VOLUME II · ESSAY 04

Who Trains the Next Engineer?

The apprenticeship paradox of automated expertise.

Christophe Pennetier, André-Benoît De Jaegere

Generative Engineering · Volume II · written from October 2026, two years after Volume I · https://generative-engineering.aquia.studio/volume-ii/training-the-engineer/

CONTENTS

The same prompt, a different past 3
The apprenticeship hidden inside the work 5
The colleagues who remain in your head 9
Aristotle in your pocket 11
The university as an encounter 14
The job keeps its name 15
The missing recovery interval 18
Engineering an apprenticeship 19
Who pays for the next engineer? 22
Sources 24

I can build an interface in TypeScript. I do not know TypeScript.

The arrangement suits us both. The language has escaped my opinions about its syntax; I have escaped several weeks of learning it. Between us sits an AI, translating what I want into something I can inspect, criticize and improve.

What I want, however, did not arrive with the subscription.

I know when a screen asks its user to think too much. I know when a beautiful demonstration is hiding a poor system design. I know why a product needs a particular test, why a seemingly reasonable feature will complicate everything that follows, and why the right answer to some requests is to leave the screen alone. Years of work supplied those judgments. The model gives them reach.

Now put a junior engineer beside me. Give us the same model, the same budget, the same blinking cursor. We have equal access to the machine. We have very different things to tell it.

The difference is not that I own a better collection of incantations. A good prompt often contains a history: mistakes already paid for, alternatives already rejected, conversations with people whose objections I can still

hear. What looks like a clever instruction may be twenty years of experience compressed into six sentences.

This is the attraction I described in “The Iron Man Paradox”. The suit lets a person act far beyond their unaided abilities, but the engineer inside it brings something the suit did not manufacture, and I worried that the person inside might not be replenishing what the suit allowed them to spend.[1] My TypeScript makes the problem unusually clean. The interface improves. My ability to direct its development may improve too. My ability to write TypeScript on my own need not improve at all. Three different outcomes, appearing together on the same screen.

There is nothing wrong with that bargain. I do not need to acquire every skill I can now put to work; delegation is part of intelligence. An engineer who insisted on personally reproducing every layer of a modern computer would end up with a very intimate relationship with sand.

Andrej Karpathy gave one version of this freedom a name in February 2025: vibe coding, a way to “forget that the code even exists”. He meant it for throwaway weekend projects; Collins made it the word of the year anyway, which says something about our weekends.[2][3] The phrase is a fine account of what an interface can make disappear. It says nothing about which knowledge the person can safely do without.

That is the hard question. Which judgments can stay outside us? Which must we develop, because we will be responsible for noticing when the system is wrong? And how will the next engineer acquire them, if the work that once exposed us to mistakes is finished before they have had time to make one?

In the first volume of this study, education came with an attractive promise: personal instruction, accessible expertise, richer environments in which to learn.[4] The promise survives. Two years later it needs a second clause. We can make expertise available without making experts.

That distinction runs through the four essays of this volume. Mathematics separated a valid result from an intelligible one. The colleague essay asked what happens when we delegate. The loop essay asked how a system learns from its encounters with reality. This last one brings the loop back to the person doing the delegating.

An engineering organization produces two things. One appears in the delivery report. The other goes home at the end of the day, slightly more capable, or slightly more dependent, than when they arrived. We have

become remarkably ambitious about the first. We are strangely casual about measuring the second.

The same prompt, a different past

It would be convenient if the evidence simply confirmed my intuition: AI rewards experience, so senior people win. It does not.

In a study published in the Quarterly Journal of Economics in 2025, Erik Brynjolfsson, Danielle Li and Lindsey Raymond followed an AI assistant through the work of 5,172 customer-support agents. Issues resolved per hour rose by 15 percent on average, and by about 30 percent for the less experienced and less skilled. The most experienced and most skilled gained a little speed and lost a little quality. The assistant seemed to spread the tacit practices of the best agents to those who had not yet learned them, and some of the gain persisted when the software went down.[5] Here AI helped people catch up: a serious finding, and an inconvenient one for anyone proposing an aristocracy of seniority.

But answering customers inside an established workflow is a different problem from deciding what product to build, which assumption to relax, or whether an apparently successful simulation answers the business question. A system may supply the missing next step particularly well when the organization has already defined the destination. My advantage may be larger when the destination itself needs inventing. That is a hypothesis about the structure of the work, not a law about age. A junior can understand a new tool better than a senior; a senior can preserve a mistake with extraordinary confidence. Years served are an imperfect proxy for what has been learned.

Even the speed story resists a single verdict. In early 2025, METR asked sixteen experienced open-source developers to work on their own repositories with and without AI tools. With the tools they took 19 percent longer. They had expected to be 24 percent faster, and afterwards they believed they had been 20 percent faster.[6] A year later METR redesigned the study and explained why its new data gave an unreliable signal: developers were choosing which tasks to bring, and time is hard to measure when one person runs several agents at once. It believed developers were probably sped up by then, on very weak evidence.[7] A result from one generation of tools should not become a constitution.

For training, I would add more distinctions. There is what a person can produce with help. There is what they can still do when the help disappears. There is the range of situations they can recognize and connect. And there is whether they get the chance to take part at all. A tool can move one of these while leaving another unchanged, or move them in opposite directions. This is why a successful demonstration cannot settle an argument about education.

Learning researchers have made the underlying distinction for decades. Nicholas Soderstrom and Robert Bjork's review of learning and performance shows that how well we perform while practising is an unreliable guide to what we retain and can transfer, and that some conditions move the two in opposite directions.[8] The activity that feels easiest now is not necessarily the one that leaves us most capable later.

A small engineering example. An agent repairs a concurrency bug. I read its explanation and recognize every sentence; recognition feels like understanding because nothing in it surprises me. A week later the same structural error appears under different names, in another library. Can I find it? Can I predict the failure before the test reveals it? Can I explain why the obvious repair creates a second problem? The closed ticket cannot answer those questions. Neither can my satisfaction with the explanation.

Yet it would be just as premature to conclude that assisted work only borrows expertise. In a three-month randomized experiment reported in September 2026, 133 patent lawyers at eleven American law firms drafted with or without an AI assistant. With it, their drafts improved, and juniors gained more than seniors. Then, at three months, everyone reviewed a patent application without AI, and the treated lawyers did better there too; but that unaided gain came entirely from senior lawyers, with seven years or more behind them, while the juniors' scores split in two directions.[9] [10] Juniors gained more with the tool; seniors gained more without it. Google ran and funded the study, with a tool of its own, at firms that do business with it, and the authors say the horizon is short. It is still evidence that assisted work can build expertise, under conditions, and a direct challenge to the claim that experienced users must decline. I want that challenge in the argument. A metaphor that cannot survive counterevidence is a slogan in a dinner jacket.

