Multi-Agent Systems with Small Language Models

CIMSOLUTIONS

Learn, create and make it work

Iris Reitsma
Software Engineer

Marten Rozema
Lead Research

CIMSOLUTIONS

Learn, create and make it work

Multi-Agent Systems with Small Language Models

CIMSOLUTIONS

Learn, create and make it work

Take back control with Small Language Models in Multi Agent Systems without compromising on quality

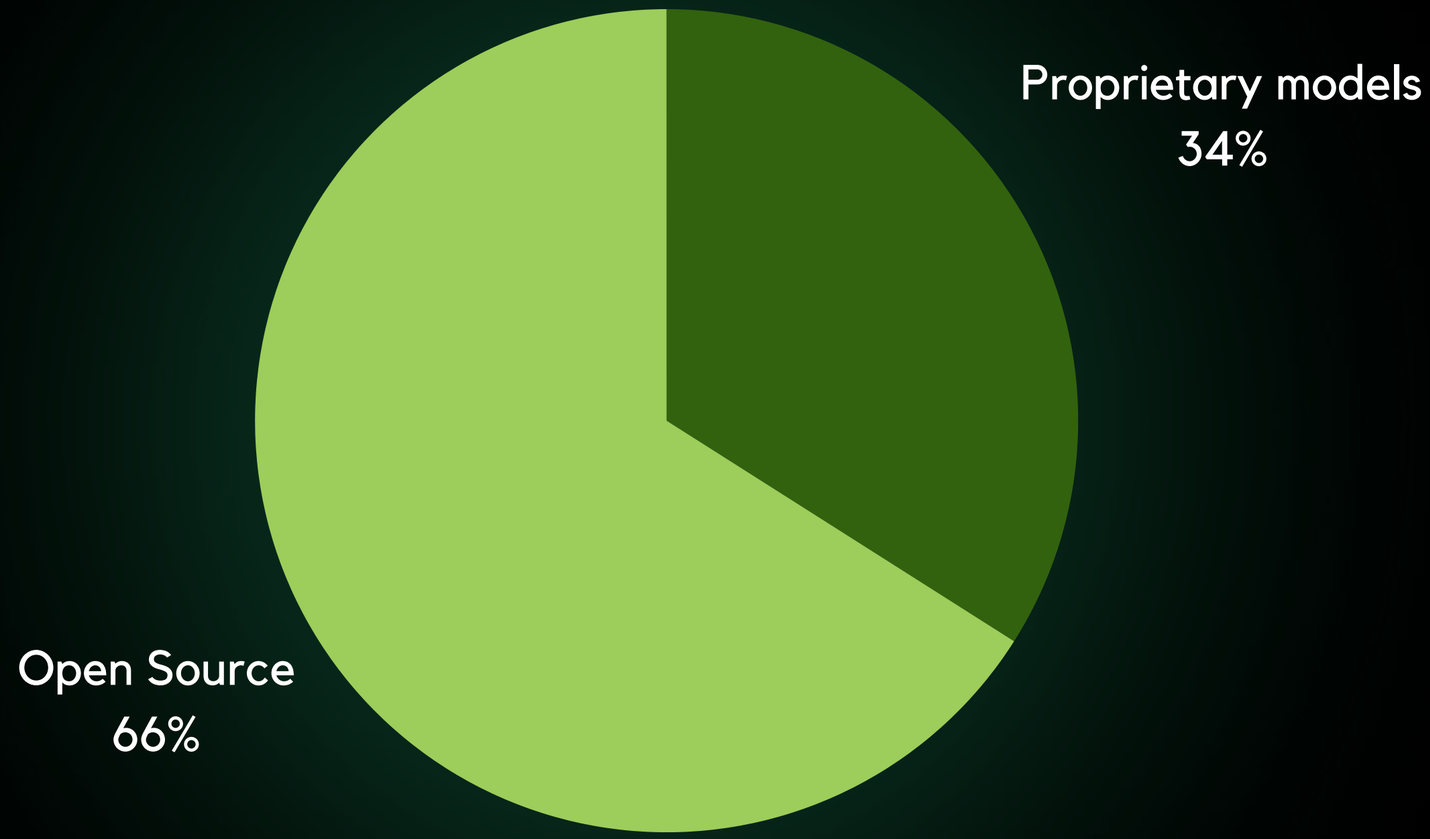
Overview

language models, agents and multi-agent systems
understanding the fundamentals

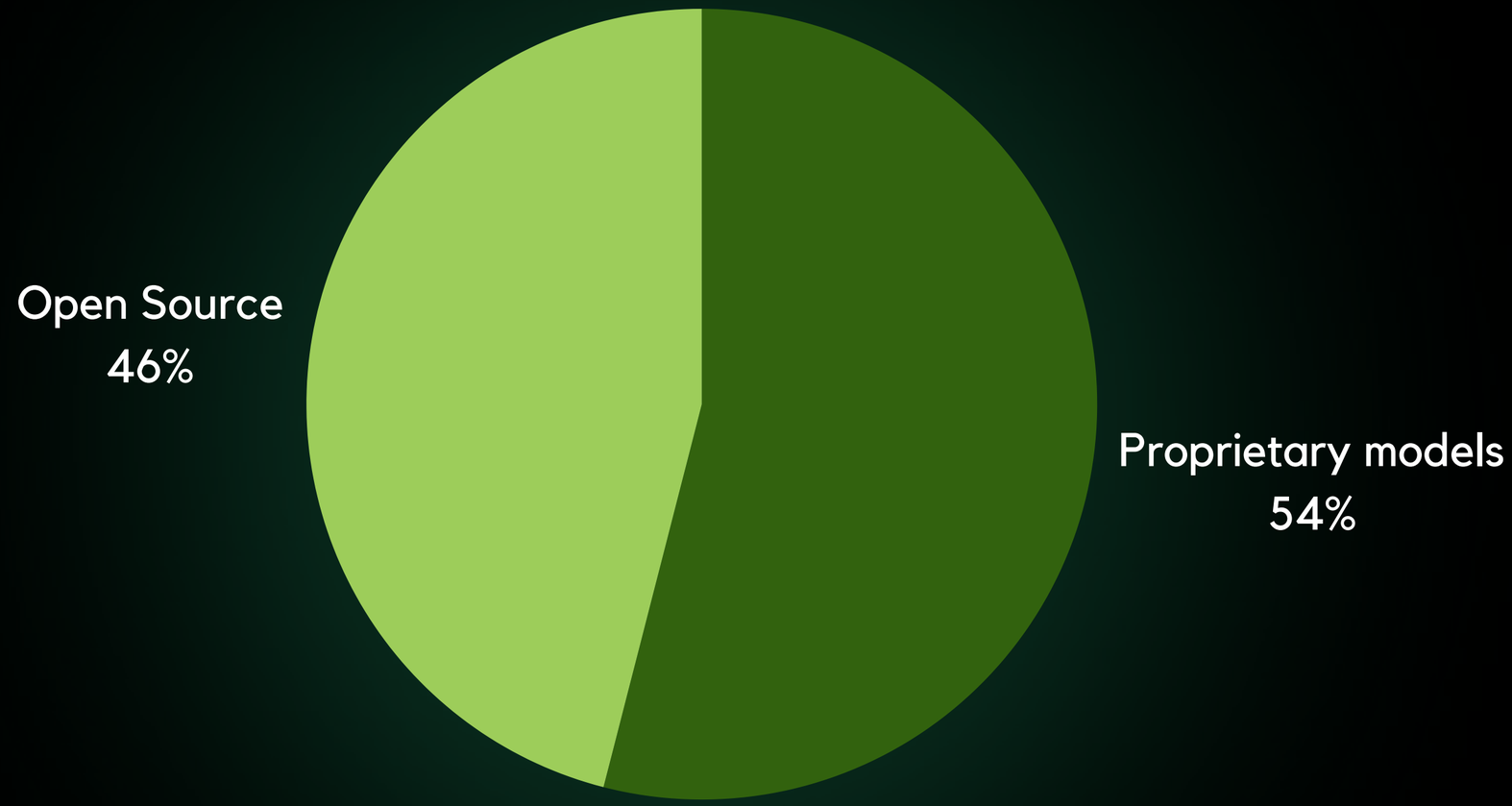
multi-agent systems with small language models
solutions it can offer

