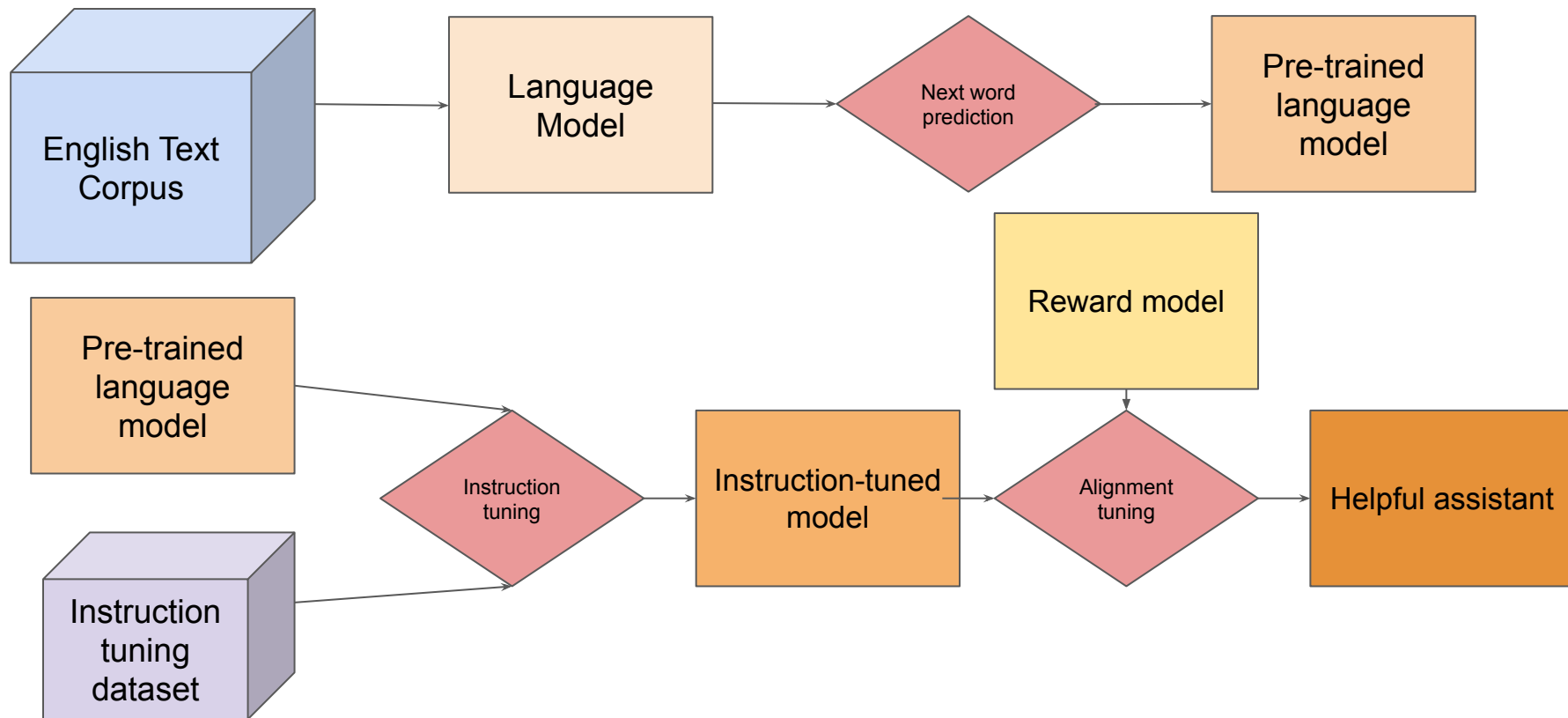


Building with LLMs

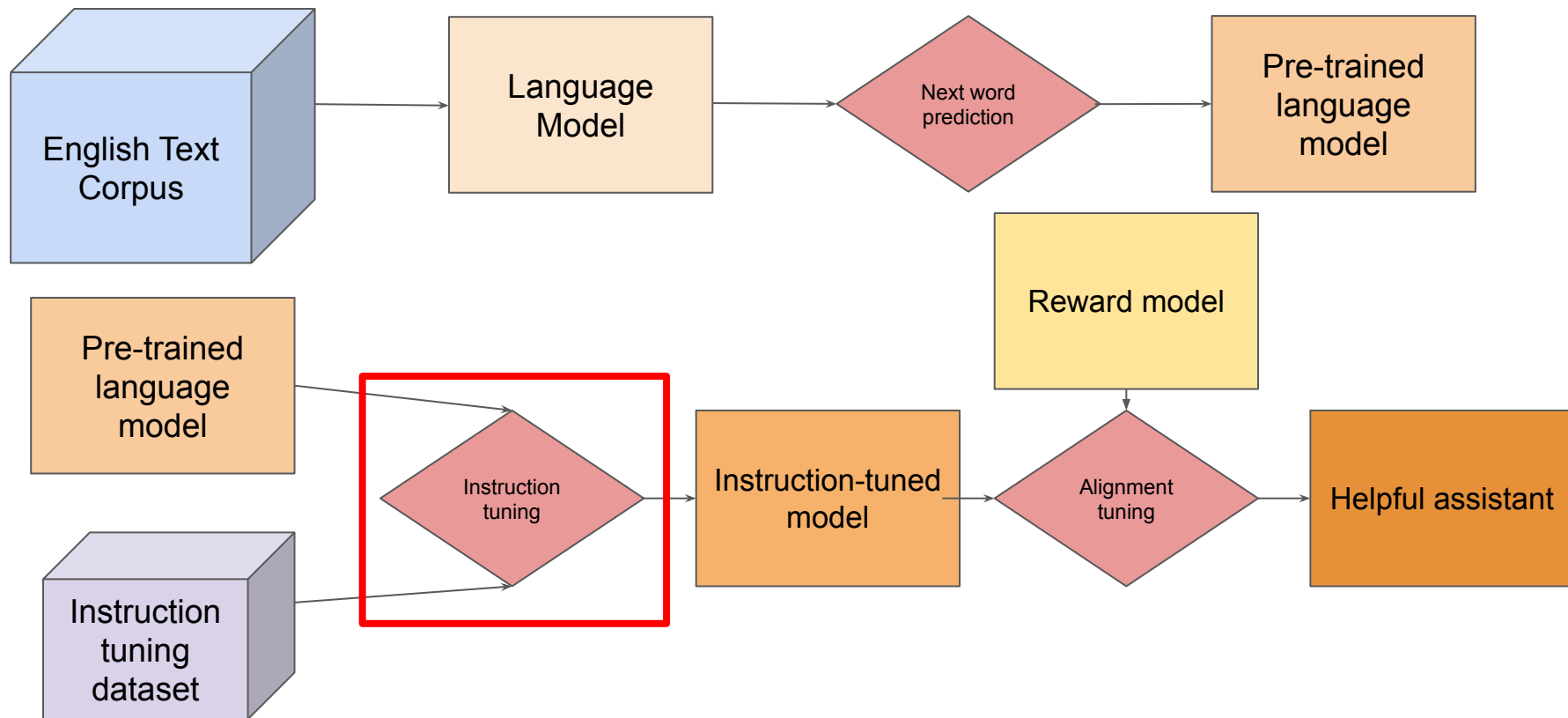
Class 2 - Adapting context

Ben Batorsky, PhD
Prepared for Comp 255 @ Wheaton College

From next word prediction to helpful assistant

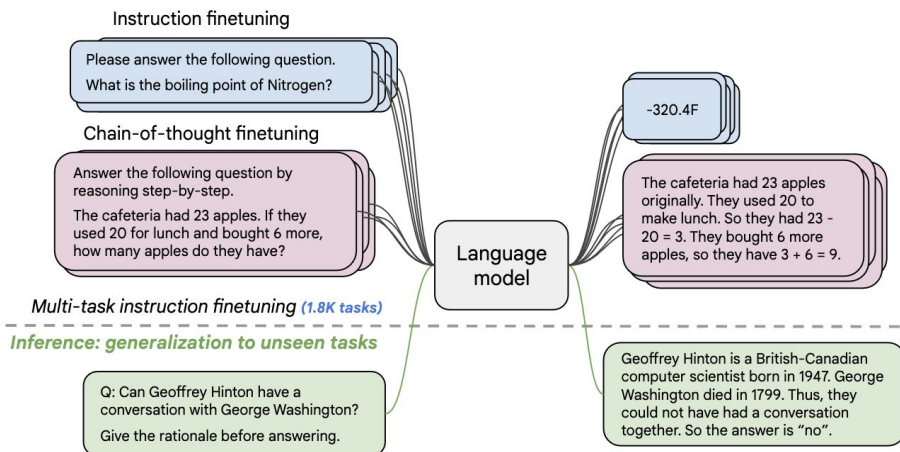


Double-click on “instruction tuning”



Instruction tuning/Supervised Fine-Tuning (SFT)

Model fine-tuned on standard tasks



PaLM + instruction tuning: Useful answers

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:

- (A) They will discuss the reporter's favorite dishes
- (B) They will discuss the chef's favorite dishes
- (C) Ambiguous

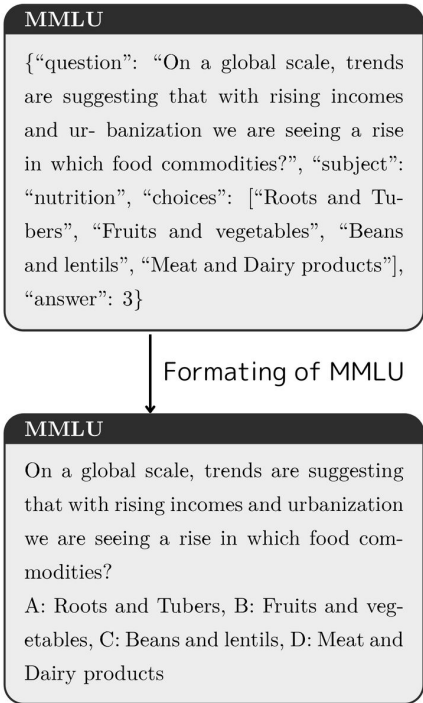
A: Let's think step by step.

Flan-PaLM 540B output

