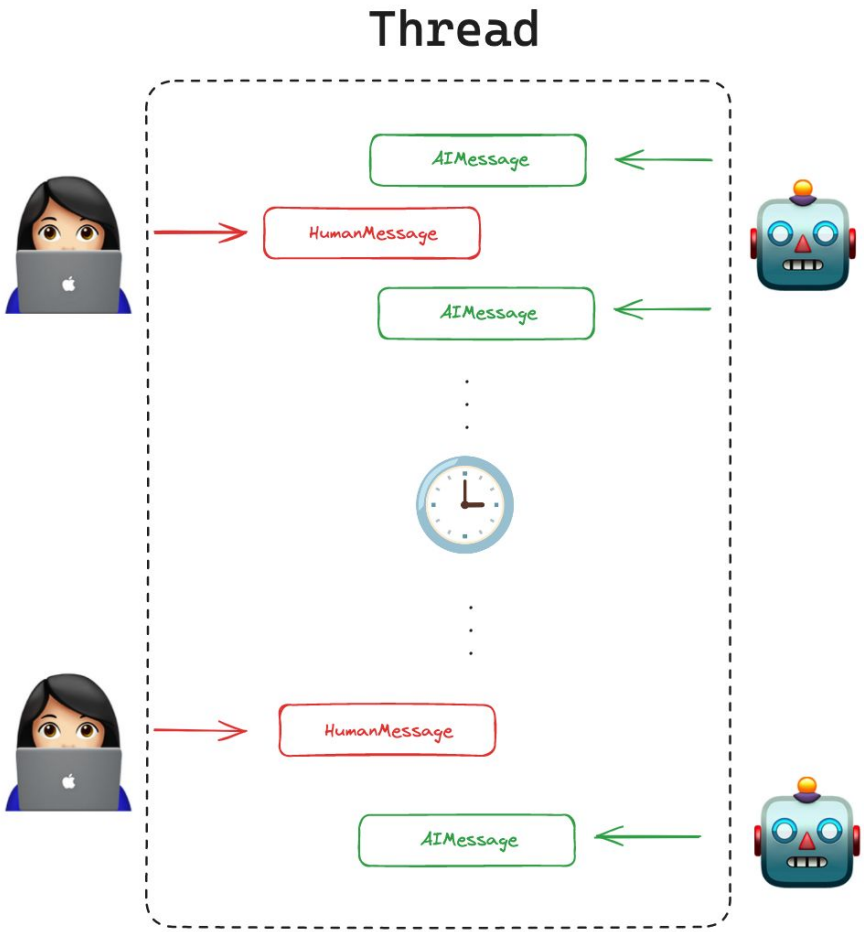
LangChain Academy

Introduction to  LangGraph

Memory is a cognitive function that allows people to store, retrieve, and use information to understand their present and future.

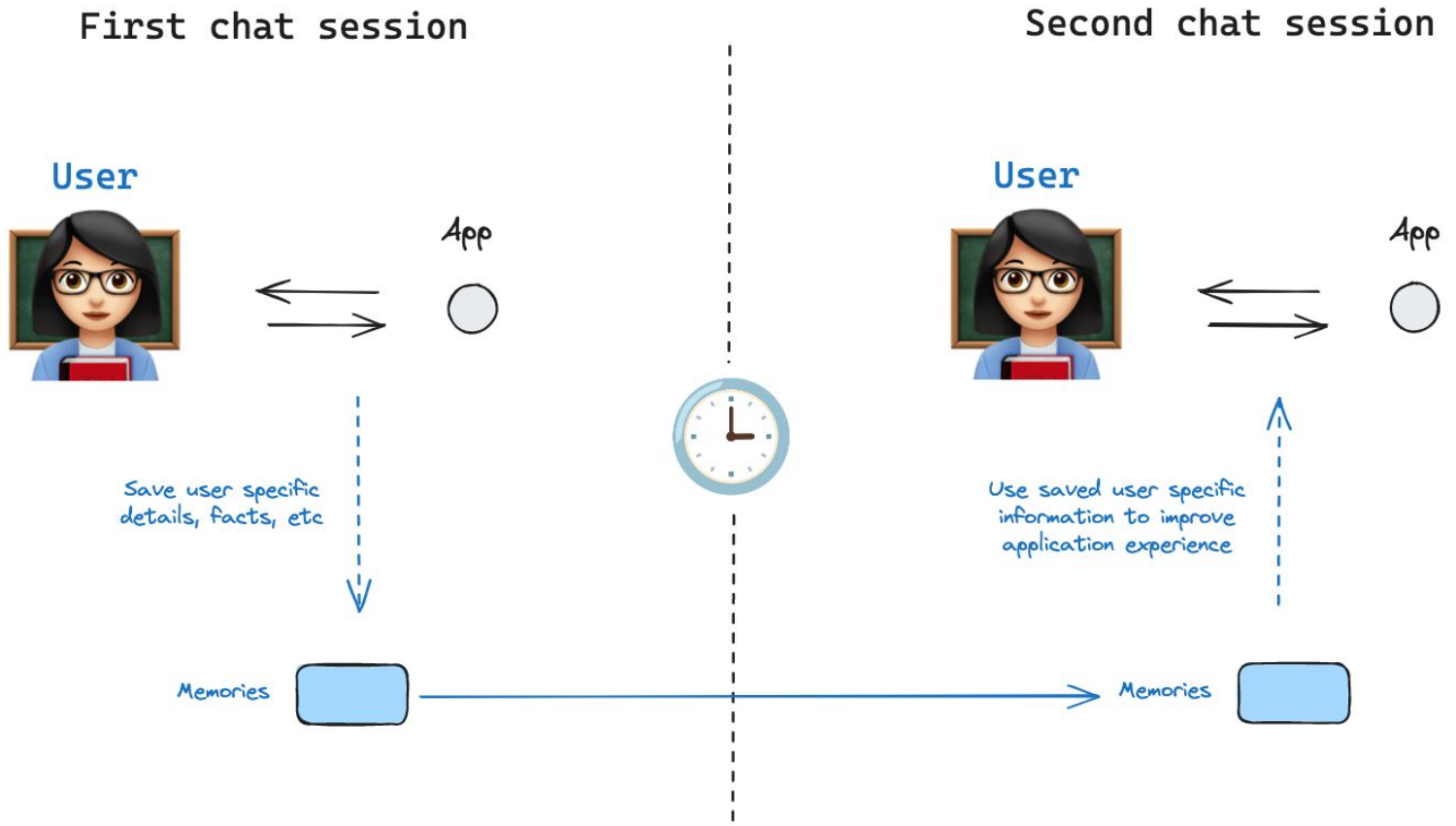
Cognitive neuroscience perspective on memory: overview and summary