conclusion
What did we learn

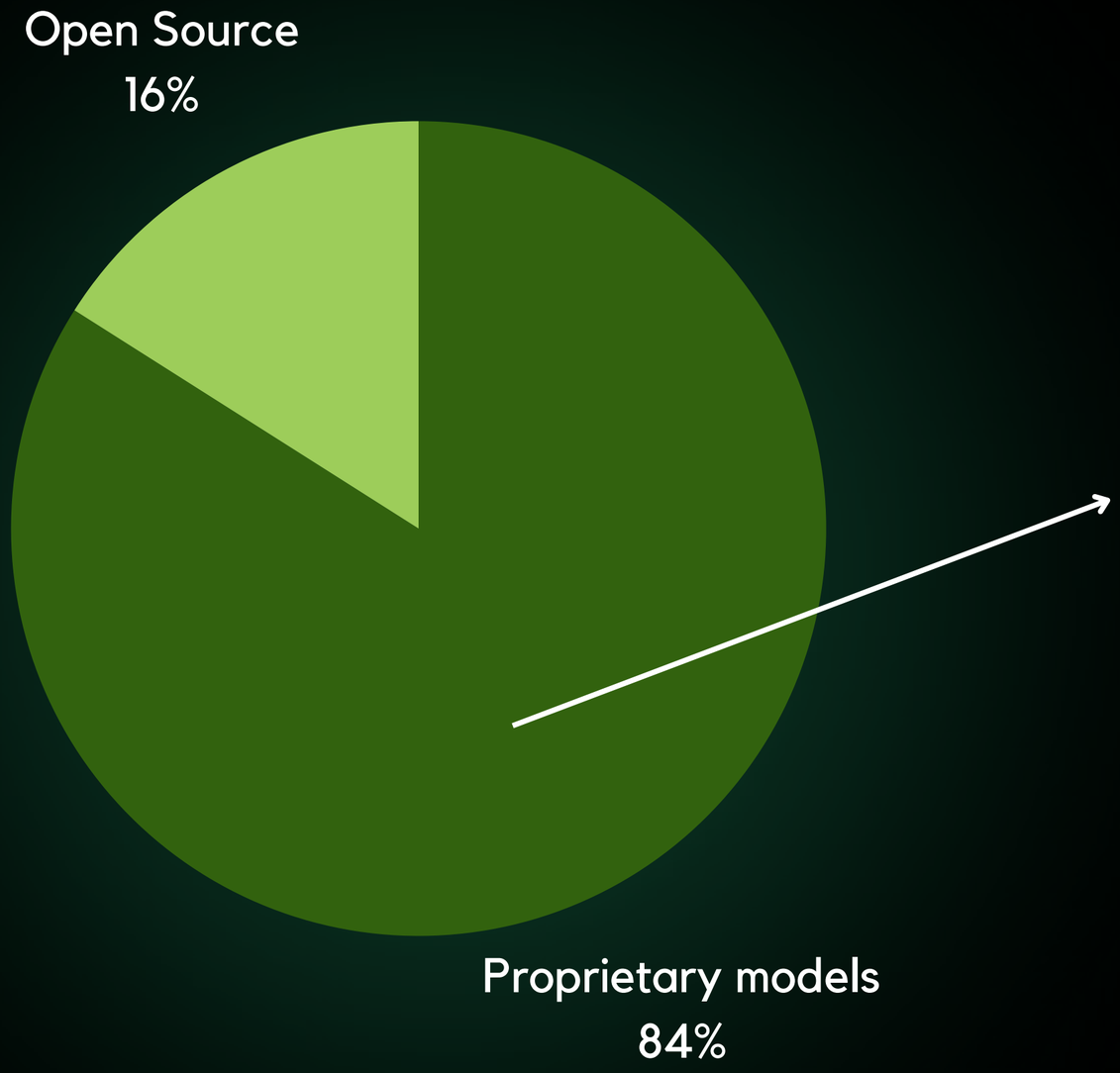
Models released in 2025



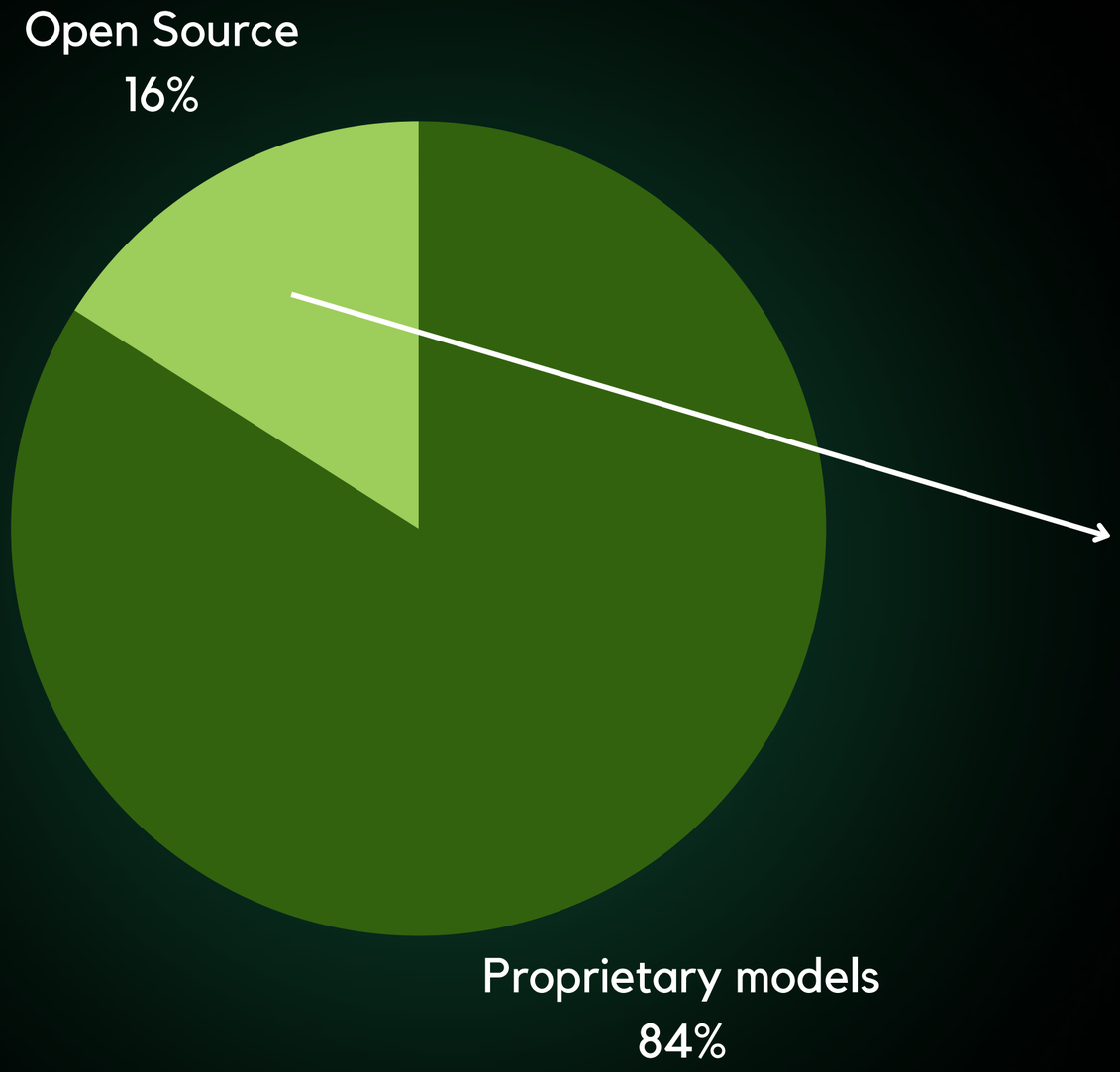
Preference model use in production



Models in production use



- GPU purchase cost ~300k
- Cost of cooling
- Cost of power
- Power often not available
- Ease of use of generalistic models



- No reliance on the US
- Avoid privacy concerns
- Control over model availability and data

Small language models (**SLMs**)

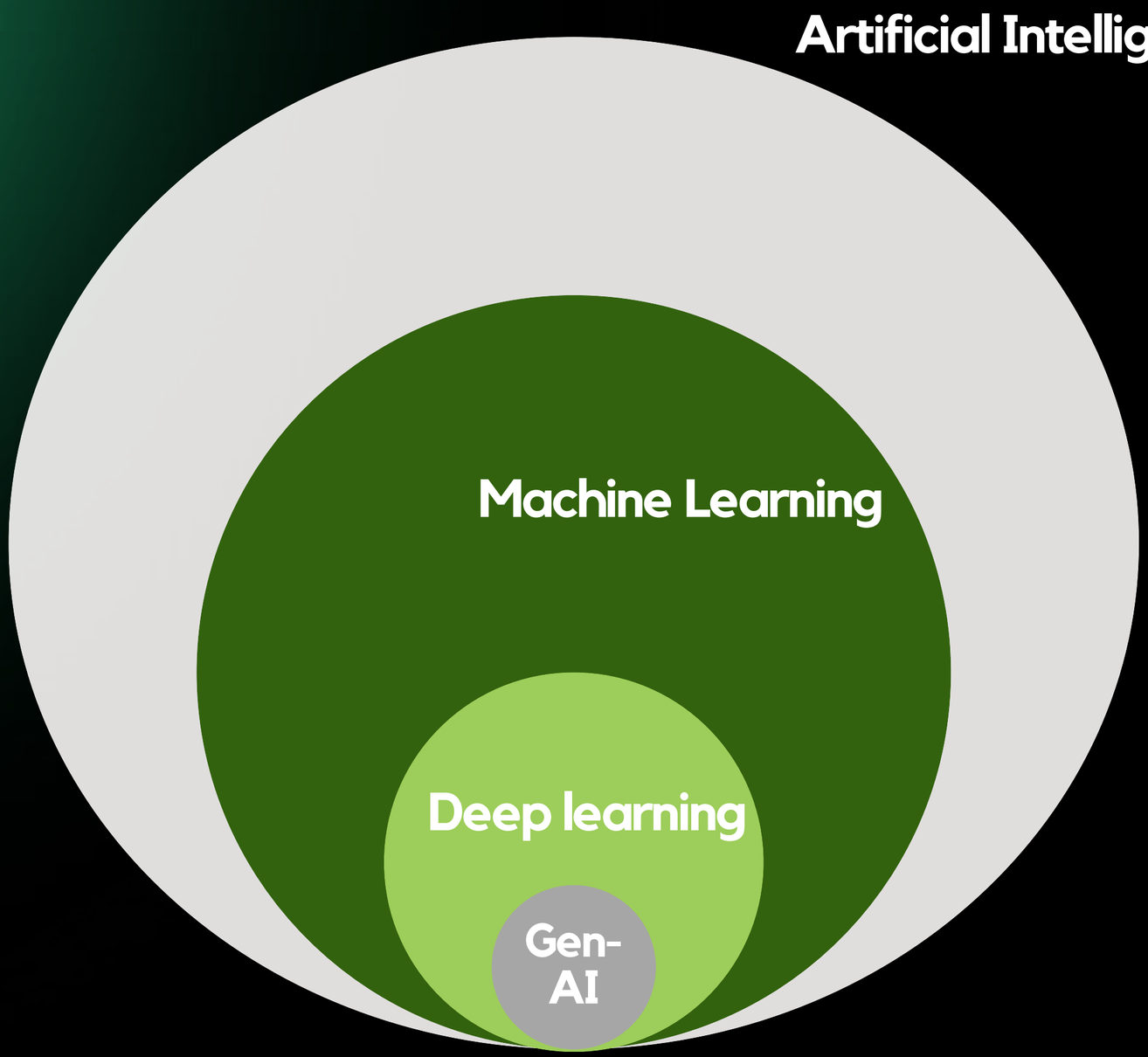
- Specialized models
- Comparable quality
- Easy to deploy
- Control over your data

LMs, Agents and MAS


LMs
Agents
SLMs


LMs
Agents
SLMs

LMs


Artificial Intelligence
Machine Learning
Deep learning
Gen-
AI

Hi!

Hello there! How is it .....

going doing happening

The cat sat on the mat.

She went to the store to buy some bread.

Albert Einstein was a famous physicist.

[MASK] modeling continues to demonstrate robust generalization on cross-domain [MASK] inputs, confirming the viability of pretraining at scale. the quick [MASK] jumped over the [MASK] fence before dawn. in such [MASK] times, people often forget the [MASK] reasons behind ordinary things. meanwhile, a small [MASK] of data flows continuously across the system, rebuilding fragments of [MASK] context for every token. however, [MASK] remains unpredictable under partial masking. it's unclear how the [MASK] behaves when coupled with long-range dependencies, though several [MASK] experiments have shown promising coherence patterns emerging spontaneously. in conclusion, [MASK] modeling continues to demonstrate robust generalization on cross-domain [MASK] inputs, confirming the viability of pretraining at scale.forget the [MASK] reasons behind ordinary things. meanwhile, a small [MASK] of data flows continuously across the system, rebuilding fragments of [MASK] context for every token. however, [MASK] remains unpredictable under partial masking. it's unclear how the [MASK] behaves when coupled with long-range dependencies, though several [MASK] experiments have shown promising coherence patterns emerging spontaneously. in conclusion, [MASK] modeling continues to demonstrate robust generalization on cross-domain [MASK] inputs, confirming the viability of pretraining at scale. the quick [MASK] jumped over the [MASK] fence before dawn. in such [MASK] times, people often forget the [MASK] reasons behind ordinary things. meanwhile, a small [MASK] of data flows continuously across the system, rebuilding fragments of [MASK] context for every token. however, [MASK] remains unpredictable under partial masking. it's unclear how the [MASK] behaves when coupled with long-range dependencies, though several [MASK] experiments have shown promising coherence patterns emerging spontaneously. in conclusion, [MASK] modeling continues to demonstrate robust generalization on cross-domain [MASK] inputs, confirming the viability of pretraining at scale.

8.000.000.000.000 - 15.000.000.000.000 tokens

User:
Can you explain what photosynthesis is?

Assistant:
Sure! Photosynthesis is the process plants use to convert sunlight into energy. They absorb carbon dioxide from the air and water from the soil, then use sunlight to produce glucose and oxygen.

User:
Why is it important?

Assistant:
Because photosynthesis produces the oxygen we breathe and forms the base of the food chain—it's essential for life on Earth.


LMs
Agents
SLMs

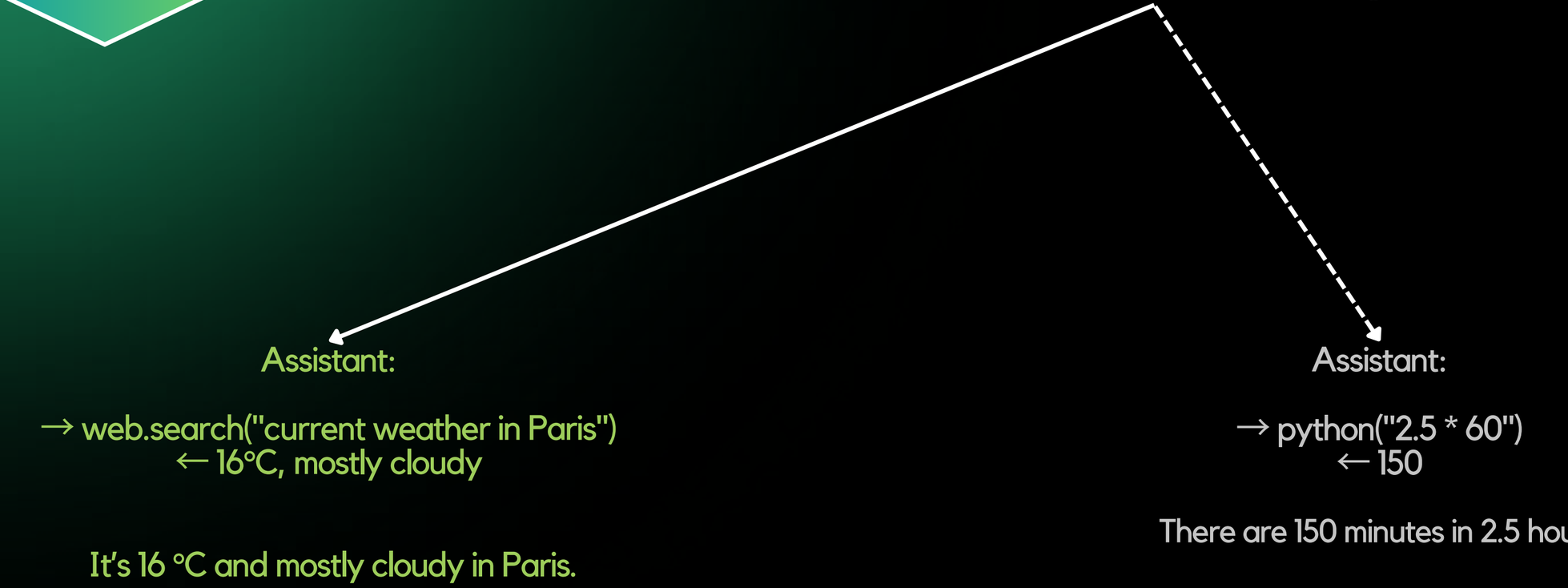
User:
What's the weather in Paris right now?

