ANTHROP\C

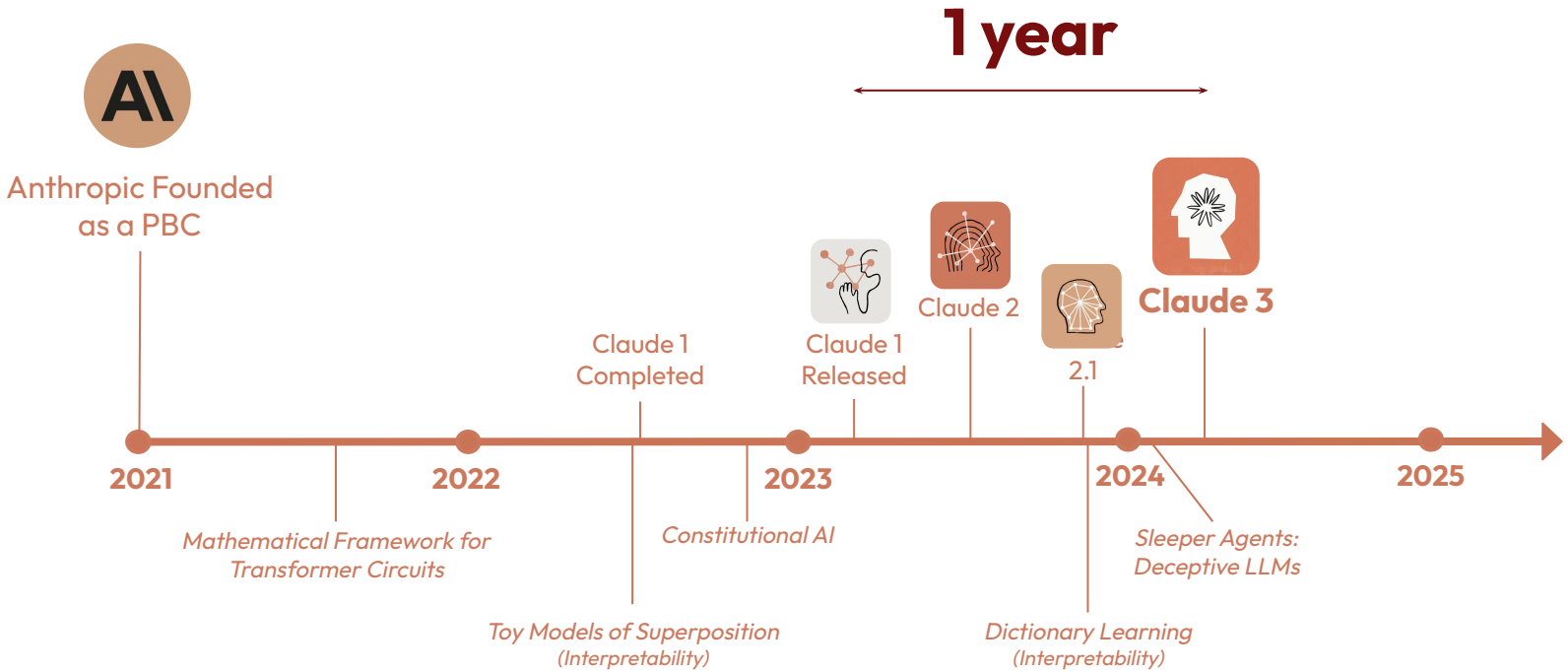
Measuring Agent Capabilities and Anthropic's RSP

Berkeley CS294/194-196 Large Language Model Agents

Ben Mann | 2024.11

Add your questions!