1 Within session (thread) memory



Thread-scoped memory

1 Across session (thread) memory



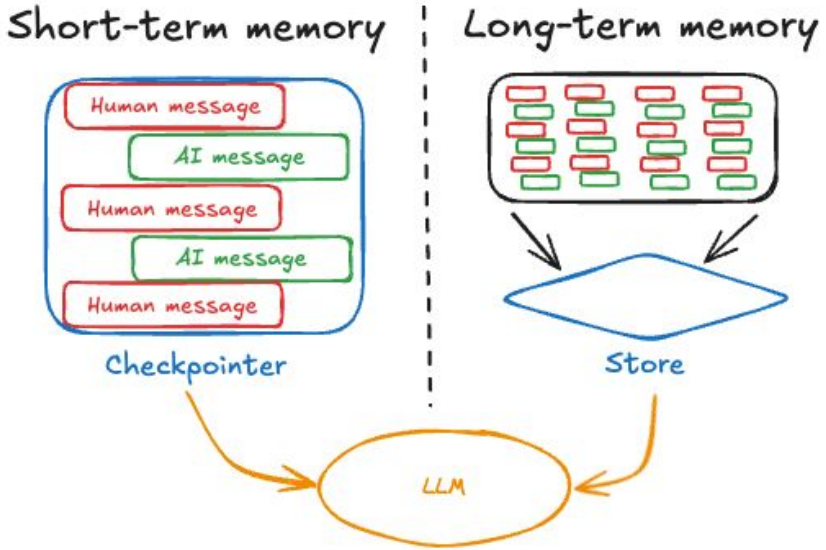
[Across-thread memory]

1 Comparison

	Short-term	Long-term
Scope	Within session (thread)	Across session (thread)
Example use-case	Persist conversational history, allow interruptions in a chat (e.g., if user is idle or to allow human-in-the-loop)	Remember information about a specific user across all chat sessions
LangGraph usage	Checkpointer	Store

