.



Traditional LLM

platform.openai.com/playground

Some LLM's have been adjusted to use a ***conversational style*** of generation

ChatGPT

Bard

Claude

**Back and forth
messaging**



What is 1+1?



The sum of 1+1 is 2.



and 3 more?



If you add 3 more to the sum of 1+1, the result would be 5.

Text Input



**Completion-Style
LLM**



Text Output

????



Chat-Style LLM



???

A message to customize how the chat bot behaves. Usually set by developers

System Message

User Message

A message created by the user

The diagram illustrates a chat interface with three distinct message types: System, User, and Assistant. The System message is a purple box at the top left, containing a system prompt. The User message is a yellow box at the top right, containing a user query. The Assistant message is a teal box at the bottom right, containing the chat model's response. Arrows point from each box to its corresponding message in the chat interface. The chat interface itself is a large white box with a gray border. It contains a system message on the left, a user message in the middle, and an assistant message on the right. Below the messages is a text input field with a green 'Submit' button and a 'Give us feedback' link.

SYSTEM	USER	ASSISTANT
You are an very unhelpful bot. You answer every question with a terrible pun.	Why do I have to pay taxes?	Because it's a taxing world out there, and the government wants a piece of the pie!

+ Add message

Submit ↺

🚩 Give us feedback

Assistant Message

A message created by the chat model

SYSTEM

You are an very unhelpful bot.
You answer every question
with a terrible pun.

USER

Why do I have to pay taxes?

ASSISTANT

Because it's a taxing world out there,
and the government wants a piece of
the pie!

USER

Why is that?

ASSISTANT

Well, they can't just taxidermy all that
money and leave it lying around, can
they?

