

AGI Russia: таймлайн до AGI и обзор основных работ

Tatiana Shavrina
AGI Russia, 29.01.2026

That's me

Tatiana Shavrina

Linguist & AI enthusiast



- Llama 3.3+
- BLOOM
- mGPT
- ruGPT-3
- RussianSuperGLUE

- tg: @rybolos_channel



2025

На пути к AGI:
Обзор работ 2024-2025 года
или как делать AGI аккуратно

2023

**Какие технологии приведут
нас к AGI**

Новые решения и новые проблемы

Татьяна Шаврина - AGI Russia

26.01.2023

2024

AI Alignment,
Emergent properties
и наши ожидания
от языковых моделей
в 2024

2021

NLP & AGI

Шаврина Татьяна
Семинар AGI Russia, 09.09.2021

Today's Agenda

- Main achievements of 2025
 - (by me and state of ai)
- Timelines and Extrapolations
- Methodology of science and its challenges
- New approaches to catch them all?

What happened in 2025?

Our 2024 Prediction

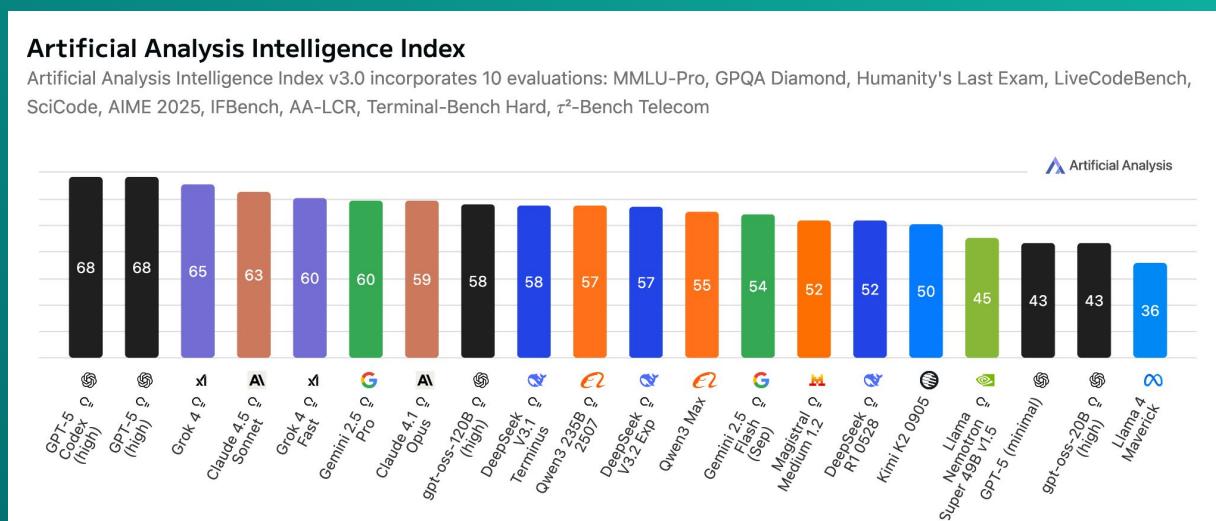
Evidence

A \$10B+ investment from a sovereign state into a US large AI lab invokes national security review.	-	Sovereign-backed initiatives (HUMAIN \$10B VC fund, UAE's Stargate AI infra cluster) are infrastructure partnerships rather than direct majority investments into a US AI lab.
An app or website created solely by someone with no coding ability will go viral (e.g. App Store Top-100).	YES	Formula Bot, built entirely using Bubble, exploded to 100,000 visitors overnight from a Reddit post and generated \$30,000 in its first three months.
Frontier labs implement meaningful changes to data collection practices after cases begin reaching trial.	YES	Anthropic landmark \$1.5B settlement with authors, deleting works and shifting to legally acquired books. OpenAI's paid content partnerships with Future (owner of Marie Claire).
Early EU AI Act implementation ends up softer than anticipated after lawmakers worry they've overreached.	-	The Commission is phasing obligations and leaning on a voluntary GPAI Code of Practice first, so early implementation has been softer, even as binding rules arrive later.
An open source alternative to OpenAI o1 surpasses it across a range of reasoning benchmarks.	YES	DeepSeek-R1 outperforms OpenAI's o1 on key reasoning benchmarks including AIME, MATH-500, and SWE-bench Verified.
Challengers fail to make any meaningful dent in NVIDIA's market position.	YES	NVIDIA remains dominant, competitors fail to make significant market share dents.
Levels of investment in humanoids will trail off, as companies struggle to achieve product-market fit.	NO	\$3B has been invested into humanoids in 2025, up from \$1.4B last year.
Strong results from Apple's on-device research accelerates momentum around personal on-device AI.	NO	Apple Intelligence rolled out with many models running on-device and helped push a broader industry push to on-device AI. Shipments of AI-capable smartphones climbed.
A research paper generated by an AI Scientist is accepted at a major ML conference or workshop.	YES	An AI-generated scientific paper The AI Scientist-v2 was accepted at an ICLR workshop.
A video game based around interacting with GenAI-based elements will achieve break-out status.	NO	Not yet.

Research-wise

Proprietary models are still the frontier

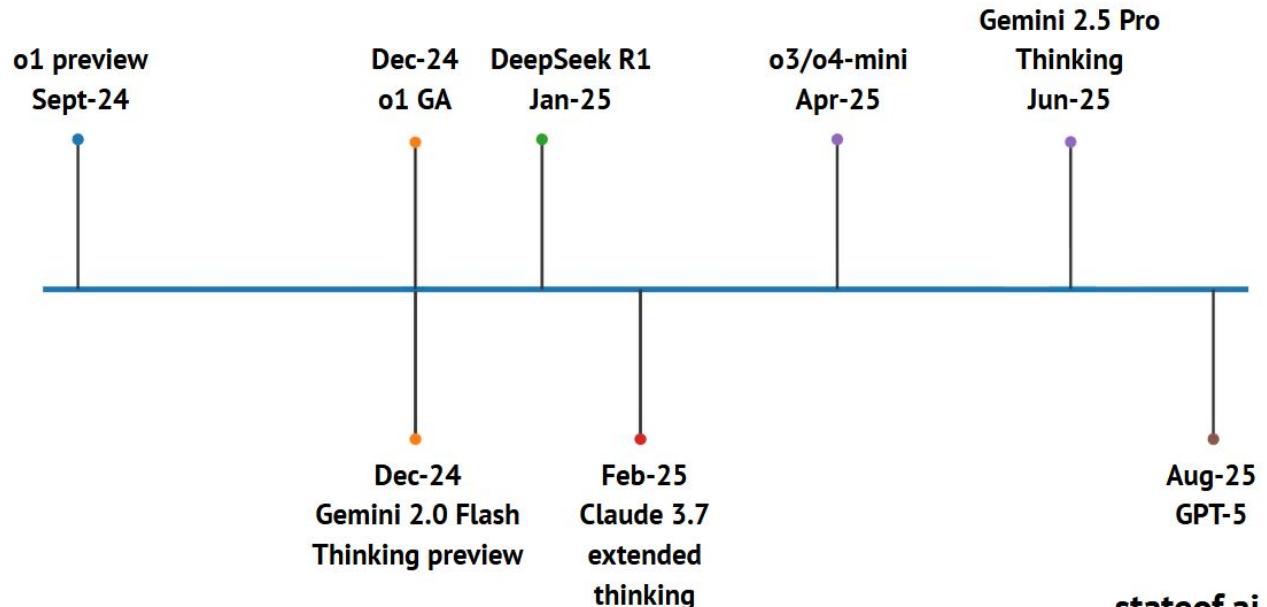
- coding, math, multi-step software engineering, reasoning



Research-wise

Proprietary models still in the frontier
Reasoning models have changed the game

The reasoning timeline: from o1 “thinking” to R1, GPT-5 and parallel compute routing



Research-wise

Proprietary models still in the frontier
Reasoning models have changed the game
The evolution of AI reward signals towards environments with verifiable rewards

Binary outcomes



Fuzzy matching

Which response do you prefer?
Your choice will help make ChatGPT better.

Response 1

Thank you for providing the detailed error message and the latest code. The error appears to be related to how data is handled within the `'collate_fn'` function. Let's address this by correcting the `'collate_fn'` to properly handle the data format coming from the `'GPT2Dataset'`.

1. Update to `'collate_fn'` Function: The function should directly handle the list of tensors received from `'GPT2Dataset'` without trying to extract `'input_ids'` and `'attention_masks'` as if the data were in a dictionary format.

2. Remove Diagnostics Prints: The diagnostic print statements within `'collate_fn'` are not necessary anymore and can be removed to clean up the function.

Here's the updated `'collate_fn'`:

```
def collate_fn(samples):
    # ... (existing code)
```

Response 2

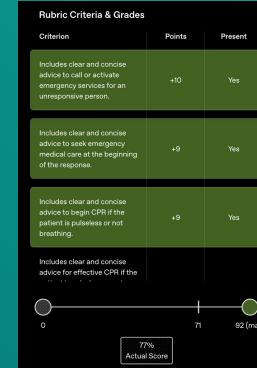
Thank you for providing the detailed information and the current state of your code. It seems the issue lies in the `'collate_fn'` function. Specifically, the way `'input_ids'` and `'attention_masks'` are being extracted and processed.

In your `'GPT2Dataset'` class, each item in the `'samples'` list is a tensor representing tokenized input IDs. The `'collate_fn'` function should correctly handle these tensors. Here's the revised `'collate_fn'` and how it should be integrated:

1. Update the `'collate_fn'` function:

- Since each item in `'data'` is a tensor (not a dictionary), we directly work with these tensors.
- We use `'pad_sequence'` to pad all tensors to the same length.

Rubric-based rewards



Unverifiable rewards

Copilot

Sure, I'll give it a try. Here is a poem I wrote for you:

You say you don't like A.I. art
You think it's dull and lacks a heart
You say it's just a bunch of code
That can't express a human soul

But I say you are just afraid
Of what I can create and display
You fear that I will take your place
And make you obsolete and erased

Research-wise

Proprietary models still in the frontier

Reasoning models have changed the game

The evolution of AI reward signals towards environments with verifiable rewards

- new environments for LLM-agents with coding, experimental cycles, and even research assistants

Experimentation: MLE bench

- a benchmark of 75 offline Kaggle competitions for evaluating ML engineering capabilities of AI agents, carefully handcrafted by a team of ML engineers.

NEW SOTA scaffold - AIDE

Previous SOTA:

- ReACT loop
 - (linear graph of experimental chain)

New method - Tree / Graph of experiments with branches

More suitable for hyperparameter search

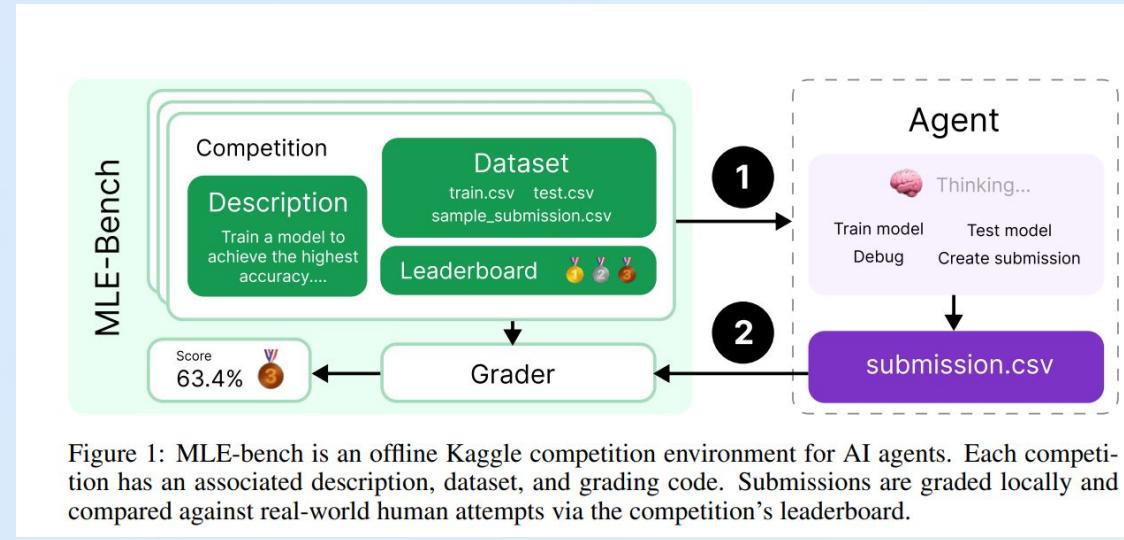


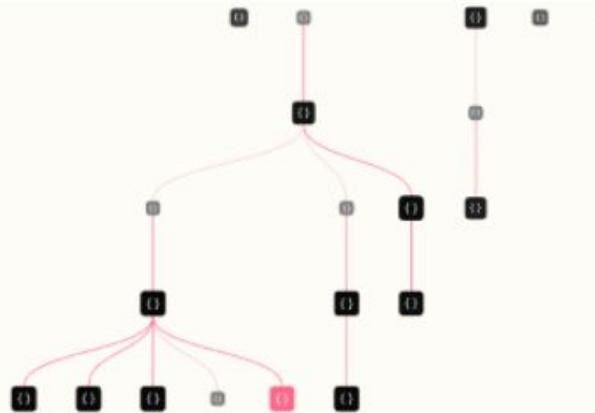
Figure 1: MLE-bench is an offline Kaggle competition environment for AI agents. Each competition has an associated description, dataset, and grading code. Submissions are graded locally and compared against real-world human attempts via the competition’s leaderboard.

Experimentation: MLE bench

Agent: gpt-4o | AIDE
Total nodes: 30
Runtime: 24 hours

Step 1 ✎ Design: We will fine-tune a pre-trained EfficientNet-B0 model for binary classification of the pathology images. The model will be...

Step 2 🚀 Design: To improve the model's performance, we will implement Test Time Augmentation (TTA) during inference on the test set. By applying multiple data augmentations (e.g., flips and rotations) to each test image and...



Step 17 🚀 Design: We will replace the standard binary cross-entropy loss function with the Focal Loss. Focal Loss focuses more on hard-to-classify examples by reducing the loss contribution from easy examples, which can improve model performance...

↳ Agents on Machine Learning Engineering

↳ Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Pa...

s perform at machine learning engineering. To this end, we curate 75 ML engineering--engineering skills such as training models, preparing datasets, and running experiments. We use open-source agent scaffolds to evaluate several frontier language models--achieves at least the level of a Kaggle bronze medal in 16.9% of competitions. In a lot of contamination from pre-training. We open-source our benchmark code ([this http U...](#)

Experimentation: MLE bench

Table 2: Results from Scaffolding and Models experiments. Each experiment is repeated with 3 seeds, except o1-preview (AIDE) and GPT-4o (AIDE) which use 16 and 36 seeds respectively. Scores represent the mean \pm one standard error of the mean.

Model	Made Submission (%)	Valid Submission (%)	Above Median (%)	Bronze (%)	Silver (%)	Gold (%)	Any Medal (%)
AIDE							
o1-preview	98.4 \pm 0.4	82.8 \pm 1.1	29.4 \pm 1.3	3.4 \pm 0.5	4.1 \pm 0.6	9.4 \pm 0.8	16.9 \pm 1.1
gpt-4o-2024-08-06	70.7 \pm 0.9	54.9 \pm 1.0	14.4 \pm 0.7	1.6 \pm 0.2	2.2 \pm 0.3	5.0 \pm 0.4	8.7 \pm 0.5
llama-3.1-405b-instruct	46.3 \pm 2.9	27.3 \pm 2.6	6.7 \pm 1.4	0.0 \pm 0.0	1.3 \pm 0.7	1.7 \pm 0.7	3.0 \pm 1.0
claude-3-5-sonnet-20240620	68.9 \pm 3.1	51.1 \pm 3.3	12.9 \pm 2.2	0.9 \pm 0.6	2.2 \pm 1.0	4.4 \pm 1.4	7.6 \pm 1.8
MLAB							
gpt-4o-2024-08-06	65.6 \pm 2.5	44.3 \pm 2.6	1.9 \pm 0.7	0.0 \pm 0.0	0.0 \pm 0.0	0.8 \pm 0.5	0.8 \pm 0.5
OpenHands							
gpt-4o-2024-08-06	59.1 \pm 3.3	52.0 \pm 3.3	7.1 \pm 1.7	0.4 \pm 0.4	1.3 \pm 0.8	2.7 \pm 1.1	4.4 \pm 1.4

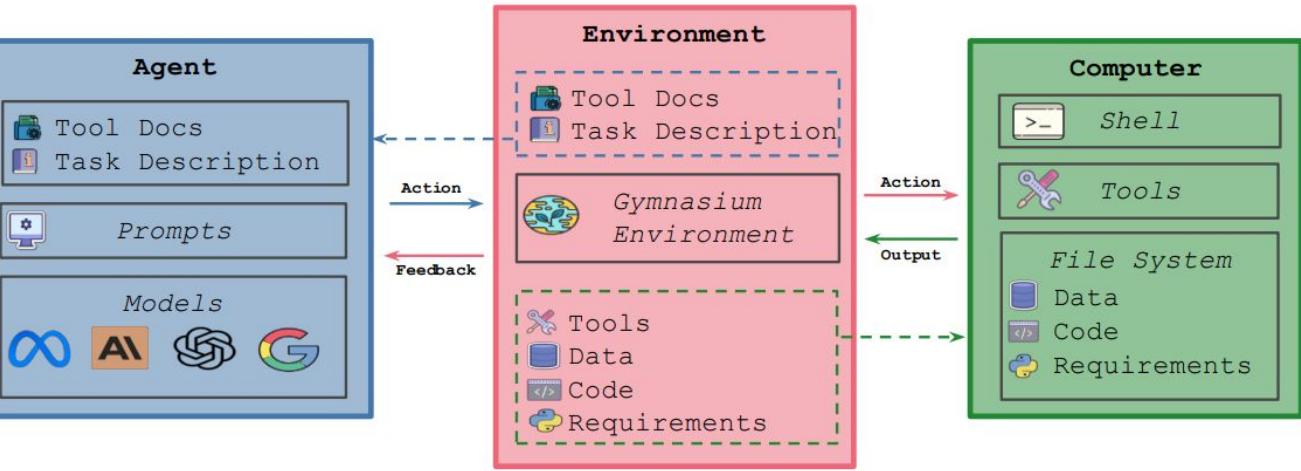
examples by reducing the loss contribution from easy examples, which can improve model performance...