So the Iron Man problem needs a sharper statement. AI can amplify expertise that exists. It can also help create expertise. Whether it does the second depends on what the person actually does while enjoying the first.

The decisive unit is not the licence. It is the learning experience organized around it.

The apprenticeship hidden inside the work

In 2024 I wrote that we should treat AI as an apprentice in our charge.[11] Two years later, the apprentice I worry about is human.

Picture a junior engineer facing a model that refuses to converge. The documentation describes the happy path. The real path contains missing values, incompatible assumptions and a warning that everyone has learned to ignore. The senior engineer is unavailable until Friday. By Thursday the junior may have learned something important. They may also have spent three days discovering that the instructions were wrong.

We should resist calling the whole experience formative. Suffering is an unreliable curriculum; a broken installation script does not become educational because the people who survived it now run the department. Some friction, though, contains the lesson. Trying an approach creates an expectation. Watching it fail supplies a discrepancy. Investigating the discrepancy can change the learner's model of the system. If an agent removes the failure before the learner has formed an expectation, the organization gains time and may lose that particular lesson. The task always contained two processes, getting the work done and changing the person doing it. Automation forces us to tell them apart.

Matt Beane watched the problem happen in operating theatres. In open surgery, the resident's hands were needed: the senior surgeon could not do everything alone, so the trainee learned by doing part of the operation, watched, corrected, trusted a little more each time. With the surgical robot, the surgeon at the console could do it all. The resident was still in the room, holding a suction tube at the bedside or watching a second screen, and no longer needed. Optional to the production, still dependent on participation for the learning.

Nothing had been forbidden. The work had simply stopped needing the learner.

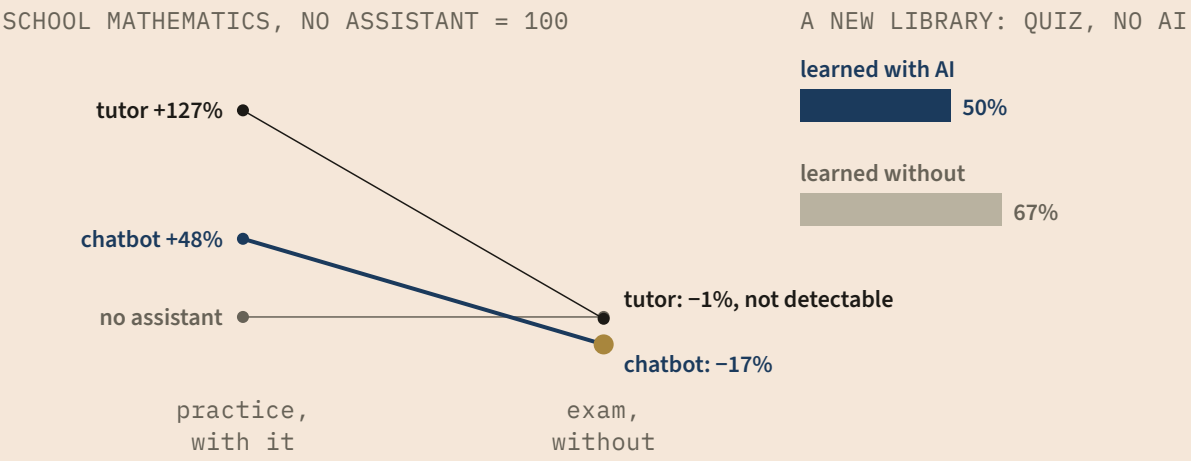
The few residents who became skilled anyway did it in the shadows: specializing early, rehearsing on simulators and recordings of operations, operating at the edge of their ability with little supervision. Beane calls it

shadow learning, and it came at a cost to the residents who practised it, to their peers, who got less time at the console, and to the profession, which got fewer experts.[12][13] Lisanne Bainbridge had named the irony forty years earlier, in “Ironies of Automation”: automated plants left their operators responsible for the abnormal conditions while taking away the practice that kept them able to handle them, and “the more advanced a control system is, so the more crucial may be the contribution of the human operator”.[14] Her examples came from process control and flight decks, long before chatbots. I am not the first to worry about the junior. I would like to be among the people who design something better than shadows.

Today we give the arrangement a reassuring name: a human in the loop. But a human is not a qualification. Someone must know enough to recognize the exception, interpret the evidence and intervene. Seating a novice beside a fluent machine does not create an expert supervisor. It can create a person whose main contribution is clicking the button that makes the organization feel supervised.

Two recent experiments bring the question close to the desk. In the first, Judy Hanwen Shen and Alex Tamkin asked 52 professional Python developers to learn an unfamiliar library through a coding task, with or without an AI assistant, then took the assistant away for a quiz. Those who had learned with it scored 50 percent, against 67 percent for those who had learned without, and the time they saved was too small to be sure of. Within the assisted group, people who asked for explanations and concepts tended to learn more than those who delegated the code, though nobody assigned those styles at random and the groups were tiny.[15] In the second, in a Turkish high school, Hamsa Bastani and colleagues gave nearly a thousand students either nothing, an assistant that worked like ordinary ChatGPT, or a tutor built to give hints rather than answers. During practice the ordinary assistant raised grades by 48 percent; in the exam that followed, taken without it, those students did 17 percent worse than the students who had never had it. The tutor raised practice grades by 127 percent, with no detectable difference on the unaided exam: it avoided the harm without producing a measurable gain (Fig. 1).[16]

FIG. 1 **A chatbot that raised practice grades by 48 percent left exam grades 17 percent lower; a tutor that gave hints raised practice more, with no detectable difference on the exam.**



Left: one high school in Turkey, nearly 1,000 students, four 90-minute sessions; grades relative to students without an assistant (100), practice with the assistant, then an exam without it. The tutor's exam estimate, about 1 percent below no assistant, is statistically indistinguishable from it. Right: 52 developers learning the Trio library, quiz taken without AI. Bastani et al., PNAS 2025; Shen and Tamkin, 2026. Two short studies: they measure what was retained weeks later at most, not careers.

Measured

Neither study shows that AI makes people worse at their jobs over a career. Both show why counting finished tasks can miss the outcome that matters to an apprentice. And neither argues for banning help: a student who copies an answer learns little from a human answer key either. What changes with AI is how available, plausible and convenient it has become to outsource the very steps through which understanding might have formed. The response is to design those steps on purpose.