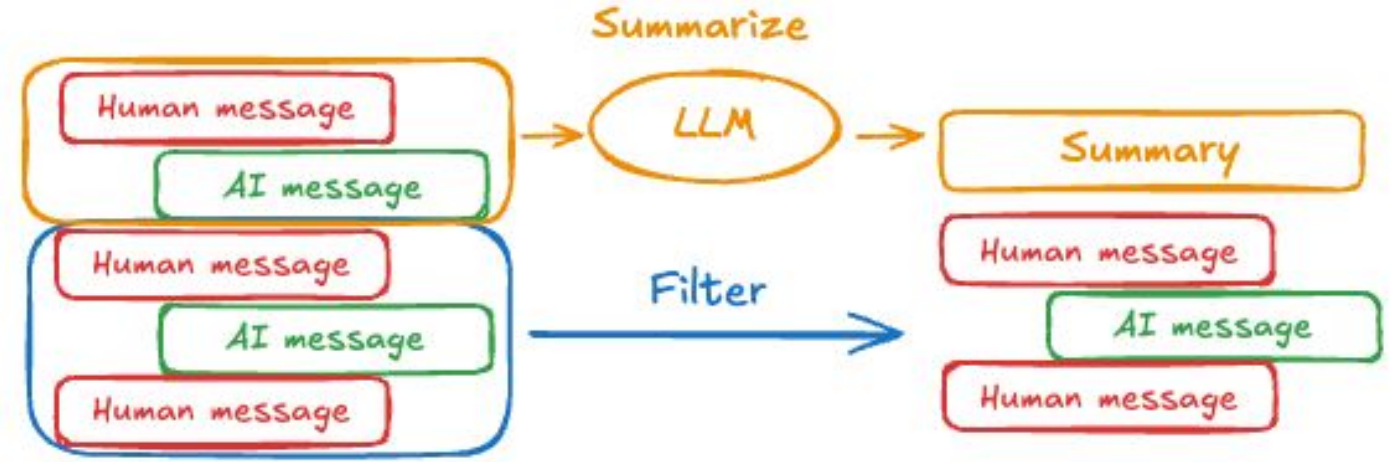
Memory conceptual documentation

1 Comparison



Memory conceptual documentation

2 Short-term memory

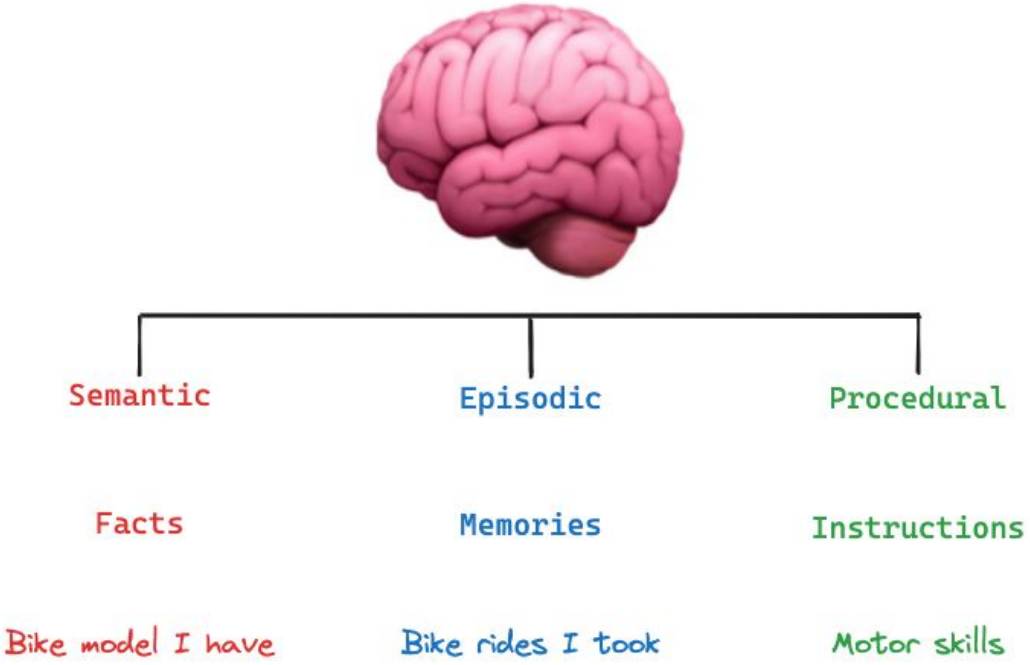


[Short-term memory conceptual documentation]

What is the type of memory?

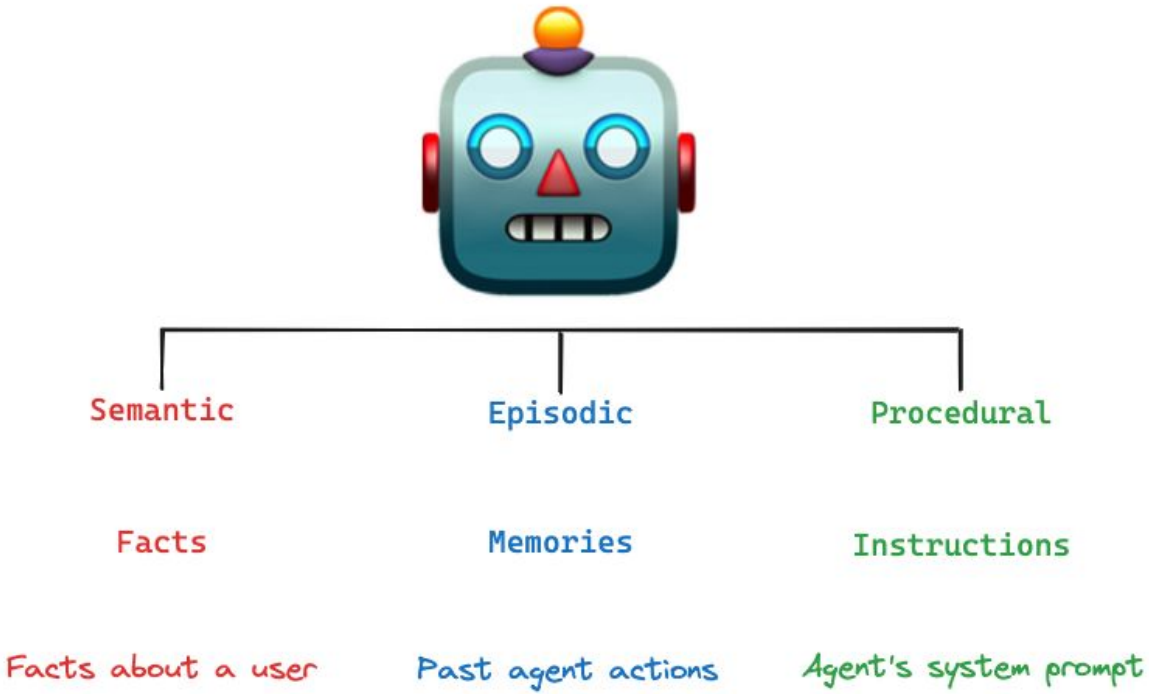
When do you want to update memories?

