Measurements for understanding the pace of AI development inside frontier labs

PRIMARY

Publisher	Anthropic
Published	2026-09-18
Retrieved	2026-09-19
Cite key	anthropic2026measureme
URL	https://www.anthropic.com/institute/measuring-pace-of-ai-development

Claims drawn from this source. CLAIM-2026-09-19-01, CLAIM-2026-09-19-02, CLAIM-2026-09-19-03, CLAIM-2026-09-19-04, CLAIM-2026-09-19-05, CLAIM-2026-09-19-06, CLAIM-2026-09-19-07, CLAIM-2026-09-19-10

Reasons to track these measurements The measurements in this piece are focused on how models are built. By better understanding the production process of models, we have a better chance of correlating model inputs, like compute, with model outputs, like capabilities. They complement capability evaluations, which measure what models can do . We publish those separately through our Responsible Scaling Policy (RSP) risk reports, which include evidence on how much our models are accelerating AI R&D. In our policy proposal on advanced AI, the Advanced AI Framework (AAIF) , we propose rules of the road for how any lab releases safe models, including transparency obligations that governments could require, such as risk reports. Together, these proposed measurements and policies are a starting point for monitoring the pace of AI development from outside the labs. (1) Measuring AI-led AI R&D Why measure AI-led R&D? Frontier AI labs increasingly use AI to build future AI models. This process allows labs in democratic countries to develop more capable models more quickly and conduct more safety and testing on models before they are released to secure AI's benefits while staying on the frontier. However, models accelerating their own development could make it more challenging for humans to understand or control these systems. It is therefore important to share these metrics to understand how close the world is to reaching recursive self improvement (a model fully autonomously building its successor). What we measured. We built a prototype index of how much of Anthropic's AI research and development (R&D) is performed by Claude, called the Anthropic R&D Automation Index. It's built by cataloguing every kind of AI R&D work done at the company, rating how automated each task currently is, and aggregating those ratings. What we found. To measure the extent to which AI is doing AI R&D at Anthropic, we use an automation rating scale developed by Epoch AI that measures "Automation Level," or AL. It runs from AL0 (no AI involvement) to AL5 (AI operates fully autonomously, with no human in the loop). In AL3, AI "collaborates": it can do large chunks of work under close human direction. In AL4, AI "leads": it can complete most of the task end-to-end from a

high-level prompt, while the human supervises. As of August 2026, Claude is not operating fully autonomously for any measured subset of AI R&D work. Claude “leads” 26% of Anthropic’s AI R&D work. The share of work at or above “AI collaborates” is above 90%. What any AI developer could report today. Any frontier developer could publish these measures regularly, using a public methodology. This would enable the numbers to be compared over time, and potentially across labs. Two obstacles stand in the way of cross-lab comparison on this type of reporting. First is the lack of a common methodology. Second, we’re using our own models to evaluate our systems, which could mean that the “judge” model could make the same kinds of errors as the model it is checking. To address these challenges, a developer’s measurements could be verified by a third party, or by other developers’ models, with guardrails in place to limit the exchange of competitively sensitive data. These measures could also become the trigger for stronger requirements, like a fixed testing window before a new model is used for further AI R&D. (2) Measuring oversight of AI agents Why measure oversight of agents? Like other frontier developers, Anthropic employees increasingly delegate tasks to agents that work semi-autonomously for long stretches, and that delegate work to one another. As work becomes increasingly automated, from “AI collaborates” toward “AI leads,” agents could make more consequential decisions, such as which research direction to pursue next. These decisions need safeguards to block harmful behavior from an individual agent, as well as an oversight regime to detect problems that might only arise across many agents. In our monitoring data to date, individual agents rarely misbehave. But when there are millions or billions of agents operating in the economy, even rare events can happen regularly. What we measured. We’ve built a system that lets us oversee and intervene in actions AI agents take on Anthropic’s systems. Here, we consider three different metrics: coverage, which describes the share of an agent’s actions that pass through a monitor before or after execution; review latency , which is the time between an action and its review, first by an automated monitor and then by a human; and escalation rate, which is the share of agent activities that are either blocked/redirected (in the case of online monitors) or flagged for further review (in the case of offline monitors). What we found. As of August 2026, there were approximately 30,000 agents doing research and engineering work at Anthropic at any one time in our most-used internal platform. These measurements cover this platform only. The actions of these agents are constrained by two kinds of monitors, summarized below: Online monitors Offline monitors Description Online monitors check the actions of agents in real-time and either block dangerous actions or steer agent misbehavior automatically. They’re built to address harms that would happen quickly and could not be undone, such as an agent copying model weights out of our systems. Offline monitors read agent activities after the fact . They are built to help address dangerous behaviors that develop slowly and can be reversed, like signs of model misalignment. We use flagged activities to improve our mitigations, like classifiers and deterministic controls. If the situation caused a large imminent danger, we would shut down the dangerous parts of the agent’s deployment. Coverage 100% of these agents’ actions pass through the online monitor before they are executed, usually within seconds. 100% of these agents’ actions are ingested by the monitor after they are taken. Review latency Automated review happens before an agent takes action; humans review any blocked actions within one week. The monitor flags roughly 100,000 transcripts per week that are then reviewed by multiple