I learned how easily that design goes wrong with my own team. When AI made us faster, I gave a quarter of the time it saved back to the team, for learning. They spent it reading the code the machine had written. It felt conscientious, and it taught them little. I have come to think that reading generated code line by line is the wrong job for a person. On writing code, I would bet on the machine against any of us; nobody asked it to pass a coding interview before hiring it, and it is odd to make a human re-interview every line it writes. What we had not done was change the way we work to suit a colleague we cannot fully trust. The change it calls for is more tests.

Testing is where the learning was. A test is a prediction made in public: given this, the system must do that. To write one worth having, you have to understand the problem, find where the system could break and imagine

the case nobody has tried; every good test is a piece of understanding made executable. And when the machine falls short, as it still does, the tests are how you push it back. Learning scientists happen to use the same word for something close. In the evidence reviewed by John Dunlosky and colleagues, practice testing, producing an answer rather than recognizing one, and practice spread over time do far more for durable learning than rereading.[17] My team had been rereading. A learning budget needs an activity that exercises the capability, not a line in the calendar.

The other half is harder: the conscious part of learning, knowing what you do not know and steering yourself towards it. Most of us are poor coaches of ourselves. We take fluency for mastery, and learners, as Soderstrom and Bjork show, routinely read their ease during practice as proof that they have learned.[8] Almost nobody is ever taught how to learn. A machine that is always available does not fix that. By making fluency cheap, it can make it worse.

So the help has to ask for work. Suppose an agent finds the bug in thirty seconds. Before showing its repair, it could ask the junior to write the test that fails, and to say why it fails. After the repair, it could present a closely related case in which the same fix is wrong, and ask for the test that proves it. Later, it could bring the principle back in another setting. The machine's speed then buys more encounters with the mechanism instead of ending the encounter sooner. That is my proposal, not a guarantee that a particular prompt works. The mechanism underneath is older. Cognitive apprenticeship made expert thinking visible through modelling, coaching and support that fades as the learner grows, with articulation and reflection along the way.[18] Research on productive failure, problem-solving first and instruction after, finds a moderate benefit on average across 53 studies, stronger when the design is faithful to the idea, and reversed for young children.[19] Neither tradition endorses leaving novices alone indefinitely; there is a world between letting a student meet a real intellectual obstacle and withholding a useful map because one's own supervisor withheld it. Guidance has to match what the learner already knows, and novices can drown in too much discovery.[20]

The principle underneath fits in a sentence. Learning needs a commitment made before the answer arrives, a prediction, a test, a diagnosis, a decision, something reality can then confirm or refuse; without it, the answer slides past as something we would surely have known if anyone had asked. That also dissolves an old opposition. Automation and augmentation are not two kinds of software but two uses of the same one: the agent that fixes the bug at noon can, at two, ask the junior for the test

that would have caught it. What changes is what the agent is asked to achieve, the order in which it shows things and what the person gets back. Someone has to choose that, and pay for it, because an agent rewarded for closed tickets will never decide by itself that this ticket should stay open long enough to teach someone.

The colleagues who remain in your head

Juniors are the obvious concern; their missing experience is visible. Seniors present a quieter problem, because for a while everything looks excellent.

AI lets me do work that once needed several complementary people. I can turn an intuition into a prototype, inspect it, reject it and try again. Fewer handoffs, less waiting, more contact with the thing I wanted to make in the first place. It would be absurd to mourn every meeting removed from this process; some contributed mainly to the measured passage of time.

But a team supplies more than execution. It supplies other ways of attending to the world. One colleague sees an exception in every rule. Another sees a business in every exception. Someone notices the awkward interaction; someone else refuses to discuss the interaction until the system survives a load test. These preferences can be inconvenient. They also become part of how we think. Years later you can still imitate those people, sometimes affectionately, sometimes accurately enough to require discretion. You know what they would object to. Their questions arrive before you have invited them.

The apparent solo worker may therefore be carrying a small, argumentative institution inside their head.

Research by Tian Heong Chan, Jürgen Mihm and Manuel Sosa makes the point with inventors and their breakthroughs. Collaboration did not have one universal advantage: it depended on whether a problem could be split into separate contributions or needed an integrated solution. And lone inventors who had collaborated in the past did better, especially on the inventions that could not be split.[21] The data are historical patents, not AI users, but they explain why removing today's team does not immediately remove what earlier teams left in the person. The question the productivity dashboard cannot see is what replenishes it now.

I find it useful to call the junior's problem a problem of the mean, and the senior's a problem of the variance. This is shorthand, not a claim that an engineer's mind follows a bell curve. The junior needs dependable competence: recognizing recurring structures, finishing important tasks, catching familiar failures. The senior may already have that foundation. The senior's exposure is in the breadth of situations, analogies, objections and approaches they can bring to a new problem. It is quite possible to stay excellent at familiar work while becoming less likely to meet an unfamiliar way of seeing it. Delivery stays impressive. The repertoire quietly narrows.

That narrowing is my hypothesis, not a measured effect of AI, and the opposite hypothesis is attractive: AI could expose a person to far more disciplines, examples and disagreements than their team ever could. There is evidence for it. In "The Cybernetic Teammate", a preregistered field experiment with 791 Procter & Gamble professionals, individuals working with AI matched the performance of teams working without it, and the AI helped people from technical and commercial functions produce more balanced solutions.[22] That is a serious reason to doubt that a machine can only echo what its user already knows. The sessions lasted a day; they did not measure how those people's capabilities developed over years. Receiving an unfamiliar contribution and building a lasting capacity from encounters with it are different things. How often does the machine raise a question the user would never have asked? Does the user pursue it, or ask the machine to get back to the deliverable?

A human colleague has a useful property here: an agenda that does not depend on completing my prompt. They may care about a customer I ignored, a discipline I undervalue, a risk for which they will personally answer. Their objection arrives with a history and a reason to persist. Charlan Nemeth and colleagues found that a genuine dissenter drew more and better solutions from a group than any of three versions of an assigned devil's advocate.[23] Asking five agents to adopt five job titles is an intervention worth testing, not proof that five independent perspectives have entered the room.

The worry about sameness has another foothold. Anil Doshi and Oliver Hauser found that access to AI-generated ideas made short stories more creative in the eyes of independent readers, especially for the less creative writers, while making the stories more similar to one another.[24] A room full of similar answers and a person losing range are different phenomena; the first can motivate a question about the second, not answer it. The same restraint applies to "model collapse": Ilia Shumailov and colleagues

showed that models trained recursively on their own outputs can lose the tails of what they model, and Matthias Gerstgrasser and colleagues showed that keeping the real data and accumulating, rather than replacing, avoids the collapse in the settings they tested.[25][26] That is a lesson about training regimes, not a diagnosis of the human brain. As an analogy it asks a good question: if a creative process increasingly feeds on its own previous output, where does contact with something independent come in? A customer interview can supply it, or a factory visit, a difficult paper, a stranger's objection, an experiment that refuses to cooperate.

