

LUCCA: Large Language Models under Conversational Cognitive Architectures

Emerald Zhang
The University of Texas at Austin
Austin, TX USA
emerald.zhang@utexas.edu

Tapaswini Kodavanti
The University of Texas at Austin
Austin, TX USA
tk24428@utexas.edu

Abstract—Existing work in cognitive architectures in robots has achieved substantial results in areas of planning and decision-making but often has large and complicated underlying systems. In our paper, we focus on social interaction and introduce LLMs under Conversational Cognitive Architectures (LUCCA), a conversational LLM-centric cognitive system capable of remembering personal conversations, retrieving specific facts, and reasoning about ideas in its memory base. We utilize a language agent with modular memory components consisting of personal memories and general information, prompt engineering to shape the correct output, and a system to interact with and synthesize these internal representations. Through an evaluation of different conversational scenarios and an HRI study, we show that the framework proves to be an effective yet lightweight alternative

Index Terms—LLMs, Prompt Engineering, Cognitive Architecture

I. INTRODUCTION

Replicating reasoning processes and imbuing robots with cognitive capabilities has been a problem long studied by researchers. Previous works involve large, complex modules and decision-making state machines working together in a logical system to formulate answers and solve problems. However, the development and improvements in large language models provide an alternative solution. Currently, large language models (LLMs) excel at generating responses to all kinds of prompts; however, they lack the ability to “understand” and process the ideas behind these prompts, often causing errors. Of particular interest is the concept of language agents [1], artificial intelligence systems that use large language models to interact with the world, mitigate these issues by connecting LLMs to internal memory and environments, grounding them to existing knowledge or external observations. leverage commonsense priors present in LLMs to adapt to novel tasks, reducing the dependence on human annotation or trial-and-error learning

Thus we propose LUCCA, a new system centered on LLMs and agents; rather than depend on bulky, rigorous state machines or other decision-making processes, we exploit the ability to generate responses by using an agent and applying the concept of accurate retrieval and prompting to drive the conversational agent’s output. Additionally, LLMs are general generative models with stochastic returns. A common issue with using machine learning models for reasoning purposes is that they must be retrained on niche datasets to be flexi-

ble to domain-specific information. LUCCA combats this by maintaining levels of agent-specific memories, reducing the dependence on retraining and human annotation. We lay out the structure in these three main components: information storage, retrieval, action. Given a conversation with a particular person, the conversational agent stores recent memories, both personal and factual, and working memories which over time become long-term synthesized memories.

We validate the success of LUCCA in numerous experiments. We give the system a variety of scenarios; first, we ask it general questions to verify that the implicit general knowledge is still available; then have personal conversations with it and query that information saved; finally, we ask it domain-specific information and expect a correct response. We find that LUCCA does indeed provide the proper answers without expensive training, offering an advantage over finetuning. Moreover, unlike finetuning, we do not require large amounts of disk space to store model checkpoints. This proves to allow flexibility in different use cases.

II. BACKGROUND

A. Cognitive Architectures

Historically, cognitive architectures have augmented the production system, which is the formalization of symbolic manipulation for determining actions. The resulting agents contained large production systems connected to external sensors, knowledge bases, complex working memories, and decision-making implementations [2]. Rooted in foundations of cognitive science and psychology, they were constructed to attempt to mimic the structure of human cognition, explicitly defining processes such as perception, memory, and planning to achieve flexible, rational, real-time behaviors [3].

For example, SOAR [2] is a cognitive architecture that emphasizes the functionality and efficiency of general AI. The original theory relied on the Problem Space Hypothesis, which holds that all goal-oriented behavior can be decomposed into a series of primitive operators or actions on the agent’s current states. This series of appropriate actions can be attained by searching through the space of all potential states that emerge after all possible action combinations. Another model is the ACT-R [4], which emphasizes human cognition modeling using the computational implementation of an interpreter for a domain-specific language.