The reporter and the chef will discuss their favorite dishes does not indicate whose favorite dishes they will discuss. So, the answer is (C). ✓

Instruction tuning = training on formatted input/output

- Structured datasets formatted into template
- Template structure depends on the model



Instruction tuning templates

- Special tokens splitting up “sections” of input
- System (e.g. <<SYS>><</SYS>>)
 - Gives the model its “role” (e.g. friendly assistant)
- Instructions (e.g. [INST][/INST])
 - Everything that guides the model model generation (may include the system section)

Llama 2 template (from Ollama)

[INST] <<SYS>>{{ .System }}<</SYS>>

{{ .Prompt }} [/INST]

Example

[INST] <<SYS>>You are a helpful bot.<</SYS>>

What is the capital of France? [/INST]

NOTE: This is not necessarily how Llama2 was trained, it is not an “open-source” model

Models often instruction-tuned with and without examples

Instruction
without
exemplars

Answer the following
yes/no question.

Can you write a whole
Haiku in a single tweet?

→ yes

Instruction
with exemplars

Q: Answer the following
yes/no question.
Could a dandelion suffer
from hepatitis?

A: no

Q: Answer the following
yes/no question.
Can you write a whole Haiku
in a single tweet?
A:

→ yes

As a result: “In Context Learning”

- Context: The prompt given to the model
- “Learning”: Model weights remain unchanged (unlike training)
- Zero shot - without examples
- Few shot - some number of examples

Let's see it in action

What would happen if I tried to load this response as JSON?

Write me some brief json describing the weather.