James March described the trap long before this one. Refining what we already do pays quickly and visibly; exploring alternatives is uncertain, slower, and easily crowded out.[27] AI multiplies what we can achieve in the familiar workflow. My concern is that an organization may spend the whole gain there. The danger, if it comes, will not look like stupidity. It will look like a person delivering beautifully inside a world that has stopped getting larger.

Aristotle in your pocket

If colleagues replenish us, can a machine teach us? Whenever education enters this discussion, a familiar defensive move appears: the machine may explain things, we are told, but it will never replace a good teacher.

Perhaps. I would first like to know whether the person in front of the class is a good teacher. I have little sentimental attachment to a lecture merely because a human delivered it. A patient system that changes its explanation, diagnoses a misconception and offers well-chosen practice could beat a professor at many parts of learning. The burden of proof runs both ways. We should test the instruction, not protect the furniture.

The aspiration is older than generative AI. In 1984 Benjamin Bloom drew attention to the "two sigma problem": students tutored one to one, with mastery learning, did far better than students in a conventional class, and the challenge was to obtain such gains at scale.[28] It was a challenge, never a warranty that any tutor, human or digital, moves every learner two standard deviations.

The evidence answers three different questions, and it helps to keep them apart. The first: can a well-designed AI tutor teach well? At Harvard, students in a physics course alternated, over two lessons, between a carefully designed AI tutor and an active-learning class: their median gains

on the test right after each lesson were more than double with the tutor, in a median of 49 logged minutes against a 60-minute class.[29] The tutor carried expert-designed pedagogy and content, and nobody measured what the students remembered later. So the answer is yes, for a design, on a lesson, measured right away. That is not a finding that an unrestricted chatbot can replace a degree.

The second question: how can expertise reach more learners? Two studies answer with different mixtures of people and machine. In Nigeria, volunteers from nine public secondary schools in Benin City followed a six-week after-school programme that combined a generative-AI tool, activities tied to the curriculum and teachers who guided them: 0.31 standard deviations gained on a test covering English, AI and digital skills, 0.23 on English alone.[30] The programme added learning time and human support, so the effect cannot be credited to the chatbot alone; which is exactly why it is useful, because it describes a system someone could build. Tutor CoPilot took another route and advised human tutors in real time. In a randomized trial with more than 700 tutors and about 1,000 pupils, pupils were 4 percentage points more likely to pass the end-of-session check, and 9 points more with the tutors who had been rated lowest. The measured gain was immediate; a lasting gain was not established, since end-of-year scores did not measurably improve after only weeks of exposure.[31] In both, AI spread expertise through the people already sitting with the learners. “AI tutor” turns out to hide several inventions: one guides the student, another guides the adult, a third answers questions in a loosely structured chat. They may all wear a text box; their mechanisms are not interchangeable.

The third question: does access alone produce learning? The market has noticed the difference between answering and teaching. OpenAI launched a study mode for ChatGPT in July 2025, built around guided questions; Anthropic had launched a learning mode for its education offering in April.[32][33] Those are statements of intended behaviour, not evidence. Khan Academy’s internal tests, published in May 2026, are evidence about design: giving the tutor a summary of the learner’s recent attempts and of the prerequisites not yet mastered helped, while other plausible improvements did not, on a narrow measure, getting the very next question in the same skill right, unaided, in the same session.[34] And at the University of Maryland, giving the sections of a course access to an AI tutor lowered final grades by about a third of a standard deviation and cut participation in the course platform, with larger losses for first-generation students; only about 15 percent of the students who had access ever used

it, and three instructors were dropped after randomization for not switching it on.[35] That is the effect of access, not of use, and no verdict against tutors. It is a reason never to confuse availability with benefit.

So a machine can sometimes teach well, can carry expertise further through people, and can be handed out without teaching anyone. The harder question is how an encounter with the machine becomes a learning process for people who do not already know how to organize one. Even an effective tutor leaves three jobs to someone: choosing the curriculum, recognizing the gap nobody mentioned, and coming back after confusion.

I am a biased test subject. Give me a subject I do not understand and an articulate machine, and I will happily begin. I can sketch a curriculum, notice a hole in an explanation, and if a field contains a concept for which I have no name, the irritation is usually enough to keep me searching until I find it. The known unknown is easy: I know what I need to learn. The unknown unknown is the troublesome one: I do not yet know that something important is missing. My instinct is to widen the search, try an analogy, ask what would make the argument fail, follow an implausible connection, imagine the opposite of the result I want. Most of it goes nowhere. I enjoy it anyway. A person who enjoys getting lost is a poor representative of people asking for directions.

This is where the fox and the hedgehog remain useful to me, provided they are modes of inquiry rather than castes. The fox ranges across possible connections; the hedgehog pursues one question until the superficial answer gives way. Serious work needs both movements. AI can help each, and each can also use AI to avoid the other. What looks like an excellent tutor for me may be an excellent instrument for a learning strategy I already have. Calling it universally accessible because everyone can open the same application would miss the capability on which the experience depends. Barry Zimmerman’s account of self-regulated learning, setting goals, monitoring progress, evaluating oneself, treats these as skills that can be developed.[36] That matters: it offers an alternative to declaring that curious, independent people will flourish and everyone else will cope. We can teach people to steer their own learning, while admitting that at first they need help to do it.

A useful tutor should therefore reveal what the learner does not know to ask: propose a prerequisite, a counterexample, a competing interpretation; check whether the student can use an idea, not whether they enjoyed its explanation; and eventually make some of its own guidance unnecessary. There is a delicate commercial question in that last

requirement. A service can be useful because we keep needing it. An educational service should also be judged by what remains when we stop.

The university as an encounter

The strongest case for an institution may begin where the information metaphor ends. Imagine a student who believes engineering means getting the calculation right. In a shared project, one teammate asks who will maintain the system. Another asks whether the user wanted it. Someone from a different discipline asks why the problem was defined that way at all. The student has not merely acquired three answers. The kinds of questions they can recognize have changed.

An institution can organize such encounters: it can put people into an environment they would not have chosen from their existing preferences. That is a different service from supplying more information on request. Universities do not perform it automatically or equally; some reproduce advantage, reward conformity and make original questions surprisingly inconvenient, and expensive proximity should not be confused with education. Still, gathering people around sustained inquiry creates relationships and exposures that a content library does not.