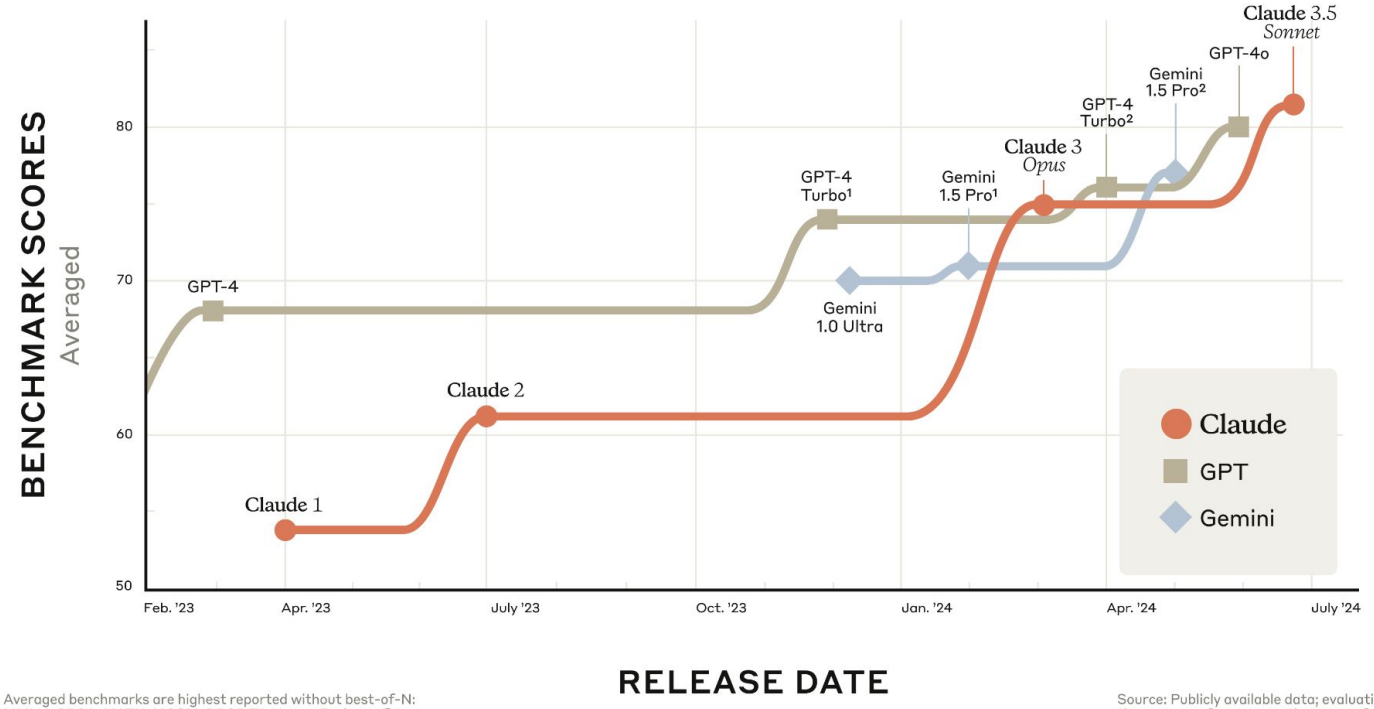
Anthropic's history



Overview

- Why measurement matters
- Our approach to AI development
- Themes: agents, safety, evaluation