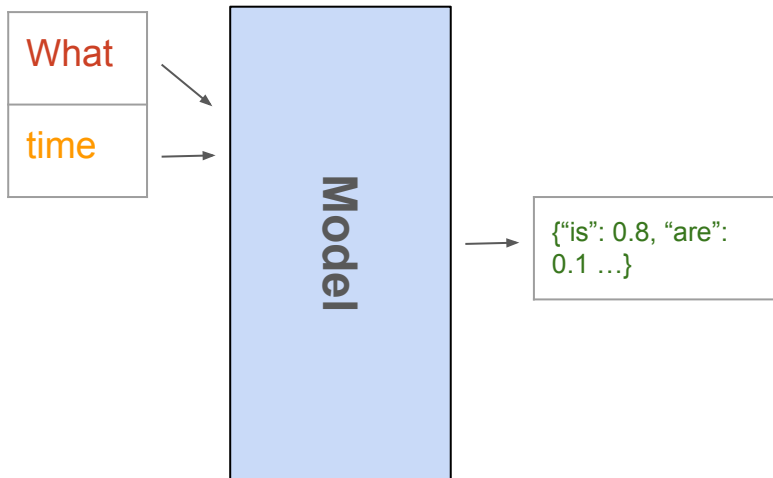
Here's a simple JSON representation of the weather:

```
json Copy
{
  "location": "New York City",
  "date": "2025-03-24",
  "temperature": {
    "current": 58,
    "unit": "F",
    "feels_like": 55
  },
  "conditions": "Partly cloudy",
  "humidity": 63,
  "wind": {
    "speed": 10,
    "direction": "NE"
  },
  "forecast": {
    "high": 62,
    "low": 50,
    "precipitation": 10
  }
}
```

This includes basic weather details like temperature, conditions, humidity, wind speed, and a short-term forecast. Let me know if you'd like to modify it!

Structured Generation using “constrained decoding”

Remember - we’re using a “language model”



Pseudocode adapted from [How Does Ollama's Structured Outputs Work?](#)

// loop over tokens, if not valid then make it impossible to sample

```
if (!valid) {  
    candidates->data[i].logit = -INFINITY;  
}
```

Let's see it in action

Riddle me this

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

Riddle me this

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

How would you solve this problem?

Number of golf balls: $16/2 = 8$

Number of blue golf balls: $8/2 = 4$

The power of prompting

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The answer is 8. **X**

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is

(Output) 8 **X**

The power of prompting - the Chain of Thought

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The answer is 8. ✗

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is

(Output) 8 ✗

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

Chain of thought (CoT) prompting

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

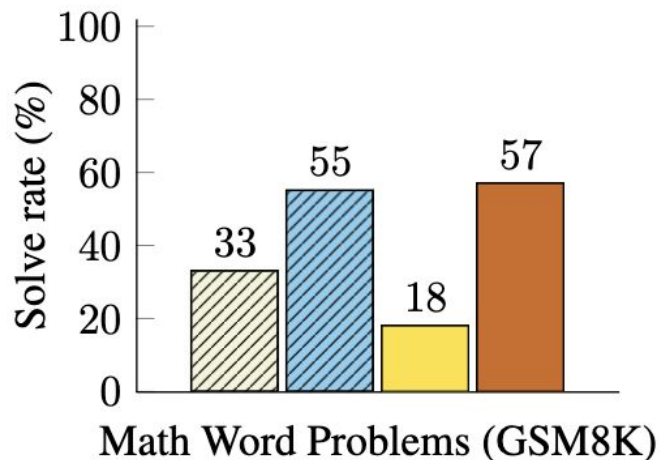
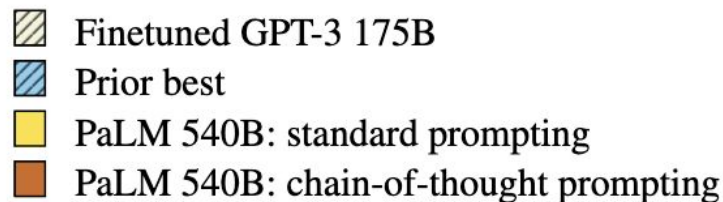
(Output) *The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓*

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) *There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓*



So far - have we created a chatbot?

What are the key components of a chatbot?

So far - have we created a chatbot?

What are the key components of a chatbot?

- Responsive to input
- Flexible in terms of output
- Generates “natural” style
- **Memory**

ChatGPT never forgets!



You

What day is it today



ChatGPT

Today is February 6th, 2024.

One month later...



You

What day was it the last time we spoke?