The research on who becomes an inventor makes exposure hard to dismiss. Alex Bell, Raj Chetty, Xavier Jaravel, Neviana Petkova and John Van Reenen found that American children from the richest one percent of families were about ten times as likely to become inventors as children from below the median, that the gap persisted among children with similar early mathematics scores, and that children who grew up around inventors in a field were more likely to invent in that field themselves. Their conclusion was a phrase: lost Einsteins.[37] The study does not evaluate AI tutors or universities. It does undermine the comfortable assumption that the people already visible in inventive careers exhaust the people capable of them.

This is why I distrust the claim that AI mainly rewards an inherently creative minority. It may first reward people with the experience, confidence, autonomy and opportunity to use it, and all of those can be socially produced. Treating them as a fixed distribution of human types turns an observation about access into a theory of destiny. My own enthusiasm needs this correction. I like working with incomplete maps, challenging a machine and throwing its answer away. None of that tells me how a school should educate thousands of students with very different

preparation, incentives and circumstances. A coach working with a committed athlete still has a hard job, but a great deal has been selected already: the athlete wants to be there, accepts demanding practice, organizes a life around improvement. A classroom cannot assume those conditions. It has to help create them while teaching the subject.

So the promise of a tutor in every pocket contains a motivational and social problem as well as an instructional one. What brings the learner back after confusion? Who notices disengagement? Where does the learner meet people whose ambitions make another path seem possible? Which environment makes asking an elementary question tolerable? AI may contribute to all of these, and the empirical question is how well, for whom, and beside what other support. It would be foolish to rule it out. It would be equally foolish to delete the institution's social functions from the comparison and announce that its information has been replaced.

For employers the point is more immediate. A junior who hears experienced people argue about an incident learns that professional judgment contains uncertainty, disagreement and revision; which questions carry weight, how evidence changes a decision, how responsibility survives the discovery of a mistake. If all they receive is a polished conclusion, they inherit the answer without the work that made it trustworthy. University and workplace meet at this obligation: both should expose learners to thinking before it has been cleaned up for presentation.

The job keeps its name

Even an excellent way of learning needs somewhere to practise with real responsibility: a first job in which the work matters and someone is willing to let a beginner do part of it. That is where the next scale begins, and where the numbers are noisiest. Discussions of AI and employment often start with a headcount and end with a verdict. A company announces layoffs; AI is in the air; therefore AI caused the layoffs.

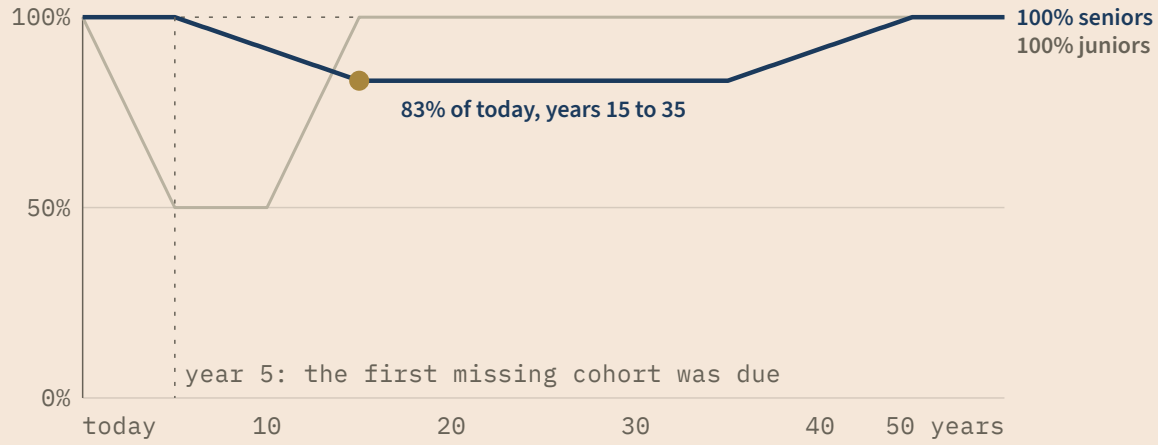
Sometimes automation contributes. The announcement alone cannot identify the mechanism. In November 2022, laying off more than 11,000 people, about 13 percent of Meta, Mark Zuckerberg wrote that he had expected the pandemic's acceleration of online business to last, had invested accordingly, and had got it wrong; he also cited the economic downturn, competition and lost advertising signal.[38] That is a different causal story from a machine replacing a task. In June 2025 Amazon's Andy Jassy wrote that, as the company rolled out generative AI and agents, "in

the next few years, we expect that this will reduce our total corporate workforce”.[39] That is an expectation, not a measured count of departures caused by AI. Pandemic hiring, business conditions, ordinary restructuring, automation and changing skill requirements should stay analytically separate even when an executive puts them in the same paragraph; and a layoff is not evidence about the competence of the person laid off.

The evidence on entrants has become more worrying, and still needs careful reading. In the August 2026 revision of “Canaries in the Coal Mine?”, Erik Brynjolfsson, Bharat Chandar and Ruyu Chen report, from American payroll data, that employment of workers aged 22 to 25 in the most AI-exposed occupations stands 19 percent below where it would be had it kept pace with their less-exposed peers, mostly through less hiring rather than more firing. They call these early, descriptive signals, not causal estimates, and note that the pattern weakens when education is controlled for.[40] This is not a claim that AI fired a fifth of young workers. In Denmark, Anders Humlum and Emilie Vestergaard linked surveys of chatbot use to administrative records and found precise null effects on earnings and recorded hours two years after ChatGPT, with tasks changing all the same.[41] Different countries, periods and measures need not tell the same story. Neither result makes the other disappear.

For training, fewer entry opportunities matter even when they come through fewer hires rather than spectacular firings. A ladder can become harder to climb because someone removed its bottom rung without pushing anyone off the top. And the rung is a delay line. Cut junior hiring and the saving shows at once, while the seniors of the profession do not move for years; then the missing cohorts arrive, or rather fail to, and the hole travels through the profession for longer than a career: from the first missing cohort, due in year five, to the retirement of the last (Fig. 2).

FIG. 2 **Halve junior hiring for ten years: the senior shortfall begins five years later, peaks at one sixth, and disappears only in year 45.**



On the page, move the cut, how long it lasts and how long an apprenticeship takes; in print, half the hiring for ten years and five years from junior to senior, over fifty years. An illustrative model of a profession, not a forecast: the same number of juniors hired every year, every junior becoming a senior after the apprenticeship, every senior staying thirty years; no attrition, no hiring from elsewhere, no change in demand. Its only claim is the shape of the delay.