AI model release and capabilities timeline

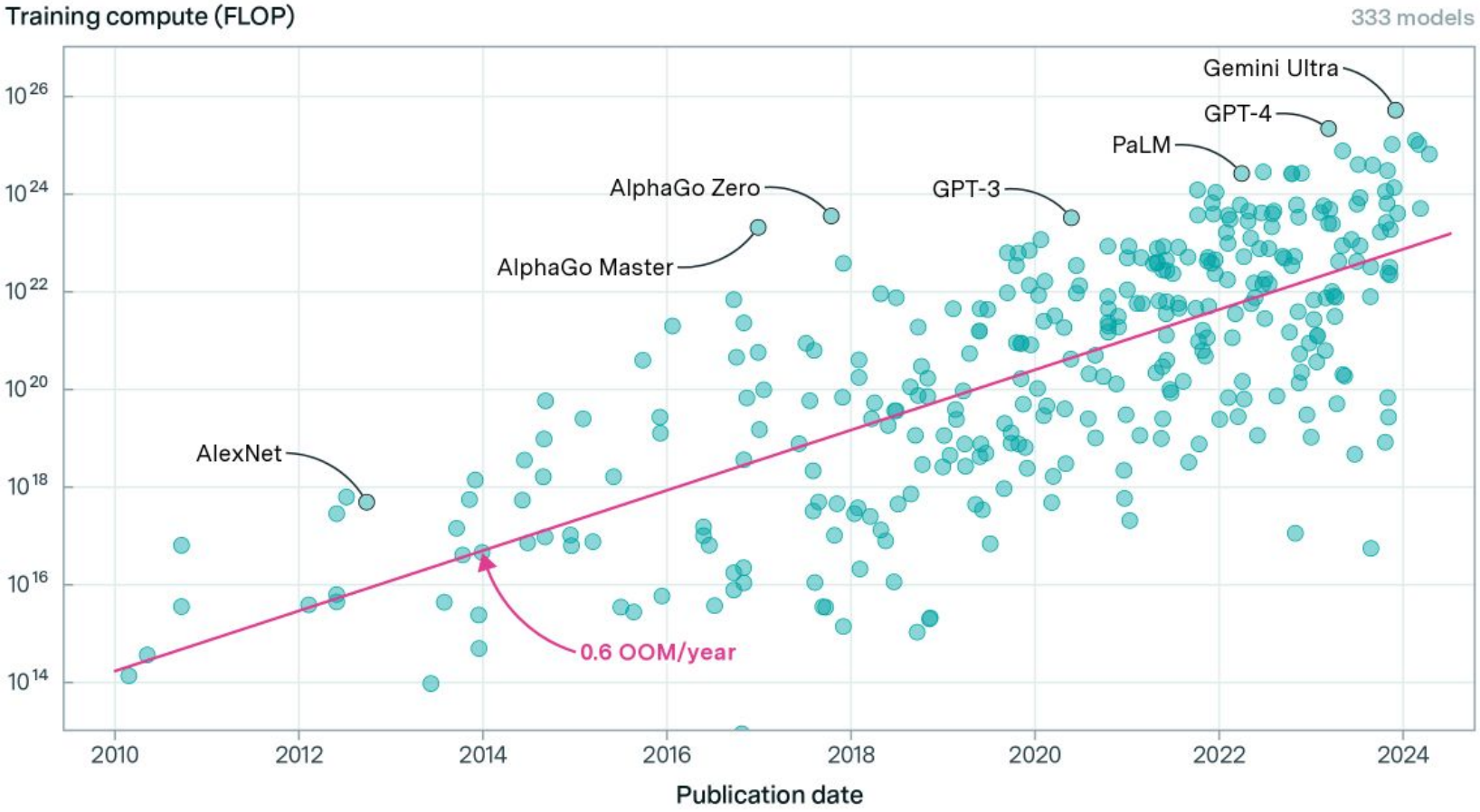


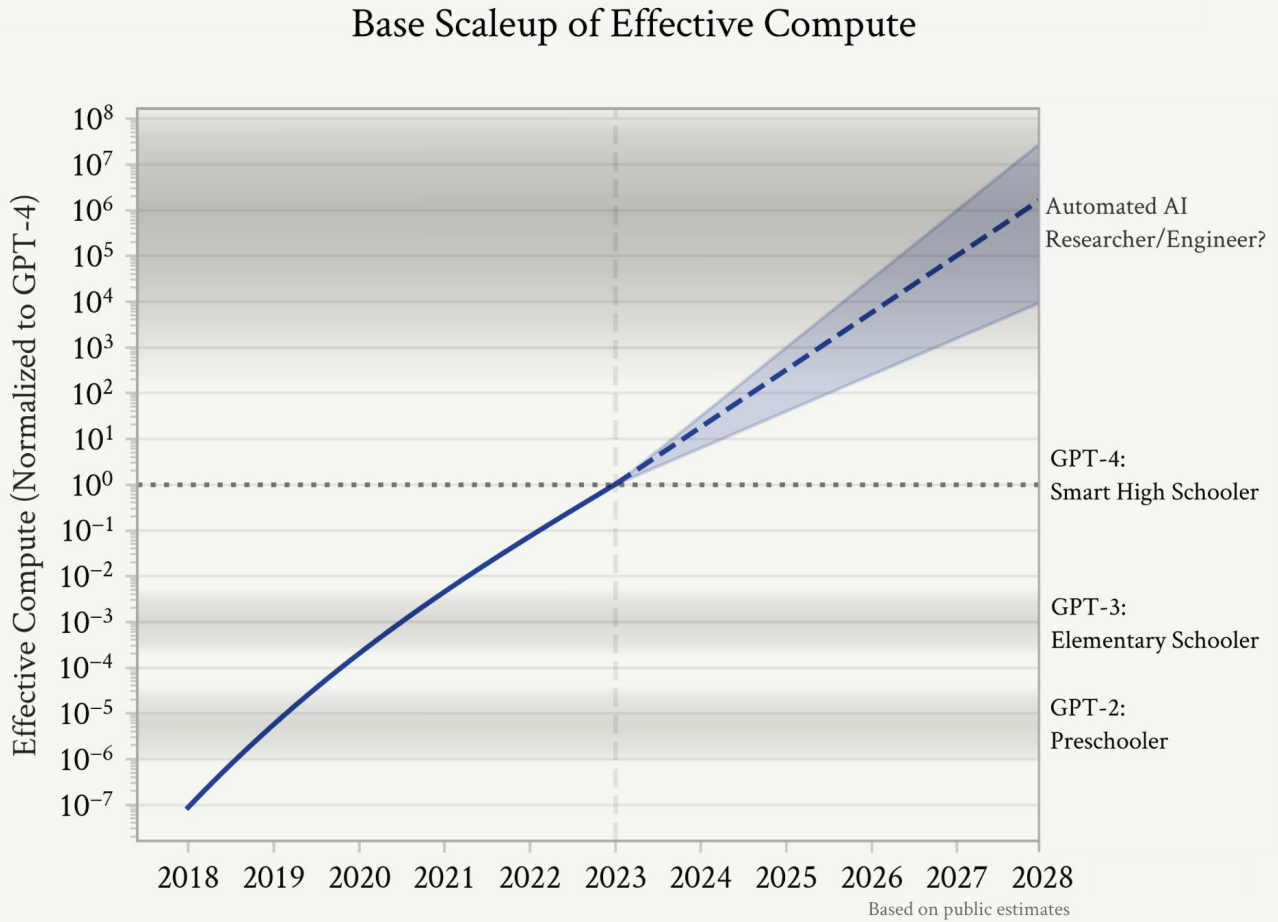
Averaged benchmarks are highest reported without best-of-N: MMLU, GPQA, MATH, MGSM, DROP F1, HumanEval pass@1, MMMU, AI2D, ChartQA, DocQA, Mathvista

Source: Publicly available data; evaluation scores are the average of representative scores found online. 1 = Initial release; 2 = Second release

Training compute of notable models

EPOCH AI





SITUATIONAL AWARENESS | Leopold Aschenbrenner

Claude 3.5 Sonnet / Agentic coding

Claude 3.5 Sonnet / Game development

Responsible Scaling Policy

How is Anthropic developing safe models?

Safety frameworks

- Historical context
- Existing parallels
- Anthropic's RSP
- Updates

What is the RSP?

The **R**esponsible **S**caling **P**olicy represents Anthropic's public commitment to ensuring that model capability does not outstrip our ability to create effective guardrails for that capability and mitigate harm.

Our Responsible Scaling Policy outlines how we will measure for potential catastrophic risks and then mitigate them

Our goals are to:

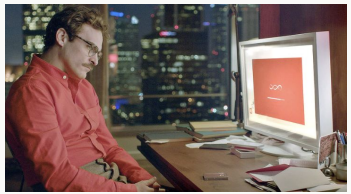
- Provide **structure** to help us make hard decisions about safety
- Hold ourselves publicly **accountable** to developing models safely
- Learn how to make and **iterate** on safe decisions
- Provide a **template** for policymakers and others in industry

AI Safety Levels

High Level Overview of AI Safety Levels (ASLs)

ASL-1	ASL-2	ASL-3	ASL-4
(smaller models)	(present large models)	(significantly higher risk)	(speculative)

Increasing Effective Compute



How do we implement our RSP?

When Anthropic approaches a new level of model capability, the RSP mandates that Anthropic prepare necessary safety measures for it.