3 Long-term memory: Humans

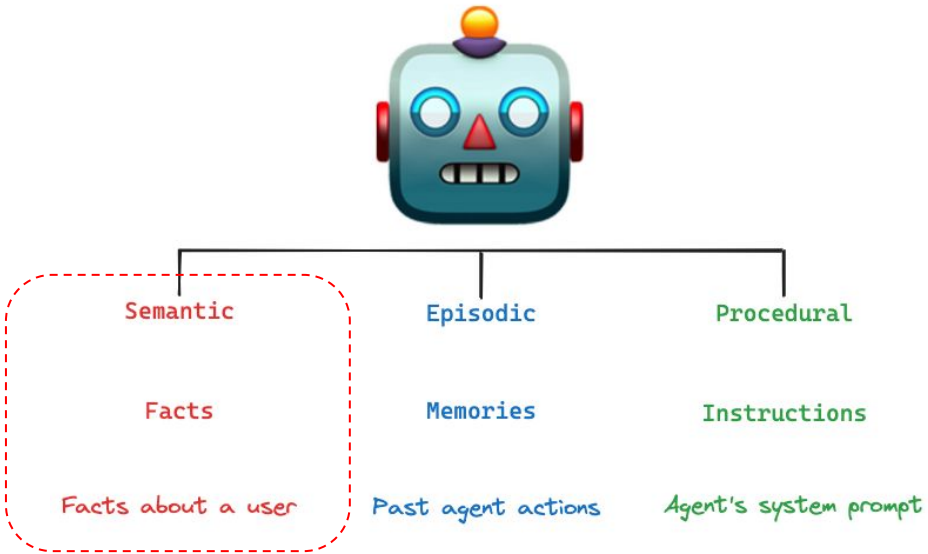


https://www.simplypsychology.org/long-term-memory.html

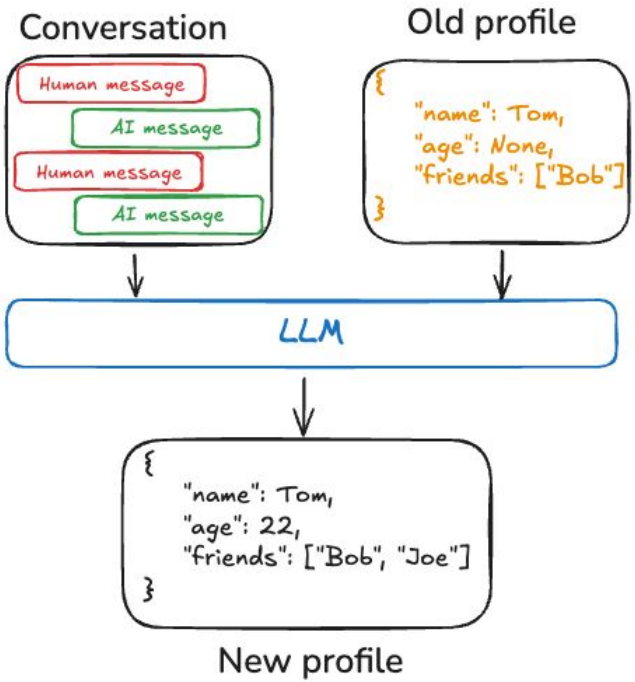
3 Long-term memory: Agents



How to structure facts?