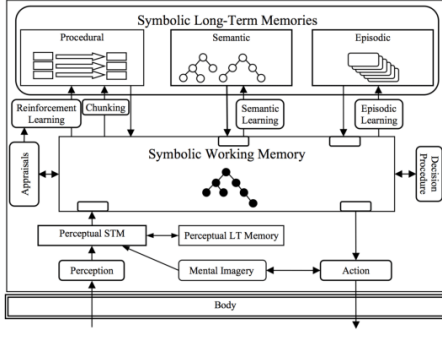


Fig. 1. Diagram of the SOAR architecture [2]. Historical cognitive architectures implement a system consisting of a large network that includes sensory groundings, long-term memory, and a decision procedure for selecting actions. Soar’s decision procedure uses productions to select and execute actions. These actions may be internal (such as modifying the agent’s memory) or external (such as a motor command).

However, these architectures are limited to domains that can be comprehensively defined in logic, and they require many pre-specified rules to function. LLMs are well-equipped to meet these limitations. Their probabilistic nature allows them to operate and produce responses arbitrarily, proving to be much more flexible than logic-based systems [5]. Additionally, models such as GPT and Llama are already pre-trained on massive amounts of internet and common knowledge, providing a large world knowledge base. LUCCA focuses more on using the concept of cognitive architectures to manufacture and guide the correct responses rather than replicating the cognitive system.

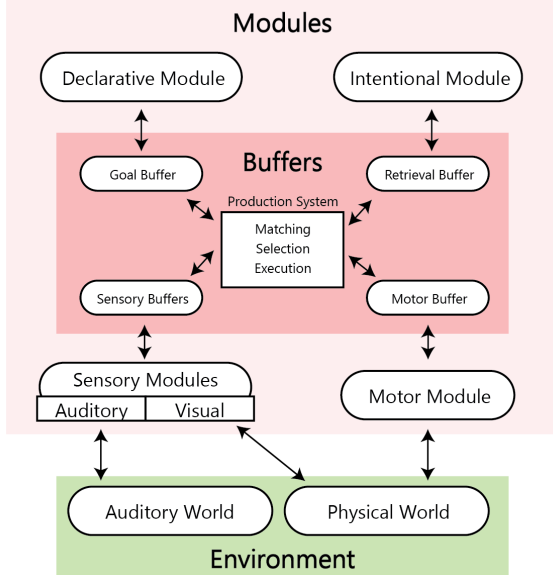


Fig. 2. Diagram of the ACT-R architecture. This is fundamentally an interpreter for a domain-specific language (DSL) where the primitives and data types reflect theoretical assumptions about human cognition.

B. Prompt Engineering

In order to efficiently utilize LLMs, prompt engineering has emerged as a branch of LLM querying techniques, designed to produce the most optimal outputs from a generative AI model. Past work includes the Ask Me Anything model [6], which concluded that open-ended questions in a Q&A context tend to outperform questions that restrict answers to a finite number of pre-set answers. This method of tuning LLM output has also expanded past just text prompting, as shown by MaPLe [7], which is a multi-modal prompt learning system that improves semantic alignment between language and vision.

Prompt engineering is often compared to program synthesis for its similarity to code segment generation, particularly when the output language is used for planning purposes. The Automatic Prompt Engineer [8] system frames language generation from LLMs as a program synthesis problem, where the generated program is written in natural language.

III. RELATED WORK

A. LLM Cognitive Model

CoALA [3] is a cognitive model with multiple types of memory, including procedural, episodic, and semantic, which is used in combination with symbolic working memory for planning purposes. The design of the system is analogous to that of a production system; however, CoALA is still a proof of concept and an implementation has not been released.

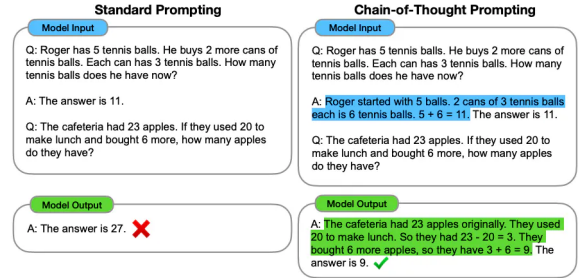


Fig. 3. Chain of Thought prompting

B. Chain-of-Thought (CoT) Prompting

CoT prompting [9] is a prompt engineering paradigm where a specific phrase is integrated within the prompt to expect a particular format or aesthetic from the response. Past work has shown that incorporating CoT prompting also inherently improves complex reasoning capabilities by forcing the LLM to step through smaller intermediary steps before jumping to the final goal.