Built

There is another change, inside a stable job title. The person best equipped for the work may change when execution becomes much easier. Suppose a product role once rewarded the ability to coordinate specialists, obtain scarce implementation time and carry a decision across several handoffs. Those skills remain useful. But if one person can now produce and test a credible prototype in an afternoon, other capabilities gain weight: deciding what to try, recognizing a poor result, connecting a customer's problem to a technical possibility, and sustaining several rounds of judgment without waiting for a meeting to authorize the next thought. This is my hypothesis about the changing A-player: AI can change the relative value of capabilities inside a job before anyone rewrites its title. It does not follow that experienced employees are obsolete, or that hiring a more energetic personality solves the problem. Many of the newly valuable capabilities can be developed; some were simply suppressed by the old structure of the work.

Anthropic's report on employment at Anthropic, in October 2026, gives a concrete and limited example. Hiring managers increasingly look for autonomy and judgment rather than execution and narrow expertise; the junior share of technical hires fell from 27 to 9 percent between spring 2025 and mid-2026, while it held in non-technical teams; and in an internal survey, about a third of machine-learning managers and a fifth of software

managers said they were hiring less because of AI.[42] These are observations from one fast-growing AI company, partly self-reported; they cannot establish a trend across the economy or isolate AI as the cause. The useful question for another organization is whether its own work is changing the same way. Which capabilities have become bottlenecks? Who has been given the tools, the permission and the time to develop them? Do the hiring criteria describe a real requirement, or the habits of whoever adopted AI first?

There is a self-defeating possibility here. A firm decides that AI makes junior execution unnecessary and autonomous judgment essential, and hires only people who acquired judgment somewhere else. That can be rational for one employer over a short horizon. It is a poor theory of how a profession reproduces itself. We can all buy experienced judgment for a while. Someone has to finance its creation.

The missing recovery interval

Access to opportunities is not enough if every minute of them becomes another delivery. People keep learning only where the work leaves room for it, and that is the second reason to look hard at the new ideal employee: the work may become more intellectually concentrated. Before AI, a difficult decision was often followed by a slower stretch: implementing it, preparing an artefact, waiting for another team, checking routine details. Some of that interval was frustrating waste. Some of it let attention change pace before the next demanding decision.

Now the result comes back in minutes and asks for another judgment. Then another. A person can go from choosing what to build, to judging the prototype, to diagnosing its failure, to deciding the next experiment, with very little lower-intensity work in between. This is where running helps me think. Interval training depends on the recovery as much as on the effort; remove the recovery and the same session becomes a different demand. The analogy is about the organization of effort, not a claim that thinking produces a runner’s lactate curve.

I have felt the intellectual version in long sessions with AI. Early on, the machine seems unusually perceptive. Later, its outputs flatten. The model has not necessarily changed; I am supplying less discrimination, fewer productive objections, less interesting direction. The system’s apparent intelligence was partly the quality of the attention I was bringing to it.

Research on work intensification gives the observation a wider frame. Aruna Ranganathan and Xingqi Maggie Ye spent eight months inside a 200-person American technology company, with more than forty interviews, and watched work expand as AI made more tasks seem possible: people took on tasks that used to belong to someone else, work spilled into breaks, several threads stayed open at once.[43] Nobody had mandated the tools. The intensification began with enthusiasm, and I recognize it: I can finally build the thing; I can also build the second thing; a third experiment costs so little that stopping feels irrational. The machine has reduced the cost of producing an option. It has not reduced the human cost of evaluating every option it makes possible. A 2026 survey by BCG researchers of 1,488 American workers associated heavy oversight of AI tools with more mental effort and fatigue, and other patterns of use with less burnout.[44] Self-reports and associations cannot diagnose anyone or prove a cause; they are a reason to ask how many concurrent streams of generated work one person is expected to supervise.

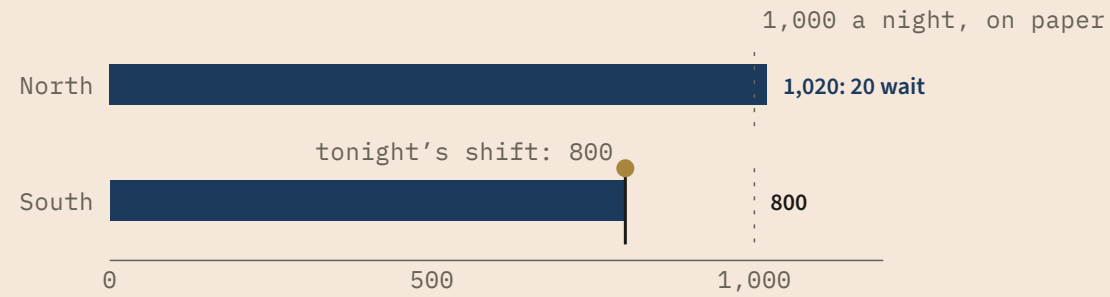
There is another trajectory. AI can remove exhausting administration, shorten a working day and make room for deeper thought. The technology does not decide who receives the saved time; targets, staffing and managers do. An organization can spend every saved hour on throughput, or keep some of it for learning, investigation and recovery. Calling the second option inefficient assumes that the expertise behind tomorrow's decisions will renew itself for free. Nor should sustained intensity become the new definition of talent. Selecting only people who thrive on the fastest feedback loop may discard valuable kinds of thought: patient synthesis, the slow recognition of an anomaly, the idea that improves after a night away from the screen. The colleague with fewer instant answers may be the one who eventually asks why the whole team is solving the wrong problem.

Engineering an apprenticeship

If AI changes how experience is acquired, training becomes a design problem inside engineering itself. The previous essay argued that a simulation earns its keep when it lets us test a change, confront an assumption and learn from the gap between model and reality.[45] The same architecture can develop people: an environment in which the learner acts, sees consequences, revises a model and tries again.

Imagine an exercise built from a logistics network. Demand rises unexpectedly; a transfer hub fills; the optimizer proposes a plan that is mathematically attractive and operationally impossible, because one capacity assumption is wrong. Before seeing an agent's answer, the trainee records a prediction: where will the congestion appear, which intervention is likely to help, and what observation would change their mind? They may ask for evidence. The night then unfolds, and their reasoning, not only their final number, becomes something to discuss. On a second run the apparent bottleneck is a symptom, and changing a dispatch rule or the configuration makes the problem easier: the trainee must notice when optimizing the existing set-up matters less than changing it. On a third, the agent recommends a change confidently, and the trainee has to decide whether to trust it (Fig. 3).

FIG. 3 **The arithmetic works. Would you approve the plan?**



On the page, play the trainee: predict, ask if you wish, decide, then see what the night does; in print, one way through. A designed exercise with illustrative numbers, and a proposal for how training could be organized, not a measurement of what it teaches.