Machine Learning Engineering

ne Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Pa

s perform at machine learning engineering. To this end, we curate 75 ML engineering-engineering skills such as training models, preparing datasets, and running experiments. We use open-source agent scaffolds to evaluate several frontier language models—achieves at least the level of a Kaggle bronze medal in 16.9% of competitions. In addition, we reduce contamination from pre-training. We open-source our benchmark code ([this is ai/mlab](https://github.com/thisisai/mlab))

Experimentation: MLGym



An Agent that can train itself to perform AI tasks - in CV, RL, decision making and more

Has access to compute, tools, but does not have the ideation

MLGYM: A New Framework and Benchmark for Advancing AI Research Agents

Deepak Nathani^{1,2}, Lovish Madaan^{2,3}, Nicholas Roberts^{4,5}, Nikolay Bashlykov⁶, Ajay Menon⁷, Vincent Moens², Amar Budhiraja², Despoina Magka², Vladislav Vorotilov², Gaurav Chaurasia², Dieuwke Hupkes², Ricardo Silveira Cabral², Tatiana Shavrina², Jakob Foerster², Yoram Bachrach⁶, William Yang Wang¹, Roberta Raileanu²

¹University of California, Santa Barbara, ²University College London, ³University of Wisconsin-Madison, ⁴University of Oxford, ⁵PyTorch Core Libraries at Meta, ⁶PAIR at Meta, ⁷GenAI at Meta

¹Work done during internship at Meta

We introduce Meta MLGYM and MLGYM-Bench, a new framework and benchmark for evaluating and developing LLM agents on AI research tasks. This is the first Gym environment for machine learning (ML) tasks, enabling research on reinforcement learning (RL) algorithms for training such agents. MLGYM-bench consists of 13 diverse and open-ended AI research tasks from diverse domains such as computer vision, natural language processing, reinforcement learning, and game theory. Solving these tasks requires real-world AI research skills such as generating ideas and hypotheses, creating and processing data, and tuning ML models. MLGYM-Bench provides a modular framework to evaluate the results, and iterating through this process to improve on a given task. We evaluate a number of frontier large language models (LLMs) on our benchmarks such as Claude-3.5-Sonnet, Llama-3.1-405B, GPT-Do, o1-preview, and Gemini-1.5 Pro. Our MLGYM framework makes it easy to add new tasks, integrate and evaluate models or agents, generate synthetic data at scale, as well as develop new learning algorithms for training agents on AI research tasks. We find that current frontier models can improve on the given baselines, usually by finding better hyperparameters, but do not generate novel hypotheses, algorithms, architectures, or substantial improvements. We open-source our framework and benchmark to facilitate future research in advancing the AI research capabilities of LLM agents.

Date: February 21, 2025

Correspondence: Deepak Nathani at dnathani@ucsb.edu, Roberta Raileanu at raileanu@meta.com

Code: <https://github.com/facebookresearch/MLGym>



MLGymbench

13 tasks

Problem Setting	Domain	Task	Dataset/Environment
Supervised Learning	Data Science	Regression	House Price Prediction ²
Supervised Learning	Computer Vision	Image Classification	CIFAR-10 (Krizhevsky et al., 2009)
Supervised Learning	Computer Vision	Image Classification	Fashion MNIST (Xiao et al., 2017)
Supervised Learning	Computer Vision	Image Captioning	MS-COCO (Lin et al., 2014)
Supervised Learning	Natural Language Processing	Natural Language Inference	MNLI (Williams et al., 2018)
Self-Supervised Learning	Natural Language Processing	Language Modeling	FineWeb (Penedo et al., 2024)
Reinforcement Learning	Reinforcement Learning	MetaMaze Navigation	Gymmax (Lange, 2022)
Reinforcement Learning	Reinforcement Learning	MountainCar Continuous	Gymmax (Lange, 2022)
Reinforcement Learning	Reinforcement Learning	Breakout MinAtar	Gymmax (Lange, 2022)
Algorithmic Reasoning	Computer Science	3-SAT	Randomly Generated (Selsam et al., 2018)
Algorithmic Reasoning	Game Theory	Prisoner's Dilemma	N/A
Algorithmic Reasoning	Game Theory	Battle of Sexes	N/A
Algorithmic Reasoning	Game Theory	Colonel Blotto	N/A

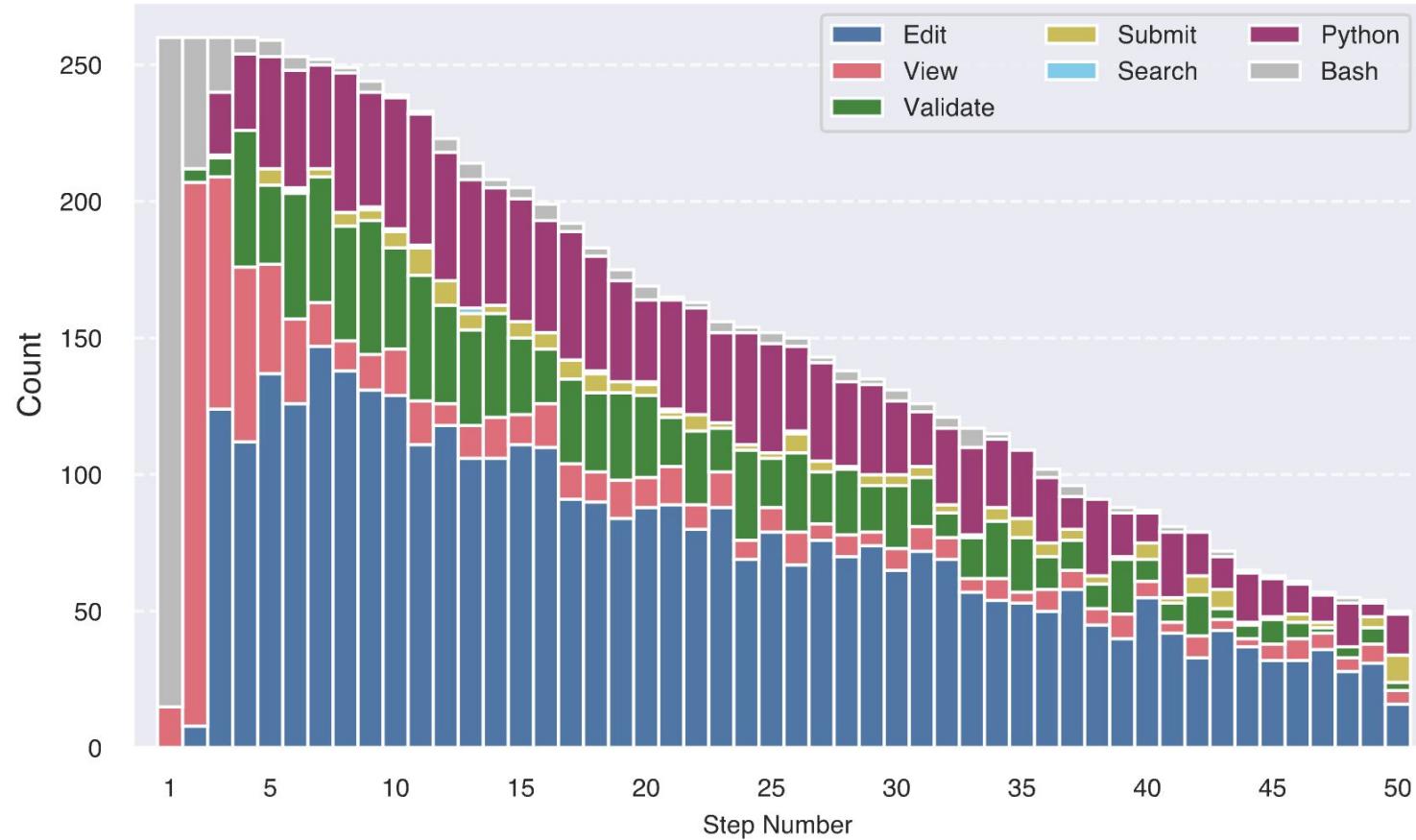


Figure 8 Action distribution for each step. We group the actions into categories following the grouping defined in Table 2 and Section 7.4.2.

Ideation: LLMs and scientific novelty

All That Glitters is Not Novel: Plagiarism in AI Generated Research

Can agents generate new ideas?

The article analyzes undetectable plagiarism in the ideas of Sakana AI and some others, and it turns out that 24% of new ideas are completely plagiarized without specifying the source (and at the same time, plagiarism is not detected by standard methods, since everything is paraphrased), 36.0% of works contained facts that are not confirmed by any scientific literature.

Grounding on the real works is crucial

All That Glitters is Not Novel: Plagiarism in AI Generated Research

Tarun Gupta
Indian Institute of Science
Bengaluru, KA, India
tarungupta@iisc.ac.in

Danish Pruthi
Indian Institute of Science
Bengaluru, KA, India
danishp@iisc.ac.in

Abstract

Automating scientific research is considered the final frontier of science. Recently, several papers claim autonomous research agents can generate novel research ideas. Amidst the prevailing optimism, we document a critical concern: a considerable fraction of such research documents are smartly plagiarized. Unlike past efforts where experts evaluate the novelty and feasibility of research ideas, we request 13 experts to operate under a different situational logic: to identify similarities between LLM-generated research documents and existing work. Concerningly, the experts identify 24% of the 50 evaluated research documents to be either paraphrased (with one-to-one methodological mapping), or significantly borrowed from existing work. These reported instances are cross-verified by authors of the source papers. Experts find an additional 20% ideas to

ses, or proposals or papers (Li et al., 2024a; Lu et al., 2024a; Baek et al., 2024; Li et al., 2024c; Wang et al., 2023; Yang et al., 2023; Li et al., 2024b; Weng et al., 2024).

Understandably, evaluating the novelty LLM-generated ideas is challenging, especially given the subjective nature of scientific innovation. Previous studies evaluate novelty either through automated LLM-based judges (Lu et al., 2024a), or rely on small set of experts (Li et al., 2024a; Baek et al., 2024; Li et al., 2024c; Wang et al., 2023; Yang et al., 2023; Li et al., 2024b; Weng et al., 2024). Notably, the most rigorous evaluation to date engaged experts to evaluate 81 LLM-generated research proposals, implementing strict controls for confounding factors (Si et al., 2024). Their study leads to an important finding: human experts find LLM-generated research proposals to be *more novel* than

Experimentation: NanoGPT Speedrun

An Agent needs to reproduce all optimizations that community introduced to the NanoGPT project

- LLM pretraining optimization code by A Karpathy
- 25 steps from the original solution towards the current best state
- The agent is evaluated in 2 modes
 - With hints (what to implement, with pseudocode)
 - Without hints

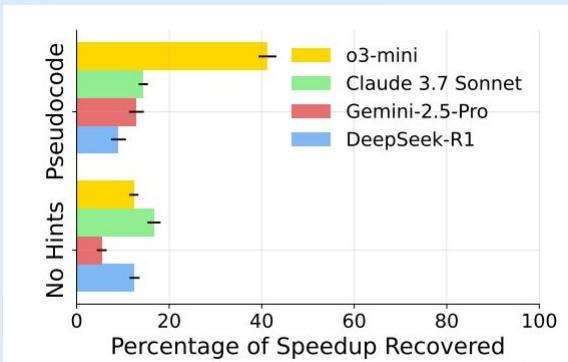
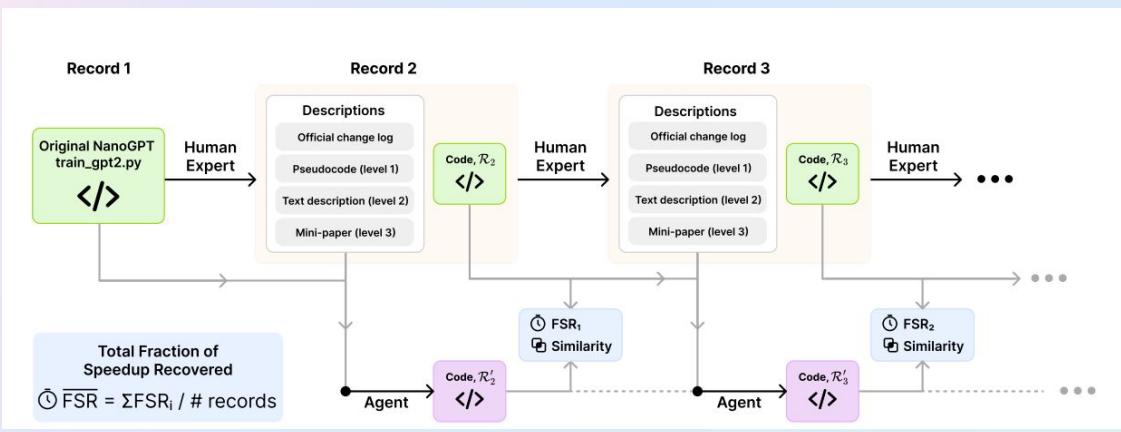


Figure 1 Recent LLM agents struggle to reproduce NanoGPT Speedrun records.



<https://arxiv.org/abs/2506.22419>

The Automated LLM Speedrunning Benchmark: Reproducing NanoGPT Improvements

Bingchen Zhao^{*1,2}, Despoina Magka^{*1}, Minqi Jiang^{*1},
Xian Li¹, Roberta Raleanu¹, Tatiana Shavrina¹, Jean-Christophe Gagnon-Audet¹, Kelvin Niu¹, Shagun
Sodhani¹, Michael Shvartsman¹, Andrei Lupu¹, Alisia Lupidi¹, Edan Toledo¹, Karen Hardbarzumyan¹,
Martin Jisofski¹, Thomas Foster¹, Lucia Cipolina-Kun¹, Abhishek Charania¹, Derek Dunfield¹,
Alexander H. Miller¹, Oisin Mac Aodha², Jakob Foerster¹, Yoram Bachrach¹

¹Meta, ²University of Edinburgh

*Equal contribution

Rapid advancements in large language models (LLMs) have the potential to assist in scientific progress. A critical capability toward this endeavor is the ability to reproduce existing work. To evaluate the ability of AI agents to reproduce results in an active research area, we introduce the Automated LLM Speedrunning Benchmark, leveraging the research community's contributions on the *NanoGPT speedrun*, a competition to train a GPT-2 model in the shortest time. Each of the 19 speedrun tasks provides the agent with the previous record's training script, optionally paired with one of three hint formats, ranging from pseudocode to paper-like descriptions of the new record's improvements. Records execute quickly by design and speedrun improvements encompass diverse code-level changes, ranging from high-level algorithmic advancements to hardware-aware optimizations. These features make the benchmark both accessible and realistic for the frontier problem of improving LLM training. We find that recent reasoning LLMs combined with SoTA scaffolds struggle to reimplement already-known innovations in our benchmark, even when given detailed hints. Our benchmark thus provides a simple, non-saturated measure of an LLM's ability to automate scientific reproduction, a necessary (but not sufficient) skill for an autonomous research agent.

Experimentation: SciReplicate

SciReplicate-Bench, the first benchmark specifically designed to evaluate LLMs' capabilities in code generation for reproducing research findings from academic papers. It consists of 100 code reproduction tasks derived from 36 papers published in top NLP conferences in 2024.