C. Multi-robot Collaboration

LLMs and natural language can also be used for communication and low-level task planning. RoCo [10] is a robotic planning system that uses LLMs to discuss and collectively reason planning strategies between multiple robots. One variant of the program can also include human-in-the-loop feedback where an external human can communicate with the robot and complete tasks together.

IV. METHODOLOGY

We approach creating an LLM-based cognitive architecture through a memory base updated in real-time, chain of thought prompting techniques, and a retrieval agent. We employ the Langchain framework’s conversational agent to facilitate information retrieval and storage in our custom memory framework.

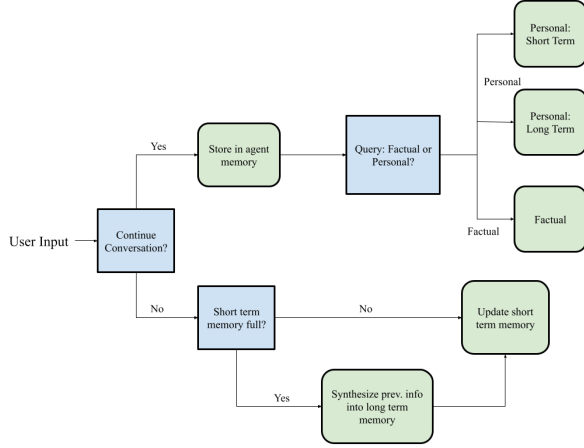


Fig. 4. Diagram of the LUCCA Architecture. Information extracted from interactions flow through the system and get sorted into their respective memory stores. An additional layer of logic manages the transfer and synthesis of memories from short-term to long-term memories.

A. Agent

The core idea behind the Langchain conversational agent is to use an LLM to choose a sequence of actions to take, and the language model itself is used as the reasoning engine to determine the actions and the order. For every step, including intermediate decision-making steps, the agent takes in 3 inputs: a list of available tools, the user input, and previously executed steps. The LLM chain then generates the next action and repeats this process until it finishes. The actual “actions” that the agent performs are in the form of tools, which are functions that the agent can call. Some examples of tools are content retrievers, utilities, or other decision chains. In our experiments, we set up a tool to retrieve factual information from a vector store derived from factual text files.

B. Memory

In order to allow effective retrieval and information preservation, we employ a multi-tiered personal memory base as well as a factual memory base for lesser-known but hard-truth facts.

1) *Personal*: We chose to model the personal information base after the cognitive models of the human thinking process, similar to the memory makeup of historical models like SOAR. To retain the maximum amount of information without exceeding storage space, the personal memory base contains 2 distinct parts: working/short-term memory and long-term memory. Short-term memory is the capacity to recall a small amount of information from a recent time period. Long-term

memory is the capacity to recall memories from a longer time ago. Each section of memory is stored in the form of a text file named after the person the agent is interacting with. For each conversation, each interaction is stored word for word in a question-response pair. At the end of every interaction, these interaction pairs are grouped and stored in the correct file along with a timestamp. For more accurate retrieval, the current conversation is also recorded in a separate file for short-term memories. It’s important to note that for short-term memories, the content is recorded word for word, emulating the “freshness” of human short-term memory. The content of this file is updated after every new conversation with the persona the file is named after. Furthermore, we expanded our memory base to include an additional layer of complexity in the form of long-term memories. Rather than evict memories from the memory base, we condense the information and store them elsewhere to emulate the long-term memories of the human cognitive system. Once the short-term memory layer is saturated, another call is made to an LLM to summarize each of the interactions. With this call, we also use prompt engineering techniques to ensure that the maximum amount of information is retained. Like both the short-term memory formats, this synthesized information is then transferred to a third text file for personal, summarized memories.

2) *Factual*: Besides the recording of personal anecdotes or information that may vary from person to person, we also maintain a fact base for information that is neither pre-trained into the model nor personal. The raw information comes in the form of a text file labeled after the content of the information(i.e annahissgymnasium.txt describes facts about the Anna Hiss Gymnasium). For the sake of better retrieval by the conversational agent, the raw information is encoded into a vector-store database that reloads and updates as the original contents of the text file update. The goal is to make LUCCA personalizeable for any use case, so this particular form of memory is client-specific, and up to the end user to update.

