Chameleon: Plug-and-Play Compositional Reasoning with Large Language Models

Pan Lu, et al

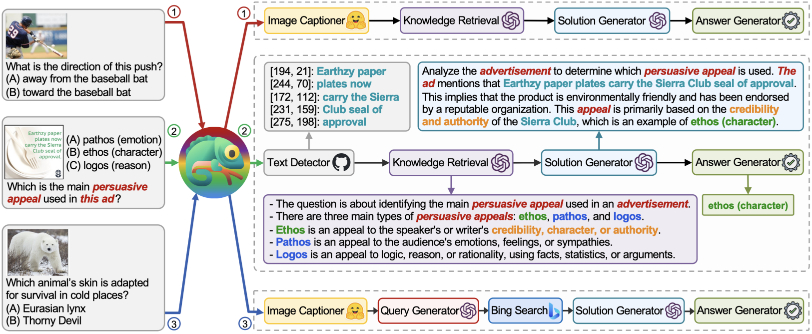
University of California, Microsoft Research

NeurIPS 2023

Abstract

- LLMs' Limitations
  - Lack of up-to-date knowledge (e.g., real-time web info)
  - Inability to use external tools (e.g., calculators)
  - Weak in precise mathematical / logical reasoning
- *Chameleon*: LLMs with *plug-and-play* modules
  - Vision models
  - Web search engines
  - Python functions
  - Heuristic-based modules
- Experiments
  - Benchmark: ScienceQA and TabMWP
  - Based on GPT-4, we achieved SOTA

Introduction



- Goal: Dynamically selecting tools
  - GPT-4 works as planner with in-context learning [1]
  - Synthesize program with basic tools

[1] Josh Achiam et al. "Gpt-4 technical report". In: *arXiv preprint arXiv:2303.08774* (2023).

Related Work

- LLM using tools
  - Codex was fine-tuned with Github codes [1]
  - Reasoning with coding was suggested [2]

- Limitation of prior works
  - Requires supervised fine-tuning
  - Fixed set of tools
  - Calls one tool at a time

- Chameleon's difference
  - In-context learning without fine-tuning
  - Plug-and-play: adapt to new tools given description
  - Composes multi-step tool sequences

[1] Mark Chen et al. "Evaluating large language models trained on code". In: *arXiv preprint arXiv:2107.03374* (2021).

[2] Wenhu Chen et al. "Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks". In: *arXiv preprint arXiv:2211.12588* (2022).

Comparision of work

Model	Tool Use						Skill Dimension					Inference & Extension		
	Size						Image	Web	Know.	Math	Table	Composition	Planning	Plug-n-Play
CoT [57]	1	✓	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Lila [39]	1	✓	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗
PoT [6]	2	✓	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗
Code4Struct [55]	1	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
PAL [10]	2	✓	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗
MathPrompter [18]	2	✓	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗
ART [43]	4	✓	✗	✗	✓	✓	✗	✓	✗	✓	✗	✓	✗	✓
Toolformer [49]	5	✗	✗	✗	✓	✗	✗	✓	✗	✗	✗	✗	natural lang.	✗
WebGPT [40]	10	✓	✗	✗	✓	✗	✗	✓	✗	✗	✗	✓	program	✗
MM-ReAct [60]	>10	✓	✗	✗	✓	✗	✓	✓	✓	✓	✓	✓	word match	✓
Visual ChatGPT [59]	>10	✓	-	-	✗	✗	✓	✗	✗	✗	✗	✓	natural lang.	✓
ViperGPT [52]	>10	✓	-	-	✗	✗	✓	✗	✓	✓	✗	✓	program	✓
VisProg [13]	>10	✓	-	-	✗	✓	✓	✗	✗	✗	✗	✓	program	✓
HuggingGPT [50]	>10	✓	✓	✗	✗	✗	✓	✗	-	✗	-	✓	natural lang.	✓
Chameleon (ours)	>10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	natural lang.	✓