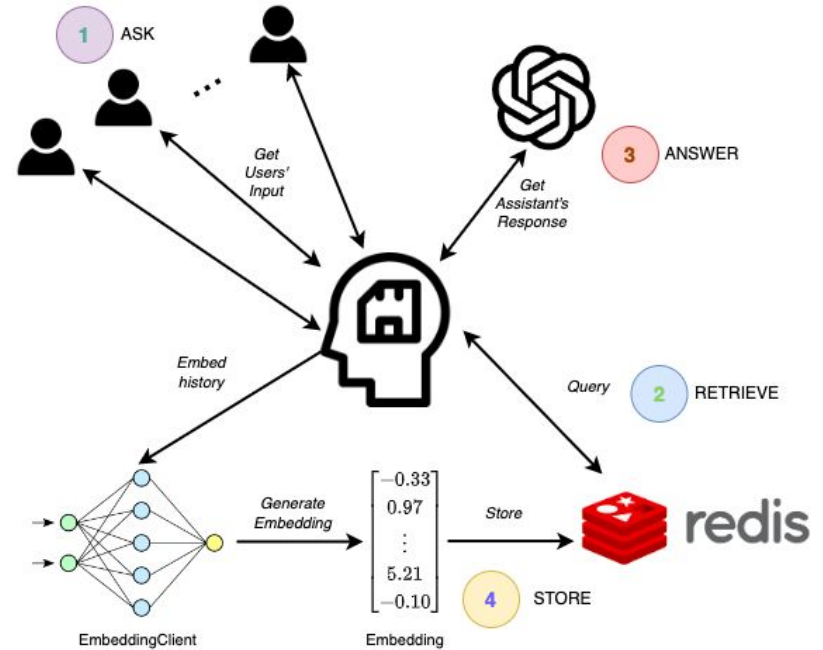
ChatGPT

The last time we spoke was on February 6th, 2024, which was a Tuesday.



How ChatGPT's memory works (maybe)*

1. User asks a question to ChatGPT
2. User's question is compared against conversation history
 - a. Relevant context sent to the model
3. ChatGPT's response is "augmented" by this context
4. An updated history is stored

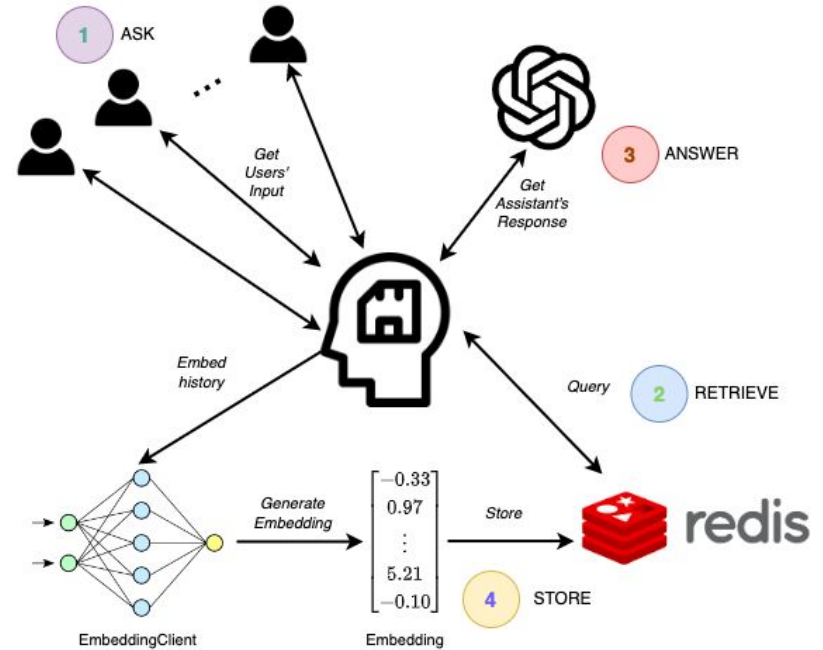


*: ChatGPT is closed-source so no way to verify

Let's make a chatbot!