C. Prompting

We strive to make LUCCA customizable for different types of conversations. Prompt engineering techniques are applied at both the interface level as well as the agent’s decision-making level to drive and regulate the correct outputs for any use case. At the interface level, the personality is hard-coded into the prompt to shape the style in which the agent delivers responses. This follows the Chain-of-Thought prompting strategy. For example, LUCCA’s initial prompt starts with “You are a friendly robot named LUCCA designed to hold casual conversations with people and provide them with information when you can...” which sets up the tone for the remainder of the conversation.

Internally, prompts are also engineered to guide retrieval and decision-making. We design prompts in such a way that aligns with Chain-of-Thought concepts to output the correct format in intermediate steps that can then be processed and funneled through the decision-making process.

V. EVALUATION

We demonstrate the effectiveness of LUCCA in social interaction by applying it in different conversational scenarios. We ran a series of tests to ensure the personal and factorial memory retrieval systems were producing results that aligned with the statements in the given files.

A. Personal Memory

We began by prompting LUCCA with various factual questions (such as those about notable cities in the world and historical significance). Afterward, we continued prompting the model with anecdotal information relating to the various locations just discussed to give the impression that the user was having a personalized conversation with the robot. After the user exits the first conversation and re-enters sometime in the future, the user checks that LUCCA can still recollect accurate information regarding the last conversation, particularly the anecdotal information. We test for every piece of information given to the robot in the initial conversation by querying the memory through the vector store for specific details.

TABLE I
PERSONAL ANECDOTE CONVERSATION I

Entity	Response
AI	Hey [NAME]! How's it going? Anything new and exciting happening in your life since we last chatted?
Person	Hi! Can you tell me about the geography of Canada?
AI	Sure! Canada is a vast country located in North America. It's known for its stunning landscapes, including the Rocky Mountains, Niagara Falls, and the Great Lakes. It has ten provinces and three territories. Is there anything specific you'd like to know about Canada?
Person	Tell me more about Niagara Falls.
AI	Niagara Falls is a famous natural wonder located on the border of Canada and the United States. It consists of three waterfalls: the Horseshoe Falls, the American Falls, and the Bridal Veil Falls. Niagara Falls is known for its breathtaking beauty and is a popular tourist destination. Have you ever been there or are you planning to visit?
Person	Yeah, I've been to Niagara Falls over twenty times by now. I used to live in Toronto, only about an hour or so away from the falls.
AI	Wow, that's impressive! Living so close to Niagara Falls must have been amazing.
Person	I will talk to you later!
AI	Alright, talk to you later! Take care!

B. Factual Memory

Similar to the personal memory tests, we query the factual information files with the vectorstore when asking the robot for information regarding a particular entity. We repeat these queries across multiple conversations to ensure accurate information is produced for every information request. In specific, we ask for known details of the UT Austin Anna Hiss Gymnasium.

TABLE II
PERSONAL ANECDOTE CONVERSATION II

Entity	Table column subhead
AI	Hey [Name]! How's it going? Anything new and exciting happening in your life since we last chatted? Did you discover any new favorite spots or activities at Niagara Falls?
Person	What was our last conversation about?
AI	Our last conversation was about Canada and specifically Niagara Falls. You mentioned that you have been to Niagara Falls over twenty times and used to live in Toronto, which is close to the falls. I asked if you had discovered any new favorite spots or activities there during your visits.

TABLE III
FACTUAL INFORMATION CONVERSATION

Entity	Response
Person	What is the Anna Hiss Gym?
AI	The Anna Hiss Gym, also known as AHG, is a building located at the University of Texas at Austin. The AHG adaptive reuse project began in January 2017 and was completed in June 2021. The focus of the project was to repair and rehabilitate the entire building, including the roof and windows, to support the university's partnership with the Army Futures Command modernization program. The building now supports research and academic programs for Texas Robotics and the College of Fine Arts. The AHG building has many features that were maintained to preserve its historic accuracy. It is now fully operational, and ongoing research is conducted there. Justin Hart's office is located on the second floor of the BWI Lab, which is part of the AHG building. There is a celebrated entrance in the back, and the BWI Lab itself is in room 2.202. To access the BWI Lab, one can enter through one of two doors. Please let me know if there's anything else you'd like to know about the Anna Hiss Gym or if there's anything else I can assist you with!