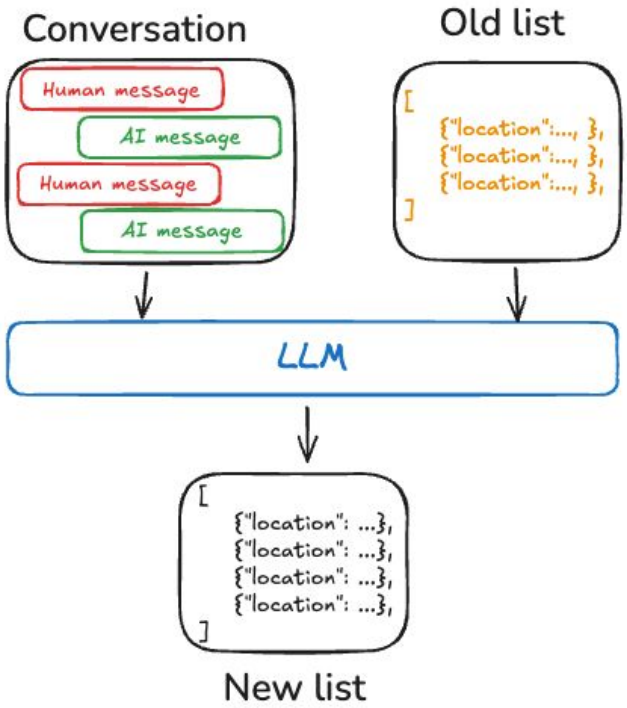


3 Long-term memory: Semantic (Profile)



Memory conceptual documentation

3 Long-term memory: Semantic (Collection)

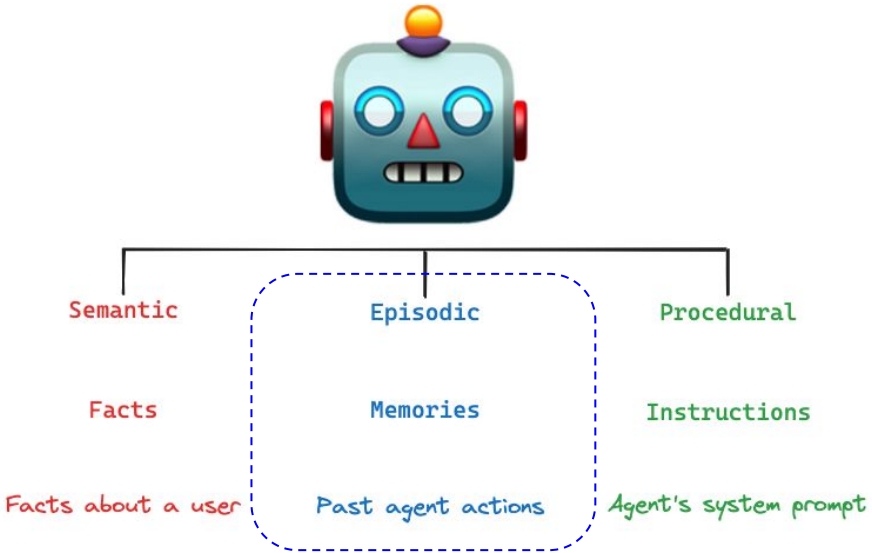


Memory conceptual documentation

	Type	Pro	Con
Profile	Single document	Easily retrieved single representation	Challenging to maintain as it grows larger
List	List of documents	Allows for smaller, narrowly scoped memories w/ easy addition of new information	Retrieval can be challenge as the list grows larger

Memory conceptual documentation

How to present past actions to an agent?



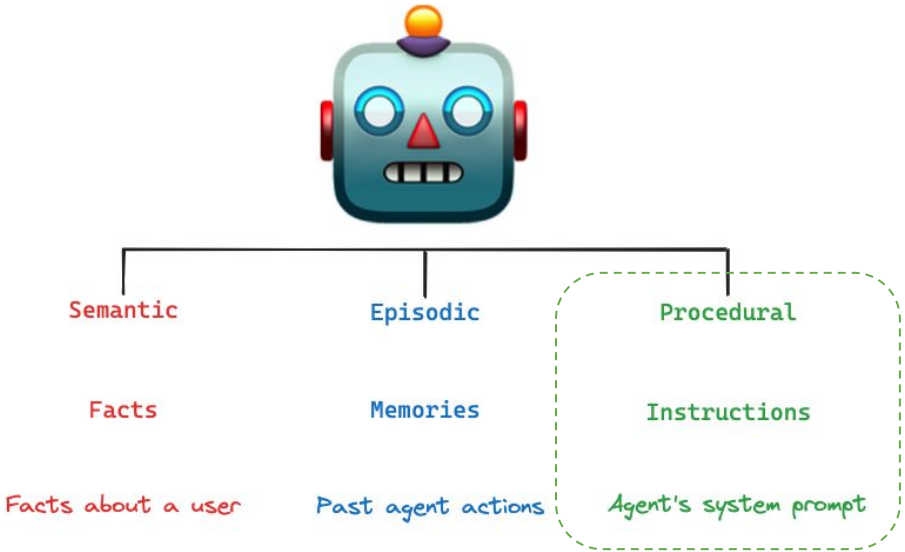
3 Long-term memory: Episodic (Few shot)

Prior reasoning trajectories

```
system: You are requested to solve math questions in an alternate
        universe. The operations have been altered to yield differ
        than expected. Do not guess the answer or rely on your inn
        of math. Use the provided tools to answer the question. Wh
        associativity and commutativity apply, distributivity does
        Answer the question using the fewest possible tools. Only
        numeric response without any clarifications. Here are some
        conversations of the user interacting with the AI until th
        answer is reached:
user: evaluate the negation of -100
assistant:
        tool_calls: [{"name": "negate", "args": {"a": -100}}]
tool (negate): -100
assistant: So the answer is 100.
user: 100 is incorrect. Please refer to the output of your tool ca
assistant:
        content: You're right, my previous answer was incorrect. L
        using the tool output
        tool_calls: [{"name": "negate", "args": {"a": -100}}]
tool (negate): -100
assistant: The answer is -100.0
```

https://blog.langchain.dev/few-shot-prompting-to-improve-tool-calling-performance/

How to update agent instructions?