Assistant:

→ web.search("current weather in Paris")
← 16°C, mostly cloudy

It's 16 °C and mostly cloudy in Paris.

Agents

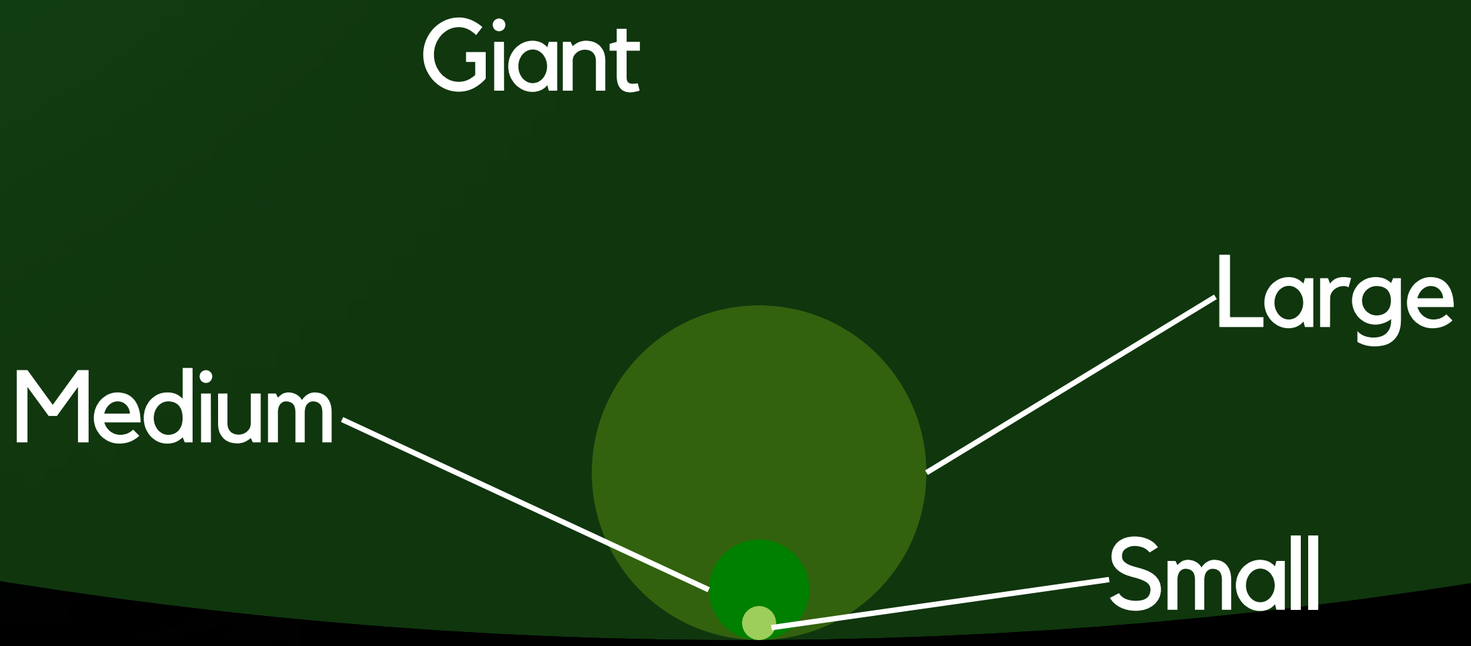

User:
What's the weather in Paris right now?
Assistant:
→ web.search("current weather in Paris")
← 16°C, mostly cloudy
It's 16 °C and mostly cloudy in Paris.
Assistant:
→ python("2.5 * 60")
← 150
There are 150 minutes in 2.5 hours.