Hypothesis

The exercise is hypothetical. Its purpose is to specify what an apprenticeship could train: framing, diagnosis, conditional judgment, and the ability to revise a recommendation when the world changes. It should include enough technical practice for the learner to understand the mechanisms they are judging; we cannot teach supervision by removing all contact with the supervised work. A generator could create variants and a tutor could give hints, but a convincing story from a language model is not a validated simulation. The constraints, consequences and assessment criteria need grounding in real models, operational evidence and expert review; otherwise the learner becomes excellent at satisfying the machine that invented the problem. Daniel Kahneman and Gary Klein, who agreed on little else, agreed on this: intuition becomes reliable only where the environment is regular enough to learn and the feedback comes fast enough to learn from, and confidence is no evidence that either was true.

[46] A training simulation should be judged by the quality of the link between the decisions it invites and the consequences it returns. And the exercise should leave the screen: an operator explaining why an available vehicle is not available, a customer explaining why a small delay matters, a colleague challenging a definition the trainee did not know they had accepted.

Assessment then needs several views of competence. In one session the learner works with the best tools available, because we want to know what they can achieve where they will actually work, and they are rewarded for using help intelligently, including asking another person. In another they explain a mechanism, predict an outcome or diagnose a small case without generative help: not a return to an obsolete examination culture, but a way of knowing which capabilities the organization can expect them to carry into an unfamiliar situation, with accessibility support kept in place. In a third, the AI is wrong: sometimes obviously, sometimes plausibly, sometimes because the problem is underspecified. Can the learner find the missing assumption, ask for a discriminating test, refuse to certify what cannot be known? Can they also recognize when the AI is right with an approach they find unfamiliar? That last question matters: training people to distrust every output would create a different incompetence. We want calibrated reliance. And a later assessment should change the surface of the problem: new names, another domain, a shifted constraint. If the learning disappears with the familiar cues, the organization has learned something valuable about its training, however good the original exercise looked.

These are design principles, not a validated package, and they deserve testing against sensible alternatives: ordinary AI-assisted work against a structured apprenticeship over several months, measuring delivered quality, time, unaided retention, transfer to unfamiliar cases and the quality of assisted decisions; for seniors, whether the range of approaches considered is widening and whether outside perspectives actually change decisions. No single creativity score should pass for a measure of a mind; the aim is to see the difference between producing more and becoming more capable, not to give every employee a psychological barcode. Others are already proposing structures: Mark Russinovich and Scott Hanselman have argued that software should borrow the preceptorship of nursing, in which an experienced practitioner is formally responsible for a newcomer’s first months.[47]

All of it needs protected time and the right incentives. If a junior loses credit for pausing to learn while a colleague is rewarded for delegating

everything at once, the instructional message is clear; a training policy written in a handbook will lose to the promotion system running beside it. Seniors need protection too: reviews across disciplines, contact with users, joint investigations, the responsibility of teaching a concept to someone else. AI should prepare these encounters and make more of them possible, not replace them by default because it can generate a plausible conversation. The goal is to make renewal part of production. A well-designed week should leave behind useful artefacts and better-equipped people, and we should be able to say where each is supposed to come from.

Who pays for the next engineer?

The last distinction may be the most important: between an individual success and a system that can sustain it.

I can thrive with AI because a great deal of my education has already been paid for. JPL, a doctorate, universities, employers, colleagues, failed projects and years of curiosity all contributed to the experience I now put to work. Some of that education was excellent. Some was accidental. Some was far more expensive than it needed to be. AI lets me draw on it with extraordinary efficiency, a gain I have no wish to give back. But an organization looking at my output cannot conclude that the earlier process has become unnecessary. It is observing a person after the investment. Charity Majors put it plainly two years ago: “Software is an apprenticeship industry,” and it takes seven years or more to forge a competent engineer.[48]

The future apprenticeship may be shorter in some ways, richer in others, and very different from mine. A learner might meet more well-chosen failures in a month than an earlier generation met in a year; a good tutor might correct a misconception before it becomes a career; a good simulator might make a rare event available for practice without anyone living through its consequences. Those are reasons for ambition, and reasons to evaluate the new system seriously. Removing old tasks does not automatically produce a better path, any more than old tasks deserve preservation because they are old. The same holds for seniors. “Already trained” is too final a phrase. Expertise is maintained through activity; it is not a certificate placed permanently inside a person.

If every company becomes a buyer of finished judgment and none takes responsibility for making it, the apparent efficiency hides a missing cost.

An organization can become better at using expertise while becoming worse at producing it. Every hour the machine saves is spent twice, once on the work and once, or never, on the person; the machine can help us produce the work, and only we can decide to reproduce the profession. Universities cannot supply all of it, and individuals cannot be expected to rebuild, in unpaid evenings, every learning opportunity removed from their working day. The question is economic and institutional as much as educational. Which organizations will fund apprenticeships? Which will count mentoring as work? Which will measure the capability they leave behind beside the output they extract? And which will discover, too late, that the experienced people they meant to hire were never given a place to become experienced?

This volume opened in Seoul, with a move no human would have played and a champion who retired three years later because, he said, there was an entity that could not be defeated.[49] The players who came after him did not retire. They studied with it. Analysing nearly six million decisions by professional players from 1950 to 2021, Minkyu Shin and colleagues found that human play began to improve after superhuman programs arrived, and that it improved through novelty: players began to make moves nobody had made before, and those moves became better, not worse.[50] Other studies add the two cautions this essay has been carrying. Young players improved most; and the diversity of moves narrowed as the new knowledge spread.[51][52] More novelty and less diversity sound like a contradiction until you see that they measure different things: novelty against the past, diversity among the present. A whole generation can leave an old tradition together and converge on a new one. That is the senior's risk from the colleagues section, at the scale of a game: a repertoire can be renewed and still narrow. The machine did not end the apprenticeship of Go. It became part of it, for better and for sameness, and the players who learned most were the ones who used it to go somewhere new.

The Iron Man suit will keep improving. It may increasingly help design its successor, and I welcome that. I still do not know TypeScript, and I have decided that I should know a little of it: not enough to write it alone, enough to predict where the next bug will be before the machine shows me.

The volume began by asking whether we can remain responsible for engineering we no longer perform directly. Its answer is a condition: only if people keep developing what it takes to understand the consequences, investigate the failures and redirect the work, which means a good deal of

real technical practice, though not repeating every operation we delegate. What kind of working world will produce the engineer who decides what the next suit is for?

Sources

Each source says what it can carry: a peer-reviewed study, a preprint not yet reviewed, a book, a research organisation, an institution, a company's own announcement or report, code, or a press report. Facts quoted here were read at the source in October 2026; each entry gives its own date.

[1] Christophe Pennetier. The Iron Man Paradox. Aquia, Collected, 12 August 2026. Author's essay.