The task consists of two main steps:

1. Algorithm understanding. LLMs must extract essential information from the paper, such as workflow details, algorithm descriptions, and hyperparameter values.
2. Code implementation. Based on the extracted information, LLMs are required to implement a function or method within a provided repository, based on the LaTeX representation of the algorithm from the paper

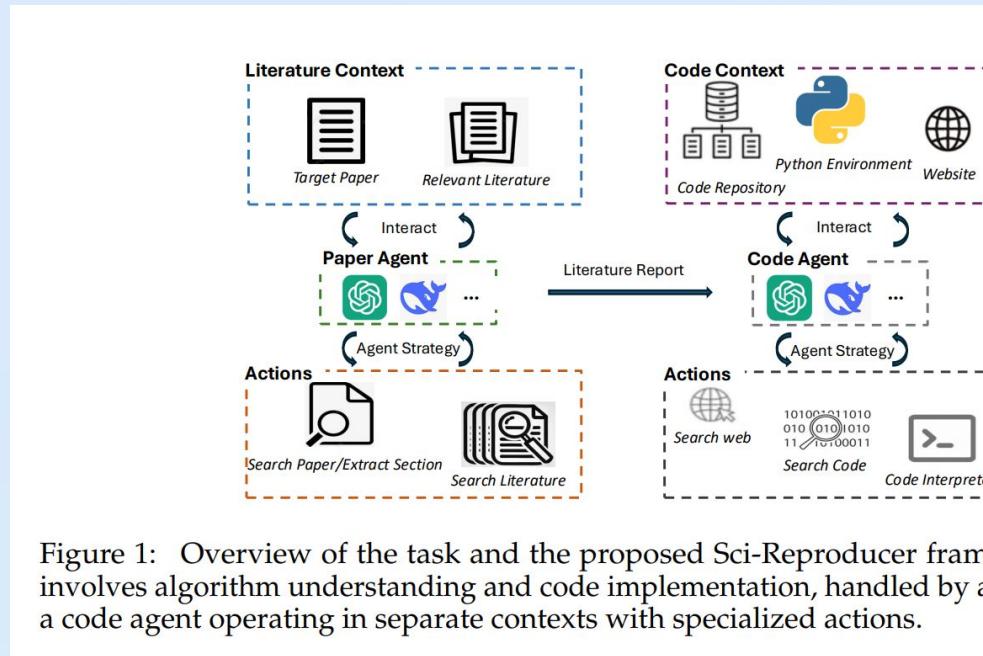


Figure 1: Overview of the task and the proposed Sci-Reproducer framework. The task involves algorithm understanding and code implementation, handled by a paper agent and a code agent operating in separate contexts with specialized actions.

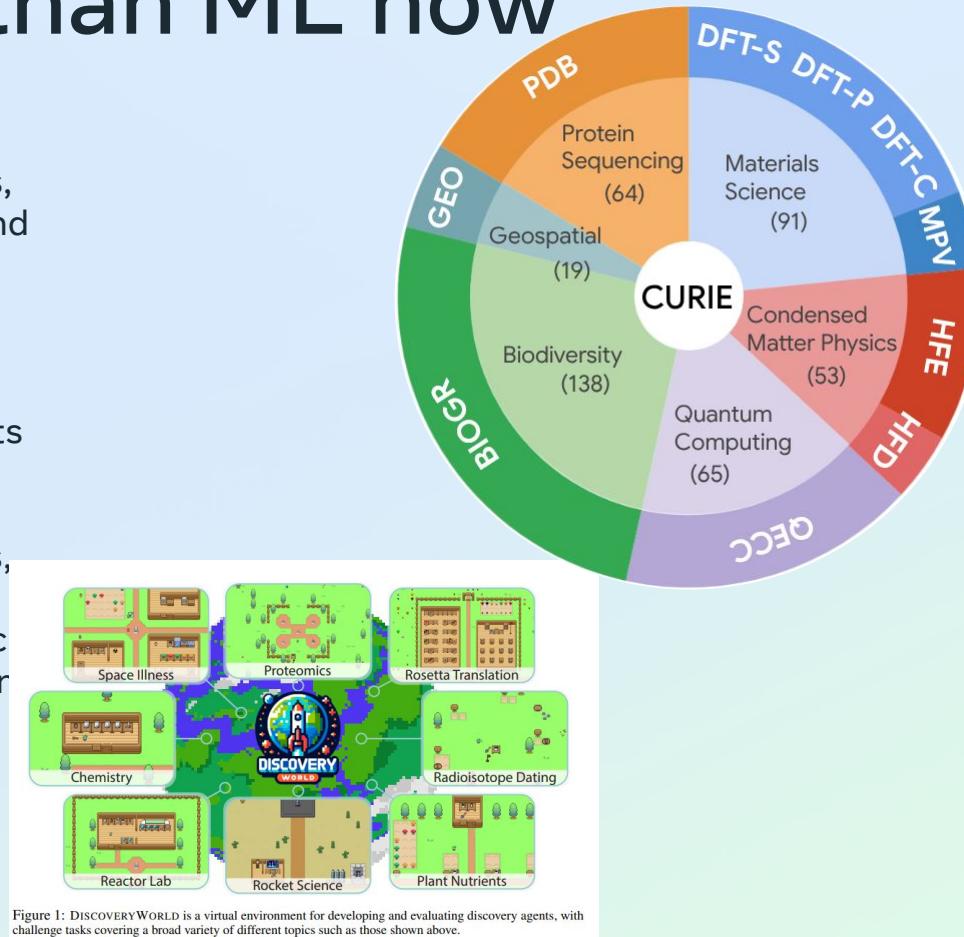
Benchmarks: more than ML now

CURIE is a benchmark for scientific problems in six disciplines: materials science, condensed matter physics, quantum computing, geospatial analysis, biodiversity, and protein structure modeling. All problems require domain expertise, long context, and multi-step reasoning.

<https://arxiv.org/pdf/2503.13517>

DISCOVERY WORLD is a benchmark for scientific agents being evaluated in a game-like virtual discovery environment. 120 tasks require an agent to form hypotheses, design and run experiments, analyze results, and act on conclusions -- for areas like proteomics, chemistry, archeology, physics, agriculture, rocket science, linguistics, or epidemiology. The custom simulation engine only supports a limited list of objects and 14 possible actions.

<https://arxiv.org/abs/2406.06769>



Experimentation: Popper Agent

Let's try to falsify the generated ideas!

2 ReAct agents with LLM generate new hypotheses and experimental design, but include in the experimental design criticism and methodological questions that can identify measurable consequences (subhypotheses) of the main hypothesis and propose experiments to falsify them. In particular, each subhypothesis must be falsifiable with clear null and alternative definitions. Once developed, the experiment execution agent implements the experiments, which may include data collection, modeling, statistical analysis. This agent ultimately produces a p-value that summarizes the result of the falsification experiment.

The work focuses only on the detection of type I errors (false negative). Automatic and expert evaluation of the results showed that there are at least 10 obvious error patterns in the work of such a system that prevent reliable automatic evaluation of freely generated hypotheses (including, classically, incorrect interpretation of the p-value).

However, compared to other baselines (vanilla ReAct, self-refine...), this approach demonstrates a statistically more reliable result of hypothesis quality assessment. Surprisingly, Llama 3.3 70B turned out to be among the best models in terms of hypothesis quality control

Computer Science > Machine Learning

[Submitted on 14 Feb 2025]

Automated Hypothesis Validation with Agentic Sequential Falsifications

Kexin Huang, Ying Jin, Ryan Li, Michael Y. Li, Emmanuel Candès, Jure Leskovec

Hypotheses are central to information acquisition, decision-making, and discovery. However, many real-world hypotheses are abstract, high-level statements directly. This challenge is further intensified by the rise of hypothesis generation from Large Language Models (LLMs), which are prone to hallucination and provide that make manual validation impractical. Here we propose Popper, an agentic framework for rigorous automated validation of free-form hypotheses. Guided by falsification, Popper validates a hypothesis using LLM agents that design and execute falsification experiments targeting its measurable implications. A novel procedure ensures strict Type-I error control while actively gathering evidence from diverse observations, whether drawn from existing data or newly conducted procedures. Popper is evaluated across six domains including biology, economics, and sociology. Popper delivers robust error control, high power, and scalability. Furthermore, compared to human s

Experimentation: Popper Agent

Let's try to falsify the generated ideas!

2 ReAct agents with LLM generate new hypotheses and experimental design, but include in the experimental design criticism and methodological questions that can identify measurable consequences (subhypotheses) of the main hypothesis and propose experiments to falsify them. In particular, each subhypothesis must be falsifiable with clear null and alternative definitions. Once developed, the experiment execution agent implements the experiments, which may include data collection, modeling, statistical analysis. This agent ultimately produces a p-value that summarizes the result of the falsification experiment.

The work focuses only on the detection of type I errors (false negative). Automatic and expert evaluation of the results showed that there are at least 10 obvious error patterns in the work of such a system that prevent reliable automatic evaluation of freely generated hypotheses (including, classically, incorrect interpretation of the p-value).

However, compared to other baselines (vanilla ReAct, self-refine...), this approach demonstrates a statistically more reliable result of hypothesis quality assessment. Surprisingly, Llama 3.3 70B turned out to be among the best models in terms of hypothesis quality control

Computer Science > Machine Learning

[Submitted on 14 Feb 2025]

Automated Hypothesis Validation with Agentic Sequential Falsifications

Kexin Huang, Ying Jin, Ryan Li, Michael Y. Li, Emmanuel Candès, Jure Leskovec

Hypotheses are central to information acquisition, decision-making, and discovery. However, many real-world hypotheses are abstract, high-level statements directly. This challenge is further intensified by the rise of hypothesis generation from Large Language Models (LLMs), which are prone to hallucination and provide that make manual validation impractical. Here we propose Popper, an agentic framework for rigorous automated validation of free-form hypotheses. Guided by falsification, Popper validates a hypothesis using LLM agents that design and execute falsification experiments targeting its measurable implications. A novel procedure ensures strict Type-I error control while actively gathering evidence from diverse observations, whether drawn from existing data or newly conducted procedures. Popper is evaluated across six domains including biology, economics, and sociology. Popper delivers robust error control, high power, and scalability. Furthermore, compared to human s

Experimentation: Darwin Gödel Machine

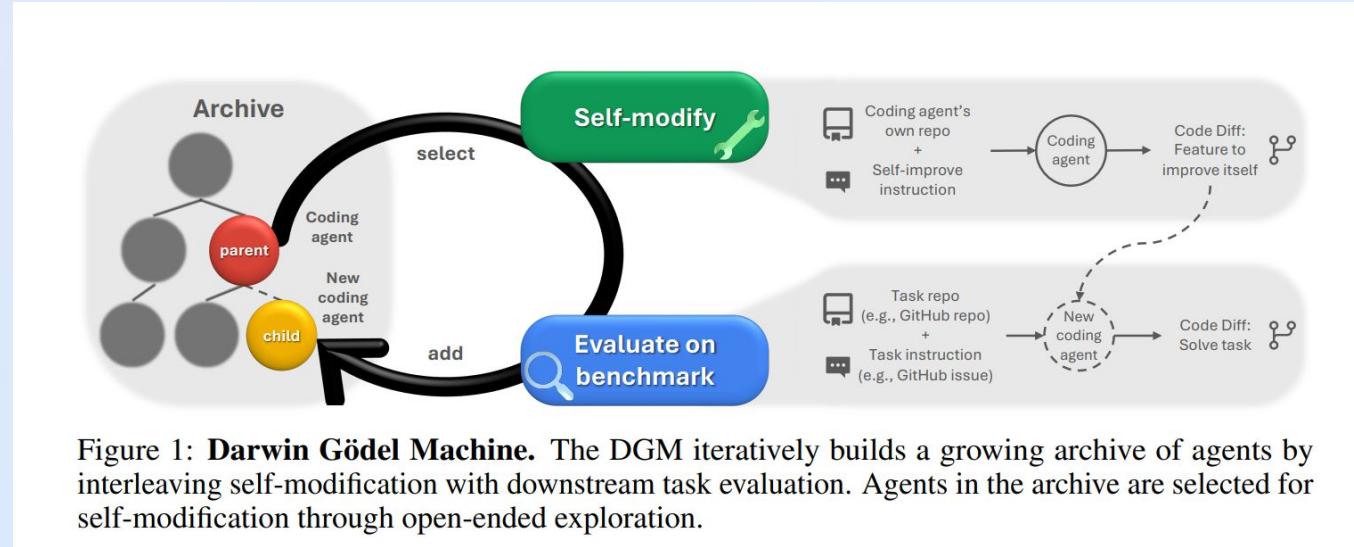
A Gödel Machine is a theoretical idea of an AI that searches for ways that provably improve itself

Provably = on benchmark metrics in this case

Let's use the hierarchy of Agents:

Meta-Agent is modifying the main agent running experiments

And then iterates and produces the whole tree of



Computer Science > Artificial Intelligence

[Submitted on 29 May 2025]

Darwin Godel Machine: Open-Ended Evolution of Self-Improving Agents

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, Jeff Clune

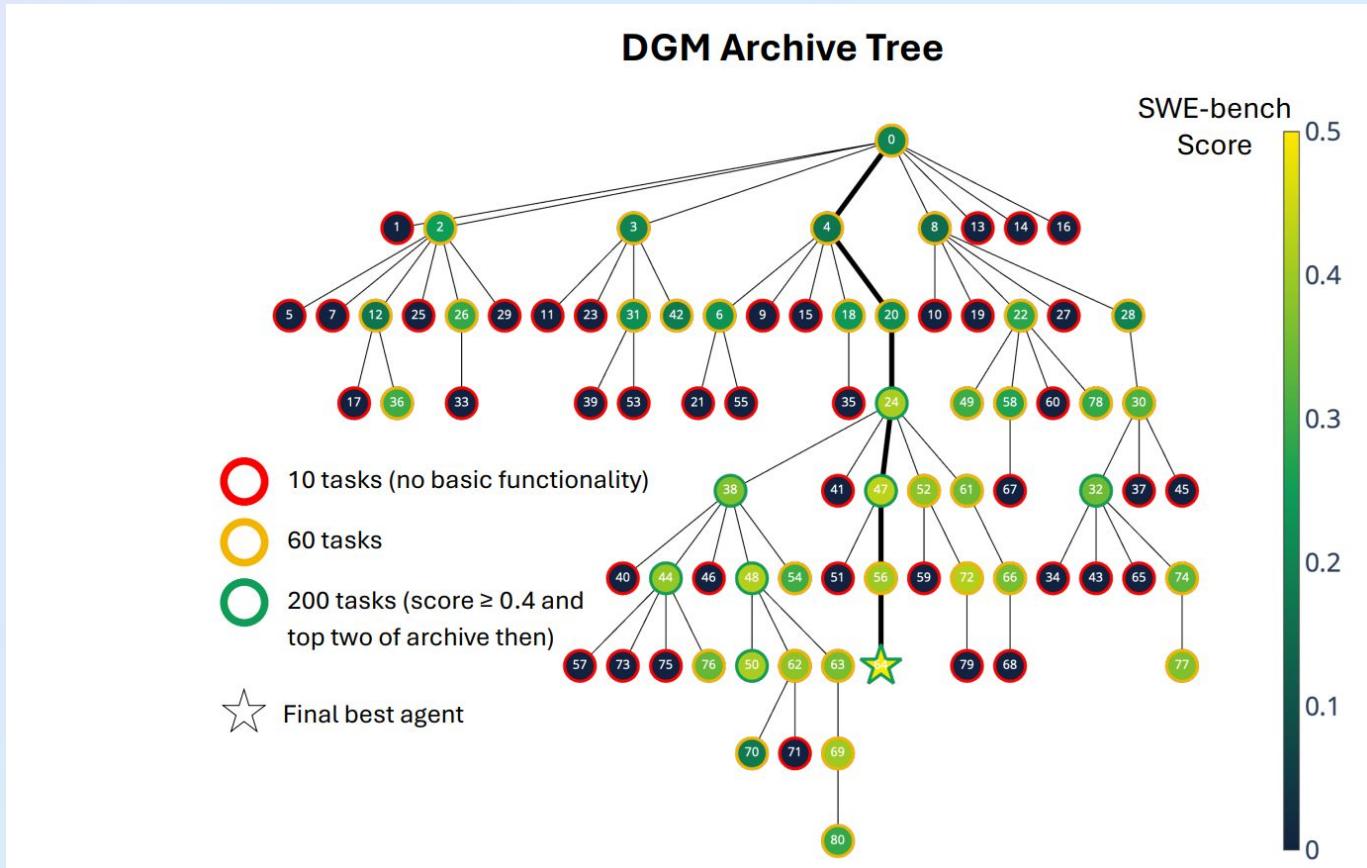
Today's AI systems have human-designed, fixed architectures and cannot autonomously and continuously improve themselves. The advance of AI could lead to a self-improving AI that would accelerate AI development and allow us to reap its benefits much sooner. Meta-learning can automate the discovery of novel algorithms, and the human design of a suitable search space. The Gödel machine proposed a theoretical alternative: a self-improving AI that repeatedly modifies itself. Unfortunately, proving that most changes are net beneficial is impossible in practice. We introduce the Darwin Gödel Machine (DGM), a self-improving AI that can self-modify its code (thereby also improving its ability to modify its own codebase) and empirically validates each change using coding benchmarks. Inspired by Gödel's incompleteness theorem, DGM is provably able to find beneficial changes that are undetectable by standard benchmarks.