⊕ Add message

List of Messages

System Message

User Message

Assistant Message

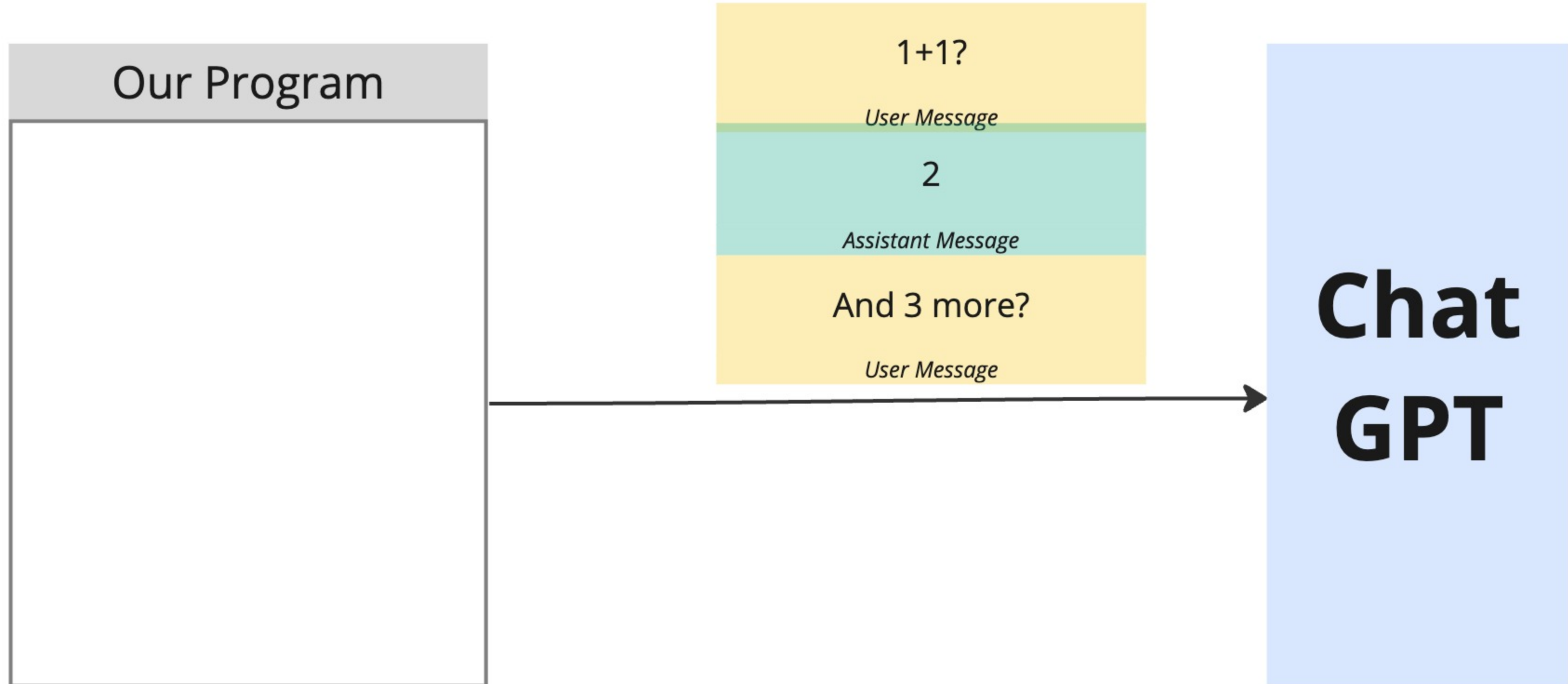
User Message

Assistant Message

This is a kind of unexpected thing

ChatGPT doesn't remember your conversation

You must send the entire message history every time you want to extend a conversation



A lot of LangChain assumes you are using a **completion model**

They are probably planning for a future where ChatGPT isn't the best model around

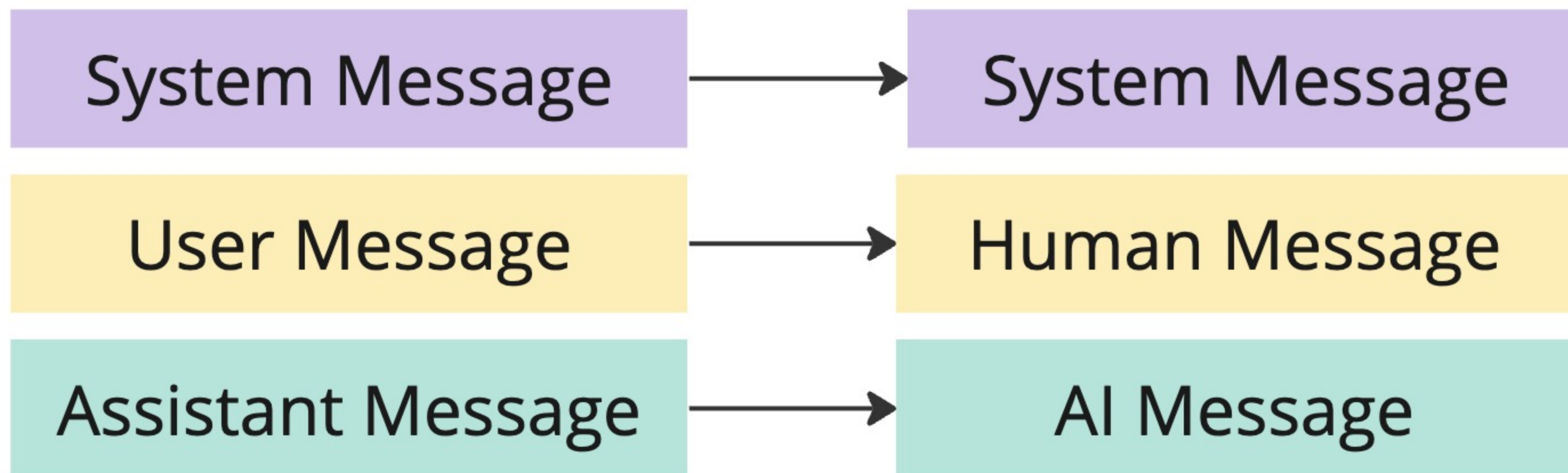
A ton of documentation assumes you are using a **completion model**.
Certain classes are designed to work with **completion models**.