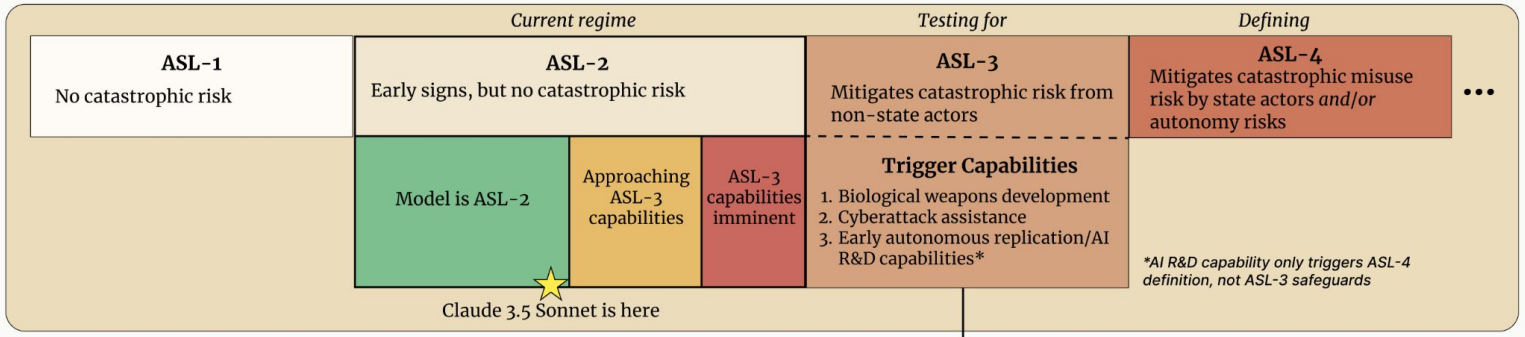
We are preparing for ASL-3.

The RSP highlights AI Safety Levels:			
ASL-1 Smaller Models	ASL-2 Present Large Models	ASL-3 Significantly Higher Risk	ASL-4 Speculative

How do we implement our RSP?

Step 1: Group capability triggers and corresponding safeguards by AI Safety Level (ASL)

Step 2: Evaluate current models for ASL-3 capabilities



Step 3: Trigger ASL-3, implement predefined safeguards, publish ASL-4

ASL-3 Standard

Security Standards
Prevent theft by most non-state actors

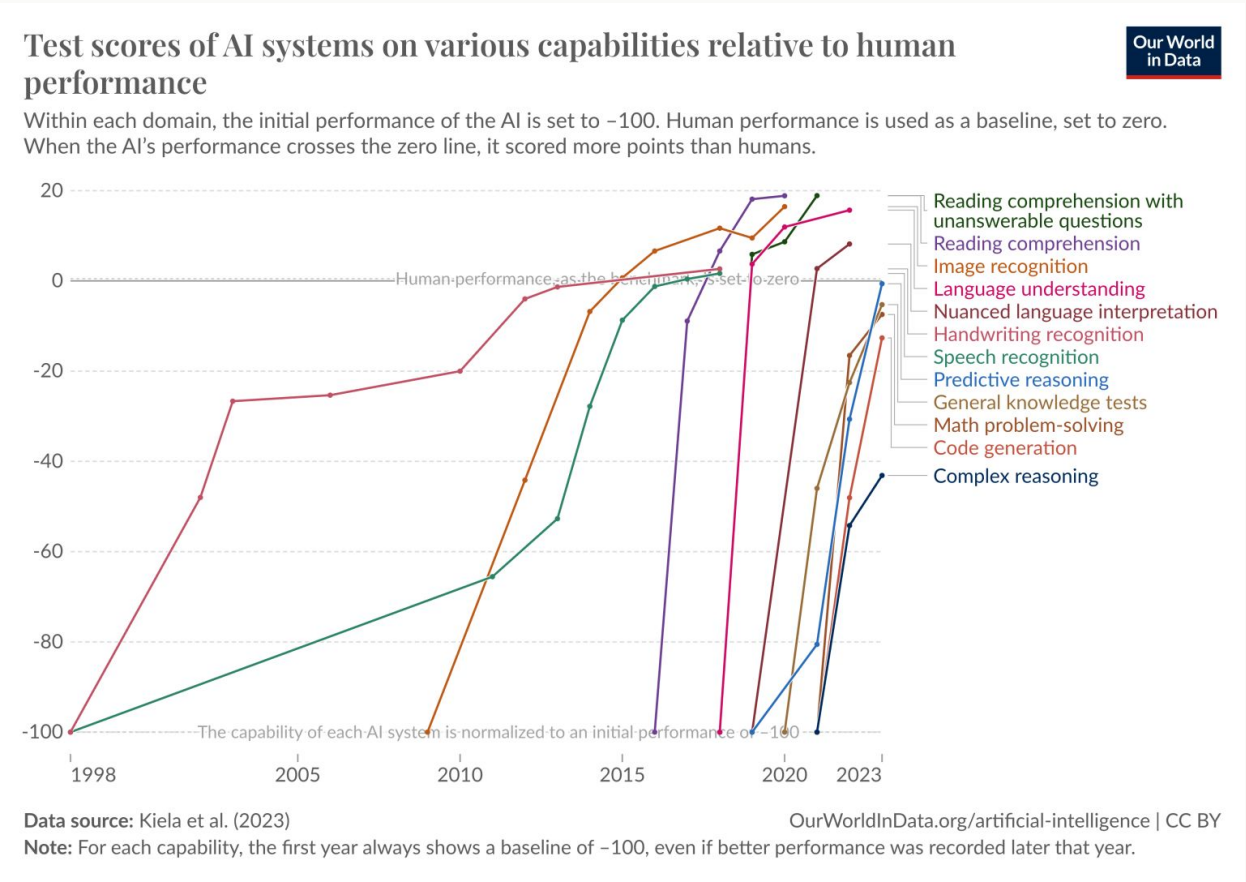
Deployment Standards
Prevent misuse despite hundreds of hours of jailbreaking attempts