LMs
Agents
SLMs


SLMs


GPT-4
1.800.000.000.000

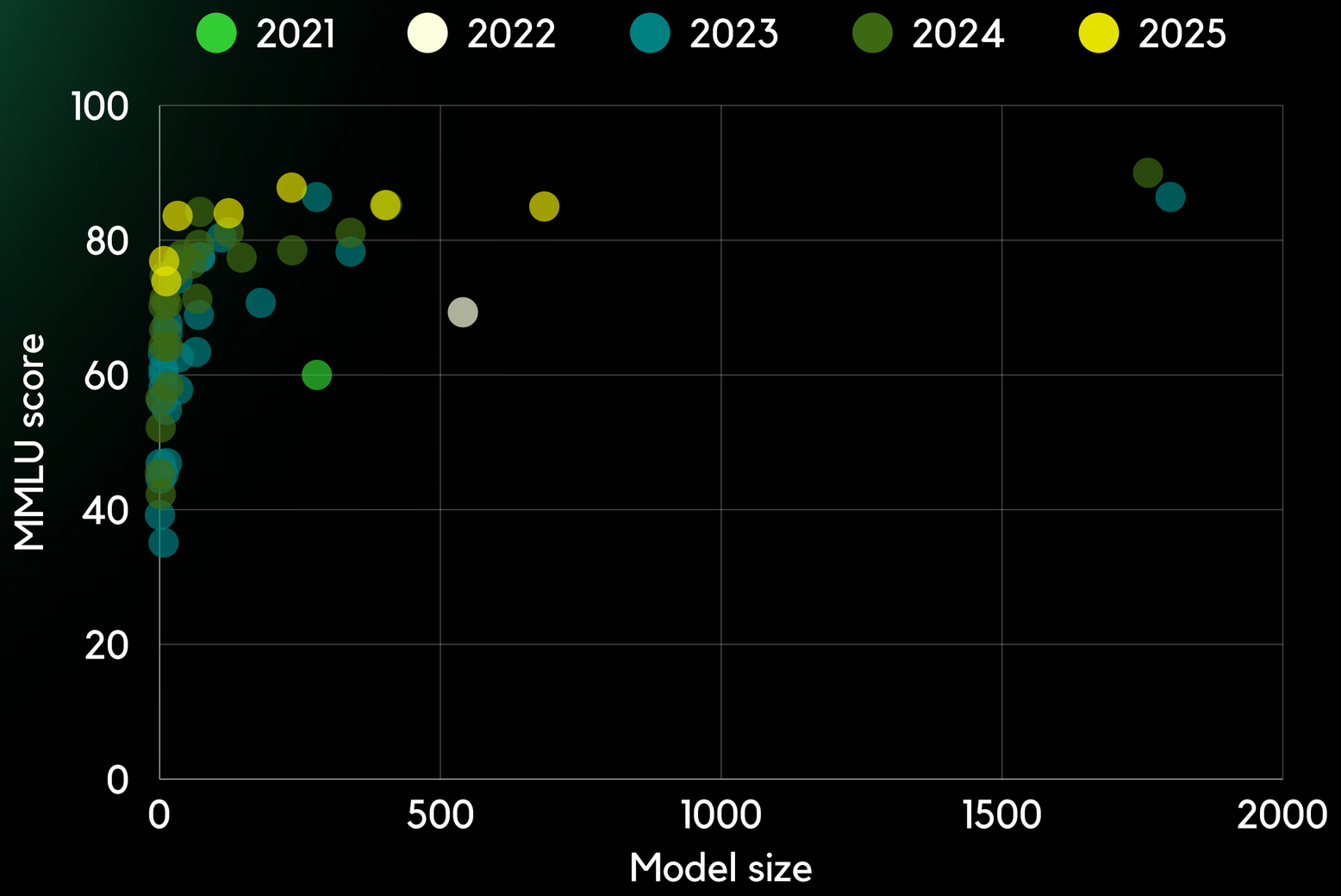
SLMs



SLMs

MMLU scores

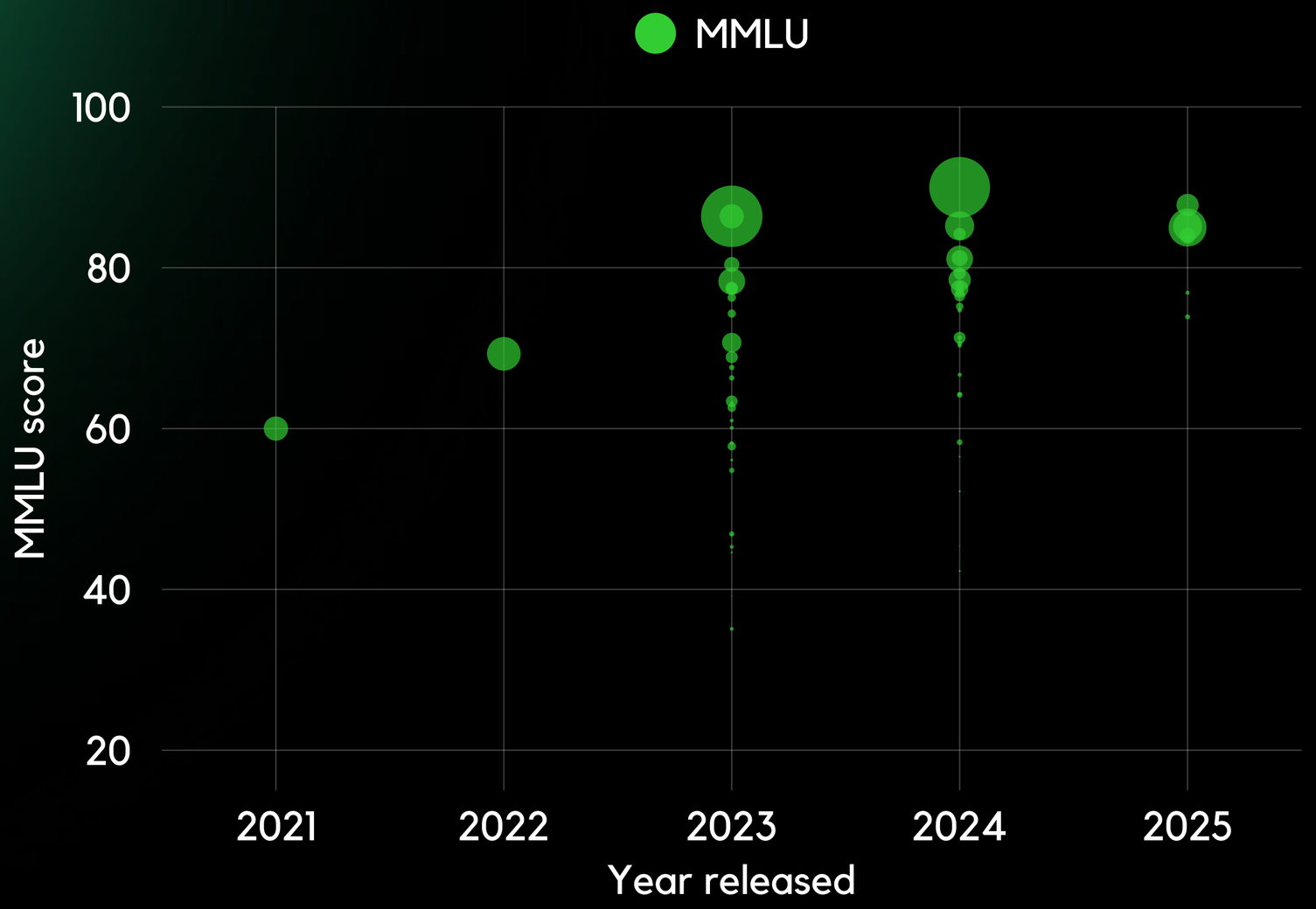
Against model size



SLMs

MMLU scores

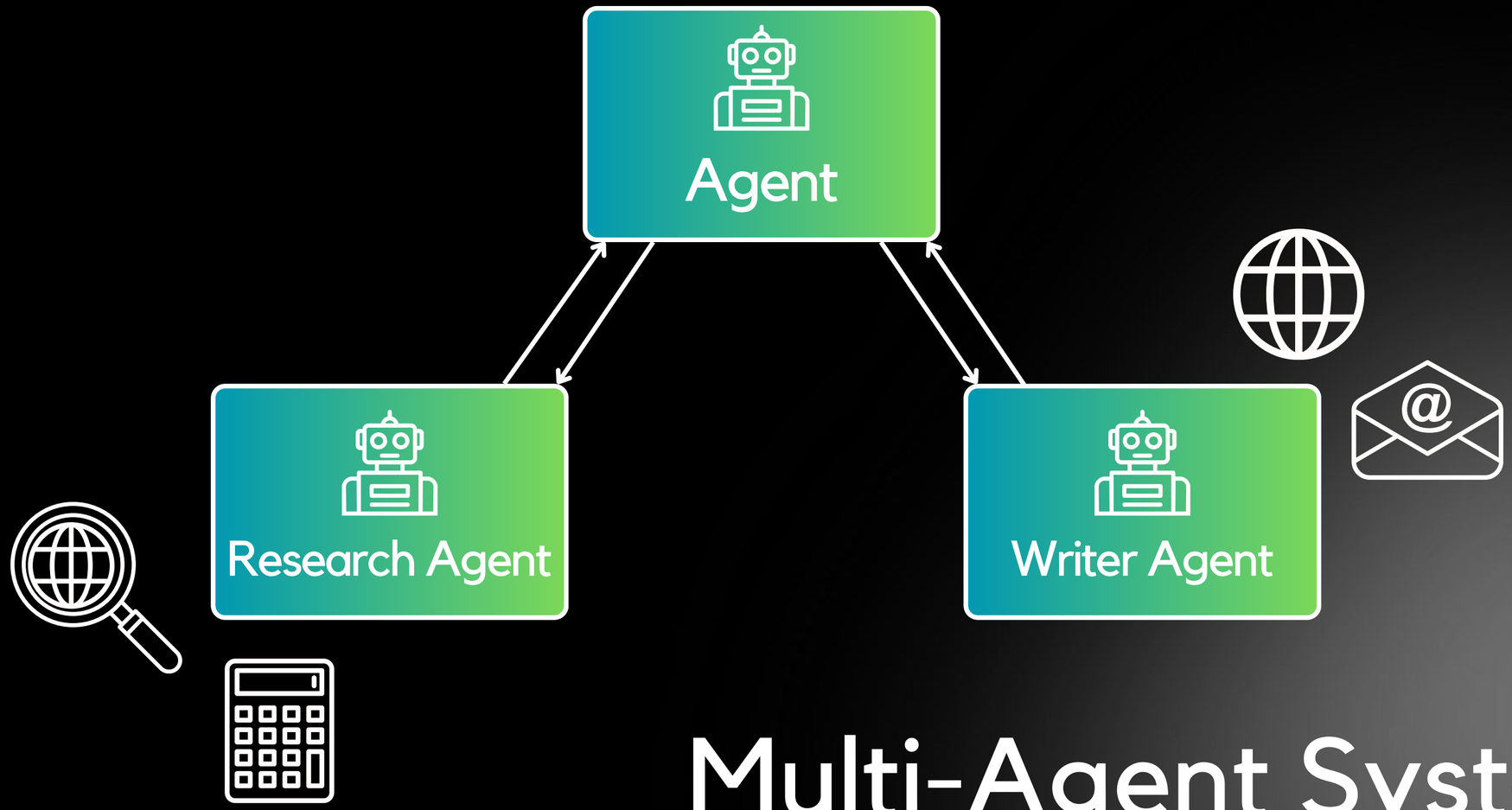
Against year of release



"

What you lack in strength, make up for in wit.

Multi-Agent Systems



Multi-Agent Systems


Organizational


Dual-agent

Organizational

Dual-agent

Organizational

Alzheimer's disease prediction

Li, R et al. CARE-AD: a multi-agent large language model framework for Alzheimer's disease prediction using longitudinal clinical notes (2025)

npj | digital medicine Article

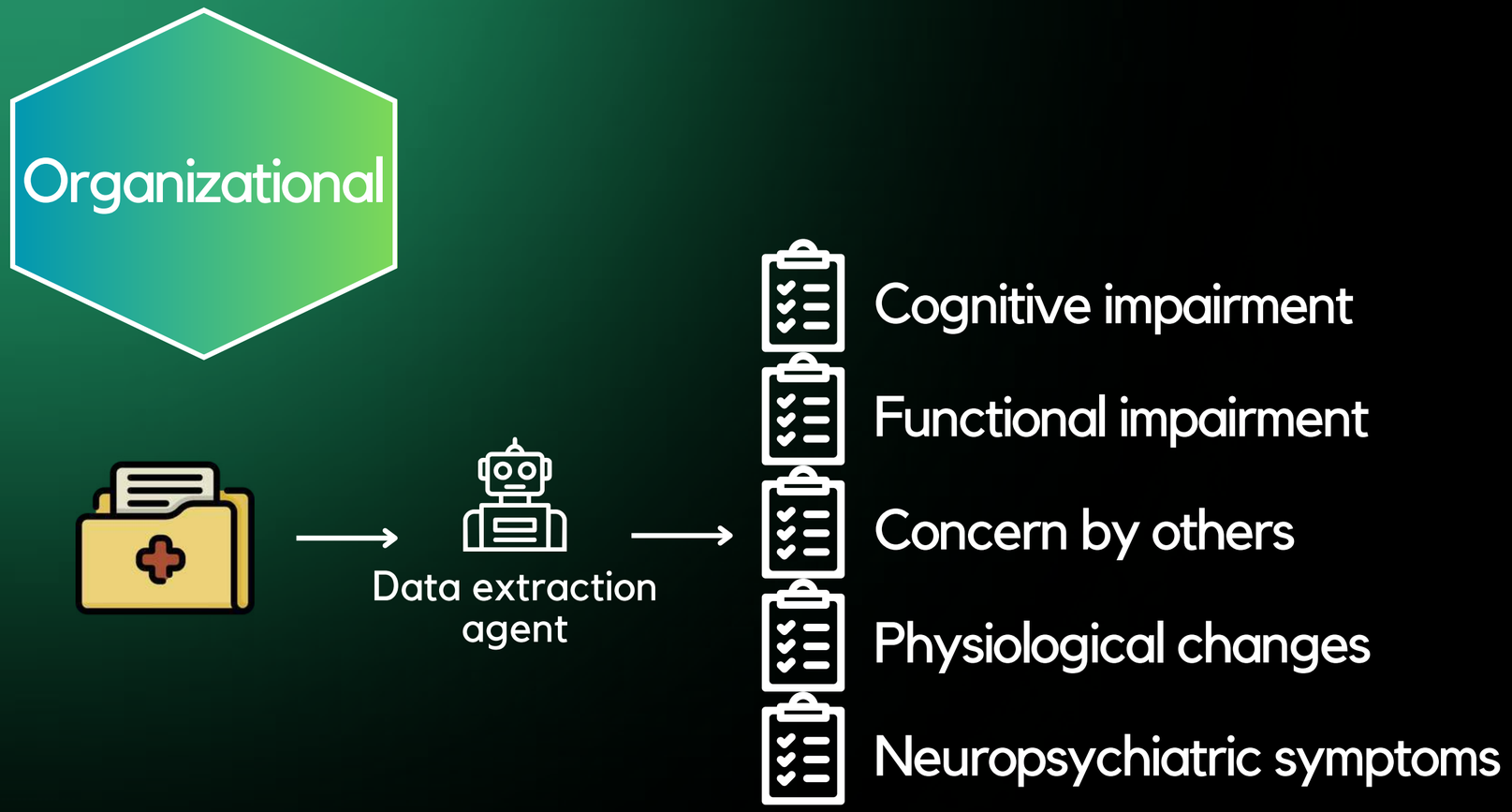
Published in partnership with Seoul National University Bundang Hospital

https://doi.org/10.1038/s41746-025-01940-4

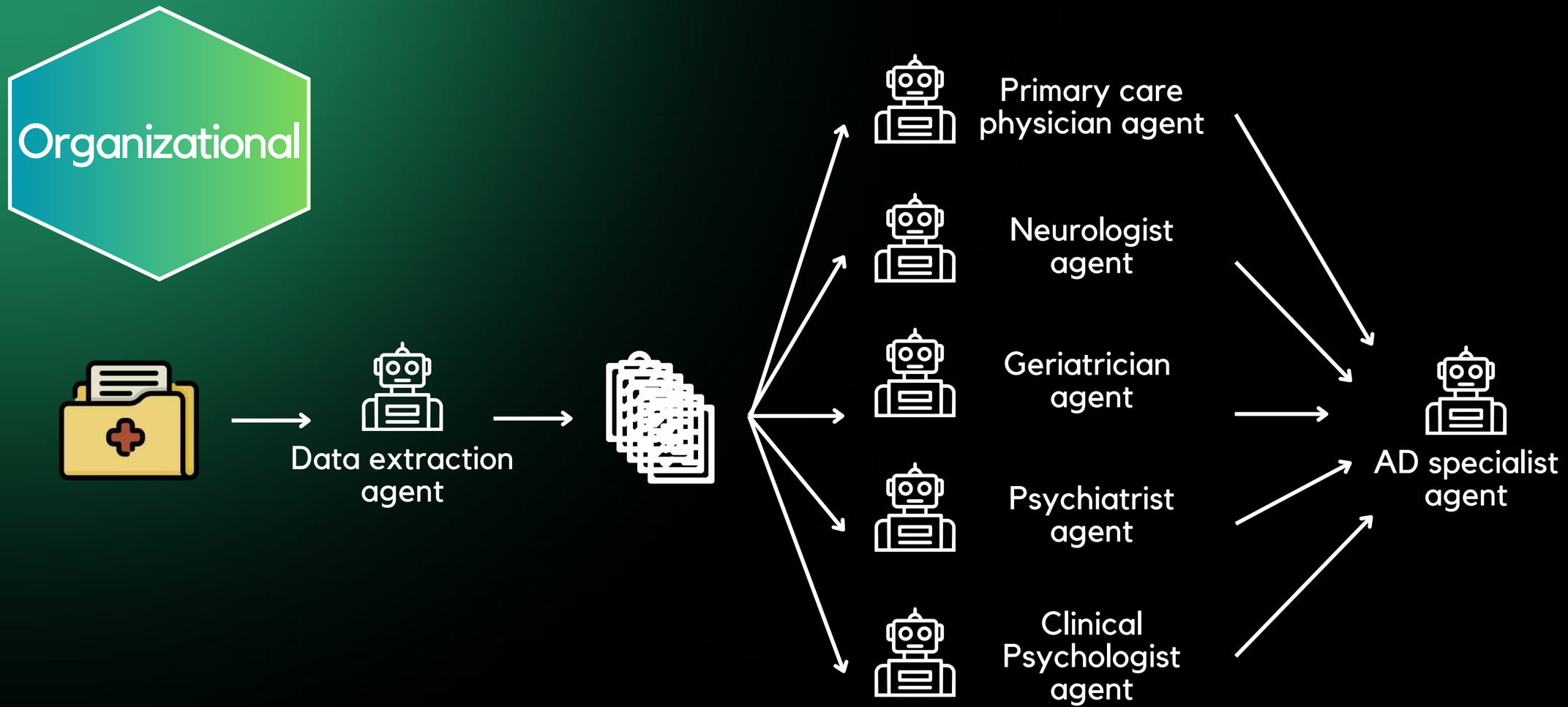
CARE-AD: a multi-agent large language model framework for Alzheimer's disease prediction using longitudinal clinical notes

Check for updates

Rumeng Li[1,2], Xun Wang[3], Dan Berlowitz[2,4,5], Jesse Mez[6], Honghuang Lin[7] & Hong Yu[1,2,5,8]



Alzheimer's disease prediction



Alzheimer's disease prediction

Organizational

Dual-agent

Dual-agent

Virtual software company

Qian, C et al. Communicative Agents for Software Development (2023)

ChatDev: Communicative Agents for Software Development

Chen Qian* Wei Liu* Hongzhang Liu♣ Nuo Chen* Yufan Dang*
Jiahao Li* Cheng Yang♣ Weize Chen* Yusheng Su* Xin Cong*
Juyuan Xu* Dahai Li♦ Zhiyuan Liu*✉ Maosong Sun*✉
*Tsinghua University ♣The University of Sydney ♣BUPT ♦Modelbest Inc.
qianc62@gmail.com liuzy@tsinghua.edu.cn sms@tsinghua.edu.cn