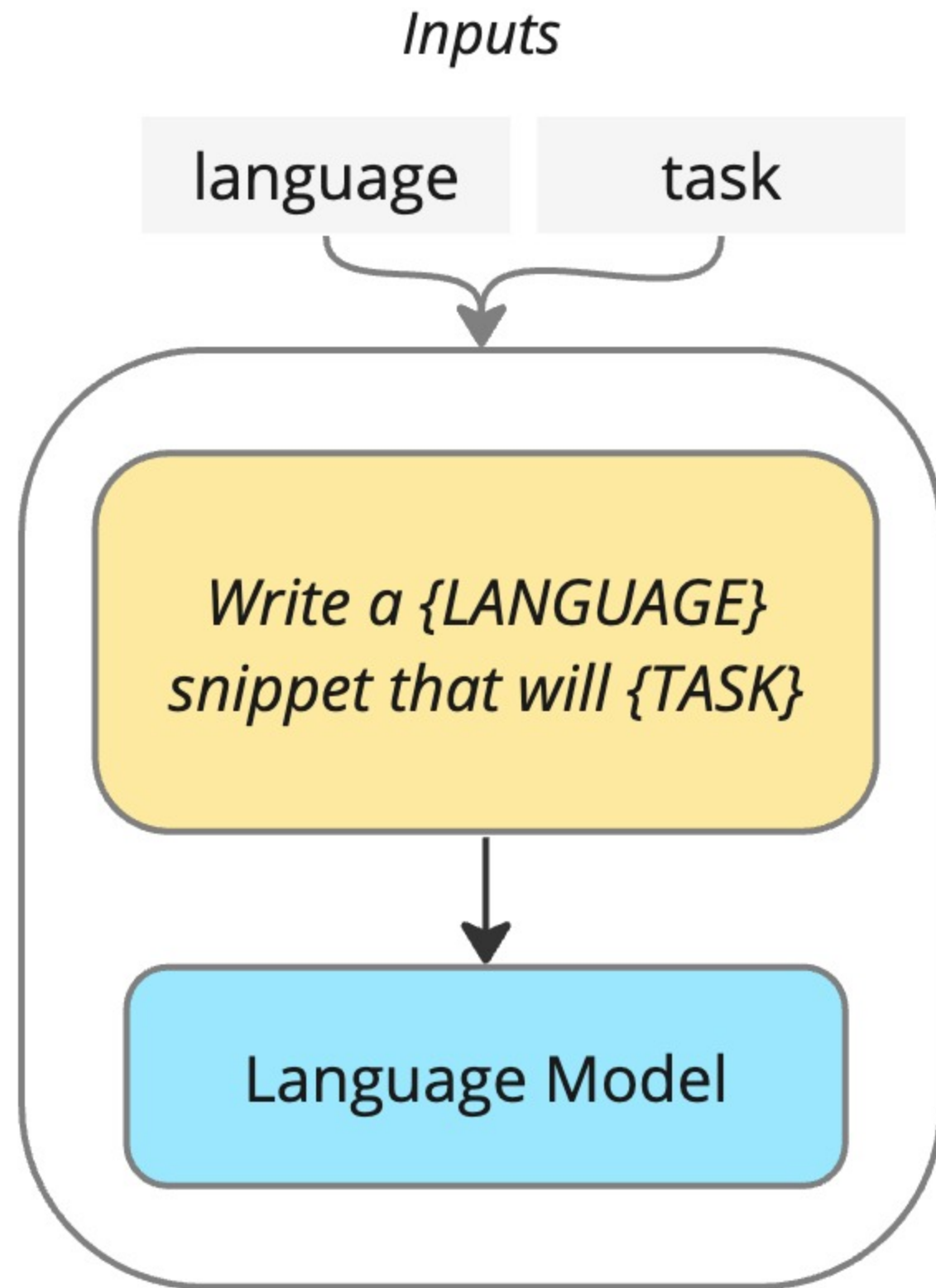
Using a **chat model** requires just a bit of extra work

This course will focus on using **chat models** because they are currently the best