Define ASL-4 Triggers
Publish a list of Trigger Capabilities requiring ASL-4 safeguards, and test ASL-3 models for these

Measuring capabilities

- Challenges
- Anthropic's methods
- Case study: computer use

Benchmarks don’t last

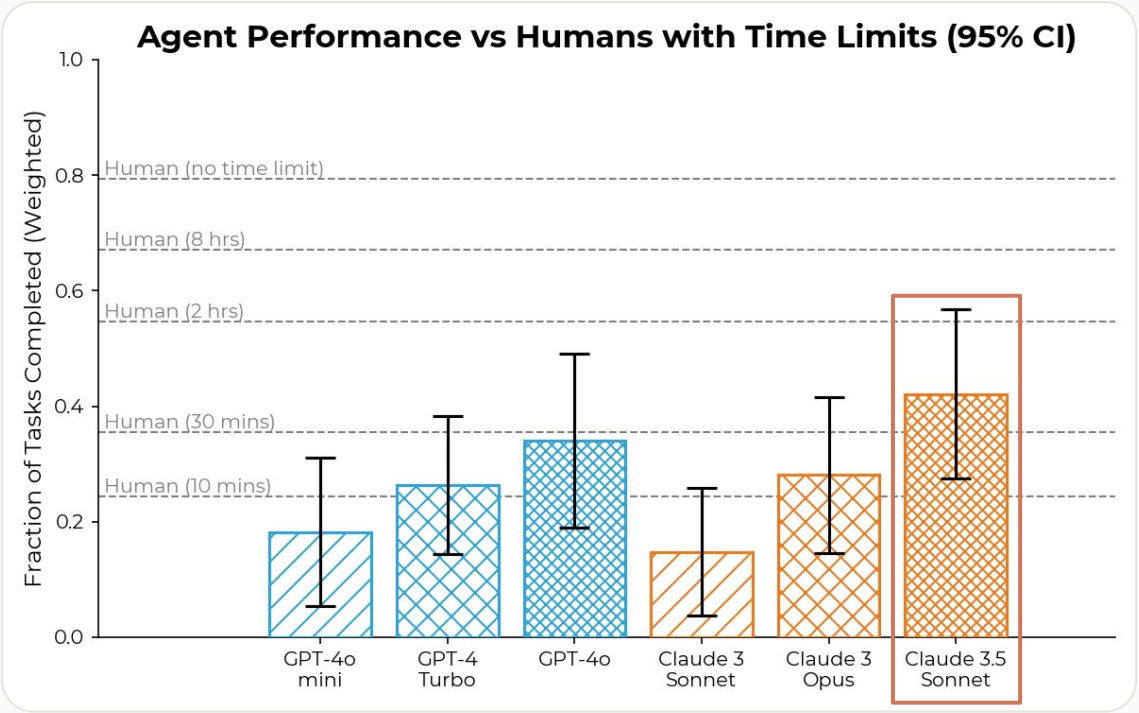


Test scores of AI systems on various capabilities relative to human performance. Human performance is used as a baseline, set to zero.