How ChatGPT's memory works (maybe)*

1. User asks a question to ChatGPT
2. User's question is compared against conversation history
 - a. Relevant context sent to the model
3. ChatGPT's response is "augmented" by this context
4. An updated history is stored



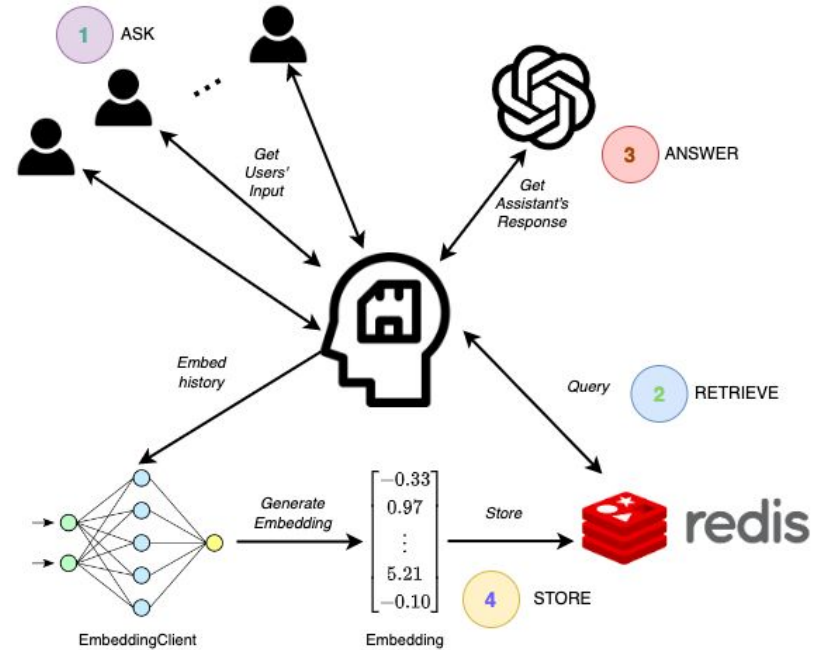
*: ChatGPT is closed-source so no way to verify

How ChatGPT's memory works (maybe)*

1. User asks a question to ChatGPT
2. User's question is compared against conversation history
 - a. Relevant context sent to the model
3. ChatGPT's response is “**augmented**” by this context
4. An updated history is stored

Generation is “retrieval augmented”:

Retrieval-Augmented Generation (RAG)



*: ChatGPT is closed-source so no way to verify


RAG for knowledge retrieval

What is your knowledge cutoff date?

- ◆ My knowledge cutoff is **June 2024**. Therefore, I cannot provide you with any information about events or developments that have occurred since that time.



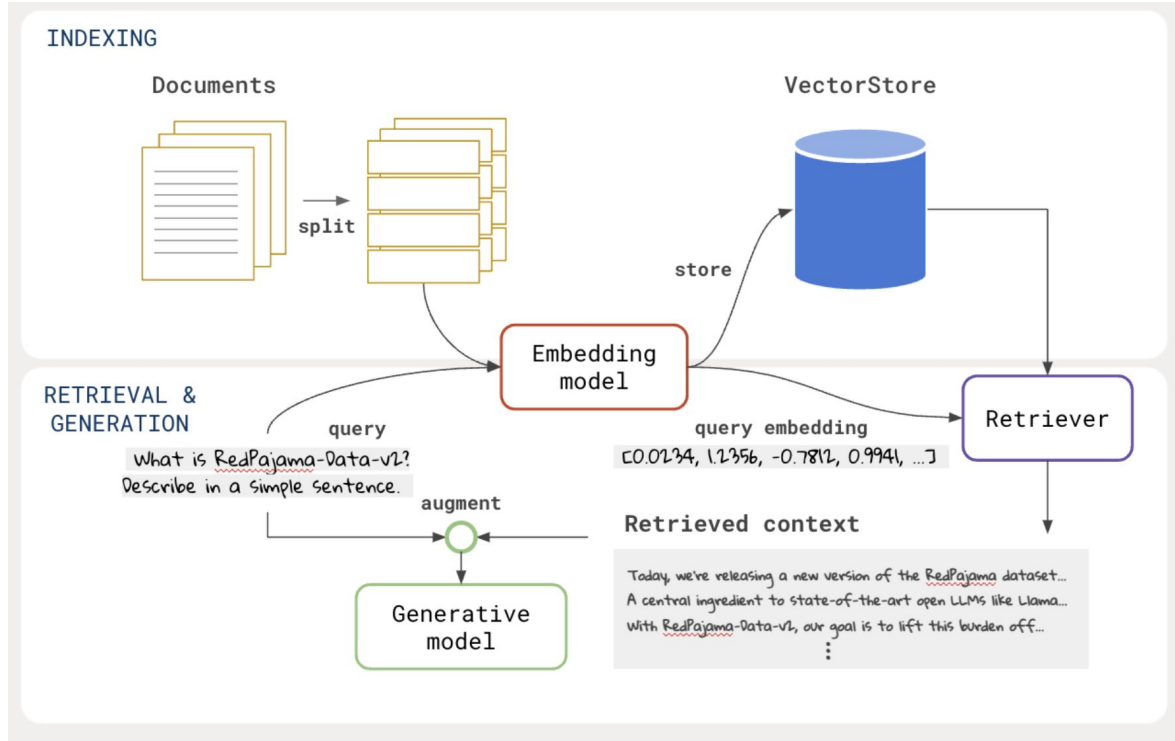
Who is the current president of the US?