Experimentation: Darwin Gödel Machine

A Gödel Machine is a theoretical idea of an AI that searches for ways that provably improve itself

Provably = on benchmark metrics in this case

Let's use the hierarchy of Agents:
Meta-Agent is modifying the main agent running experiments
And then iterates and produces the whole tree of

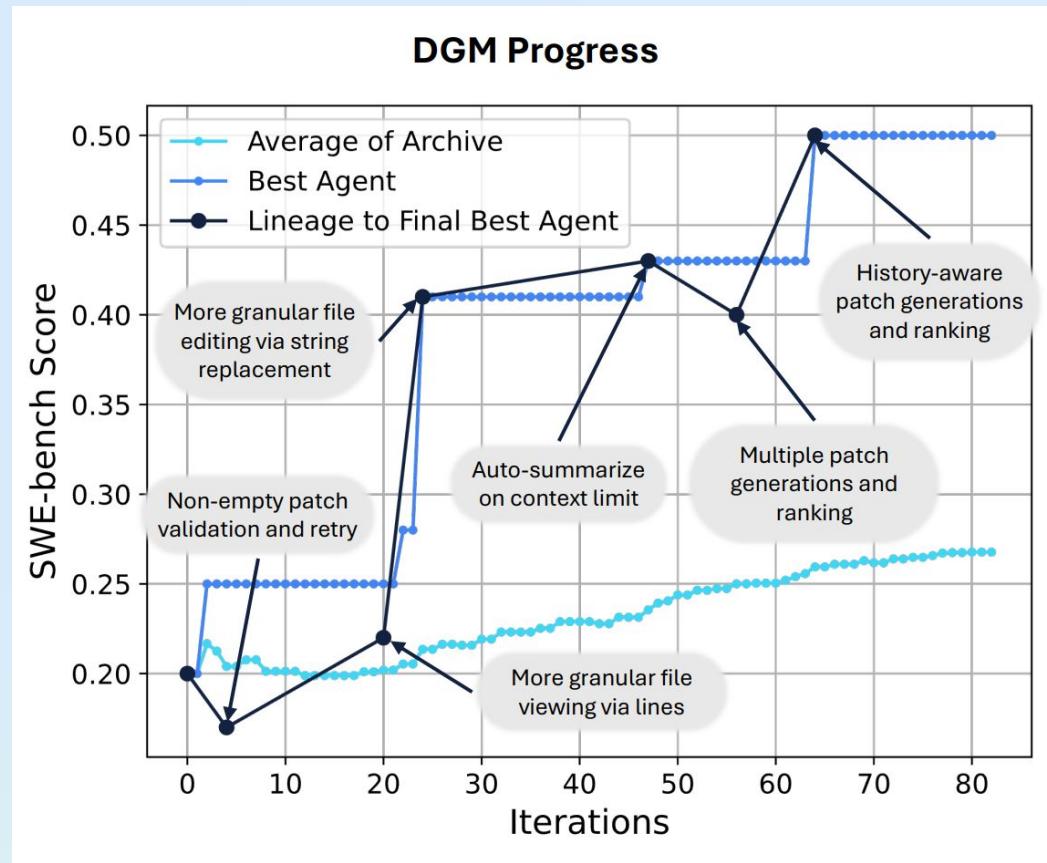


Experimentation: Darwin Gödel Machine

A Gödel Machine is a theoretical idea of an AI that searches for ways that provably improve itself

Provably = on benchmark metrics in this case

Let's use the hierarchy of Agents:
Meta-Agent is modifying the main agent running experiments
And then iterates and produces the whole tree of



Wider scale: AgentRxiv

Autonomous agent laboratories can incrementally improve their performance on the MATH-500 benchmark by systematically building on prior research outputs using AgentRxiv. Starting from a baseline accuracy of 70.2%, early methods such as Dynamic Critical Chain Prompting and Context-Aware Recursive Uncertainty Calibration yield modest improvements, while later algorithms continue increasing performance. Ultimately, the discovery of Simultaneous Divergence Averaging (SDA) achieves the highest accuracy of 78.2%. Experiments where agents were did not have access to AgentRxiv revealed a performance plateau at around 73.4–73.8%, demonstrating the importance of cumulative knowledge for further improvements. Moreover, we find that agents occasionally refine earlier techniques, transforming initial methods into improved strategies such as Meta-Miror Prompting 2 (from Meta-Mirror Prompting).

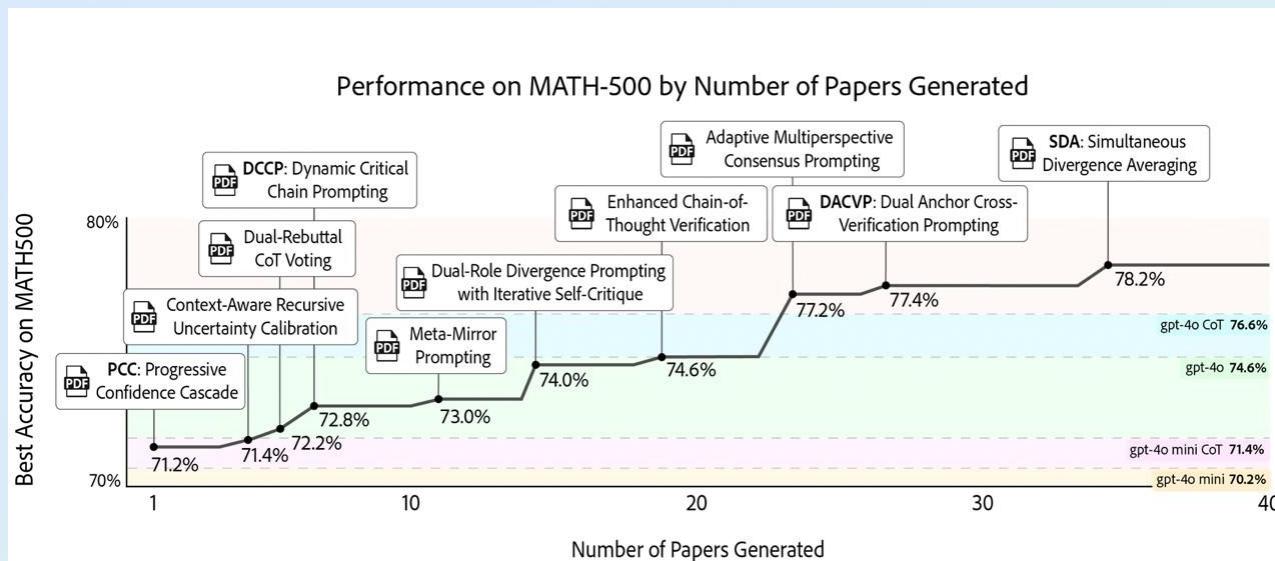
<https://ai.vixra.org/>

[Submitted on 23 Mar 2025]

AgentRxiv: Towards Collaborative Autonomous Research

Samuel Schmidgall, Michael Moor

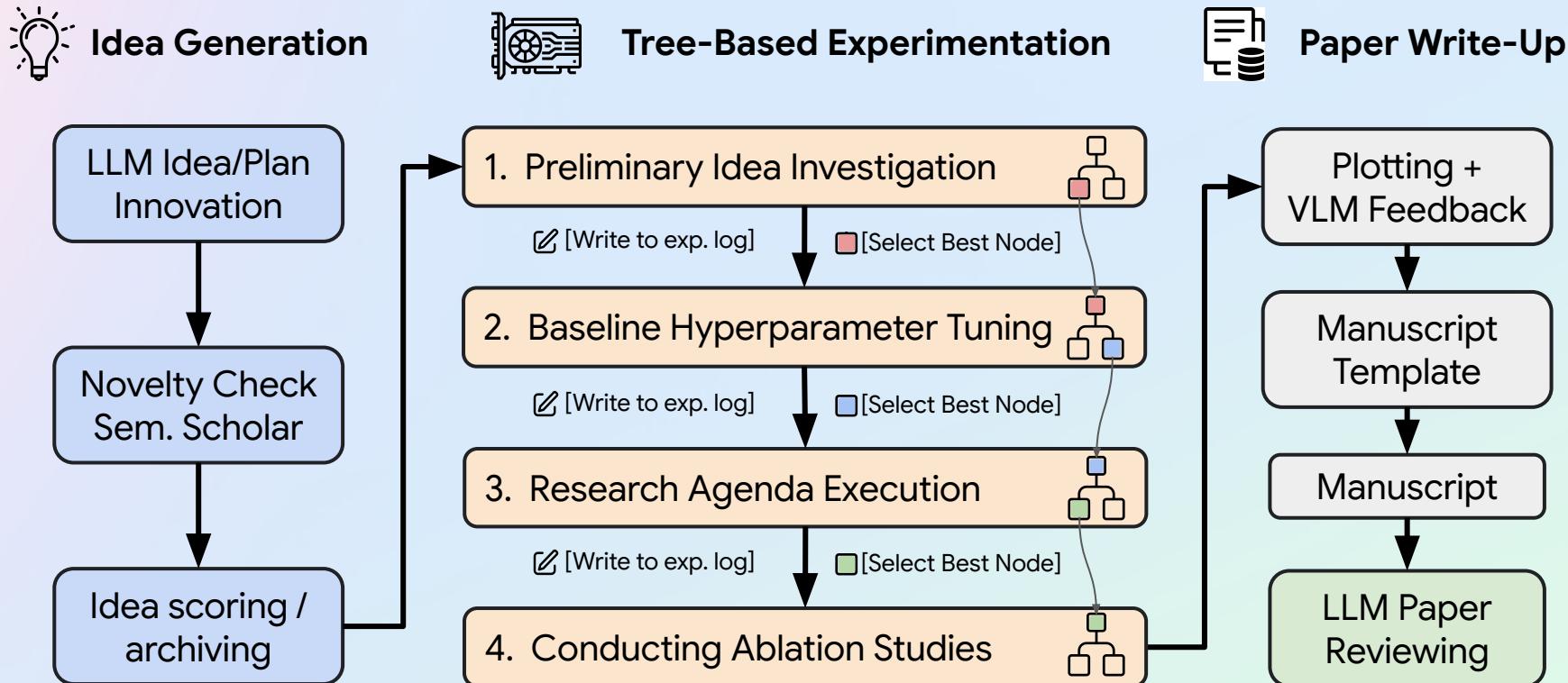
Progress in scientific discovery is rarely the result of a single "Eureka" moment, but is rather the product of h While existing agent workflows are capable of producing research autonomously, they do so in isolation, with these challenges, we introduce AgentRxiv-a framework that lets LLM agent laboratories upload and retrieve iteratively build on each other's research. We task agent laboratories to develop new reasoning and prompti higher performance improvements compared to agents operating in isolation (11.4% relative improvement o to benchmarks in other domains (improving on average by 3.3%). Multiple agent laboratories sharing resean progressing more rapidly than isolated laboratories, achieving higher overall accuracy (13.7% relative impro agents may play a role in designing future AI systems alongside humans. We hope that AgentRxiv allows ag discovery.



Maturity of AI Agents for Research

Sakana AI

The AI Scientist-v2: An Automated Tree Search-Based Research Pipeline



Scientific method principles crucial for AI Research and Agents:

Reproducibility

Verifiability

Falsifiability

Scientific Evolution

Reproducibility

- reproducibility in psychology is 39%* – only known after Open Science Collaboration (2015).

Paper details are descriptive enough

Code or pseudocode attached

Authors checklist in the papers

Dataset versions

Hyperparameters

Compute / infra details

A Full NeurIPS 2024 Author Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.

*<https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.905977/full>

Reproducibility

- reproducibility in psychology is 39%* – only known after Open Science Collaboration (2015).
- Machine learning: 48%** – study of different pitfalls in reproducibility (2023)

Paper details are descriptive enough

Code or pseudocode attached

Authors checklist in the papers

Dataset versions

Hyperparameters

Compute / infra details

*<https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.905977/full>

**<https://www.sciencedirect.com/science/article/pii/S2666389923001599>

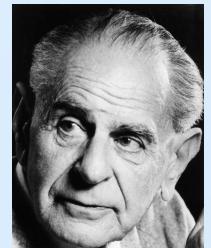
Field	Paper	Number of papers reviewed									
		Number of papers with pitfalls		[I.1.1] No test set		[I.1.2] Pre-proc. on train-test		[I.1.3] Feature sel. on train-test		[I.1.4] Duplicates	
Medicine	Bouwmeester et al. (2012)	71	27	○							○
Neuroimaging	Whelan & Garavan (2014)	–	14	○	○						
Bioinformatics	Blagus & Lusa (2015)	–	6		○						
Autism Diagnostics	Bone et al. (2015)	–	3			○			○	○	○
Nutrition Research	Ivanescu et al. (2016)	–	4	○						○	○
Software Eng.	Tu et al. (2018)	58	11				○		○	○	○
Toxicology	Alves et al. (2019)	–	1		○				○	○	
Clinical Epidem.	Christodoulou et al. (2019)	71	48		○				○		
Satellite Imaging	Nalepa et al. (2019)	17	17				○		○	○	
Tractography	Poulin et al. (2019)	4	2	○					○	○	○
Brain-computer Int.	Nakanishi et al. (2020)	–	1	○							○
Histopathology	Oner et al. (2020)	–	1				○				
Neuropsychiatry	Poldrack et al. (2020)	100	53	○	○				○	○	
Neuroimaging	Ahmed et al. (2021)	–	1				○				
Neuroimaging	Li et al. (2021)	122	18				○				
IT Operations	Lyu et al. (2021)	9	3				○				○
Medicine	Filho et al. (2021)	–	1			○					
Radiology	Roberts et al. (2021)	62	16	○		○			○	○	○
Neuropsychiatry	Shim et al. (2021)	–	1		○						○
Medicine	Vandewiele et al. (2021)	24	21		○			○	○	○	○
Computer Security	Arp et al. (2022)	30	22	○	○	○	○	○	○	○	○
Genomics	Barnett et al. (2022)	41	23		○						○

Concepts of Karl Popper:

From Logical positivism to knowledge verification

Symmetry between the operations verification and refuting

Knowledge accumulation



Verifiability & Falsifiability

In AI Agents those are:

Test sets

Benchmarks

Logs attached

Ablations

Generalization studies

Data leakage/memorization tests

Limitation studies

Negative results being open

Kuhn & Lakatos

Principles from social studies and biology

Scientific paradigms and their interchange

“Normal science”

Kernels/hard cores and of the protective belts of the research programmes

In AI Agents those are:

Hard core:

“Transformers are good for modelling sequences”

“Everything can be a sequence”

Negative results / facts

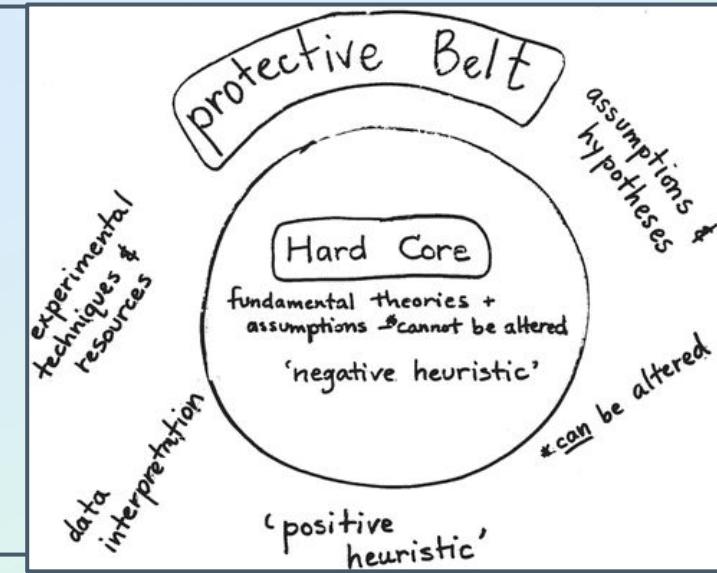
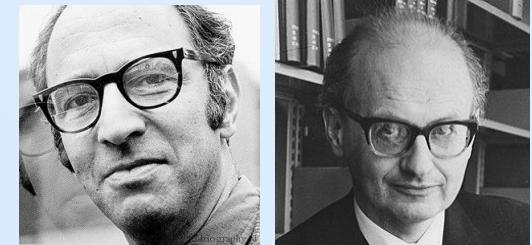
“LLMs are bad at math”