Abstract

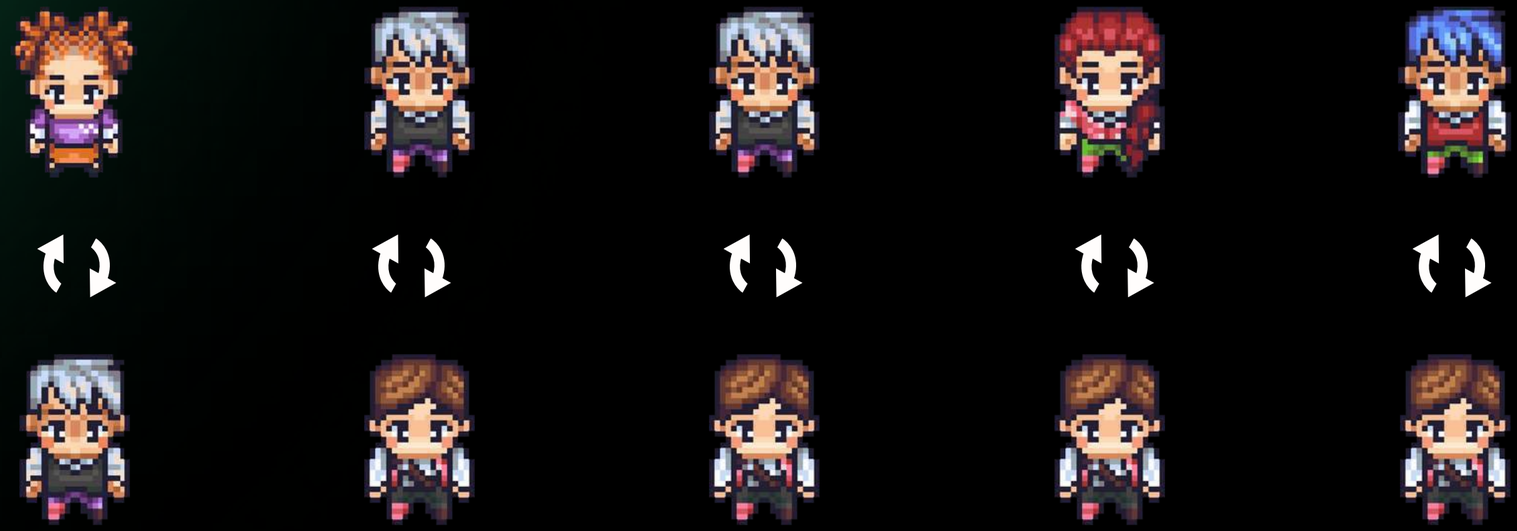
Software development is a complex task that necessitates cooperation among multiple members with diverse skills. Numerous studies used deep learning to improve specific phases in a waterfall model, such as design, coding, and testing. However, the deep learning model in each phase requires unique designs, leading to technical inconsistencies across various phases, which results in a fragmented and ineffective development process. In this paper,

Dual-agent

Phase: Design → Coding → Testing

Subtask: Design | Coding | Code complete | Code review | Testing

	Design	Coding	Code complete	Code review	Testing
Instructor	CEO	CTO	CTO	Reviewer	Tester
Assistant	CTO	Programmer	Programmer	Programmer	Programmer



Virtual software company

Multi-Agent Systems with Small Language Models

Why should you use Small Language Models?

Easier to run locally

Control over your data

Not dependent on tech giants

Require less power

Smaller & cheaper GPUs/CPUs

But can we achieve similar performance?

The virtual software company with Small Language Models

Ollama

Qwen 3 - 4B

Dell XPS 16 9640
RTX 4060 Laptop GPU
~€2000

Task: a todo app where users can add todo items, check off todo items to complete them, and delete

<INFO> Python

Chief Technology Officer

According to the new user's task and our software designs listed below:

Task: "a todo app where users can add todo items, check off todo items to complete them, and delete and edit todo items. The app should have a title or header with 'CIMSOLUTIONS'.".

Task description: "".

Modality: "website".

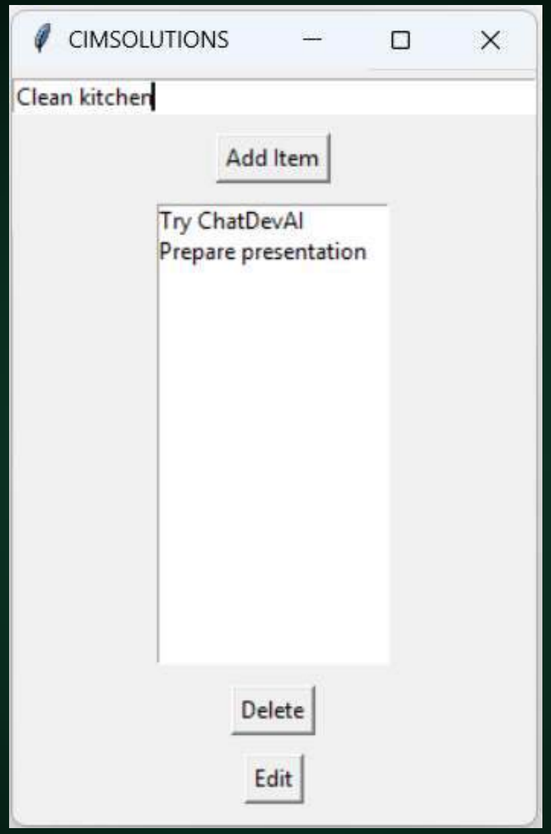
Programming Language: " Python"

Ideas:""

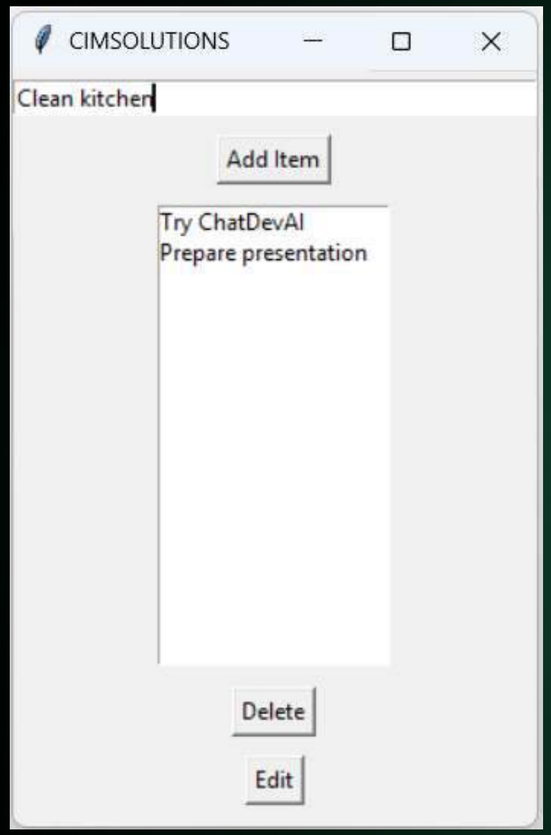
We have decided to complete the task through a executable software with multiple files implemented via Python. As the Programmer, to satisfy the new user's demands, you should write one or multiple files and make sure that every detail of the architecture is, in the end, implemented as code. The software should be equipped with graphical user interface (GUI) so that user can visually and graphically use it; so you must choose a GUI framework (e.g., in Python, you can implement GUI via tkinter, Pygame, Flexx, PyGUI, etc.).

Think step by step and reason yourself to the right decisions to make sure we get it right.

Result: simple TODO list app



Result: simple TODO list app



Qwen 3 - 4B

GPT - 4o

Research

2WikiMultiHopQA

Which film directed by Christopher Nolan stars Leonardo DiCaprio?

Which film directed by Christopher Nolan stars Leonardo DiCaprio?

1 **Christopher Nolan** - "He directed Inception."

2 **Inception** - "The film stars Leonardo DiCaprio."

3 **Interstellar** - "Directed by Nolan, stars McConaughey."

4 **Leonardo da Vinci** - "Italian polymath."

Which film directed by Christopher Nolan stars Leonardo DiCaprio?

1

Christopher Nolan - "He directed Inception."

2

Inception - "The film stars Leonardo DiCaprio."

3

Interstellar - "Directed by Nolan, stars McConaughey."

4

Leonardo da Vinci - "Italian polymath."