- Prior Tools: Lack of generalization
  - Limited tools, Manually prompting usage
- Our method: Chameleon
  - Plug-and-Play Flexibility
  - Natural language planning by LLM (Interpretable)

General Framework

- Notations
  - Input query x_0
  - Natural language planner $\mathcal{P}$
  - Task instruction $\mathcal{I}$
  - Module inventory $\mathcal{M}$ consists of modules: $\{M_i\}$
  - Constraints $\mathcal{G}$ for the sequence orders of modules
  - Few-shot examples $\mathcal{D}$

General Framework

- Workflow
  - Plan p, Time step t, Output y^t, Cache c^t
  - $p = \mathcal{P}(x_0; \mathcal{I}, \mathcal{M}, \mathcal{G}, \mathcal{D})$
  - $y^t \leftarrow M^t(x^{t-1}; c^{t-1})$
  - $x^t \leftarrow \text{update\_input}(x^{t-1}; y^t)$
  - $c^t \leftarrow \text{update\_cache}(c^{t-1}; y^t)$
  - The functions are hand-designed for each M_i

- Response
  - $r = y^T \leftarrow M^T(x^{T-1}; c^{T-1})$

Module Inventory

Tool Types	Tools
OpenAI	Knowledge Retrieval, Query Generator, Row Lookup, Column Lookup, Table Verbalizer, Program Generator, Solution Generator
Hugging Face	Image Captioner
Github	Text Detector
Web Search	Bing Search
Python	Program Verifier, Program Executor
Rule-based	Answer Generator

- Knowledge Retrieval
  - This module retrieves additional background knowledge
- Query Generator
  - It creates search engine queries based on the problem
- Row / Column Lookup
  - Reasoning process may involve tabular context
  - Focusing only on relevant section for query

Module Inventory

Tool Types	Tools
OpenAI	Knowledge Retrieval, Query Generator, Row Lookup, Column Lookup, Table Verbalizer, Program Generator, Solution Generator
Hugging Face	Image Captioner
Github	Text Detector
Web Search	Bing Search
Python	Program Verifier, Program Executor
Rule-based	Answer Generator

- Table Verbalizer
  - Converting structured tables into text

- Program Generator
  - It generates Python programs to solve queries

Module Inventory

Tool Types	Tools
OpenAI	Knowledge Retrieval, Query Generator, Row Lookup, Column Lookup, Table Verbalizer, Program Generator, Solution Generator
Hugging Face	Image Captioner
Github	Text Detector
Web Search	Bing Search
Python	Program Verifier, Program Executor
Rule-based	Answer Generator

- Image Captioner
  - Converts raw image data into a textual description
- Text Detector
  - Identifies text within a given image
- Bing search
  - It excels when broader or up-to-date information

Module Inventory

Tool Types	Tools
OpenAI	Knowledge Retrieval, Query Generator, Row Lookup, Column Lookup, Table Verbalizer, Program Generator, Solution Generator
Hugging Face	Image Captioner
Github	Text Detector
Web Search	Bing Search
Python	Program Verifier, Program Executor
Rule-based	Answer Generator

- Program Verifier
  - Checks for syntax and logical errors [1]
- Program Executor
  - Executes the program and produces the result

[1] Aman Madaan et al. "Self-refine: Iterative refinement with self-feedback". In: *Advances in Neural Information Processing Systems* 36 (2024).

Module Inventory

Tool Types	Tools
OpenAI	Knowledge Retrieval, Query Generator, Row Lookup, Column Lookup, Table Verbalizer, Program Generator, Solution Generator
Hugging Face	Image Captioner
Github	Text Detector
Web Search	Bing Search
Python	Program Verifier, Program Executor
Rule-based	Answer Generator

- Solution Generator
  - Using all the cached information, generate solution [1]
- Answer Generator
  - Executes the program and produces the result

[1] Jason Wei et al. "Chain-of-thought prompting elicits reasoning in large language models". In: *Advances in neural information processing systems* 35 (2022), pp. 24824–24837.

