School of Engineering

Contact

Bachelorarbeit 2024

Studiengang Informatik

A2C2 – Natural Language-Instructed Autonomous Agent for Computer Control

Diplomanden
Gabriel Nobel
Rebekka von Wartburg-Kottler

Supervisor
Prof. Thilo Stadelmann
Pascal Sager

Institut / Zentrum
Center for Artificial Intelligence

Recent advances in artificial intelligence (AI) have boosted progress across various domains, particularly enabling breakthroughs in the discipline of **Natural Language-Instructed Autonomous Agents for Computer Control (A2C2s).** Due to their capabilities of understanding natural language and executing actions the same way a human would, these agents have the potential to significantly simplify human-machine interaction, reduce resource requirements in business, and empower non-technical users to operate computer systems effortlessly.

Problem Statement

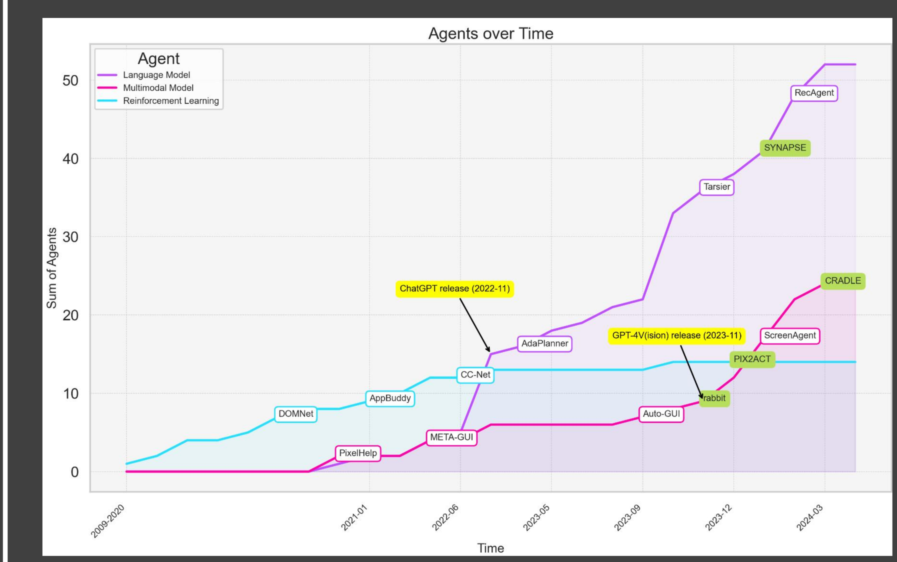
- What does the research landscape **look** like, and how can it be **categorized** to gain a clear and valuable overview?
- What are the **strengths** and **potentials** of existing systems, and how can they be exploited?
- Which **challenges** are known to be **unsolved** and could lead to breakthroughs in the field if solved?
- What does the look like **proposed system**, based on well-founded knowledge in this field and ideally representing a further step towards A2C2?

Input

Instruction Space – *what is the user instruction?* – can be affected by wording, precision and completeness.
Observation Space – *what does the agent perceive?*

- **Pixel-Based** (Screenshot, Video): generalizable, large possible action space
- **Textual** (API, HTML): structured, interpretable, domain-dependent
- **Multimodal** (Text+Image): structured, generalizable, domain-dependent
- **Preprocessed** (Minecraft): clear action space, not generalizable

Background



The research area has made a clear transition from **reinforcement learning** to **language models** to more advanced **multimodal models**. The impact of generative models is highly visible.

Learning

Neural learning – *how can the model weights be adjusted?*

- RL: predefined policy, not generalizable
- Fine-Tuning: iterative, few examples, highly available

Memory – *how is an external knowledge base built?*

- Natural Language: comprehensible, difficult to query
- Embedding: semantic query, flexible, integrated in models, resource-intensive
- Symbolic DB: structured, stable, static schema

Taxonomy

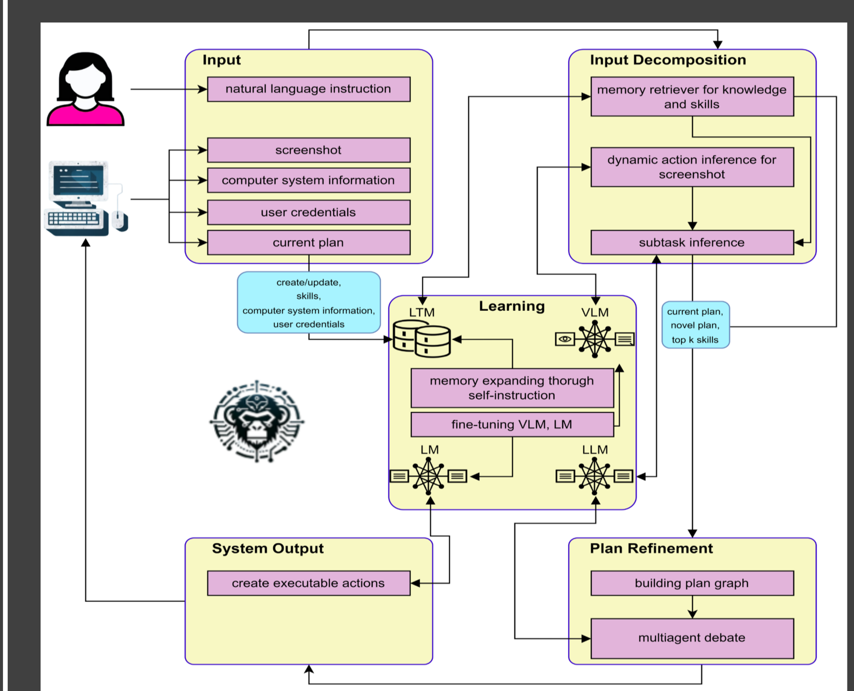
- **Input** – *providing information to the agent*
- **Learning** – *refining skills and building knowledge*
- **Input Decomposition** – *ensuring task comprehensibility*
- **Plan Refinement** – *debating and refining subtasks*
- **System Output** – *interacting with the environment*

Input Decomposition

	Size (KB)	Actions	tokens
raw HTML	224	1331 tags	76485
raw image	530	2400 x 1080px	425
filtered HTML	20	41 tags	6178
processed image	530	91 elements	425
inferred representation	2	48 elements	797

Input Decomposition can be divided into **dynamic action inference** – *how can the observation space be simplified?* – and **subtask inference** – *how can a complex instruction be partitioned?*

Conclusion



Our proposed architecture, contains identified strengths and further research areas like security, performance, and personalization optimizations.

System Output

IO Peripherals: generalizable, large action space

Executable Code: execute actions independently (task & domain), less interpretable

Tool usage: reduces complexity, determining the appropriate tool and its parameters

Pinnacle Methods

RABBIT

PIX2ACT

SYNAPSE

CRADLE

Plan Refinement

Open loop reasoning – *how good is the plan in itself?* – can correct a plan through few-shot examples without feedback.

Multiagent reasoning – *how good is the plan if compared?* – introduces debating and independent control agents.

Closed loop reasoning – *what if an unexpected behavior occurs?* – adds refinement via environment- and human feedback.