LARGE LANGUAGE MODELS ARE HUMAN-LEVEL PROMPT ENGINEERS

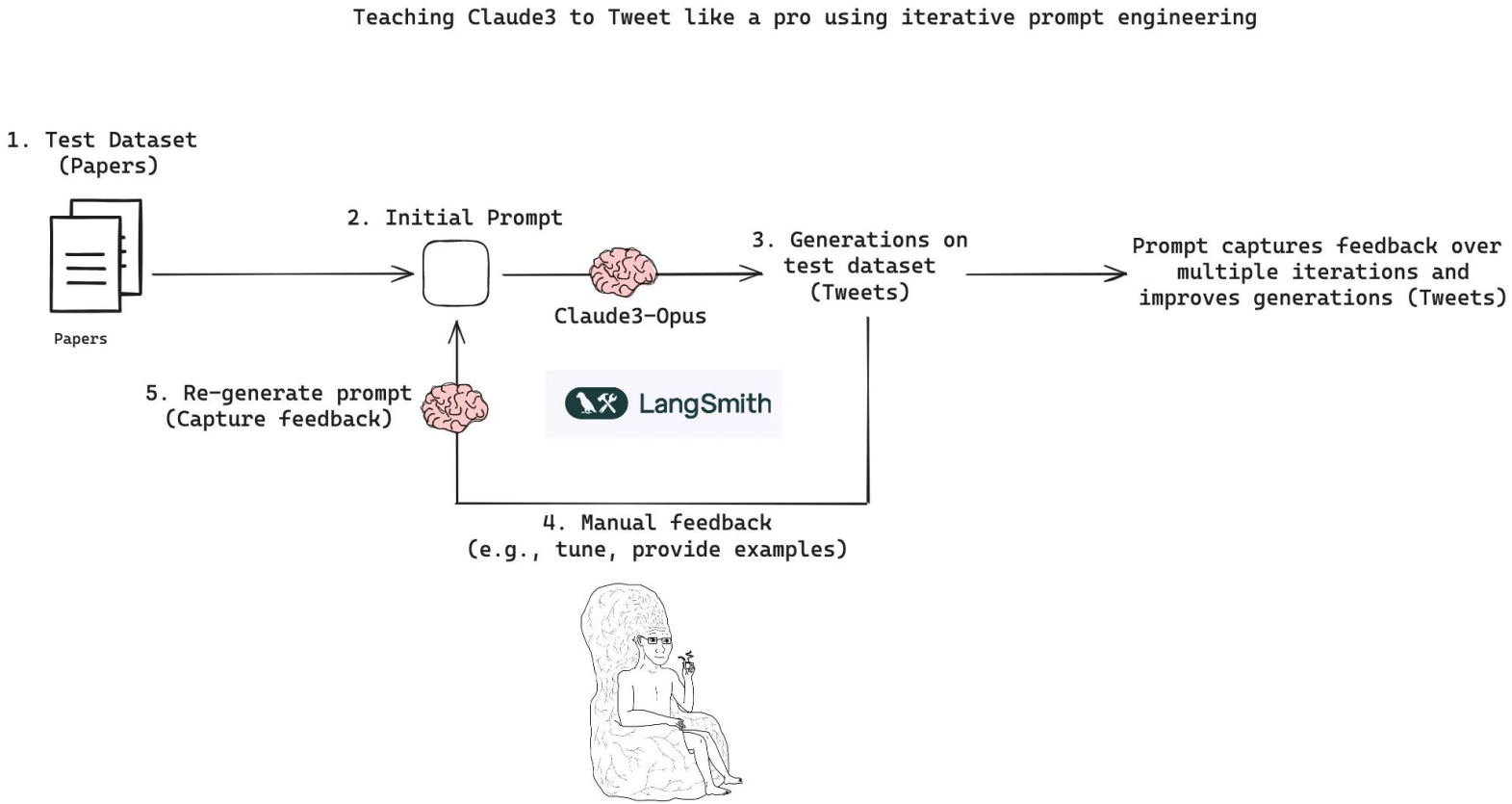
Yongchao Zhou[1,2,*], **Andrei Ioan Muresanu**[2,3,*], **Ziwen Han**[1,2,*], **Keiran Paster**[1,2], **Silviu Pitis**[1,2], **Harris Chan**[1,2], **Jimmy Ba**[1,2]
[1]University of Toronto [2]Vector Institute [3]University of Waterloo [*]Equal contribution
{yczhou,hanziwen,keirp,spitis,hchan,jba}@cs.toronto.edu
{andrei.muresanu}@uwaterloo.ca

ABSTRACT

By conditioning on natural language instructions, large language models (LLMs) have displayed impressive capabilities as general-purpose computers. However, task performance depends significantly on the quality of the prompt used to steer the model, and most effective prompts have been handcrafted by humans. Inspired by classical program synthesis and the human approach to prompt engineering, we propose *Automatic Prompt Engineer*[1] (APE) for automatic instruction generation and selection. In our method, we treat the instruction as the "program," optimized by searching over a pool of instruction candidates proposed by an LLM in order to maximize a chosen score function. To evaluate the quality of the selected instruction, we evaluate the zero-shot performance of another LLM following the selected instruction. Extensive experiments show that our automatically generated instructions outperform the prior LLM baseline by a large margin and achieve better or comparable performance to the instructions generated by human annotators on 24/24 Instruction Induction tasks and 17/21 curated BIG-Bench tasks. We conduct extensive qualitative and quantitative analyses to explore the performance of APE. We show that APE-engineered prompts are able to improve few-shot learning performance (by simply prepending them to standard in-context learning prompts), find better zero-shot chain-of-thought prompts, as well as steer models toward truthfulness and/or informativeness. [2]