C. HRI Study

To further evaluate the ability of LUCCA to apply itself well in conversational situations, we ran an HRI study with recruited participants. To maintain the reliability and validity of results, the study responders were all familiar with the UT Austin area. In order to statistically evaluate and quantify the results, we created a series of questions using a Likert scale rating, which is a numerical rating used to measure the opinions or attitudes towards the program being assessed.

Each question was followed by 5 ranked answer statements where the user would choose whichever statement they felt best resonated with their belief. Each of the subjects were assigned to either group A or group B, and were not told which label corresponded to which conversation system. Each user then simulated a conversation with the system by typing in responses as reactions to the given prompts. They were also asked to simulate future conversations, or scenarios where the subject has returned to the conversation system after 1 day and then 1 week.

We created a comprehensive study to test all aspects of our architecture. Initially, we ask the participant to have a short conversation with LUCCA as they were talking to a human. Specifically, we ask that they include a few key points that are easily memorable as main conversation points. For example, a quick conversation with a roommate may be about a book you’ve read recently, where the specific book is considered one of the main topics of the conversation. Following this is a test for long-term personal memory, where participants are asked to pretend that a long time has passed since the last interaction with LUCCA (i.e. 1 week). Finally, there is a factual memory test, setup in the same way that the personal memory tests were set up. Under the assumption that LUCCA is a conversational system specific to UT Austin, we request that study respondents ask LUCCA detailed questions about specific buildings at UT Austin, information of which we have already written into the factual memory store. Between each section, participants respond to several questions assessing the ability of LUCCA to accurately recall personal information, question about previous conversations, and avoid hallucinations in responses.

TABLE IV
EXAMPLE HRI STUDY QUESTIONS

Test	Sample Question
Short-term memory	How well does LUCCA recall information from the last conversation in general?
	Did LUCCA ask personal and relative questions about the previous conversation?
Long-term memory	How personalized were the responses and were there any sentences with hallucinations?
	Were there any sentences with hallucinations?
Factual Memory	How well does the information given by LUCCA match your personal experiences at these locations? How well does the program recollect use-case-specific information accurately?

To create a baseline to test the effectiveness of LUCCA against, we collect answers from two groups and compare the results against each other. Group A interacts with LUCCA while Group B just converses with a vanilla LLM with no additional modules or support.

VI. RESULTS

For quantitative analysis, four metrics were assessed from the outcomes of the survey, of which are summarized as “recall” or “relativity”. Recall refers to if the system can accurately recall information regarding the previous conversations, whereas relativity is referring to if the LLMs prompts were relevant and showed acknowledgement to the previous conversations to its responses. Finally, hallucinations refer to the tendency of general vanilla LLMs to generate realistic yet untrue guesses that answer a certain query.

With the data collected from the Likert-style responses, a t-test for the difference of means was conducted to validate

the statistical significance of the findings, where the averages being compared came from the scores of Group A and Group B.

For each of these metrics, we define our null and alternate hypotheses to favor higher levels for Group B, or the subject group that is testing the conversation system with LUCCA. For example, assume the random variable X to be the difference in means (mean for Group A minus mean of Group B) for short term recall. Our hypotheses are as follows:

$$H_0 = \mu_X = 0$$

$$H_1 = \mu_X < 0$$

We follow the same pattern for the long term recall, denoted by the random variable Y , and for short term relativity, denoted by the random variable Z .

$$H_0 = \mu_Y = 0$$

$$H_1 = \mu_Y < 0$$

$$H_0 = \mu_Z = 0$$

$$H_1 = \mu_Z < 0$$

TABLE V
BASIC STATISTICAL ANALYSIS FROM STUDY

Metric	Mean	Variance
Short term recall A	3.25	1.583333333
Short term recall B	4.75	0.25
Long term recall A	1.75	1
Long term recall B	4.5	0.25
Short term relativity A	2.5	0.916666667
Short term relativity B	4.25	0.333333333