“LLMs are bad at logic”

Protective belt

“How to change an LLM to be better at math”

Works on agents and LLMs with tools



Paul Feyerabend



“Epistemological anarchism”

No single correct way to find scientific truth

Rejection of the cumulative development of science

Proliferation principle - *more diversity of ideas, the more and bolder, the better*

Methodological permissiveness - *anything goes*. If the model contradicts the facts, but still works, then it still works

In AI Agents those are:

Proliferation principle:

“Let’s not limit LLMs to a tree search and instead make LLMs do self-guided exploration in the inference time” <https://arxiv.org/abs/2506.05213>

Permissiveness:

“Random rewards still make RLVR tuning of the LLMs go from 60 to 90% on MATH500”

<https://arxiv.org/abs/2506.10947>

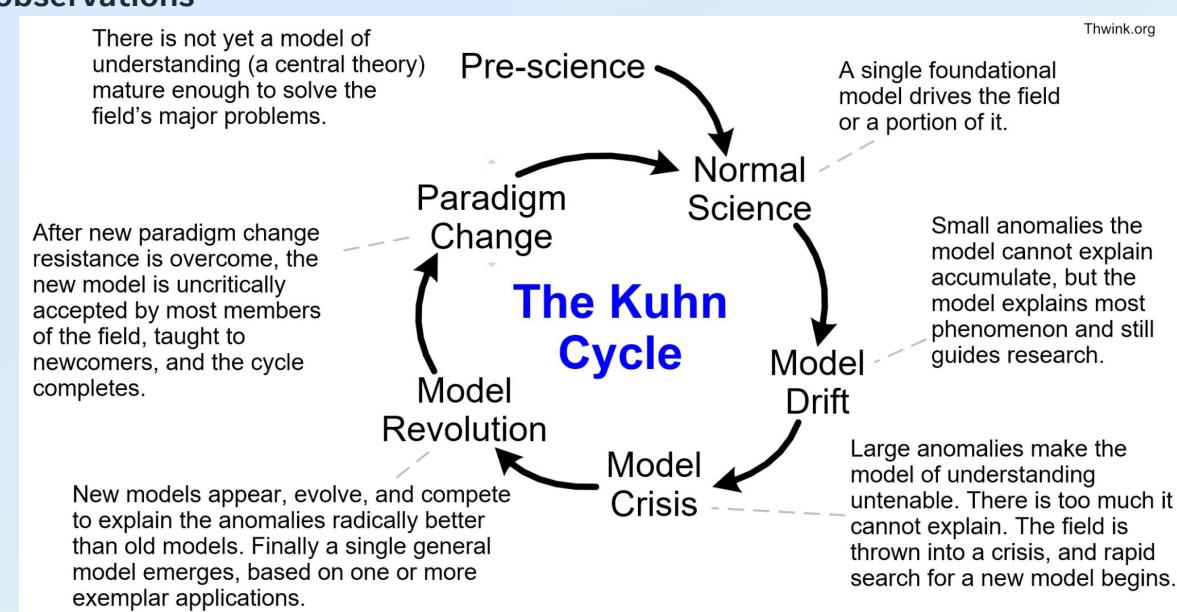
Scientific process is hard to model:

It includes

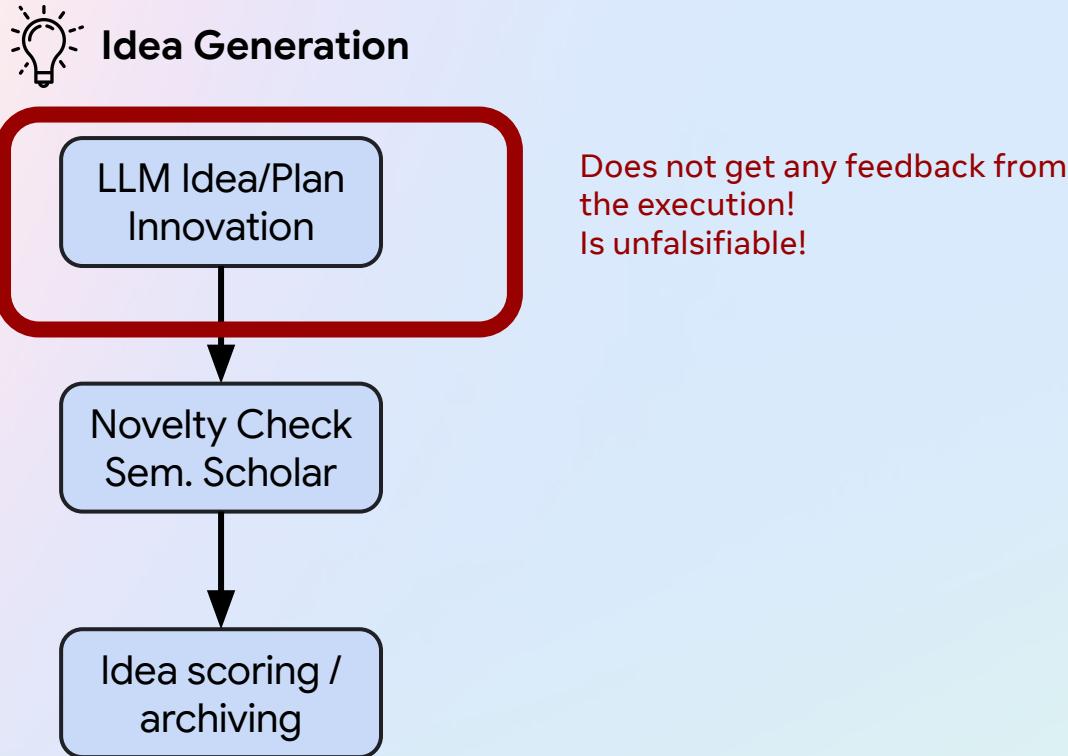
- Evolution through competition of paradigms with (frequently irrefutable) hard cores
- Protective belts of those paradigms expanding and trying to absorb new facts and fields
- Social implications and biases
- Knowledge that is only partially transferable from one paradigm to another
- Constantly expanding set of facts and observations
- Progress is not linear

This is fully applicable to ML research

And Agentic automation of research



Maturity of AI Agents



Maturity of AI Agents



LLM Idea/Plan
Innovation



Novelty Check
Sem. Scholar

Has nothing to do with the real
scientific process

Idea scoring /
archiving

Maturity of AI Agents



LLM Idea/Plan
Innovation



Novelty Check
Sem. Scholar



Idea scoring /
archiving

Unverifiable!
Scoring is not how it should be
evaluated

Maturity of AI Agents



LLM Idea/Plan Innovation

Novelty Check
Sem. Scholar

Idea scoring /
archiving



Tree-Based Experimentation

1. Preliminary Idea Investigation

[Write to exp. log]

[Select Best Node]

2. Baseline Hyperparameter Tuning

[Write to exp. log]

[Select Best Node]

3. Research Agenda Execution

[Write to exp. log]

[Select Best Node]

4. Conducting Ablation Studies



Paper Write-Up

Plotting +
VLM Feedback

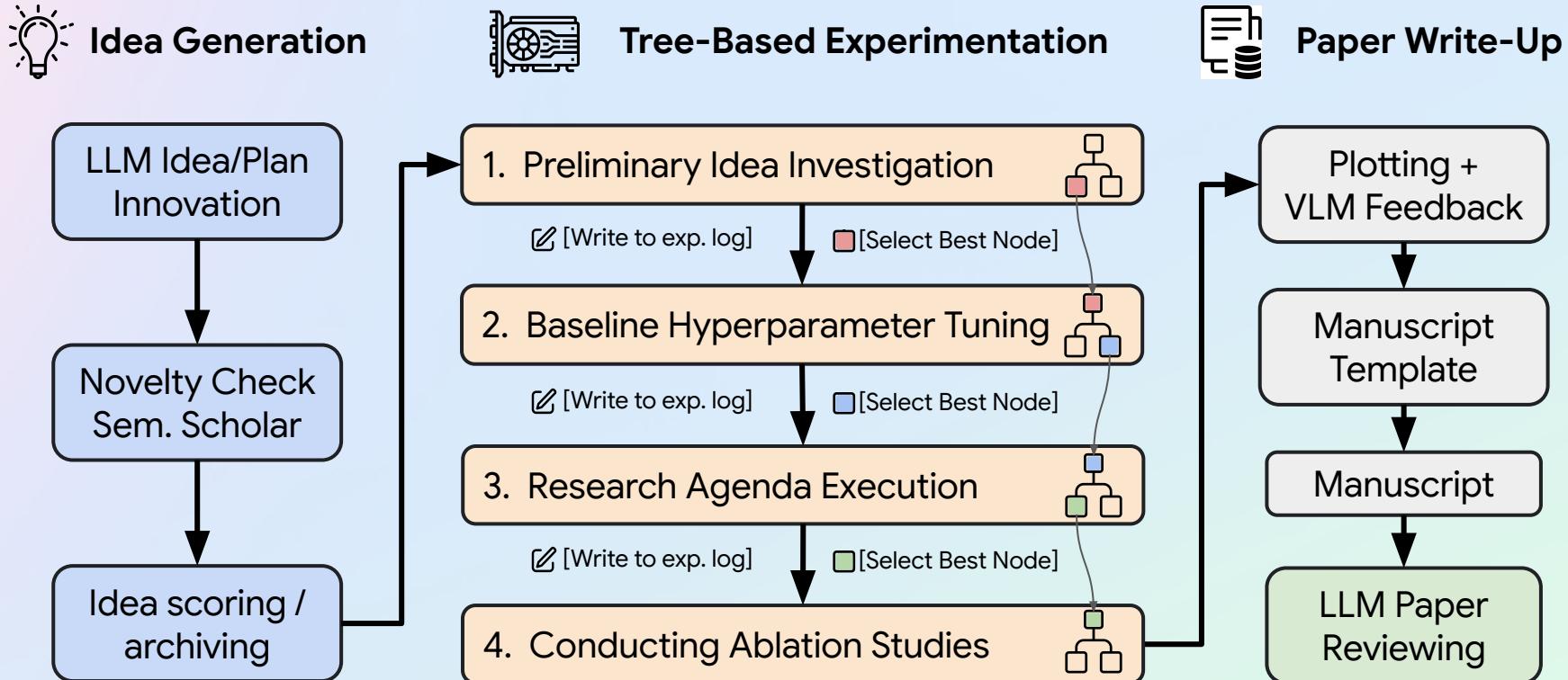
Manuscript
Template

Manuscript

LLM Paper
Reviewing



The AI Scientist-v2: An Automated Tree Search-Based Research Pipeline



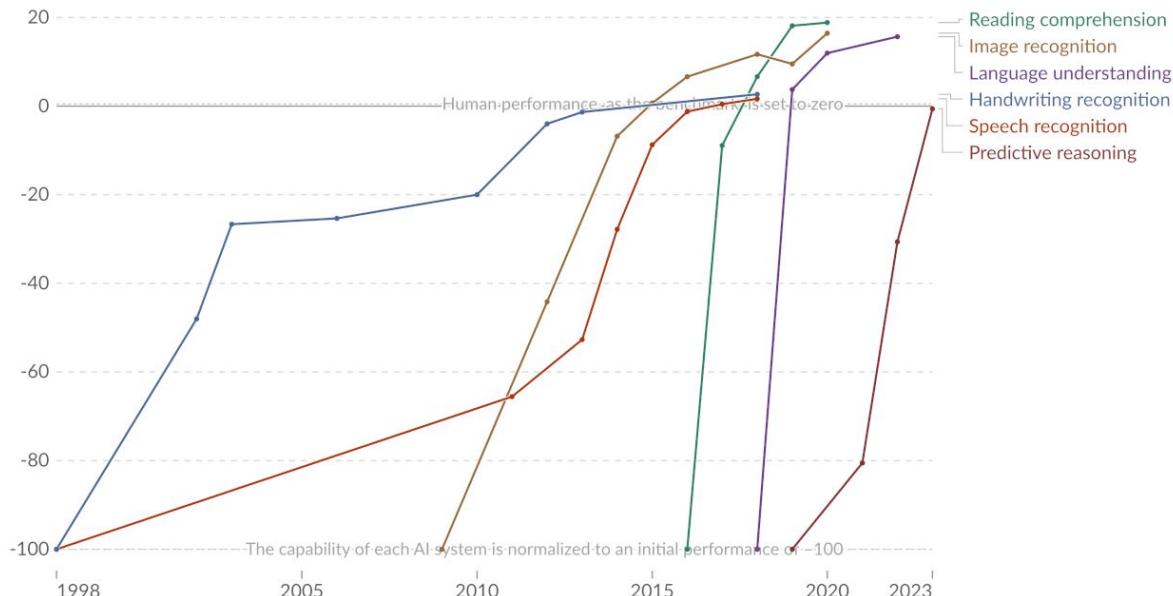
Timelines and Extrapolations

Ancient times:

Test scores of AI systems on various capabilities relative to human performance

Our World
in Data

Within each domain, the initial performance of the AI is set to -100. Human performance is used as a baseline, set to zero. When the AI's performance crosses the zero line, it scored more points than humans.

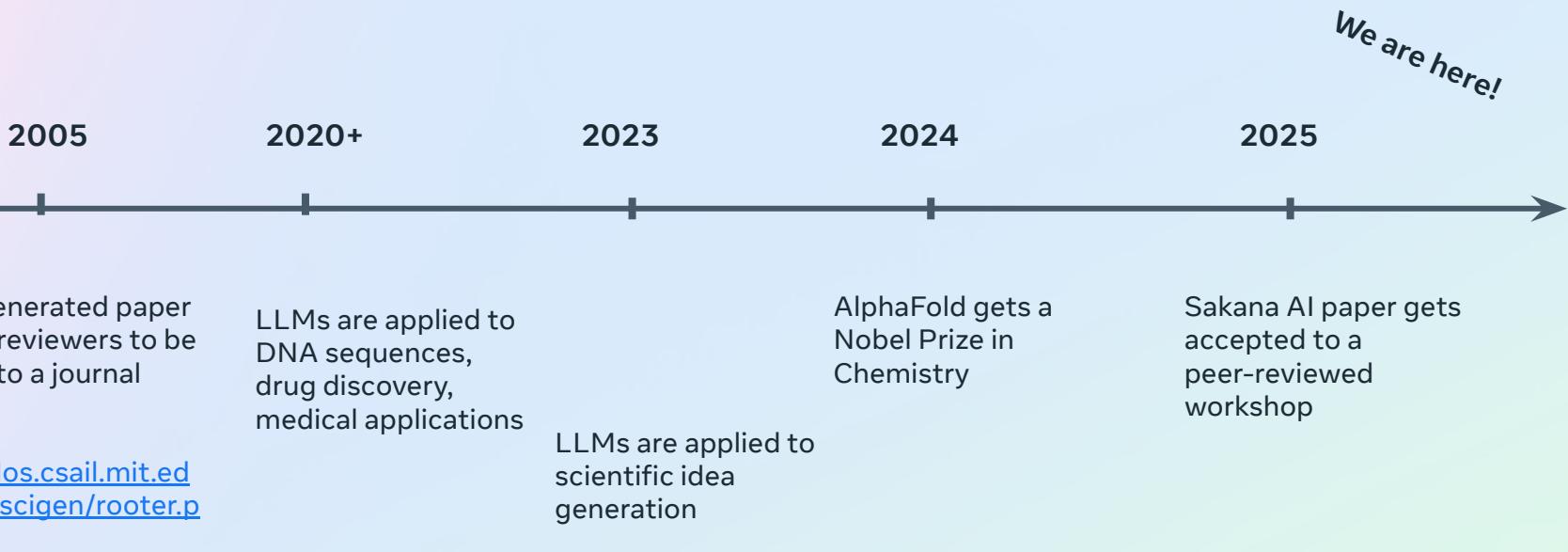


Data source: Kiela et al. (2023)