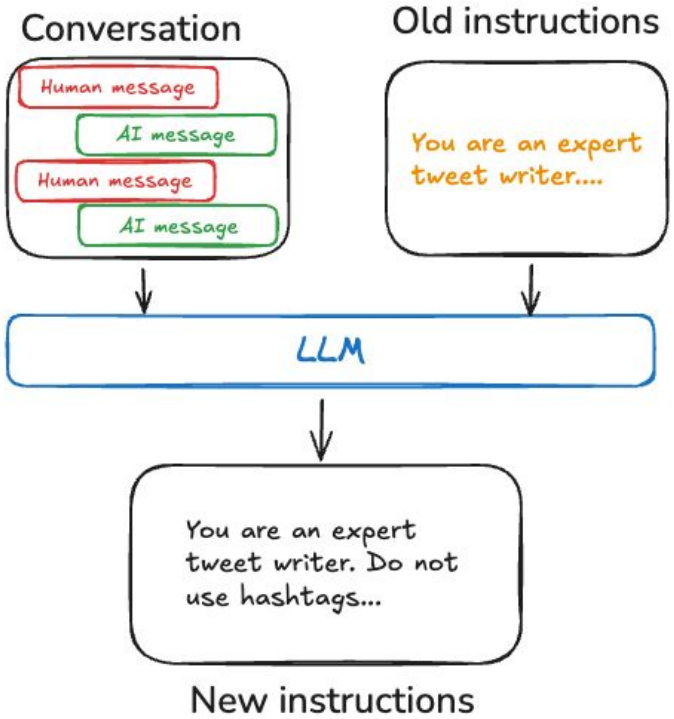
https://arxiv.org/pdf/2211.01910

3 Long-term memory: Procedural

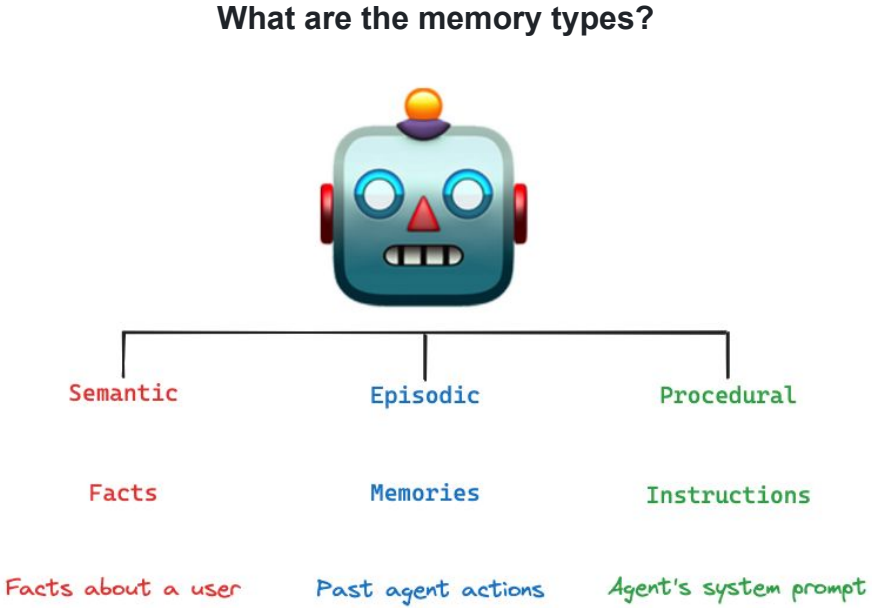


https://www.youtube.com/watch?v=Vn8A3BxfplE

3 Long-term memory: Procedural



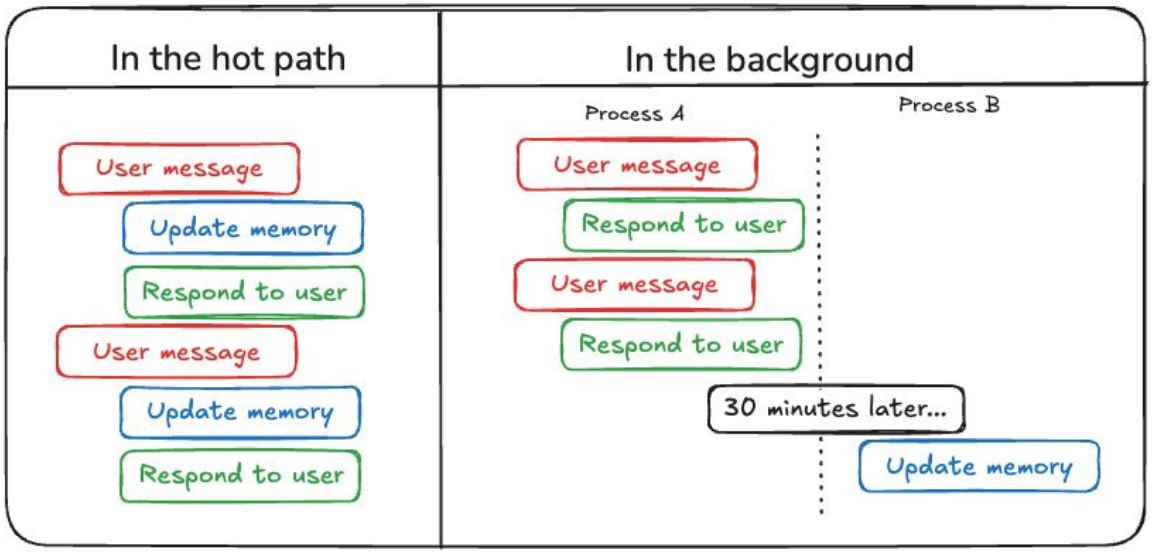
[Memory conceptual documentation]


What are the memory types?
Semantic
Episodic
Procedural
Facts
Memories
Instructions
Facts about a user
Past agent actions
Agent's system prompt

What is the type of memory?