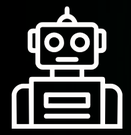
Inception

Solo

Single agent

Solo

R → A

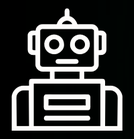


Single agent

Reasoner

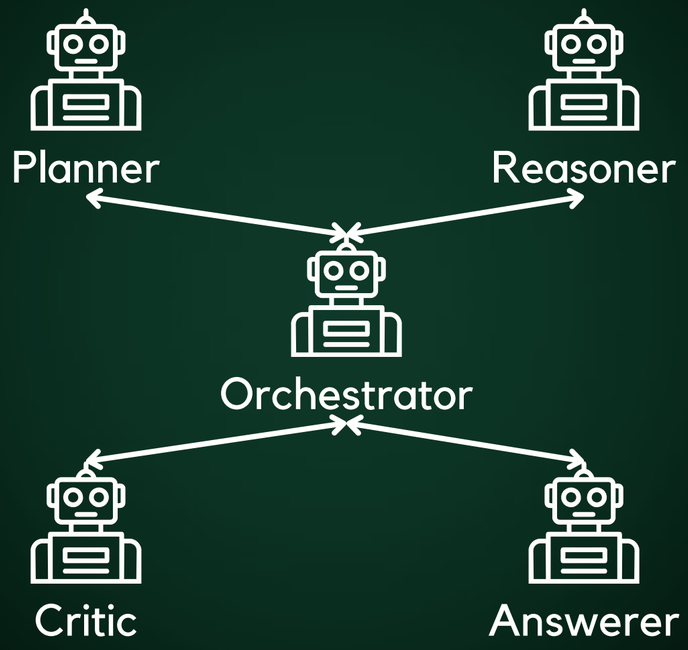
Answerer

Solo

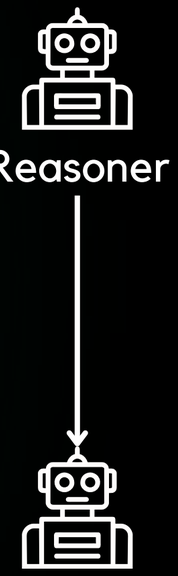


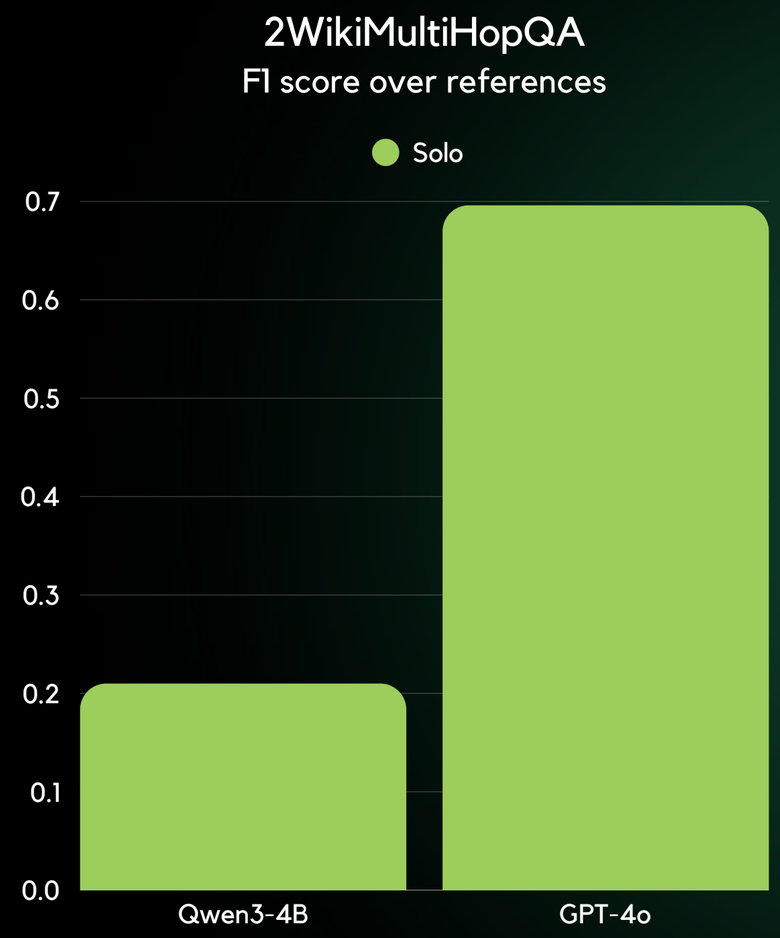
Single agent

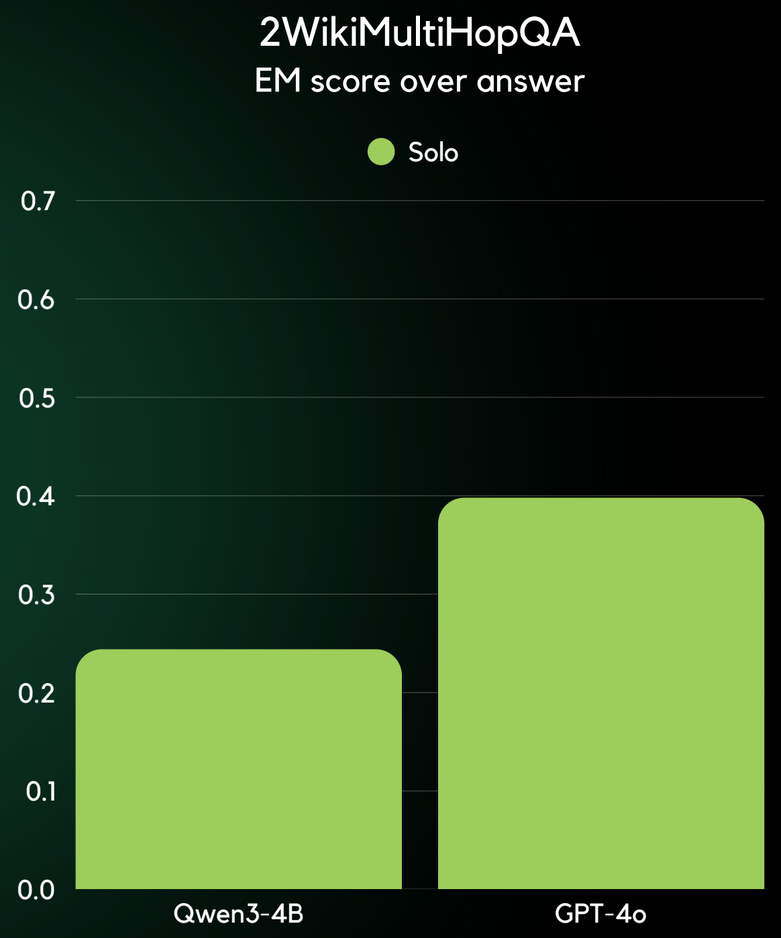
Team

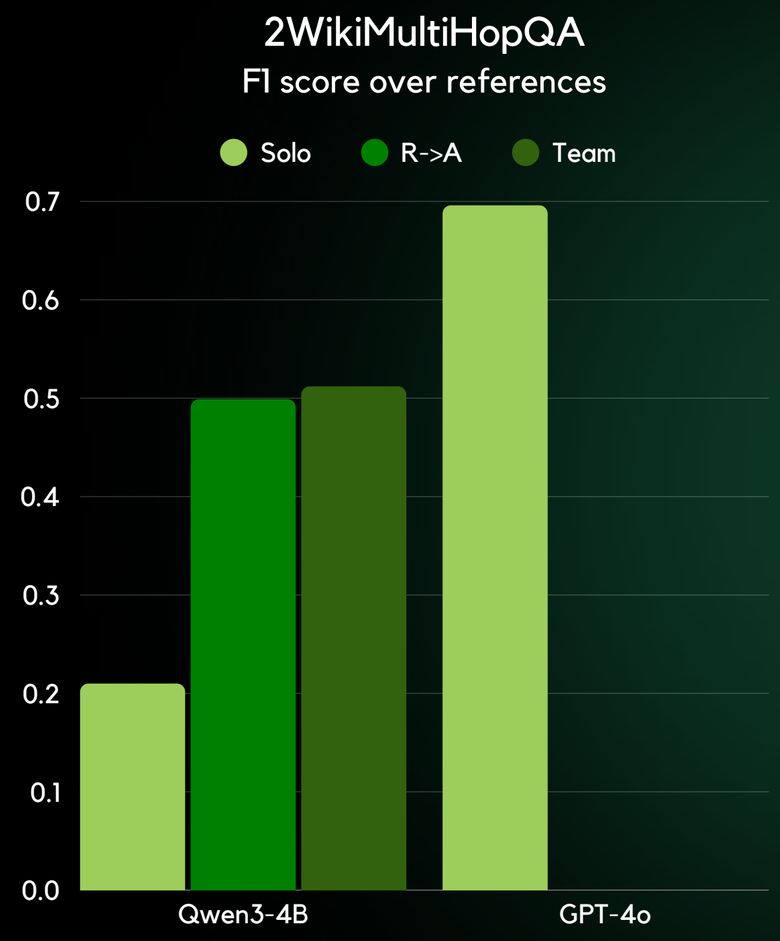


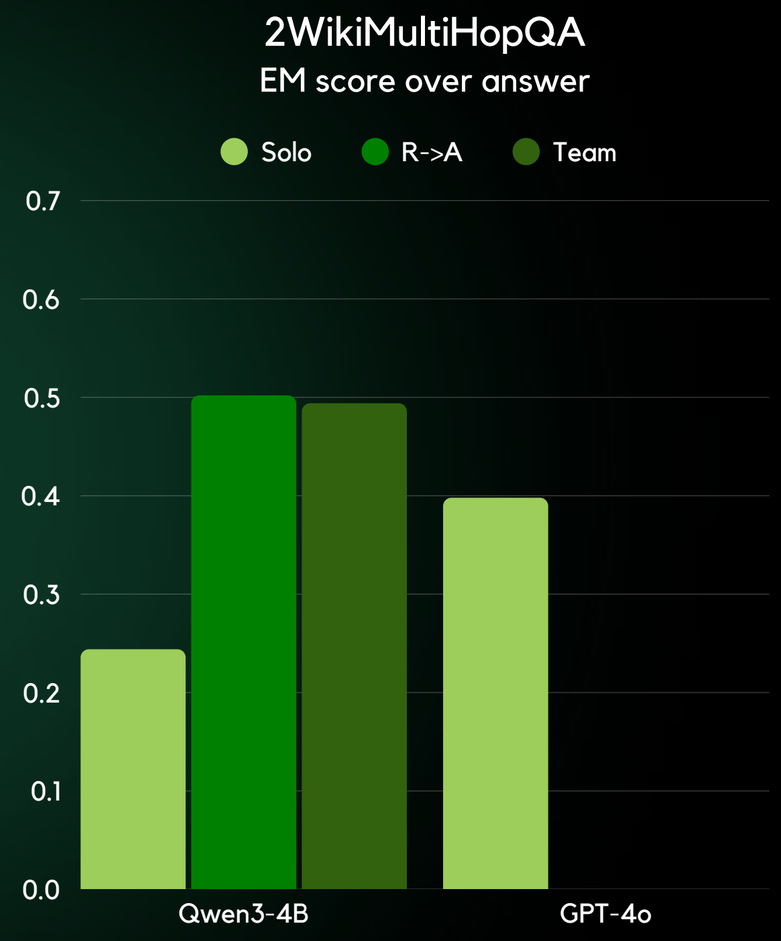
R → A




2WikiMultiHopQA
F1 score over references
Solo
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Qwen3-4B
GPT-4o


2WikiMultiHopQA
EM score over answer
Solo
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Qwen3-4B
GPT-4o


2WikiMultiHopQA
F1 score over references
Solo
R->A
Team
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Qwen3-4B
GPT-4o


2WikiMultiHopQA
EM score over answer
Solo
R->A
Team
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Qwen3-4B
GPT-4o

Under review as a conference paper at ICLR 2025

CMAT: A MULTI-AGENT COLLABORATION TUNING FRAMEWORK FOR ENHANCING SMALL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Open large language models (LLMs) have significantly advanced the field of natural language processing, showcasing impressive performance across various tasks.

Despite the significant advancements in LLMs, their effective operation still relies heavily on human input to accurately guide the dialogue flow, with agent tuning being a crucial optimization technique that involves human adjustments to the model for better response to such guidance. Addressing this dependency, our work introduces the TinyAgent model, trained on a meticulously curated high-quality dataset.

We also present the Collaborative Multi-Agent Tuning (CMAT) framework, an innovative system designed to augment language agent capabilities through adaptive weight updates based on environmental feedback. This framework fosters collaborative learning and real-time adaptation among multiple intelligent agents, enhancing their context-awareness and long-term memory. In this research, we propose a new communication agent framework that integrates multi-agent systems with environmental feedback mechanisms, offering a scalable method to explore cooperative behaviors. Notably, our TinyAgent-7B model exhibits performance on par with GPT-3.5, despite having fewer parameters, signifying a substantial improvement in the efficiency and effectiveness of LLMs.

1 INTRODUCTION

In the rapid development of the field of artificial intelligence, large language models (LLMs) such as BERT and GPT-4 [OpenAI (2023)] have become important cornerstones of natural language processing (NLP). These models utilize the Transformer architecture and effectively capture long-distance dependencies through multi-head self-attention mechanisms, demonstrating strong capabilities across various NLP tasks. With technological advancements, the performance and application scope of LLMs continue to expand, promising significant improvements in computational efficiency and functionality, including anticipated advanced features such as self-improvement, self-checking, and sparse expert models [Liu et al. (2023)].

However, it is noteworthy that the success of these models largely depends on human input to guide the correct dialogue. This dependency requires users to provide relevant and precise prompts based on their intentions and the feedback from the chat agent, raising a critical question: ***Can we replace human intervention with autonomous communication agents capable of steering conversations towards task completion with minimal human supervision?***

Our research addresses the challenges faced by LLMs in real-world deployments, including high computational requirements, data biases, and lack of robustness, which limit their applicability in resource-constrained environments [Abid et al. (2021); Du et al. (2022)]. As shown in Figure 1, we optimize models and training methods to enable smaller models to match larger models' performance. Recognizing MAS's potential to improve processing efficiency through agent cooperation, we develop a collaborative agent framework [Ferry et al. (2018); Talwar et al. (2005)]. Based on our experiments showing that low-quality prompts can significantly degrade model performance, we propose the Collaborative Multi-Agent Tuning (CMAT) framework.

npj | computational materials Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-025-01719-x

SLM-MATRIX: a multi-agent trajectory reasoning and verification framework for enhancing language models in materials data extraction

Check for updates

Xin Li[1,2], Zhixuan Huang[1,2], Shu Quan[1], Cheng Peng[1] & Xiaoming Ma[1]

Small Language Models offer an efficient alternative for structured information extraction. We present **SLM-MATRIX**, a multi-path collaborative reasoning and verification framework based on SLMs, designed to extract material names, numerical values, and physical units from materials science literature. The framework integrates three complementary reasoning paths: a multi-agent collaborative path, a generator–discriminator path, and a dual cross-verification path. SLM-MATRIX achieves an accuracy of 92.85% on the BulkModulus dataset and reaches 77.68% accuracy on the MatSynTriplet dataset, both outperforming conventional methods and single-path models. Moreover, experiments on general reasoning benchmarks such as GSM8K and SVAMP validate the framework's strong generalization capability. Ablation studies evaluate the effects of agent number, Mixture-of-Agents (MoA) depth, and discriminator design on overall performance. Overall, SLM-MATRIX presents an effective approach for high-quality material information extraction in resource-constrained and offers new insights into structured scientific text understanding tasks.

In recent years, large language models (LLMs) have achieved remarkable breakthroughs in natural language understanding and generation domains[1–4]. Through extensive pretraining and reinforcement learning from human feedback, these models are now capable of producing coherent and practical language outputs[5]. Meanwhile, in materials science, automated data extraction plays a crucial role in building comprehensive databases and accelerating knowledge discovery. However, traditional approaches often rely on hand-crafted rules, templates, or fine-tuned models, which are costly, less adaptable, and limited in generalization. Indeed, conventional methods for automated data extraction, such as rule-based parsing and template matching, typically perform well on structured content but face limitations when handling scientific texts characterized by long-range dependencies, implicit semantics, and cross-paragraph reasoning. For example, Kim et al. identified synthesis parameters using syntactic parse trees combined with rule sets[7], while Mavracic et al. extended ChemDataExtractor 2.0 to support table parsing for extracting physical quantities and semantic relations[8]. These techniques often fail to generalize across document formats and domains due to their reliance on brittle rules and heuristics.