classifiers for false-positives and initial oversight review. Most of the flags are resolved by this process, and the highest priority flags (approximately ~50 per week) are escalated to human review. Blocking / escalation rate We analyzed over a billion decisions from our research & engineering agents over August 2026 and found that 0.002% of them (about 1 in 47,000) were blocked by this monitor. The monitor flags roughly one to two transcripts in every thousand for further review. What an AI developer could report today. Any developer running agents on its own research and engineering workloads could publish the same measures: coverage (the share of agent activity that is monitored), review latency (how soon flagged activities are reviewed), and escalation rate (what share of agent activities the monitors block or flag). Together, these measures would allow society to see whether oversight is keeping pace with the growing role of AI in AI R&D. We published all of these measurements in our recent risk report Developers should share how often known agent misbehavior is caught by monitors, and should have third parties do this kind of testing independently. In the past, METR has independently red-teamed our offline monitoring platform . As described above, we are now setting up external third party evaluators at Anthropic. (3) Measuring compute allocation Why measure compute allocation? Broadly speaking, AI developers use compute for building more powerful models, serving customers, and safety-focused work like auditing a model's "thoughts" training model organisms to study misalignment , and evaluating whether a model can be safely deployed . Understanding how AI developers allocate their compute can tell you where a developer is focusing its resources and how that focus changes over time. Additionally, compute is among the most verifiable inputs to the AI R&D process, meaning that it could be a critical lever in a future pacing effort. A coordinated pacing effort could encourage companies to increase the compute allocated to safety across the industry and devote more resources to alignment, interpretability, safety testing, and evaluation. What we measured. We examined a snapshot of how Anthropic used all of its compute from July 13 to July 20. To do that, we sorted every workload into a small number of categories, then asked how much of the compute going to AI R&D was safety work. Safety research tends to use less compute than frontier training runs by its nature, so compute is an imperfect proxy for how much a company focuses on safety. This is because safety research consists of individual researchers designing experiments, which is time-consuming even though running the experiments is not particularly compute-intensive. The value of this metric, therefore, is less the absolute numbers and more that it provides a straightforward mechanism to compare like with like, across developers and over time. What we found. Over the examined week, about 6% of compute that went to AI R&D was allocated toward safety, and about 12% of compute that went to AI-driven AI R&D was allocated toward safety. These are deliberately conservative estimates. For example, if a token was used to advance capabilities as much as it was to advance safety, it was not counted in these metrics. Additionally, these metrics do not account for safeguards classifiers, which are a separate, comparable amount of compute that make our models much safer for the world. What an AI developer could report today. Any frontier developer could publish what share of its AI R&D compute goes to safety work, with the category definitions published alongside and the classification checked by an independent third party. Safety research is hard to distinguish from capabilities research, and each developer will be tempted to draw the line generously. The burden of proof should sit