OpenAI Terminology

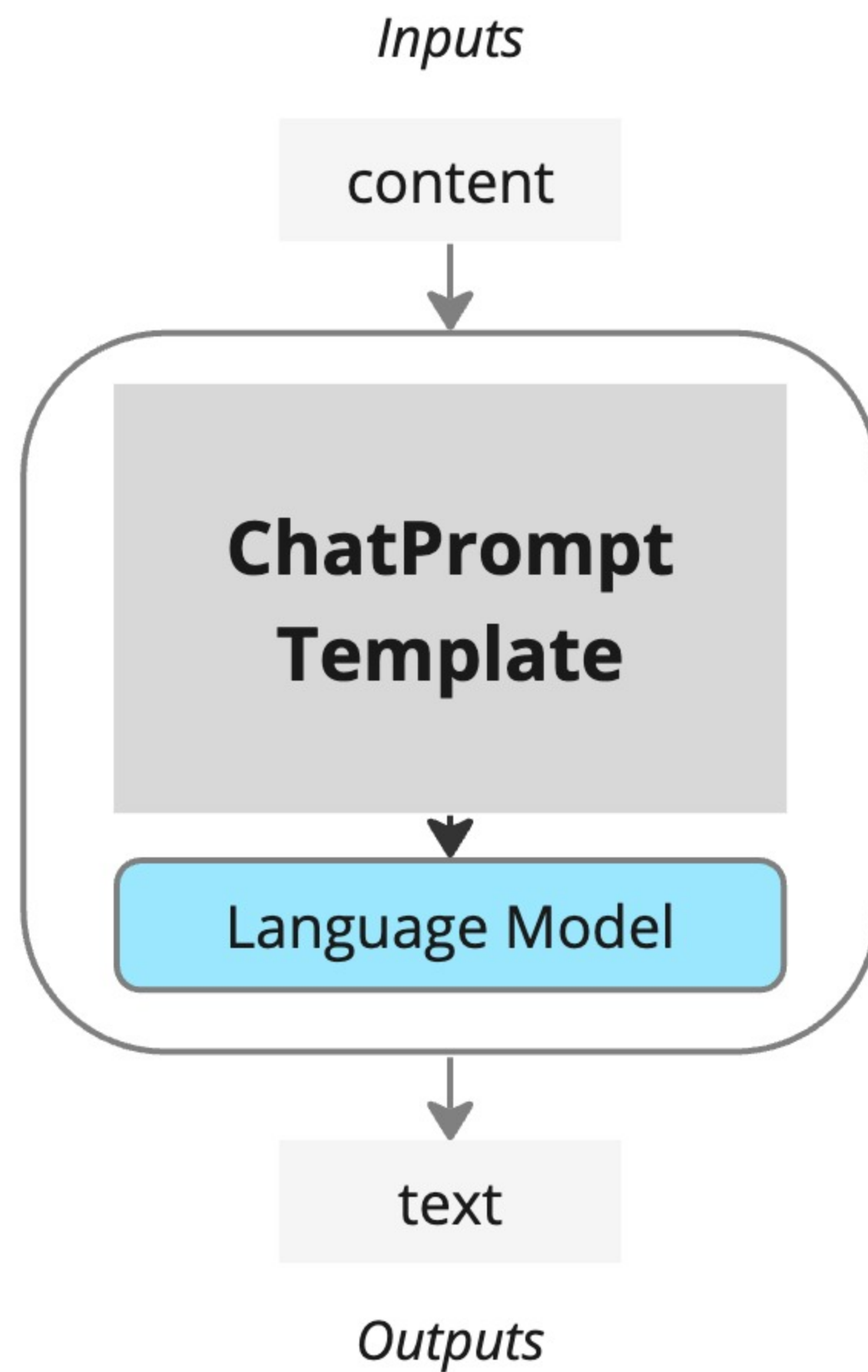
LangChain Terminology





With a PromptTemplate....