With the emergence of generative LLMs, researchers began to explore more scalable extraction methods. Consequently, the emergence of generative LLMs has enabled researchers to extract information such as material names, key property values, and corresponding units with minimal domain expertise, using only carefully designed prompt engineering strategies. For instance, Polak et al. proposed ChatExtract[9], a framework that demonstrates the potential of advanced generative LLMs for efficient data extraction. The core of this method lies in a set of engineered prompts that guide the LLM to identify and extract target information from relevant sentences. More importantly, it introduces a series of follow-up questions to repeatedly verify the accuracy of the extracted data. This interactive verification mechanism effectively mitigates the inherent hallucination problem of LLMs. In evaluations on materials-related datasets, ChatExtract combined with GPT-4 achieved precision and recall rates approaching 90%. Broadly speaking, generative LLMs have demonstrated strong performance across a wide range of natural language processing tasks, largely attributed to the Transformer architecture[10] and large-scale pretraining. The GPT series models[4] have shown excellent generalization in both zero-shot and few-shot settings, and are now widely applied in diverse domains.

[1]Environmental Finance Lab, School of Environment and Energy, Peking University Shenzhen Graduate School, Shenzhen, China. [2]These authors contributed equally: Xin Li, Zhixuan Huang. e-mail: maxm@pkusz.edu.cn

Small Language Models are the Future of Agentic AI

Peter Belcak[1] Greg Heinrich[1] Shizhe Diao[1] Yonggan Fu[1] Xin Dong[1]
Saurav Muralidharan[1] Yingyan Celine Lin[1,2] Pavlo Molchanov[1]
[1]NVIDIA Research [2]Georgia Institute of Technology
agents-research@nvidia.com

Abstract

Large language models (LLMs) are often praised for exhibiting near-human performance on a wide range of tasks and valued for their ability to hold a general conversation. The rise of agentic AI systems is, however, ushering in a mass of applications in which language models perform a small number of specialized tasks repetitively and with little variation.

Here we lay out the position that small language models (SLMs) are *sufficiently powerful, inherently more suitable, and necessarily more economical for many invocations in agentic systems, and are therefore the future of agentic AI*. Our argumentation is grounded in the current level of capabilities exhibited by SLMs, the common architectures of agentic systems, and the economy of LM deployment. We further argue that in situations where general-purpose conversational abilities are essential, heterogeneous agentic systems (i.e., agents invoking multiple different models) are the natural choice. We discuss the potential barriers for the adoption of SLMs in agentic systems and outline a general LLM-to-SLM agent conversion algorithm.

Our position[1], formulated as a value statement, highlights the significance of the operational and economic impact even a partial shift from LLMs to SLMs is to have on the AI agent industry. We aim to stimulate the discussion on the effective use of AI resources and hope to advance the efforts to lower the costs of AI of the present day. Calling for both contributions to and critique of our position, we commit to publishing all such correspondence at `research.nvidia.com/labs/lpr/slm-agents`.

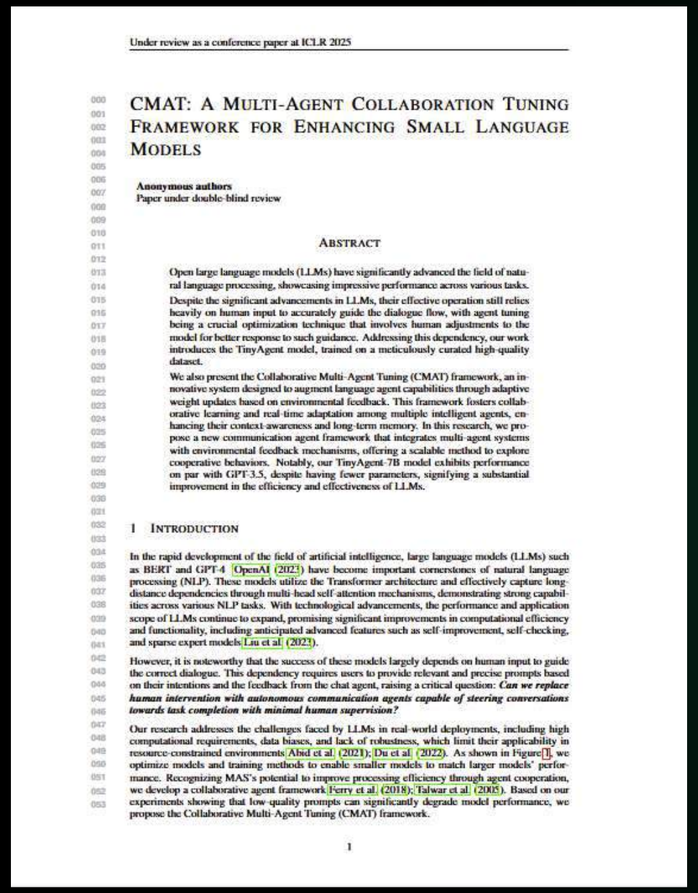
1 Introduction

The deployment of agentic artificial intelligence is on a meteoric rise. Recent surveys show that more than a half of large IT enterprises are actively using AI agents, with 21% having adopted just within the last year [14]. Aside from the users, markets also see substantial economic value in AI agents: As of late 2024, the agentic AI sector had seen more than USD 2bn in startup funding, was valued at USD 5.2bn, and was expected to grow to nearly USD 200bn by 2034 [46, 51]. Put plainly, there is a growing expectation that AI agents will play a substantial role in the modern economy.

The core components powering most modern AI agents are (very) large language models [52, 48]. It is the LLMs that provide the foundational intelligence that enables agents to make strategic decisions about when and how to use available tools, control the flow of operations needed to complete tasks, and, if necessary, to break down complex tasks into manageable subtasks and to perform reasoning for action planning and problem-solving [52, 17]. A typical AI agent then simply communicates with a chosen LLM API endpoint by making requests to centralized cloud infrastructure that hosts these models [52].

[1]The views and positions expressed in this paper are those of the authors and do not necessarily reflect the views or positions of any entities they represent.

Preprint. Under review.



Under review as a conference paper at ICLR 2025

CMAT: A MULTI-AGENT COLLABORATION TUNING FRAMEWORK FOR ENHANCING SMALL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Open large language models (LLMs) have significantly advanced the field of natural language processing, showcasing impressive performance across various tasks.

Despite the significant advancements in LLMs, their effective operation still relies heavily on human input to accurately guide the dialogue flow, with agent tuning being a crucial optimization technique that involves human adjustments to the model for better response to such guidance. Addressing this dependency, our work introduces the TinyAgent model, trained on a meticulously curated high-quality dataset.

We also present the Collaborative Multi-Agent Tuning (CMAT) framework, an innovative system designed to augment language agent capabilities through adaptive weight updates based on environmental feedback. This framework fosters collaborative learning and real-time adaptation among multiple intelligent agents, enhancing their context-awareness and long-term memory. In this research, we propose a new communication agent framework that integrates multi-agent systems with environmental feedback mechanisms, offering a scalable method to explore cooperative behaviors. Notably, our TinyAgent-7B model exhibits performance on par with GPT-3.5, despite having fewer parameters, signifying a substantial improvement in the efficiency and effectiveness of LLMs.

1 INTRODUCTION

In the rapid development of the field of artificial intelligence, large language models (LLMs) such as BERT and GPT-4 OpenAI (2023) have become important cornerstones of natural language processing (NLP). These models utilize the Transformer architecture and effectively capture long-distance dependencies through multi-head self-attention mechanisms, demonstrating strong capabilities across various NLP tasks. With technological advancements, the performance and application scope of LLMs continue to expand, promising significant improvements in computational efficiency and functionality, including anticipated advanced features such as self-improvement, self-checking, and sparse expert models Liu et al. (2023).

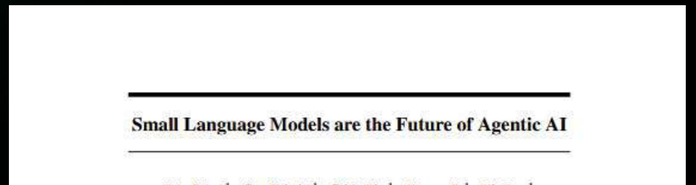
However, it is noteworthy that the success of these models largely depends on human input to guide the correct dialogue. This dependency requires users to provide relevant and precise prompts based on their intentions and the feedback from the chat agent, raising a critical question: ***Can we replace human intervention with autonomous communication agents capable of steering conversations towards task completion with minimal human supervision?***

Our research addresses the challenges faced by LLMs in real-world deployments, including high computational requirements, data biases, and lack of robustness, which limit their applicability in resource-constrained environments Abid et al. (2021); Du et al. (2022). As shown in Figure 1, we optimize models and training methods to enable smaller models to match larger models' performance. Recognizing MAS's potential to improve processing efficiency through agent cooperation, we develop a collaborative agent framework Ferry et al. (2018); Talwar et al. (2005). Based on our experiments showing that low-quality prompts can significantly degrade model performance, we propose the Collaborative Multi-Agent Tuning (CMAT) framework.

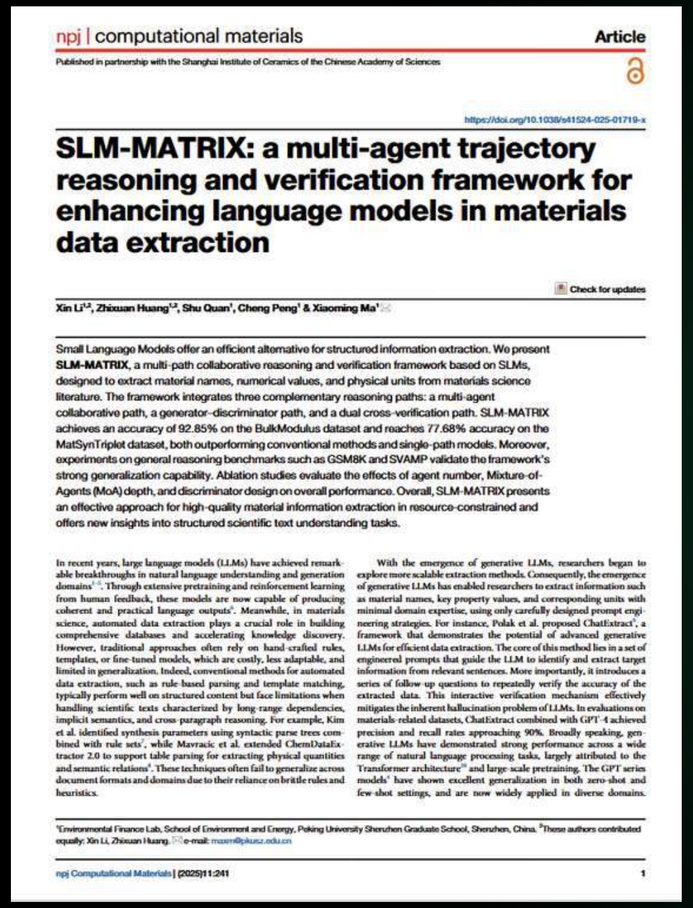
- Finetuned SLMs
- Memory
- Validation by a critic
- Achieves comparable results on benchmarks

npj | computational materials Article
Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences
https://doi.org/10.1038/s41524-025-01719-x

SLM-MATRIX: a multi-agent trajectory reasoning and verification framework for



Small Language Models are the Future of Agentic AI



npj | computational materials Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-025-01719-x

SLM-MATRIX: a multi-agent trajectory reasoning and verification framework for enhancing language models in materials data extraction

Check for updates

Xin Li[1,2], Zhixuan Huang[1,2], Shu Quan[1], Cheng Peng[1] & Xiaoming Ma[1]

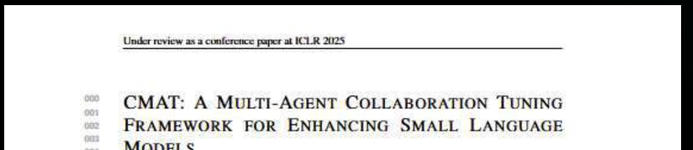
Small Language Models offer an efficient alternative for structured information extraction. We present **SLM-MATRIX**, a multi-path collaborative reasoning and verification framework based on SLMs, designed to extract material names, numerical values, and physical units from materials science literature. The framework integrates three complementary reasoning paths: a multi-agent collaborative path, a generator–discriminator path, and a dual cross-verification path. SLM-MATRIX achieves an accuracy of 92.85% on the BulkModulus dataset and reaches 77.68% accuracy on the MatSynTriplet dataset, both outperforming conventional methods and single-path models. Moreover, experiments on general reasoning benchmarks such as GSM8K and SVAMP validate the framework's strong generalization capability. Ablation studies evaluate the effects of agent number, Mixture-of-Agents (MoA) depth, and discriminator design on overall performance. Overall, SLM-MATRIX presents an effective approach for high-quality material information extraction in resource-constrained and offers new insights into structured scientific text understanding tasks.