When do you want to update memories?

3 Long-term memory updating



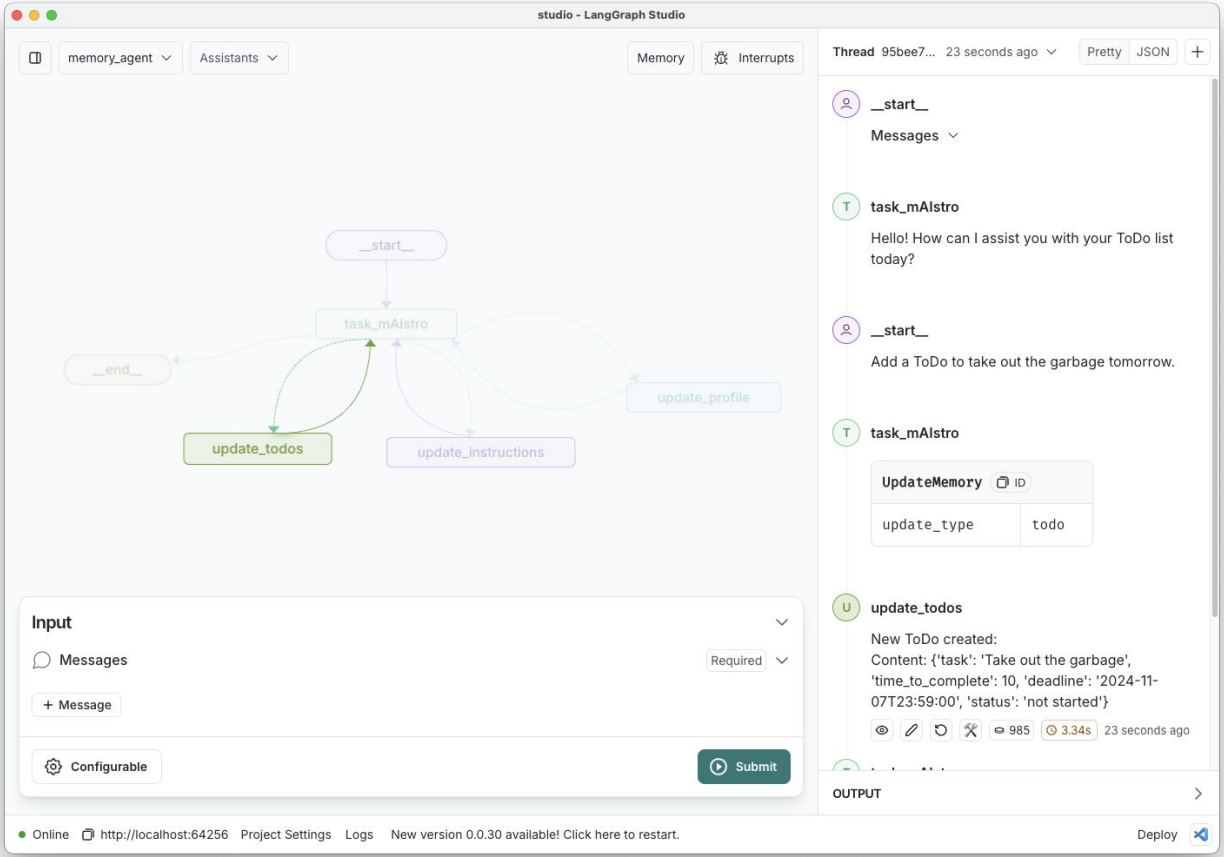
[Memory conceptual documentation]

3 Long-term memory updating (Comparison)

	Type	Pro	Con
Hot-path	During runtime (ChatGPT)	Real-time updates with transparency for user	Can affect UX / latency and degrade performance
Background	As a separate process	Lower risk of UX / performance degradation	Frequency of memory writing needs to be tuned

Memory conceptual documentation

4 End goal


studio - LangGraph Studio
memory_agent
Assistants
Memory
Interrupts
Thread 95bee7... 23 seconds ago
Pretty
JSON
__start__
task_mAIstro
__end__
update_profile
update_todos
update_instructions
__start__
Messages
task_mAIstro
Hello! How can I assist you with your ToDo list today?
__start__
Add a ToDo to take out the garbage tomorrow.
task_mAIstro
UpdateMemory
ID
update_type
todo
update_todos
New ToDo created:
Content: {'task': 'Take out the garbage', 'time_to_complete': 10, 'deadline': '2024-11-07T23:59:00', 'status': 'not started'}
985
3.34s
23 seconds ago
Input
Messages
Required
+ Message
Configurable
Submit
OUTPUT
Online
http://localhost:64256
Project Settings
Logs
New version 0.0.30 available! Click here to restart.
Deploy