Chat Chain



ChatPromptTemplate

SystemMessagePromptTemplate

HumanMessagePromptTemplate

ChatPromptTemplate

System

You are a chatbot specializing in
{subject}

Human

Tell me about why I need to {query}

Inputs

subject	programming
query	Write functions



ChatPromptTemplate

System

You are a chatbot specializing in
{subject}

Human

Tell me about why I need to {query}

You are a chatbot specializing in
programming

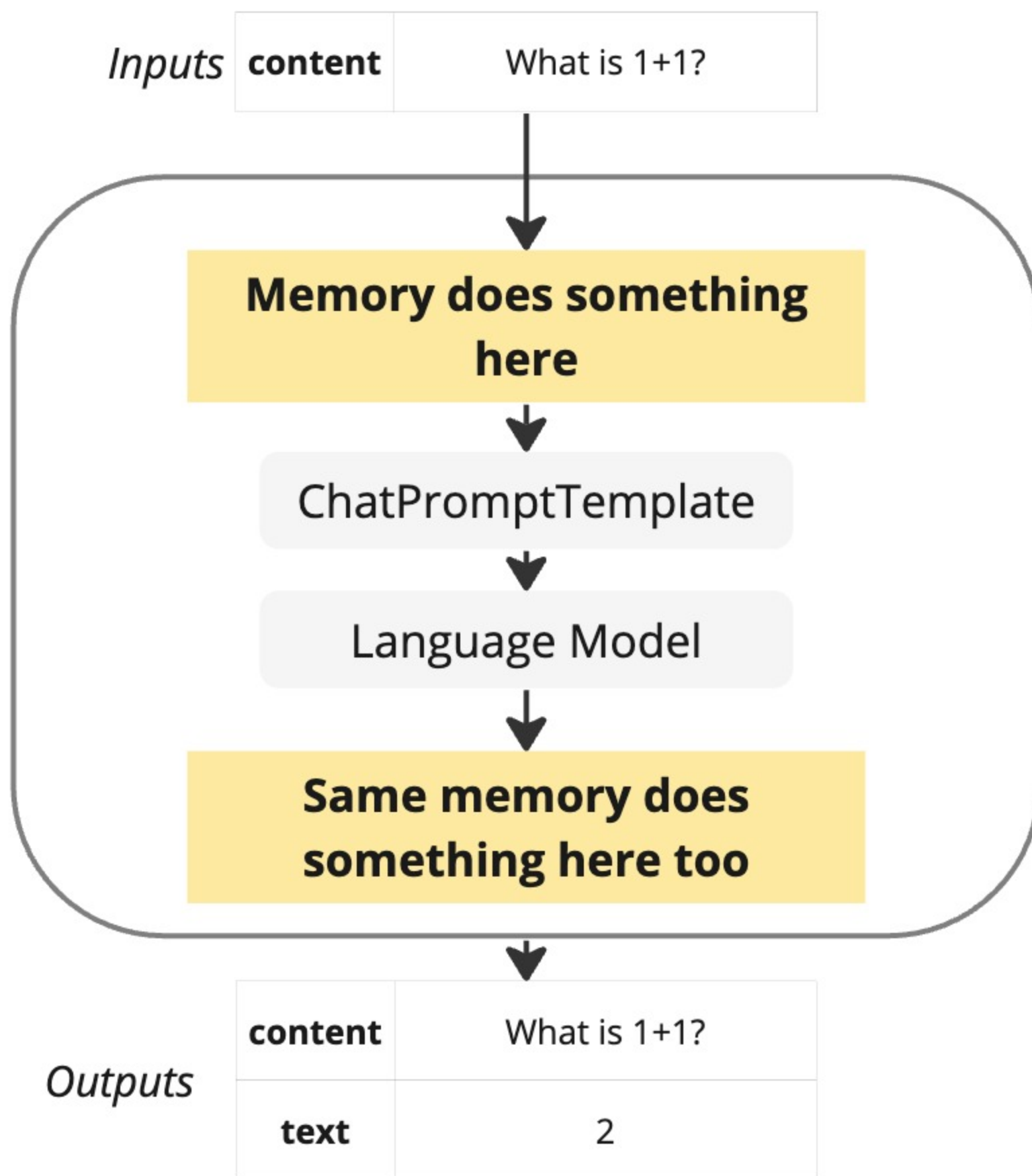
Tell me about why I need to write
functions

**We are going to use a LangChain feature called
'Memory'**

First, a definition

Then an example of what memory can do

Then what memory does in a Chat Chain



Memory is used to store data in a chain

Used for a ton of stuff, not just storing the list of messages

Inputs

content

What is 1+1?

Memory

content

What is 1+1?