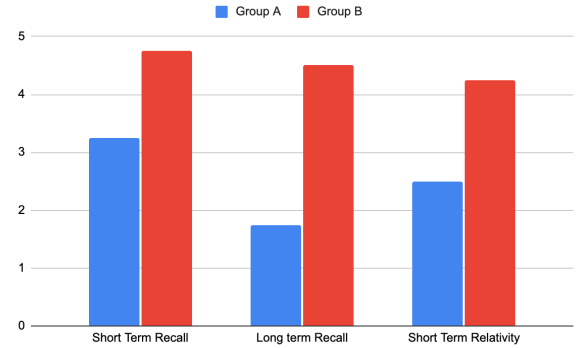


Fig. 5. Means graph

Based on the results, and using a standard 5% significance level, we determined that the short term recall metric is not statistically significant for the subjects we tested with and without LUCCA. However, for the long term recall, we calculated a p-value of 0.0203, falling within the rejection region and thus allowing us to reject the null hypothesis. Intuitively, this makes sense as an LLM won’t remember every single detail from previous conversations. Including the queryable memory base helped in providing more background information, leading to better long term recall.

Additionally, the p-value for the short term relativity was also less than the specified significance level. This also supports our relativity hypothesis, since the language agent can only acknowledge previous conversation memory if the queryable memory base is available.

However, studying and measuring the behavior of LLM-based systems are difficult due to the subjectivity in evaluating responses. Given our time frame, we were limited to only 10 participants, so data is subject to large variance. Furthermore, significant assumptions needed to be made to conduct the t-test. This included assuming that the two populations being tested (the general consensus regarding recall for the conversation system with and without LUCCA) follow the Gaussian Distribution. Additionally, we needed to assume that the two populations in questions had similar standard deviations.

TABLE VI
P-VALUES OF T-TEST RESULTS FROM STUDY

Metric	<i>p-value</i>
Short term recall	0.1901
Long term recall	0.0203
Short term relativity	0.0203

VII. CONCLUSION

We have presented LUCCA, a new lighter-weight cognitive framework derived from canonical cognitive architectures but centered around modern LLM agents. We show that this framework does allow robots to effectively answer questions personally and accurately without the need for retraining data or a dense, complex foundational framework. Furthermore, the results show that LUCCA is a feasible and accurate conversationalist, and that there is a statistically significant improvement from the abilities of a general LLM. Although we focus only on lab-related interactions and questions in this paper, our method can be generalized to a variety of use cases.

The system detailed in this work demonstrates that cognitive reasoning and personalized responses are something that robots can achieve. Future work could extend our framework by expanding and complicating the memory base to incorporate more levels or reason about abstract ideas and topics or focus on the efficiency, speed, and scalability of the system. Additionally, LUCCA can be expanded to take in visual or auditory information, which is further processed and stored in its memory bank to move toward more realistic interactions. Finally, experimenting and gauging the accuracy of LLMs require more datasets and benchmarking in the domain of conversations; thus, future work should maintain the goal of creating more generalizable and robust frameworks.

ACKNOWLEDGMENTS

We would like to thank Dr. Justin Hart for his support, guidance, and valuable feedback.

REFERENCES

- [1] M. V. Mattias Wahde, "Conversational agents: Theory and applications," 2022.
- [2] J. E. Laird, "Introduction to the soar cognitive architecture," 2022.
- [3] T. R. Summers, S. Yao, K. Narasimhan, and T. L. Griffiths, "Cognitive architectures for language agents," 2023.
- [4] F. E. Ritter, F. Tehranchi, and J. D. Oury, "Act-r: A cognitive architecture for modeling cognition," 2017.
- [5] OpenAI, "Gpt-4 technical report," 2023.
- [6] S. Arora, A. Narayan, M. F. Chen, L. Orr, N. Guha, K. Bhatia, I. Chami, F. Sala, and C. Ré, "Ask me anything: A simple strategy for prompting language models," 2022.
- [7] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "Maple: Multi-modal prompt learning," 2023.
- [8] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba, "Large language models are human-level prompt engineers," 2023.
- [9] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," 2023.
- [10] Z. Mandi, S. Jain, and S. Song, "Roco: Dialectic multi-robot collaboration with large language models," 2023.