Benchmark

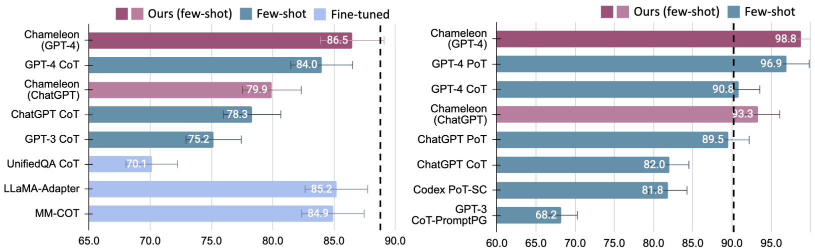
- TabMWP [1]
  - A math reasoning benchmark with tables
  - Row & Column lookup, Table Verbalizer would be needed

- ScienceQA [2]
  - A multi-modal dataset covering scientific topics
  - e.g.) Physics problem with image

[1] Pan Lu et al. "Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning". In: *arXiv preprint arXiv:2209.14610* (2022).

[2] Pan Lu et al. "Learn to explain: Multimodal reasoning via thought chains for science question answering". In: *Advances in Neural Information Processing Systems* 35 (2022).

Experiment



(a) Results on ScienceQA

(b) Results on TabMWP

- Result
  - SOTA in few-shot settings (ScienceQA)
  - Outperformed fine-tuning models (TabMWP)

Experiment

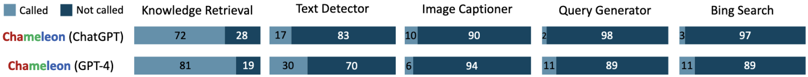


Figure 4: Tools called in the generated programs from Chameleon on ScienceQA.

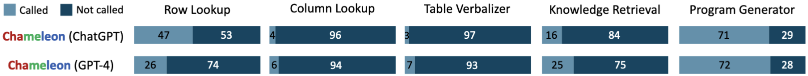


Figure 5: Tools called in the generated programs from Chameleon on TabMWP.

- Tool usage of ChatGPT-based Chameleon
  - Heavily influenced by few-shot examples
  - Strongly prefers certain tools

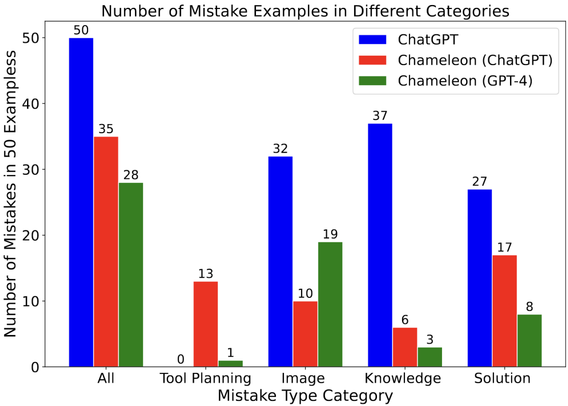
- Tool usage of GPT-4-based Chameleon
  - Distributes tool calls more objectively
  - Uses query generator and web search at the same time

Ablation study

Module	Δ (ScienceQA)	Δ (TabMWP)
Knowledge Retrieval	-7.8%	-2.2%
Bing Search	-7.4%	-
Text Detector	-8.4%	-
Image Captioner	-6.0%	-
Program Generator	-	-7.4%
Table Verbalizer	-	-0.2%

- Most of tools are vital
  - Knowledge retrieval is important in both tasks
  - Domain specific tools are important
  - Vision models for ScienceQA, Program tools for TabMWP

Error Analysis



- 50 mistakes of ChatGPT on ScienceQA
  - Chameleon reduces mistakes by tools

Conclusion

- We introduce *Chameleon*
  - Augmenting external tools
  - Plug-and-play manner

- Experiments on ScienceQA, TabMWP
  - Significant improvements in accuracy
  - Potential for addressing real-world queries