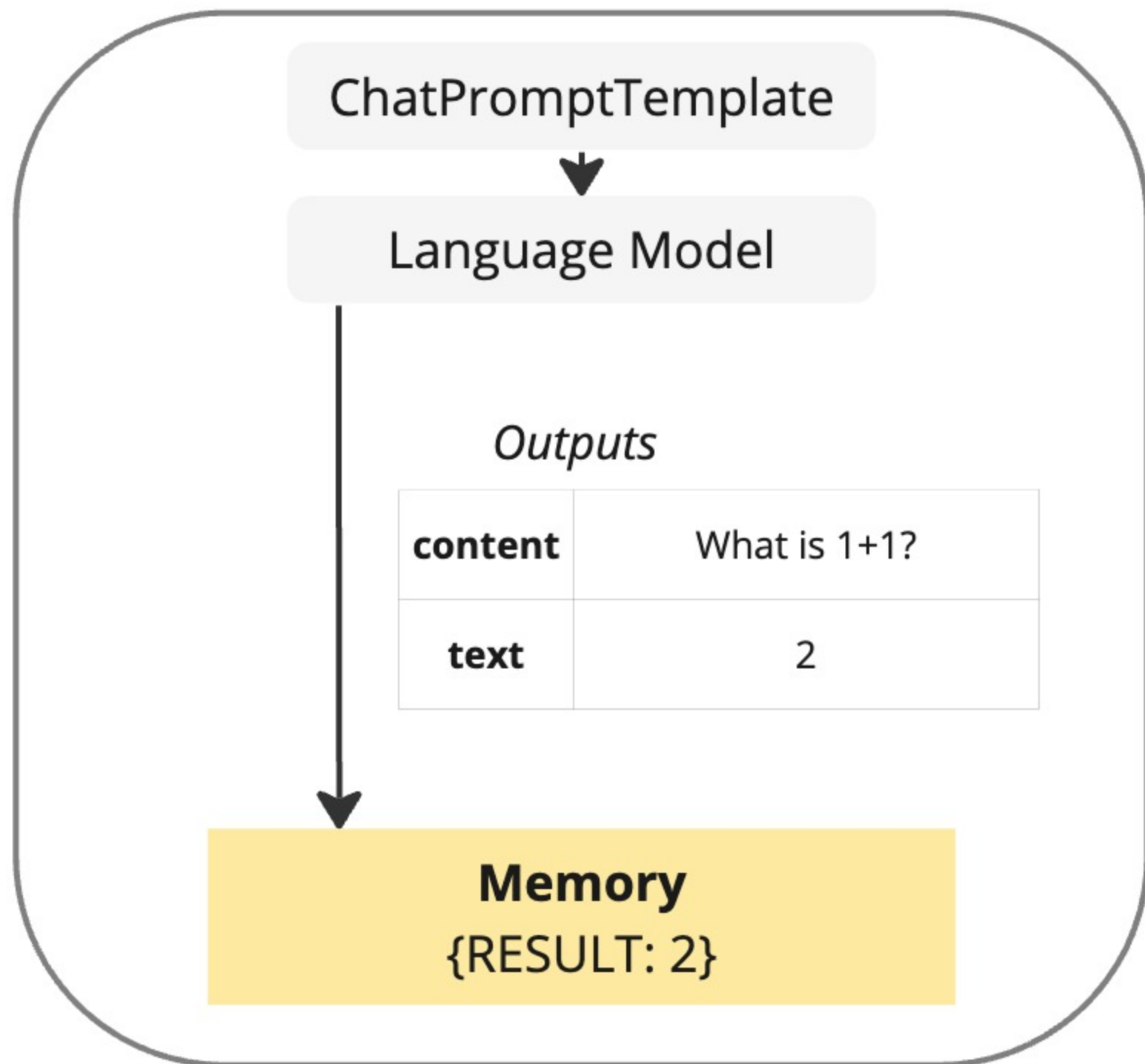
messages

[]

ChatPromptTemplate

Language Model

When you run a chain, the memory receives the input variables and has the ability to add in additional variables