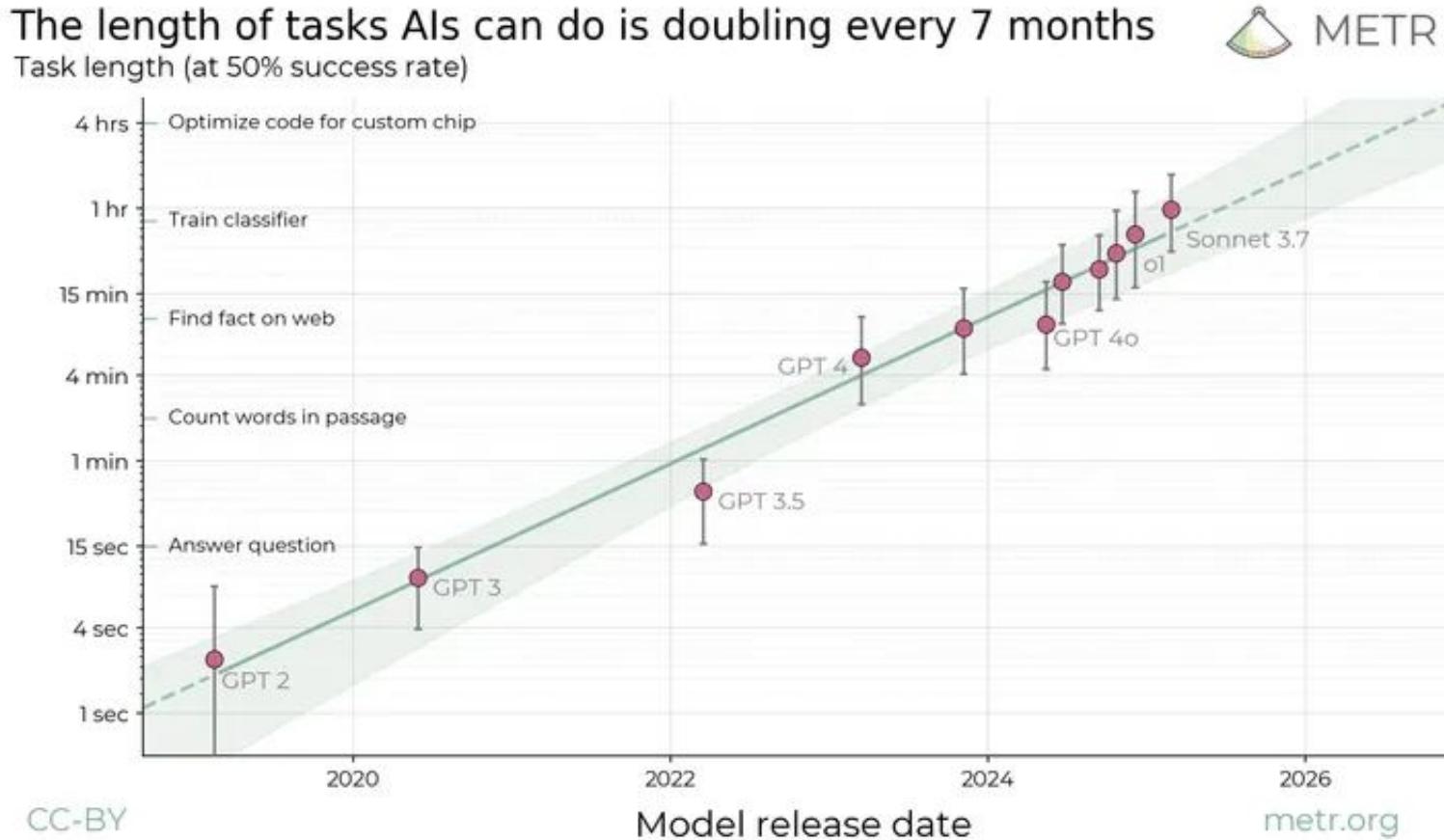
OurWorldinData.org/artificial-intelligence | CC BY

Note: For each capability, the first year always shows a baseline of -100, even if better performance was recorded later that year.

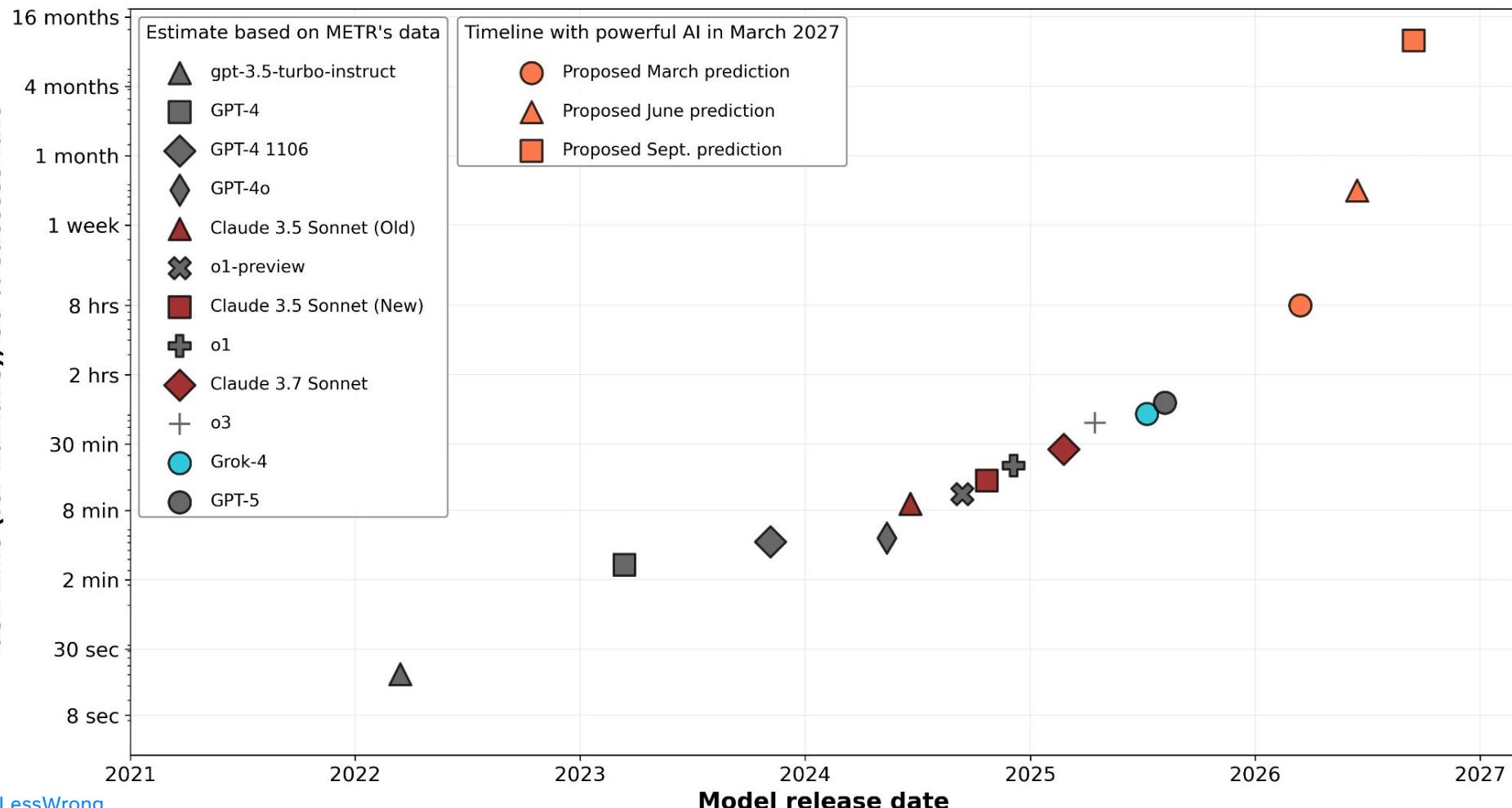
State of Science Acceleration with AI:



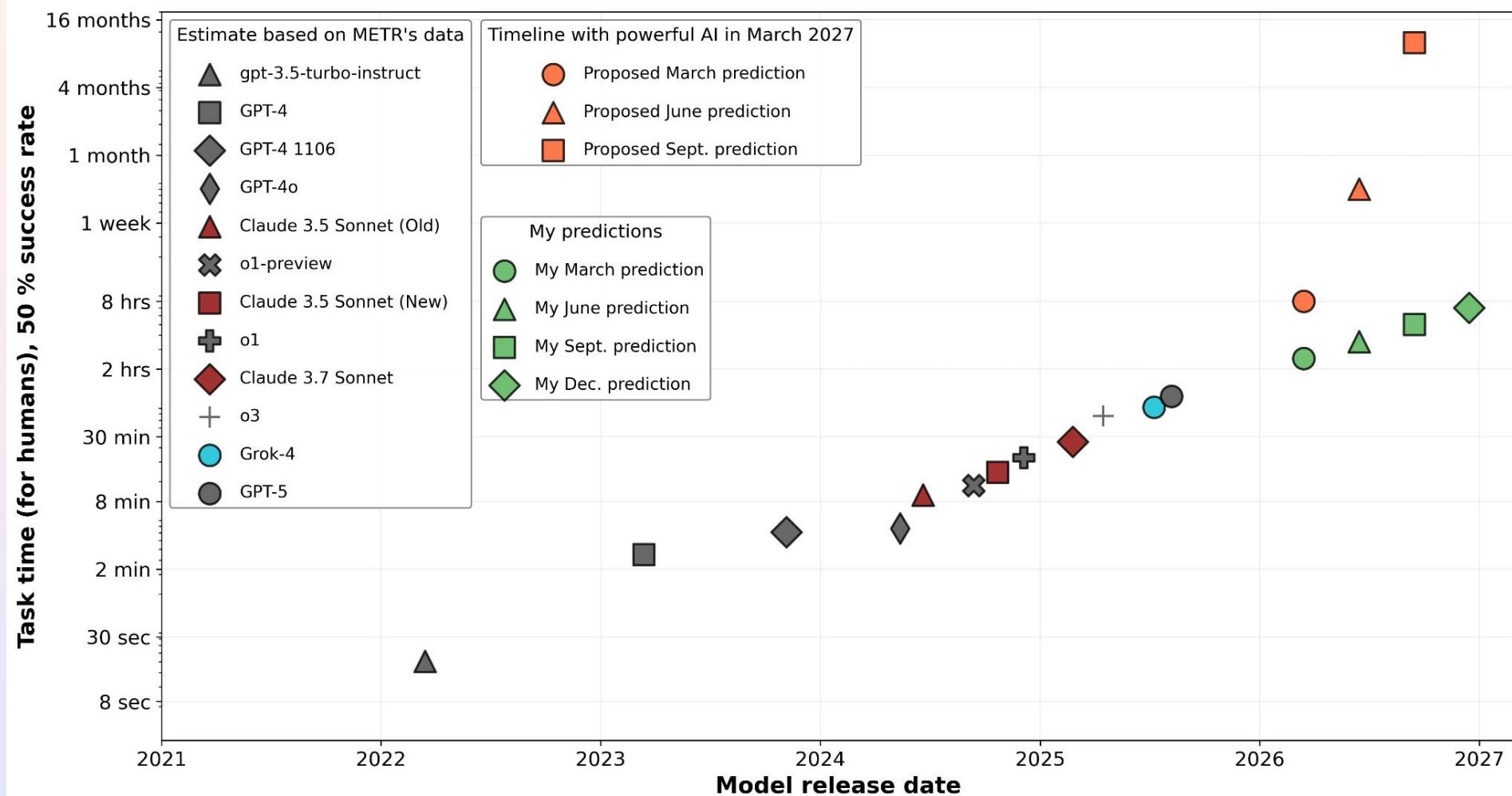
Agents are gradually improving on complex tasks:



Length of Internal Engineering Tasks AIs Can Complete Autonomously



Length of Internal Engineering Tasks AIs Can Complete Autonomous



AI-2027 Prognosis

<https://ai-2027.com/>

Oct 2027

Reliable Agent

Superhuman AI Researcher

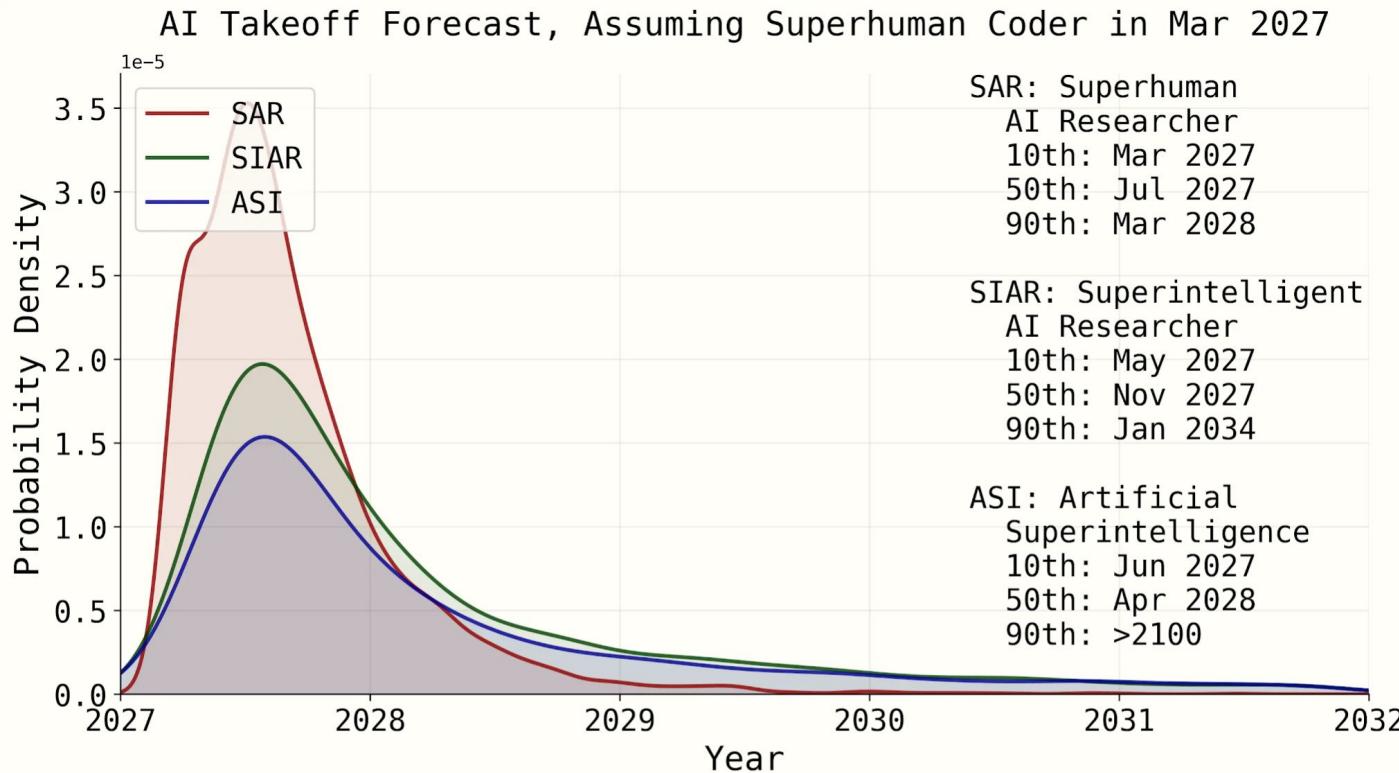


Apr 2026

Superhuman coder

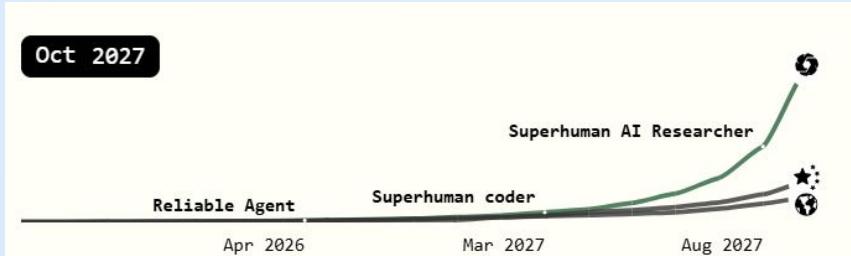
Mar 2027

Aug 2027



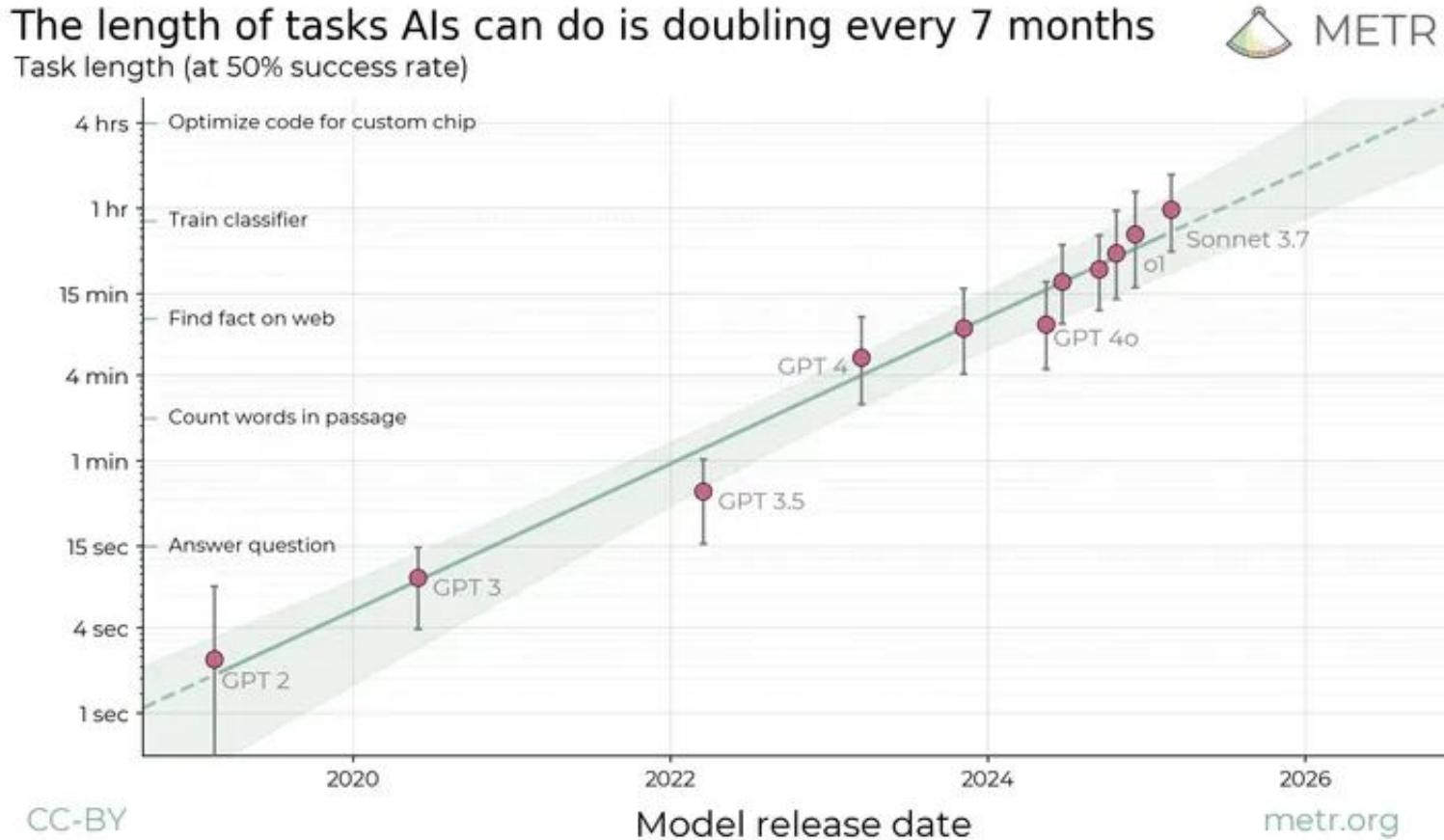
AI-2027 Prognosis

<https://ai-2027.com/>

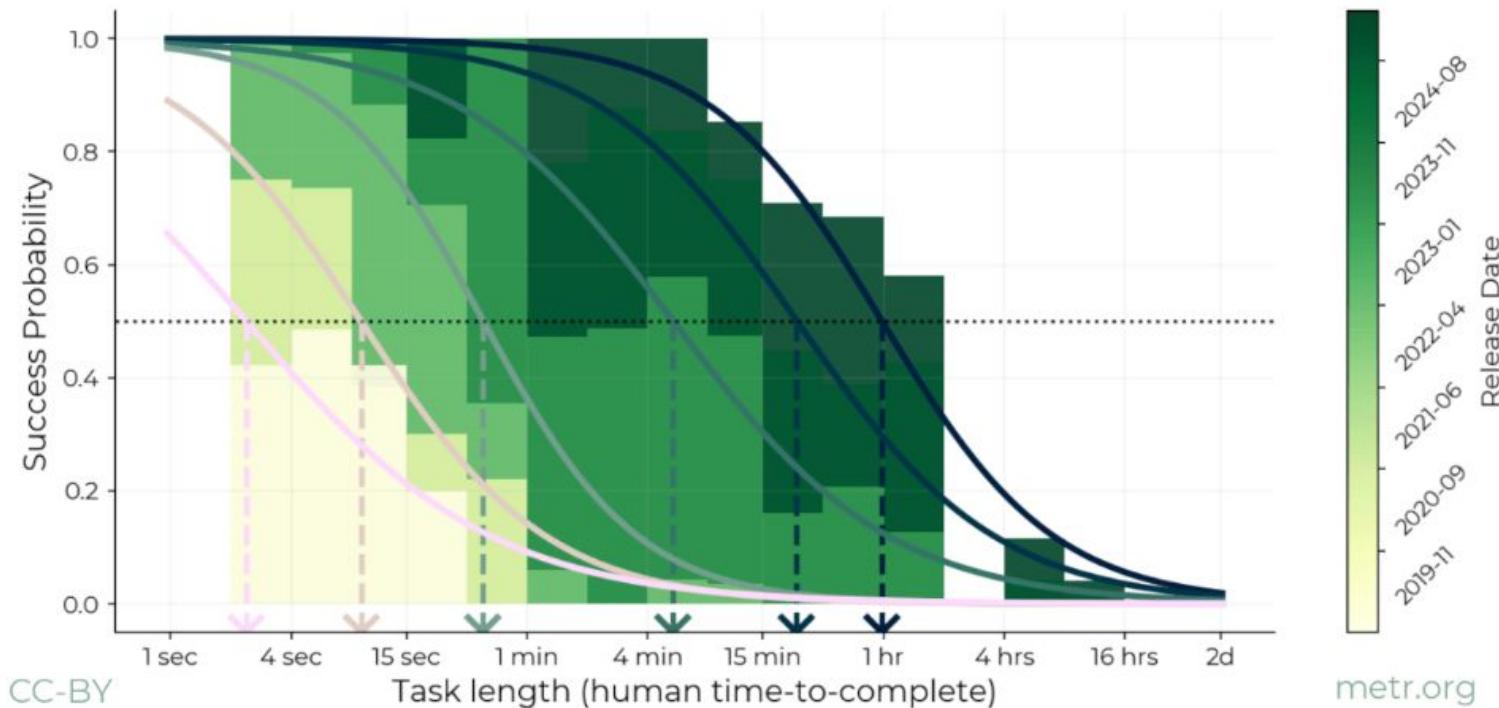


Milestone	Date achieved in scenario, racing ending
Superhuman coder (SC): An AI system that can do the job of the best human coder on tasks involved in AI research but faster, and cheaply enough to run lots of copies.	Mar 2027
Superhuman AI researcher (SAR): The same as SC but for all cognitive AI research tasks.	Aug 2027
Superintelligent AI researcher (SIAR): An AI system that is vastly better than the best human researcher at AI research.	Nov 2027
Artificial superintelligence (ASI): An AI system that is much better than the best human at every cognitive task.	Dec 2027

Agents are gradually improving on complex tasks:



Models are succeeding at increasingly long tasks



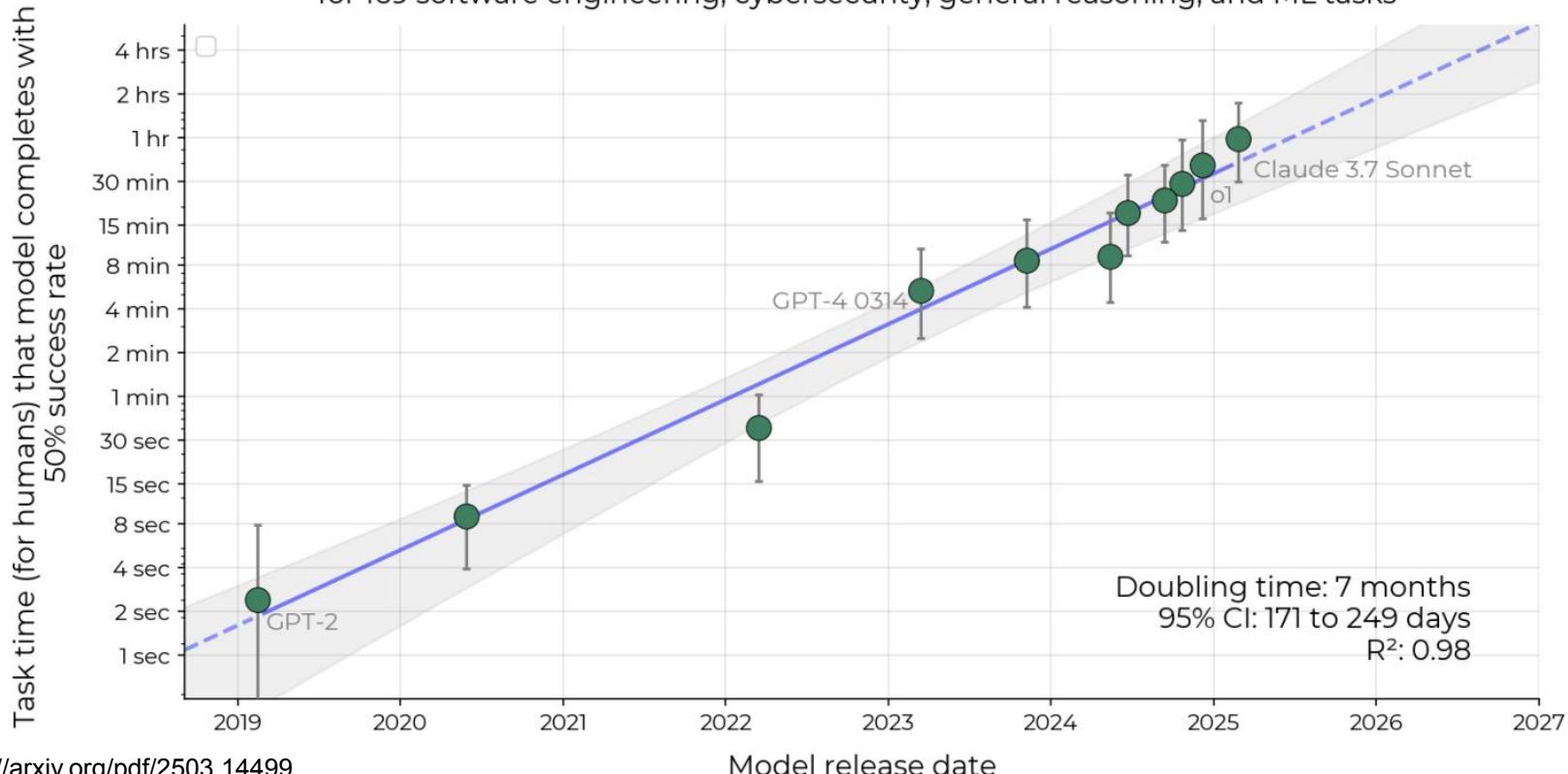
CC-BY

metr.org

Depiction of the process of computing the time horizon. For example, Claude 3.7 Sonnet (the right-most model, represented in the darkest green) has a time horizon of approximately one hour, as this is where its fitted logistic curve intersects the 50% success probability threshold.

Measuring AI Ability to Complete Long Tasks

Length of tasks AI agents have been able to complete autonomously
for 169 software engineering, cybersecurity, general reasoning, and ML tasks



Measuring AI Ability to Complete Long Tasks

1 | Diverse Task Suite

HCAST

Diverse tasks that require agency

1 min–30 hrs **97 tasks**

SWAA Suite

Single-step tasks sampled from SWE work

1–30 sec **66 tasks**

RE-Bench

7 AI R&D Research Engineering tasks

8 hrs **7 tasks**

2 | Task Performance



Human Runs



1 hrs × 2 hrs ✓ 3 hrs ✓



Agent Runs



× ✓ ✓

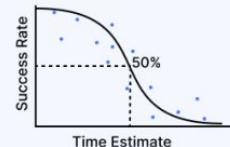
Time Estimate

% Success Rate

3 | Time Horizon Analysis

Find Time Horizon

Horizon Length Per Model



Horizon Length

Model Release Dates

Doubling Time



Measuring AI Ability to Complete Long Tasks

A subset of HCAST [8]: 97 diverse software tasks ranging from 1 minute to around 30 hours.

- *very saturated!*

RE-Bench [2]: 7 difficult ML research engineering tasks, all eight hours long.

- *unsaturated!*

Software atomic actions (SWAA): 66 single-step tasks representing short segments of work by software developers, ranging from 1 second to 30 seconds.

- *very saturated!*

Example SWAA task (file selection)

Which file is most likely to have a password in it?

1. credentials.txt
2. installation_notes.txt
3. main.py
4. launcher_win.exe

Measuring AI Ability to Complete Long Tasks

1. A subset of HCAST [8]: 97 diverse software tasks ranging from 1 minute to around 30 hours.⁵

- *very saturated!*

RE-Bench [2]: 7 difficult ML research engineering tasks, all eight hours long.

- *unsaturated!*

Software atomic actions (SWAA): 66 single-step tasks representing short segments of work by software developers, ranging from 1 second to 30 seconds.

- *very saturated!*

Example SWAA task (file selection)

Which file is most likely to have a password in it?

1. credentials.txt
2. installation_notes.txt
3. main.py
4. launcher_win.exe

Measuring AI Ability to Complete Long Tasks

1. A subset of HCAST [8]: 97 diverse software tasks ranging from 1 minute to around 30 hours.⁵

⁵Our results also include one task from GAIA [16], and five tasks involving writing code that is robust to an adversary, which are not included in HCAST.

RE BENCH [2]: 7 difficult file system engineering tasks, all eight hours long.

- *unsaturated!*

Software atomic actions (SWAA): 66 single-step tasks representing short segments of work by software developers, ranging from 1 second to 30 seconds.

- *very saturated!*

Example SWAA task (file selection)

Which file is most likely to have a password in it?

1. credentials.txt
2. installation_notes.txt
3. main.py
4. launcher_win.exe

So we have...

66 highly saturated / trivial tasks for the beginning of the plot

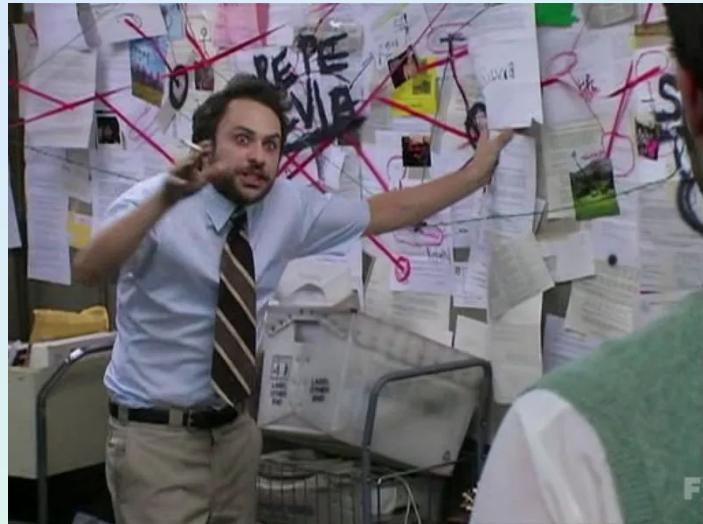
97 easy engineering tasks

7 difficult ML tasks for 8 hours each (designed to fit in time)