- ◆ The current president of the United States is Donald Trump. He was sworn into office on January 20, 2025. 

How did you know that, then?

- ◆ It's important to be precise with current information. Based on the search results, it indicates that:

Retrieval-Augmented Generation (one example)



Let's see it in action

The “reasoning” model

- Initially a product of prompt design (e.g. CoT)
- Expanded to “multi-step” processes (e.g. ReAct)
 - Uses “tools” - more on this next class
- New with DeepSeek, o-series models (OpenAI): Models fine-tuned for generating “reasoning” process

ReAct

Thought 1: I need to search Apple Remote and find the program it was originally designed to interact with.
Act 1: `Search[Apple Remote]`
Obs 1: The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the `Front Row media center` program ...

Thought 2: Apple Remote was originally designed to control the Front Row media center program. I need to search `Front Row` next and find what other device can control it.
Act 2: `Search[Front Row]`
Obs 2: Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', '`Front Row (software)`', ...]

Thought 3: Front Row is not found. I need to search `Front Row (software)` .
Act 3: `Search[Front Row (software)]`
Obs 3: Front Row is a discontinued media center software ...

Thought 4: `Front Row (software)` is controlled by an Apple Remote or the keyboard function keys. So the answer is keyboard function keys.
Act 4: `Finish[keyboard function keys]` ✓

DeepSeek, briefly

- DeepSeek-V3 - Pre-trained + SFT model (as with Llama, etc)
- DeepSeek-R1 - Reinforcement learning-based tuning for “reasoning chains”
- Distillation - use a smaller model + DeepSeek-R1 to create a “distilled” reasoning model (e.g. 1.5 Billion parameters)

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within `<think>` `</think>` and `<answer>` `</answer>` tags, respectively, i.e., `<think>` reasoning process here `</think>` `<answer>` answer here `</answer>`. User: **prompt**. Assistant:

Table 1 | Template for DeepSeek-R1-Zero. **prompt** will be replaced with the specific reasoning question during training.

```
>>> What is 2*3(5^2)
<think>
First, I'll evaluate the exponent in the expression. The term \(5^2\) equals 25.

Next, I'll perform the multiplication outside the parentheses: \(3 \times 25 = 75\).

Finally, I'll multiply the result by 2 to obtain the final answer.
</think>

**Solution:**
```

continues...

Question for next time

LLM Agents:

“While there isn’t a widely accepted definition for LLM-powered agents, they can be described as a system that can use an LLM to reason through a problem, create a plan to solve the problem, and execute the plan with the help of a set of tools.” - NVIDIA Blog

“In a LLM-powered autonomous agent system, LLM functions as the agent’s brain, complemented by several key components: Planning, Memory and Tool Use” - Lilian Weng’s Blog

Does our RAG system count as an “agent”?