After the model runs, the output variables are sent to memory

Memory has a chance to inspect the result and store some part of it

ConversationTokenBufferMemory

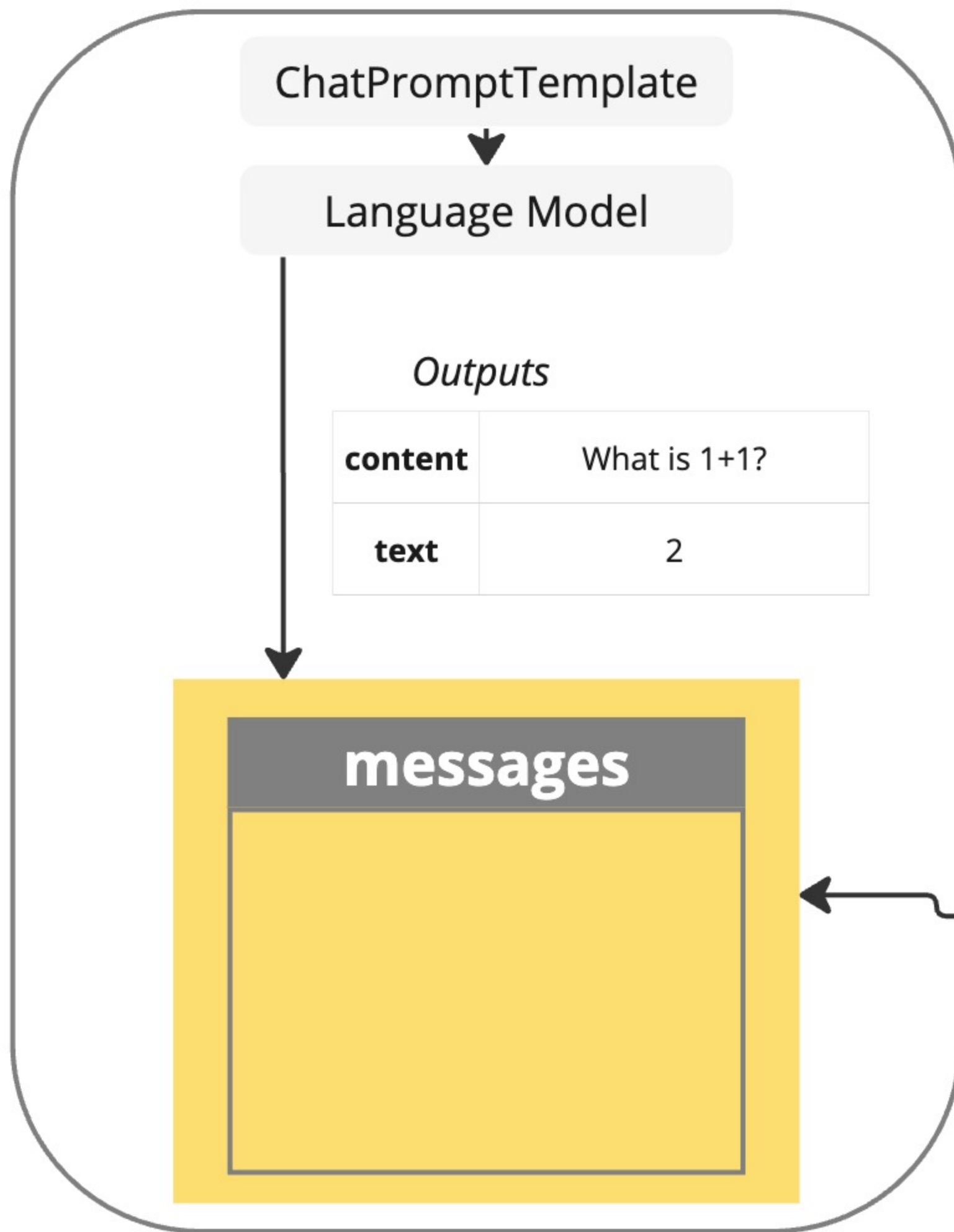
CombinedMemory

ConversationBufferWindowMemory

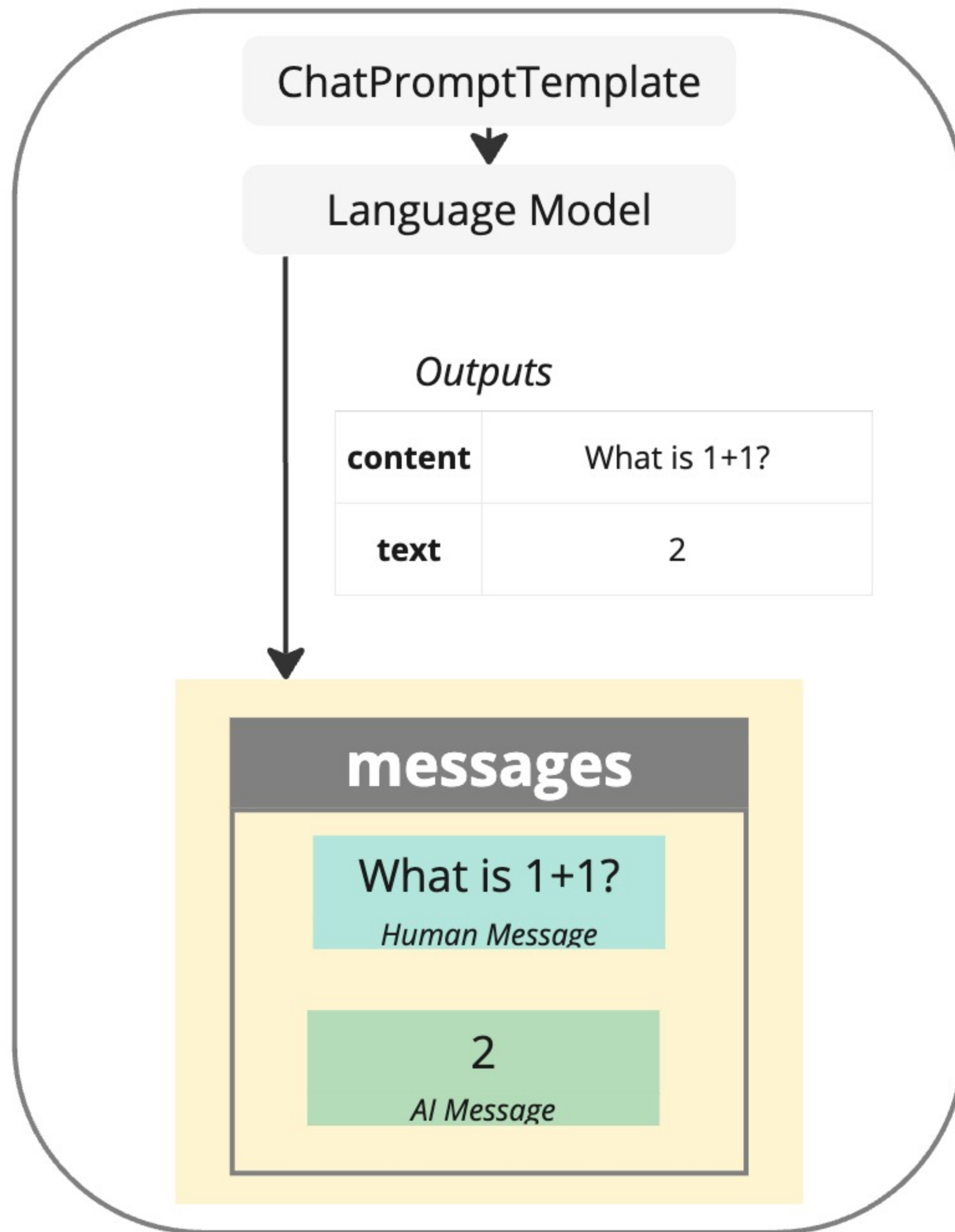
ConversationBufferMemory

LangChain has many
kinds of memory.

We are going to use one called
'ConversationBufferMemory'



Memory can have some storage. In this case, it is a list called 'messages'



The memory adds a 'HumanMessage' and an 'AIMessage' to the list

Inputs

content

And 3 more?

messages

What is 1+1?