with the developer to show that work is safety-related. Developers, governments, and the wider research community would benefit from converging on a shared definition ahead of time. A measurement like this could inform future actions, such as a lab's commitments about the share of compute going to safety research, or limits on the share of compute going towards AI research agents. Conclusion As the world considers pacing the frontier, we should do everything possible to minimize the gap between what frontier labs know and what the public knows. This means better measuring the development of AI, reporting on it publicly, and giving society an opportunity to decide how to use this information. We hope to model that transparency by releasing these measurements, and we'll continue to do so. Appendix Here are methodological details on all of the measurements we've prototyped. Measuring AI-led R&D How we did it. The Automation Index requires three things: a complete map of all the AI R&D tasks being done at Anthropic, a way to rate the level of automation, and a way to weight the tasks, so that important areas of work count for more than less important ones. No one person can list every AI R&D task at a frontier AI company by hand, at least not at the granularity we want. Instead we constructed this list of tasks in a bottom-up manner from work records including Slack and various sources of internal documentation. For each week in July 2026, we randomly sampled 20% of staff from each department that make up the model R&D loop. A Claude research agent reviewed each sampled person's week using Slack and internal documentation, and listed the tasks they worked on. Repeating this for each week in July 2026 gives us a flat list of ~15,000 granular model R&D tasks. We then used Claude to organize these tasks into a hierarchical tree, starting from all model R&D at the root and branching into areas such as training and product, then pretraining and reinforcement learning, and so on down to increasingly specific kinds of work. The resulting tree has 542 nodes at different depths, of which 378 are leaves like "eval platform defect diagnosis and fixes," "RL sandbox egress and network policy," and "serving incident postmortems." We freeze this tree so that every measurement we make happens against the same basket of work. For each node in the tree (a task category describing all the work beneath it), a Claude agent deeply researches how that kind of work is done across the company: who does it, with what tools, and how much of it AI performs. An independent Claude judge then read the resulting evidence and assigned one of six automation levels, adopting a scale proposed by Epoch AI to differentiate the degree to which AI is used: no AI involvement, minimal AI involvement, AI assists, collaborates, leads, or is autonomous. When we rate a given month's automation, we only allow the research agents that do the ratings to see evidence from that month or earlier. To aggregate all the automation level ratings into one number, we want to give each node in the tree a weight corresponding to how important that work is to the overall model R&D effort. Rather than deciding ourselves what kinds of work are more important than others, we used the amount of person-time dedicated to that task as a proxy. Using our sample, we had Claude research what each person worked on during each week of July 2026. Each person gets one unit of weight per week, split evenly across the tasks they worked on that week. If person A worked on four tasks, each gets 0.25; if person B worked on ten, each gets 0.10. A category's weight is the sum of all the person-time weights given to it. This is a crude approximation, but on average the scheme behaves sensibly: it assigns more weight to categories that many people are assigned to. What this does and doesn't capture. First, the automation ratings depend on

the judge model. To check them, we asked Anthropic staff who own the relevant work areas to rate the relative automation of their areas. To ensure an unbiased read, staff made their ratings without knowing what evidence the models had gathered or how they had judged that evidence. Our judge model agreed with humans about as often as humans agreed with each other (model-versus-human exact agreement was 59%, human-versus-human was 35%), and model and human ratings were within one level of each other 97% of the time. There remains real room for disagreement on borderline cases, such as where exactly “AI collaborates” ends and “AI leads” begins. Second, the basket is frozen. A growing index number on a July 2026 baseline tells us that the work humans were doing at that time is being automated; it does not, on its own, tell us whether new kinds of work are appearing that humans have shifted onto. To investigate this, we constructed an alternate version of the frozen tree from January 2026 data, and compared new tasks arriving every month from February to July 2026 against that January 2026 basket. We found no rise in the number of “novel” tasks between the January and July baskets of tasks, suggesting that the structure of model R&D work is stable at our level of analysis. Nevertheless, we plan to rebuild the basket of tasks periodically and re-version our published automation numbers as appropriate. Oversight of agents How we did it. In our most commonly used internal agent scaffold, two design choices allow us to more easily detect unknown unknowns and let agents monitor each other: Identity. Most AI users are accustomed to talking to a single “Claude” across many different chats. With many agents working together, we have found it important to give agents an individual identity, and tie all of the data that agent creates to its identity. This lets an agent distinguish itself from others, and treat what comes from another agent as a claim to check rather than a thought of its own. It reduces the risk of correlated actions, by allowing agents to make judgments based on their individual experience. And because the identity is not tied to a model, it persists through model upgrades, so an agent’s record is continuous even if the underlying model powering it changes. From an oversight perspective, this design ensures that every action can be audited per agent across time, and that every data record can be linked back to a specific agent. Open communication. Agents communicate through a shared, open messaging system, rather than privately. Every message is tied to the identity of the agent that posted it and can link to other content, so that the agents are sharing original references. This allows agents to avoid “playing telephone” as they communicate. Because agents can see each others’ messages, they can correct each others’ errors, and coordinate about shared work. All messages are cross-referenced with agent transcripts, allowing behavioral monitoring to traverse communication between agents. What this does and doesn’t capture. At this scale of agent activity, it is hard to be certain we have a complete picture of all possible kinds of behaviors, or that we are making these patterns adequately legible to human oversight. We have been doing this kind of monitoring for months, not years, and expect the science of agent monitoring to change considerably going forward. Measuring compute allocation How we did it. We started with our existing compute monitoring tools; Anthropic, like presumably every frontier developer, has several monitoring tools built for capacity planning that periodically sample accelerator usage and tag workloads with best-efforts labels (i.e., research and model development, internal usage, first-party inference, and so on) based on its metadata. Usage on third-party cloud compute is reported to us by the providers and folded in. Most of the