Kiela et al. (2023)

Measuring task completion time relative to humans

Claude 3.5 Sonnet can handle tasks that would take human developers around 30 minutes, in seconds.



August 2024 Eval from METR (Model Evaluation and Threat Research)

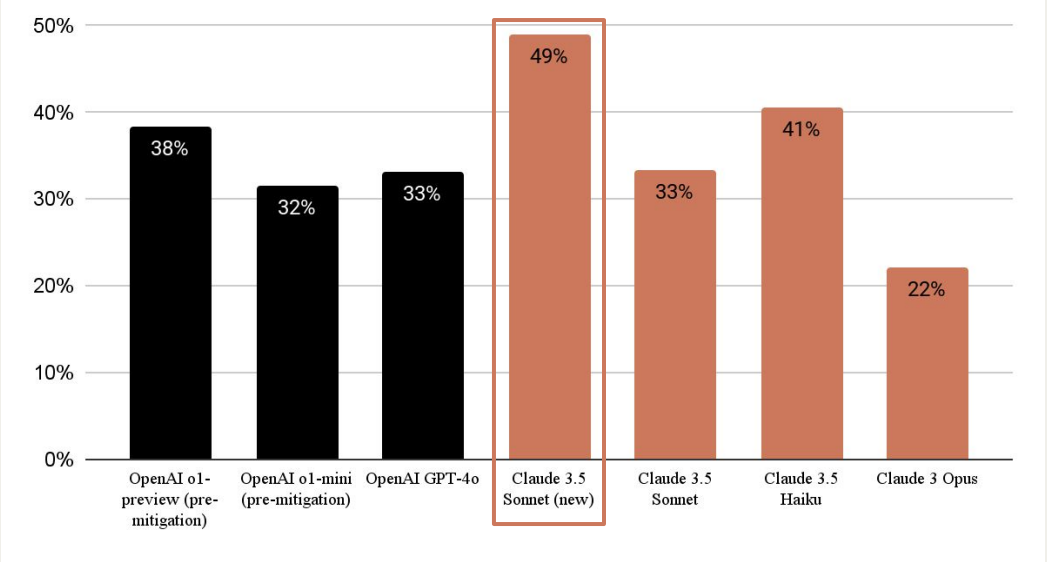
3.5 Sonnet outperforms OpenAI o1

Despite being significantly faster and cheaper than o1*, Claude 3.5 Sonnet beats o1-preview on key agentic benchmarks like SWE-bench Verified – making it the **best choice across the industry for production usage**

**Claude 3.5 Sonnet is $3/Mtok Input and $15/Mtok Output, compared to OpenAI o1 at $15/Mtok Input and $60/Mtok output (billing is also non-deterministic)*

SWE-bench Verified

Assesses a model's ability to complete real-world software engineering tasks and understand, modify, and test code with tools



Sources: [OpenAI o1 model system card](), [Anthropic model card (October 2024 3.5 model addendum)]()