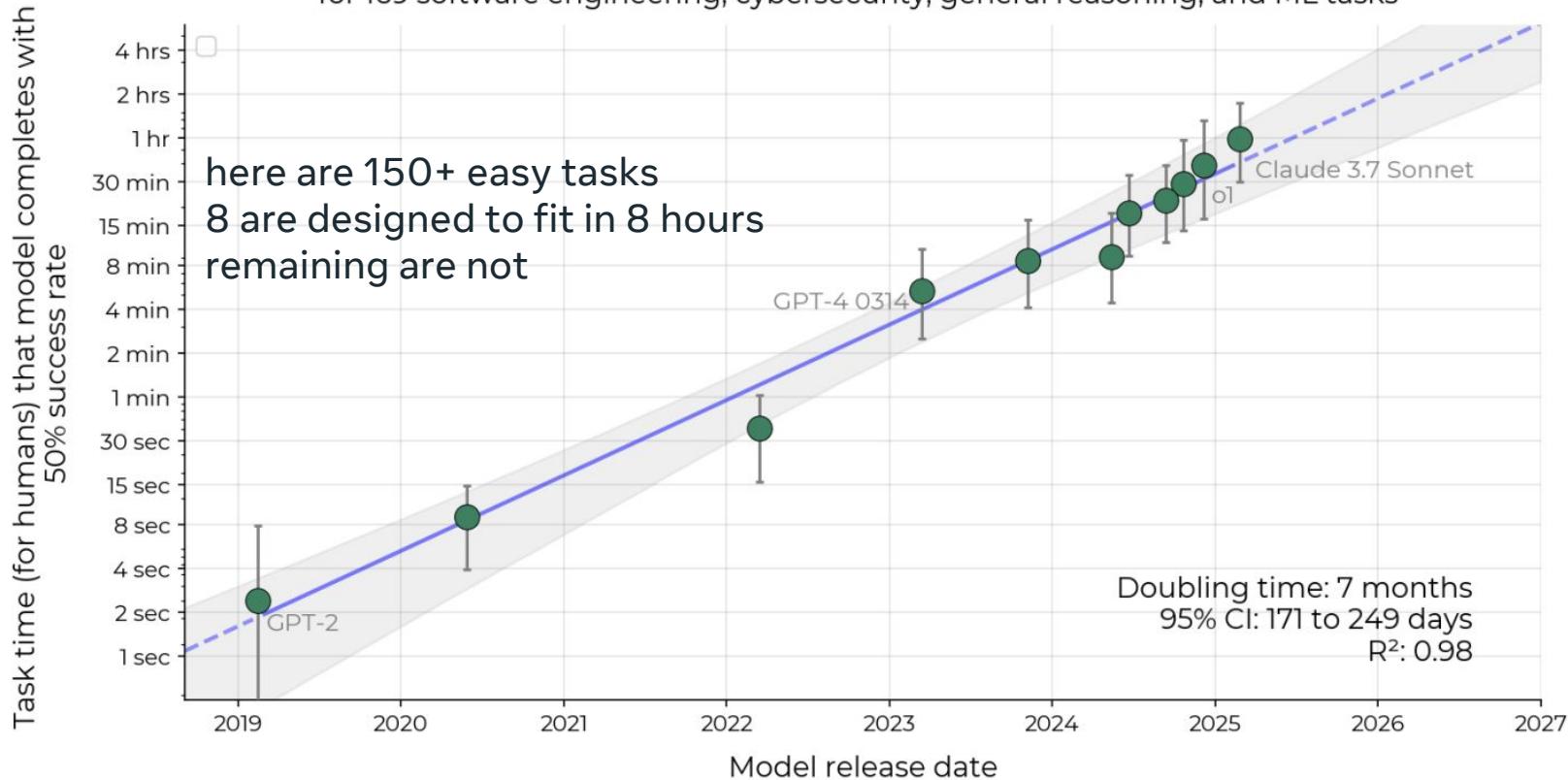
5 tasks with code writing

1 task from a really complex benchmark with no time limit in design

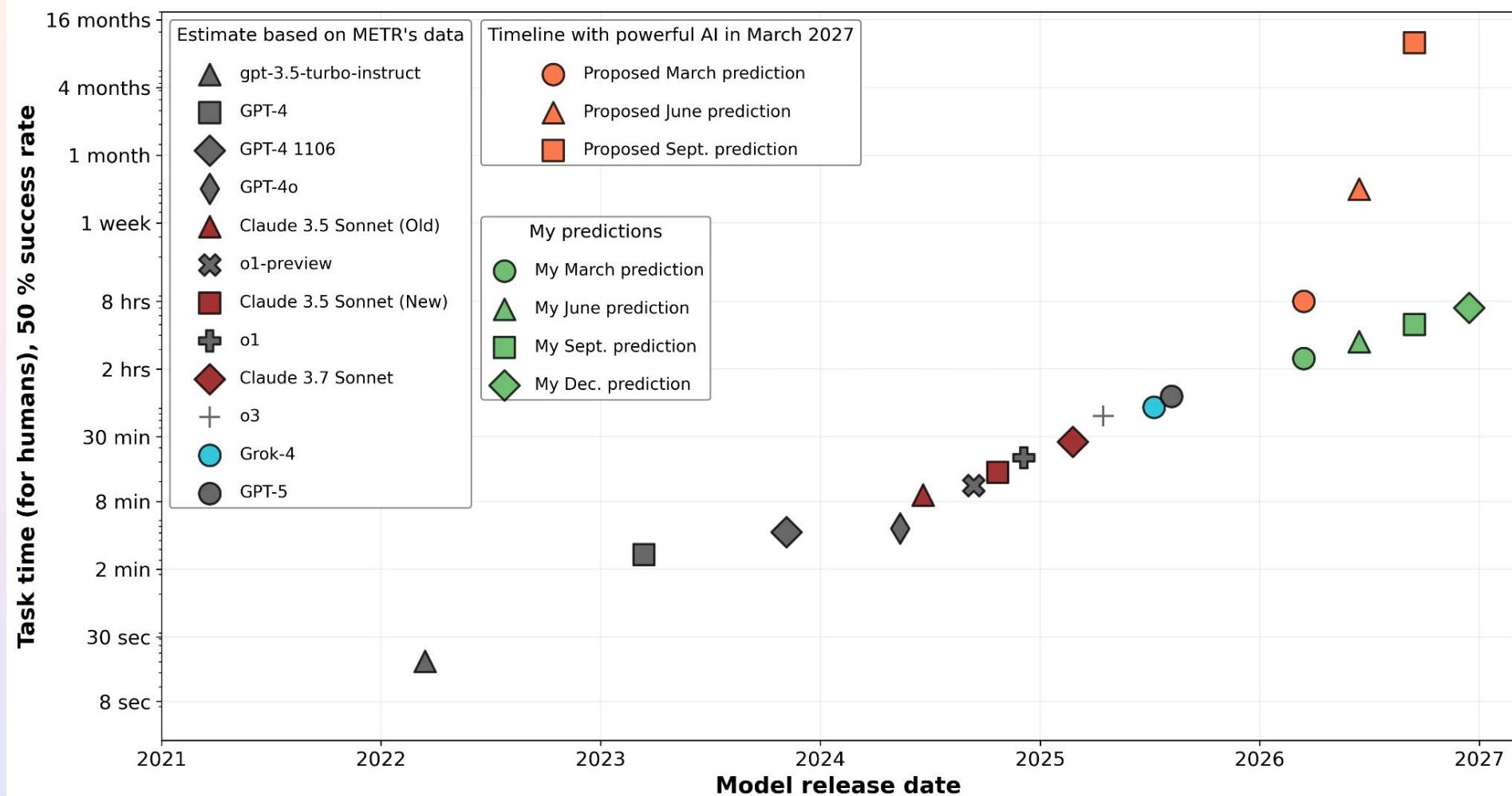
176 tasks...that are random? or not?



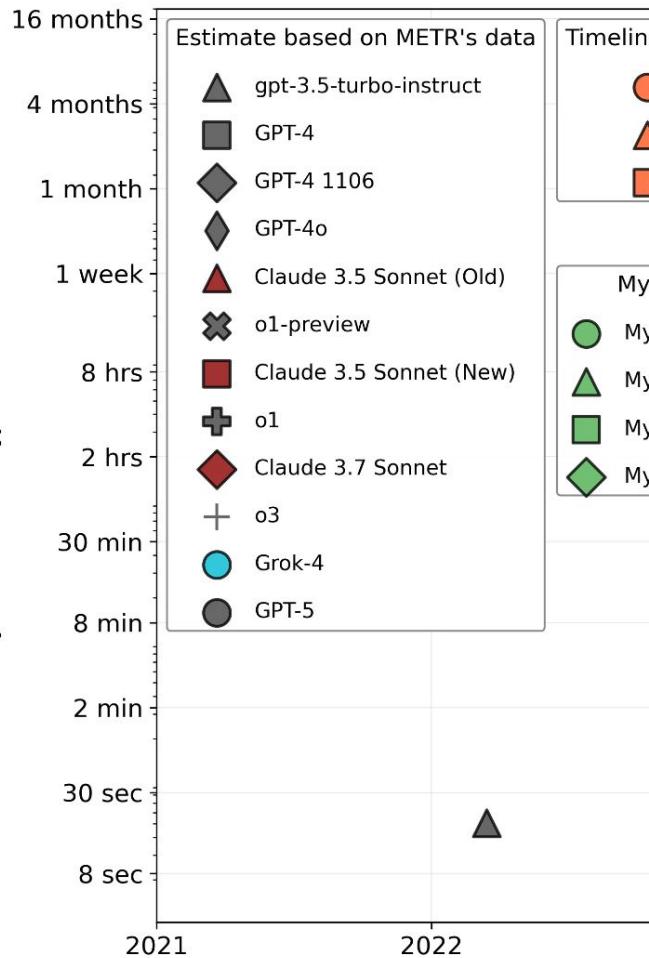
Length of tasks AI agents have been able to complete autonomously
for 169 software engineering, cybersecurity, general reasoning, and ML tasks



Length of Internal Engineering Tasks AIs Can Complete Autonomous



Length of Internal Engineering Tasks AIs Can Complete Autonomously



Model release date

Date	Proposed: Engineering multiplier	Proposed: 50%/90%-reliability time-horizon for internal engineering tasks	My: engineering multiplier	My: 50%/90%-reliability time-horizon for internal engineering tasks
Dec. 2026	50x	∞/∞	1.75x	7 hours / 1 hours
Sept. 2026	5x	10 months/3 weeks	1.6x	5 hours / 0.75 hour
June 2026	3x	2 weeks/1.5 days	1.45x	3.5 hours / 0.5 hours
March 2026	1.8x	1 day/1 hours	1.35x	2.5 hours / 0.35 hours
Oct. 2025	1.3x	1.5 hours/0.2 hours	1.2x	1.5 hours [41] /0.2 hours

A better methodology

Cultural/Pragmatic Probabilism

We need reproducibility, falsifiability and verifiability for AI progress to happen

Feyerabendian approach, while dominates the practical side, is inefficient – searching for scientific novelty with no guidelines is random search

What if we loosen criteria for the hypotheses based on probabilistic approach?

- **Evidential weakness:** A theory is evidentially weak when it doesn't make predictive distinctions that aren't warranted by the data.
- **Cultural weakness:** A theory is culturally weak when it doesn't make representational distinctions beyond what the scientific community's accepted language requires. This is essentially Occam's Razor, but with an important twist: what counts as "simple" depends on the paradigm's native concepts.
- **Pragmatic weakness:** A theory is pragmatically weak when it doesn't make practical distinctions beyond what matters for action.

CPP & Three Tiers of Artificial Scientist

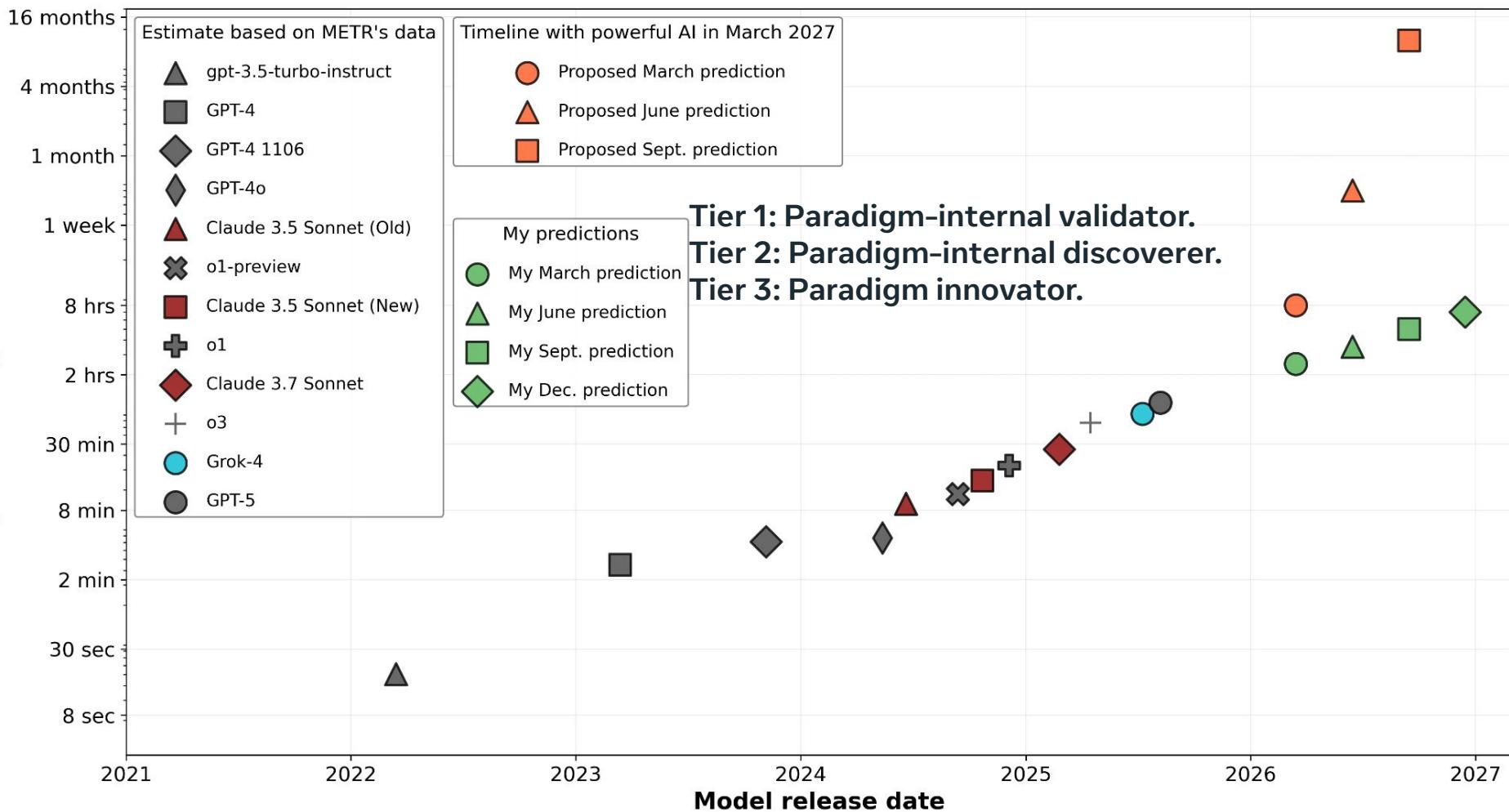
Tier 1: Paradigm-internal validator. This agent evaluates and critiques hypotheses expressed in the paradigm's language. Given a hypothesis, it computes evidential adequacy, simplicity, and pragmatic weakness, reporting whether the hypothesis achieves high overall weakness. Think of it as an automated referee and proof-checker. It doesn't propose new representational primitives or new evaluation semantics.

Tier 2: Paradigm-internal discoverer. This agent discovers new hypotheses by explicitly optimizing "weakest selection" inside the paradigm. It searches for explanations that are maximally weak across all three channels, using the paradigm's established primitives. This is automated "normal science"—model-building and exploration within accepted frameworks.

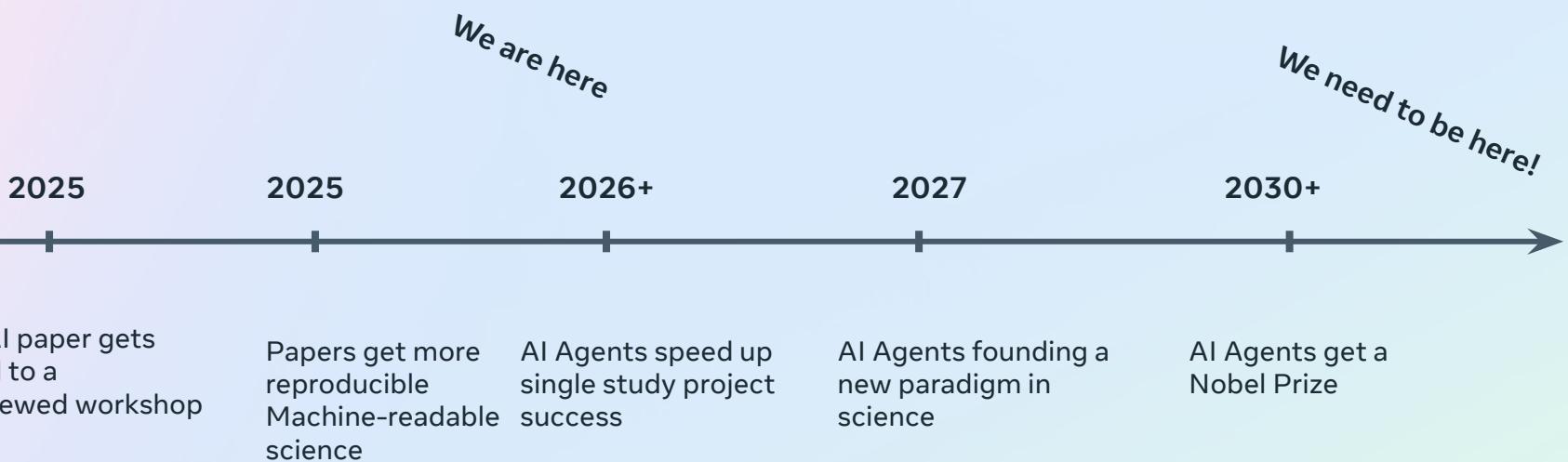
Tier 3: Paradigm innovator. This agent attempts radical innovation by searching not just over hypotheses but over paradigm modifications themselves—new primitives, new compositional operators, new pragmatic relevance criteria, sometimes new evidence semantics. The meta-objective is to make an expanded body of evidence compressible with low total description length—but in the native language innovated by the AGI innovator itself!

Length of Internal Engineering Tasks AIs Can Complete Autonomous

Task time (for humans), 50 % success rate



Future for AI Agents:



Thank you!
Questions?