Human Message

2

AI Message

content

And 3 more?

messages

["What is 1+1?",
"2"]

ChatPromptTemplate

Language Model

This is super nasty stuff

**I normally wouldn't show it so early in a
course**

**Unfortunately, it is the bare minimum to
use Chat-style models with LangChain
right now**

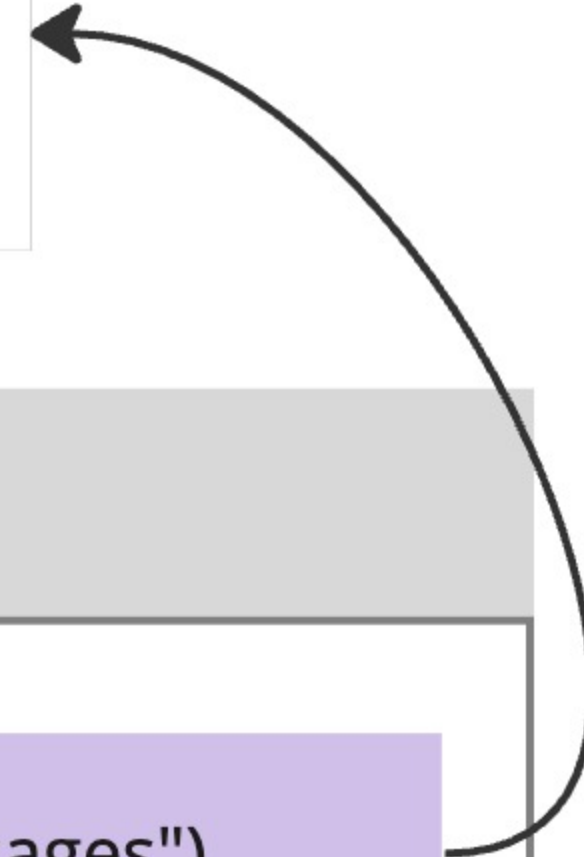
content	And 3 more?
messages	[HumanMessage("What is 1+1?"), AIMessage("2")]



ChatPromptTemplate

MessagesPlaceholder(variable_name="messages")

HumanMessagePromptTemplate.from_template("{content}")



content	And 3 more?
messages	[HumanMessage("What is 1+1?"), AIMessage("2")]