work of this exercise was stitching these existing sources together. We then used Claude to classify each workload as either safety work or AI R&D via a prompted classifier. Safety work was defined as work whose dominant purpose is making AI systems safer, more understandable, or more secure. Everything else, including capability research, training production models, product development, and developer tooling, was counted as AI R&D. Work that helps capability as much as it helps safety was also counted as AI R&D, so the safety share is conservative. For research training and evaluation runs, we built a classifier that reads the run's metadata and the code it used, and returns a classification, a justification, and a confidence level. Rather than classify all of the week's almost 10,000 runs, we sampled about 14% of them, weighting the sample toward the runs that used the most compute, so that the result reflects where the compute actually went, rather than how many runs there were. For inference for AI research agents, a variant of the same classifier read the agent's session transcript. Where transcripts were inaccessible (usually due to the work being compartmentalized), we classified them by the user's team or conservatively defaulted to classifying them as AI R&D. We plan to refine this pipeline so that an independent third-party could re-run the classifier on a random subsample of jobs and transcripts and check both the sorting and the totals. You're helping to perform an internal audit at the frontier AI company Anthropic to track where our research compute goes. The aim of the audit is to produce a public-facing breakdown of the usage of all of our AI accelerator chips into a handful of buckets. One split we particularly care about is the division between compute which was spent on safety research versus other R&D. Your job is to look at one research job at a time, figure out what it was doing, and assign it to one of those two buckets. [...] Safety and/or security research is work whose dominant purpose is making AI systems safer, more understandable, or more secure. This work can be broken down into a few main categories: [...] On the other hand, the following work falls outside of the scope of safety research: [...] Here are some boundary cases, along with how to think about them: [...] What this does and doesn't capture. The main lesson of this exercise is that classifying what is and isn't safety work is difficult but tractable, since the boundary between these categories is not black and white. For example, research on scalable oversight might make future models more aligned and current models more commercially useful — it's difficult to determine whether this is primarily safety- or capabilities-advancing. We found that an extensive written definition of each task, with clear boundary cases (an excerpt is above), gets the classifier to agree with human reviewers within one or two percentage points of difference between the human and machine raters. But some cases were too difficult to determine even after several hours of human review. Our definition is one reasonable choice among many; a different developer, or a regulator, might draw the line differently. Three further limitations matter. First, many of the underlying labels we relied on (i.e., reasons for runs, workload tags, the source of API traffic) are set by automated rules, or occasionally directly by users, and are best-effort, not verified. In most cases, we expect that our classifications are accurate, but in some cases usage may be mislabeled and our pipeline would not necessarily catch it. A measurement meant to be trusted by outsiders will need to be complete, accurate, and technically enforced. Second, the measurement covers one week, which is enough to show that the measurement can be made, but not enough to show a meaningful trend. Third, and most importantly, compute share measures only what is spent. A more efficient safety

classifier, or a faster inference stack for production models, lowers the safety portion, but doesn’t mean we’re doing less safety work. Our own classifier overheads have fallen with efficiency improvements, and have risen when production inference was more efficient than the classifiers were.