In recent years, large language models (LLMs) have achieved remarkable breakthroughs in natural language understanding and generation domains[1–4]. Through extensive pretraining and reinforcement learning from human feedback, these models are now capable of producing coherent and practical language outputs[5]. Meanwhile, in materials science, automated data extraction plays a crucial role in building comprehensive databases and accelerating knowledge discovery. However, traditional approaches often rely on hand-crafted rules, templates, or fine-tuned models, which are costly, less adaptable, and limited in generalization. Indeed, conventional methods for automated data extraction, such as rule-based parsing and template matching, typically perform well on structured content but face limitations when handling scientific texts characterized by long-range dependencies, implicit semantics, and cross-paragraph reasoning. For example, Kim et al. identified synthesis parameters using syntactic parse trees combined with rule sets[6], while Mavracic et al. extended ChemDataExtractor 2.0 to support table parsing for extracting physical quantities and semantic relations[7]. These techniques often fail to generalize across document formats and domains due to their reliance on brittle rules and heuristics.

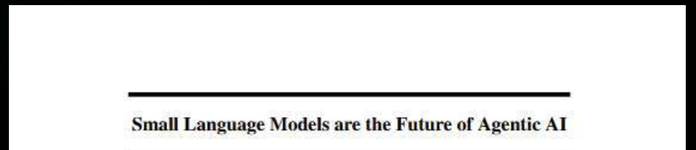
With the emergence of generative LLMs, researchers began to explore more scalable extraction methods. Consequently, the emergence of generative LLMs has enabled researchers to extract information such as material names, key property values, and corresponding units with minimal domain expertise, using only carefully designed prompt engineering strategies. For instance, Polak et al. proposed ChatExtract[8], a framework that demonstrates the potential of advanced generative LLMs for efficient data extraction. The core of this method lies in a set of engineered prompts that guide the LLM to identify and extract target information from relevant sentences. More importantly, it introduces a series of follow-up questions to repeatedly verify the accuracy of the extracted data. This interactive verification mechanism effectively mitigates the inherent hallucination problem of LLMs. In evaluations on materials-related datasets, ChatExtract combined with GPT-4 achieved precision and recall rates approaching 90%. Broadly speaking, generative LLMs have demonstrated strong performance across a wide range of natural language processing tasks, largely attributed to the Transformer architecture[9] and large-scale pretraining. The GPT series models[5] have shown excellent generalization in both zero-shot and few-shot settings, and are now widely applied in diverse domains.

[1]Environmental Finance Lab, School of Environment and Energy, Peking University Shenzhen Graduate School, Shenzhen, China. [2]These authors contributed equally: Xin Li, Zhixuan Huang. e-mail: mazm@pkusz.edu.cn

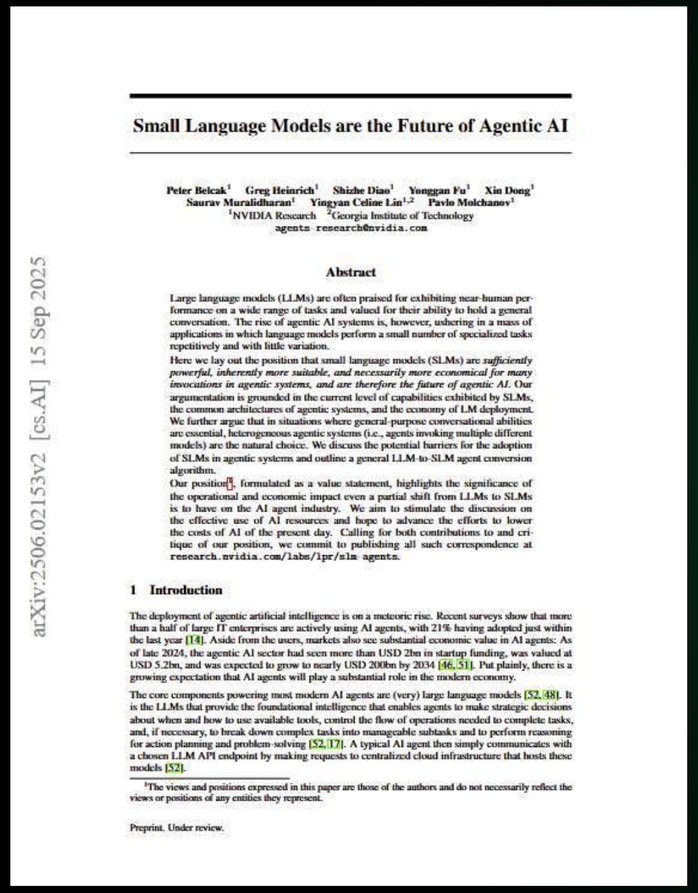
- Multiple SLMs generating answer
- SLM as discriminator
- Comparable results to GPT-4



CMAT: A MULTI-AGENT COLLABORATION TUNING FRAMEWORK FOR ENHANCING SMALL LANGUAGE MODELS



Small Language Models are the Future of Agentic AI



Small Language Models are the Future of Agentic AI

Peter Belcak[1] Greg Heinrich[1] Shizhe Diao[1] Yonggan Fu[1] Xin Dong[1]
Saurav Muralidharan[1] Yingyan Celine Lin[1,2] Pavlo Molchanov[1]
[1]NVIDIA Research [2]Georgia Institute of Technology
agents-research@nvidia.com

Abstract

Large language models (LLMs) are often praised for exhibiting near-human performance on a wide range of tasks and valued for their ability to hold a general conversation. The rise of agentic AI systems is, however, ushering in a mass of applications in which language models perform a small number of specialized tasks repetitively and with little variation.

Here we lay out the position that small language models (SLMs) are *sufficiently powerful, inherently more suitable, and necessarily more economical for many invocations in agentic systems, and are therefore the future of agentic AI*. Our argumentation is grounded in the current level of capabilities exhibited by SLMs, the common architectures of agentic systems, and the economy of LM deployment. We further argue that in situations where general-purpose conversational abilities are essential, heterogeneous agentic systems (i.e., agents invoking multiple different models) are the natural choice. We discuss the potential barriers for the adoption of SLMs in agentic systems and outline a general LLM-to-SLM agent conversion algorithm.

Our position[1], formulated as a value statement, highlights the significance of the operational and economic impact even a partial shift from LLMs to SLMs is to have on the AI agent industry. We aim to stimulate the discussion on the effective use of AI resources and hope to advance the efforts to lower the costs of AI of the present day. Calling for both contributions to and critique of our position, we commit to publishing all such correspondence at `research.nvidia.com/labs/lpr/slm-agents`.

1 Introduction

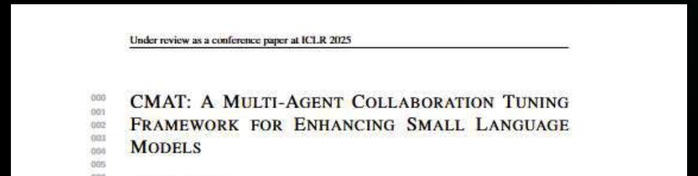
The deployment of agentic artificial intelligence is on a meteoric rise. Recent surveys show that more than a half of large IT enterprises are actively using AI agents, with 21% having adopted just within the last year [14]. Aside from the users, markets also see substantial economic value in AI agents: As of late 2024, the agentic AI sector had seen more than USD 2bn in startup funding, was valued at USD 5.2bn, and was expected to grow to nearly USD 200bn by 2034 [46, 51]. Put plainly, there is a growing expectation that AI agents will play a substantial role in the modern economy.

The core components powering most modern AI agents are (very) large language models [52, 48]. It is the LLMs that provide the foundational intelligence that enables agents to make strategic decisions about when and how to use available tools, control the flow of operations needed to complete tasks, and, if necessary, to break down complex tasks into manageable subtasks and to perform reasoning for action planning and problem-solving [52, 17]. A typical AI agent then simply communicates with a chosen LLM API endpoint by making requests to centralized cloud infrastructure that hosts these models [52].

[1]The views and positions expressed in this paper are those of the authors and do not necessarily reflect the views or positions of any entities they represent.

Preprint. Under review.

- Overkill of generalists
- Are faster, cheaper, easier to fine-tune
- They can understand us
- We often do not do rocket science



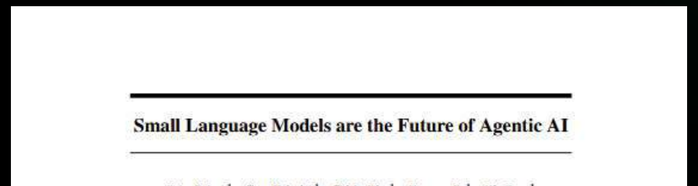
Under review as a conference paper at ICLR 2025

CMAT: A Multi-Agent Collaboration Tuning Framework for Enhancing Small Language Models

npj | computational materials Article
Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences
https://doi.org/10.1038/s41524-025-01719-x

SLM-MATRIX: a multi-agent trajectory reasoning and verification framework for

When small language models collaborate like specialists in a team, they can reach the power of giants.



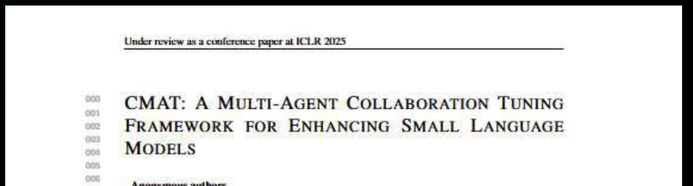
Small Language Models are the Future of Agentic AI

npj | computational materials Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-025-01719-x

SLM-MATRIX: a multi-agent trajectory reasoning and verification framework for



Under review as a conference paper at ICLR 2025

CMAT: A MULTI-AGENT COLLABORATION TUNING FRAMEWORK FOR ENHANCING SMALL LANGUAGE MODELS

Conclusion

Takeaway points

Using LLMs brings concerns about privacy and security

Companies lack the infrastructure to run LLMs themselves

SLMs are getting better and better

Using SLMs in MAS can achieve similar performance as LLMs

In many cases, LLMs are overkill

Come and visit us at stand 14

CIMSOLUTIONS

Learn, create and make it work

Used sources

CARE-AD (Alzheimer prediction): https://www.nature.com/articles/s41746-025-01940-4
ChatDevAI (virtual software company): https://chatdev.ai/
Small Language Models are the Future of Agentic AI: https://arxiv.org/abs/2506.02153
How Small Language Models Are Key to Scalable Agentic AI: https://developer.nvidia.com/blog/how-small-language-models-are-key-to-scalable-agentic-ai/
SLM-MATRIX: https://www.nature.com/articles/s41524-025-01719-x
CMAT: https://openreview.net/forum?id=e8JgXGeuqJ
MMLU: https://medium.com/%40makarenko.roman121/what-you-need-to-know-about-llm-rankings-in-2025-bfdac3d0e466
Qwen3-235B-A22B: https://www.actuia.com/en/news/alibaba-launches-qwen3-235b-a22b-instruct-2507-and-breaks-away-from-hybrid-reasoning/
LLM Leaderboard: https://www.vellum.ai/llm-leaderboard
Qwen3 technical report: https://arxiv.org/pdf/2505.09388
Gemma3: https://deepmind.google/models/gemma/gemma-3/
Gemini2.5 Pro: https://deepmind.google/models/gemini/pro/
Llama 4 herd: https://developer.puter.com/encyclopedia/llama-4/
Deepseek v3.1: https://www.cometapi.com/what-is-deepseek-v3-1/
Reflective Multi-Agent Collaboration based on Large Language Models: https://proceedings.neurips.cc/paper_files/paper/2024/hash/fa54b0edce5eef0bb07654e8ee800cb4-Abstract-Conference.html
A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges: https://link.springer.com/article/10.1007/s44336-024-00009-2
Multi-Agent Collaboration Mechanisms: A Survey of LLMs: https://arxiv.org/abs/2501.06322
Symbiotic Agents: A Novel Paradigm for Trustworthy AGI-driven Networks: https://arxiv.org/abs/2507.17695

CIMSOLUTIONS

Learn, create and make it work