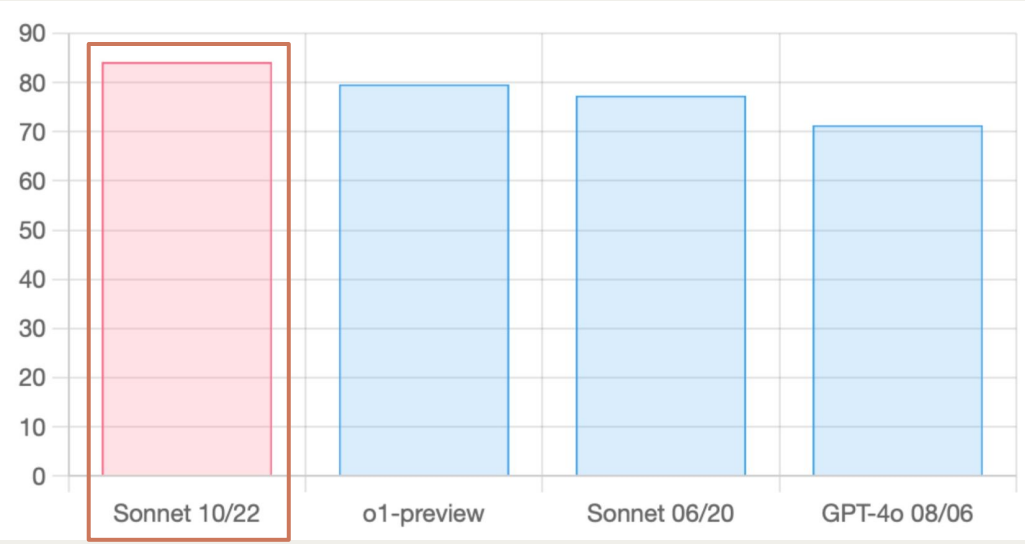
3.5 Sonnet outperforms OpenAI o1

Despite being significantly faster and cheaper than o1*, Claude 3.5 Sonnet beats o1-preview on key agentic benchmarks such the **aider code editing benchmark** – making it the practical choice

Claude 3.5 Sonnet is $3/Mtok Input and $15/Mtok Output, compared to OpenAI o1 at $15/Mtok Input and $60/Mtok output (billing is also non-deterministic)

Aider's code editing benchmark

Measures the LLM's coding ability, ability to write new code that integrates into existing code, and apply changes to the source file without human intervention



Sources: Aider LLM Leaderboards

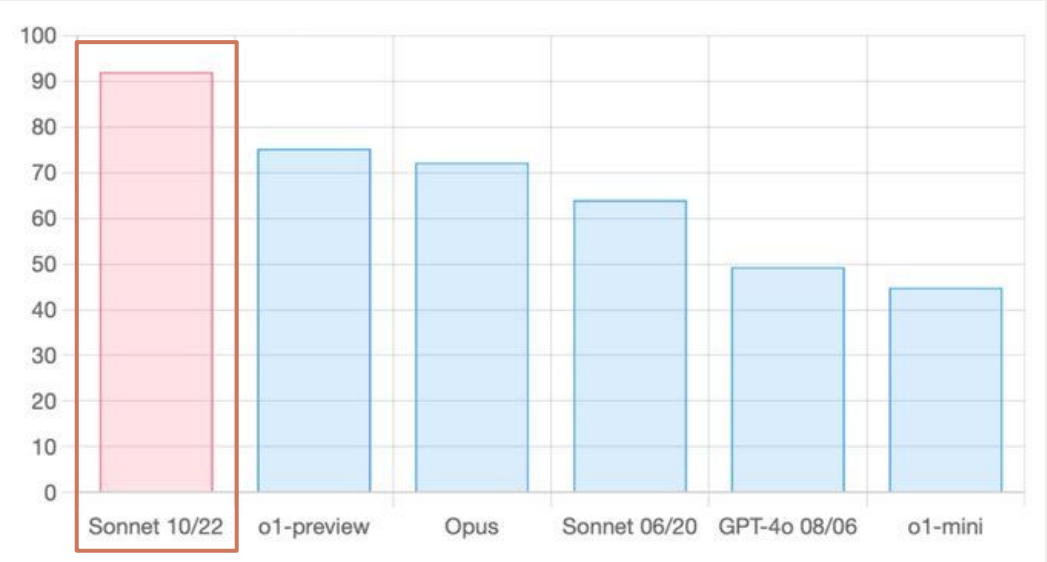
3.5 Sonnet outperforms OpenAI o1

Despite being significantly faster and cheaper than o1*, Claude 3.5 Sonnet beats o1-preview on key agentic benchmarks such as an **18% lead in the aider refactoring benchmark** – making it the practical choice

**Claude 3.5 Sonnet is $3/Mtok Input and $15/Mtok Output, compared to OpenAI o1 at $15/Mtok Input and $60/Mtok output (billing is also non-deterministic)*

Aider's refactoring benchmark

A more challenging benchmark which tests the model's ability to output long chunks of code without skipping sections or making mistakes



Sources: Aider LLM Leaderboards

Case study

Computer use

Computer use

- Demos!
- Technical implementation
- Safety considerations
- Future implications

Claude 3.5 Sonnet (upgraded)

Computer use demo: Coding

https://www.youtube.com/watch?v=jqx18KgIzAE

Claude 3.5 Sonnet (upgraded)

Computer use demo: Operations

https://youtu.be/ODaHJzOyVCQ

Practical safety measures

- ASL standard implementation
- Security measures
- Deployment safeguards
- Lessons learned in year 1

Future

- Scaling governance
- Capability measurement
- Academic collaboration

Add your questions!

Thank you