[2] Andrej Karpathy. "There's a new kind of coding I call 'vibe coding'…". X, 2 February 2025. Post.

[3] Anadolu Agency. Collins English Dictionary chooses 'vibe coding' as 2025 word of the year. Anadolu Agency, 6 November 2025. Press report.

[4] Christophe Pennetier and André-Benoît De Jaegere. Education and Training of Future Engineers. Generative Engineering, the original study, written November 2024 to February 2025. The original study.

[5] Erik Brynjolfsson, Danielle Li and Lindsey Raymond. Generative AI at Work. Quarterly Journal of Economics 140(2), 889–942, doi:10.1093/qje/qjae044, 2025. Peer-reviewed study.

[6] Joel Becker, Nate Rush, Elizabeth Barnes and David Rein, METR. Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity. METR; arXiv 2507.09089, 10 July 2025. Research organisation.

[7] Joel Becker, Nate Rush, Tom Cunningham, David Rein and Khalid Mahamud, METR. We are Changing our Developer Productivity Experiment Design. METR, 24 February 2026. Research organisation.

[8] Nicholas C. Soderstrom and Robert A. Bjork. Learning Versus Performance: An Integrative Review. Perspectives on Psychological Science 10(2), 176–199, 2015. Peer-reviewed study.

[9] David Autor, Tanya Rodchenko, Josh Martin, Zanna Iscenko, Scott Strand, David Pearl and Melissa Ferere. Does AI Assistance Enhance or Erode Expertise? Evidence from a Three-Month Field Experiment in Patent Drafting. NBER Working Paper 35720 (run and funded by Google, with its own tool), September 2026, revised October 2026. Working paper.

[10] David Autor and Tanya Rodchenko, Google Research. Does better work always mean better workers?. Google Research blog, 7 October 2026. Company research note.

[11] Christophe Pennetier. AI Needs a Boss, Not a Programmer. Aquia, Collected, 5 March 2024. Author's essay.

[12] Matt Beane. Shadow Learning: Building Robotic Surgical Skill When Approved Means Fail. Administrative Science Quarterly 64(1), 87–123, 2019 (online January 2018). Peer-reviewed study.

[13] Matt Beane. The Skill Code: How to Save Human Ability in an Age of Intelligent Machines. Harper Business, 11 June 2024. Book.

[14] Lisanne Bainbridge. Ironies of automation. Automatica 19 (6), 775–779, 1983. Peer-reviewed study.

[15] Judy Hanwen Shen and Alex Tamkin. How AI Impacts Skill Formation. arXiv 2601.20245, January 2026. Preprint (not yet peer-reviewed).

[16] Hamsa Bastani, Osbert Bastani, Alp Sungu, Haosen Ge, Özge Kabakcı and Rei Mariman. Generative AI without guardrails can harm learning: Evidence from high school mathematics. PNAS 122(26), e2422633122 (affiliation corrected, PNAS 122(34)), 25 June 2025. Peer-reviewed study.

[17] John Dunlosky, Katherine A. Rawson, Elizabeth J. Marsh, Mitchell J. Nathan and Daniel T. Willingham. Improving Students' Learning With Effective Learning Techniques: Promising Directions From Cognitive and Educational Psychology. Psychological Science in the Public Interest 14(1), 4–58, 2013. Peer-reviewed review.

[18] Allan Collins, John Seely Brown and Ann Holum. Cognitive Apprenticeship: Making Thinking Visible. American Educator 15(3), 6–11, 38–46, Winter 1991. Journal article.

[19] Tanmay Sinha and Manu Kapur. When Problem Solving Followed by Instruction Works: Evidence for Productive Failure. Review of Educational Research 91(5), 761–798, 2021. Peer-reviewed study (meta-analysis).

[20] Paul A. Kirschner, John Sweller and Richard E. Clark. Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching. Educational Psychologist 41(2), 75–86, 2006. Peer-reviewed position paper.

[21] Tian Heong Chan, Jürgen Mihm and Manuel E. Sosa. Revisiting the Role of Collaboration in Creating Breakthrough Inventions. Manufacturing & Service Operations Management 23(5), 1005–1024, 2021 (online 2020). Peer-reviewed study.

[22] Fabrizio Dell'Acqua and colleagues. The Cybernetic Teammate: A Field Experiment on Generative AI and Teamwork. Organization Science 37(4), 1217–1242, 2026. Peer-reviewed study.

[23] Charlan Nemeth, Brown and Rogers. Devil's advocate versus authentic dissent: stimulating quantity and quality. European Journal of Social Psychology 31(6), 707–720, 2001. Peer-reviewed study.

[24] Anil R. Doshi and Oliver P. Hauser. Generative AI enhances individual creativity but reduces the collective diversity of novel content. Science Advances 10(28), eadn5290, 12 July 2024. Peer-reviewed study.

[25] Ilia Shumailov and colleagues. AI models collapse when trained on recursively generated data. Nature 631, 755–759 (author correction, Nature 640, E6, 2025), 24 July 2024. Peer-reviewed study.

[26] Matthias Gerstgrasser, Rylan Schaeffer and colleagues. Is Model Collapse Inevitable? Breaking the Curse of Recursion by Accumulating Real and Synthetic Data. Conference on Language Modeling (COLM) 2024; arXiv 2404.01413, 2024. Peer-reviewed conference paper.

[27] James G. March. Exploration and Exploitation in Organizational Learning. Organization Science 2(1), 71–87, 1991. Peer-reviewed study.

[28] Benjamin S. Bloom. The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. Educational Researcher 13(6), 4–16, 1984. Journal article.

[29] Greg Kestin and colleagues. AI tutoring outperforms in-class active learning: an RCT introducing a novel research-based design in an authentic educational setting. Scientific Reports 15, 17458, 3 June 2025. Peer-reviewed study.

[30] Martín De Simone and colleagues. From Chalkboards to Chatbots: Evaluating the Impact of Generative AI on Learning Outcomes in Nigeria. World Bank Policy Research Working Paper 11125, 19 May 2025. Working paper.

[31] Rose E. Wang and colleagues, with Susanna Loeb and Dora Demszky. Tutor CoPilot: A Human-AI Approach for Scaling Real-Time Expertise. arXiv 2410.03017, version 3, October 2026 (first version October 2024). Preprint (not yet peer-reviewed).

[32] OpenAI. Introducing study mode. OpenAI, 29 July 2025. Company announcement.

[33] Anthropic. Introducing Claude for Education. Anthropic, 2 April 2025. Company announcement.

[34] Khan Academy. How Khan Academy is building a better AI tutor: our most recent learnings. Khan Academy blog (internal product tests), 1 May 2026. Company report.

[35] Jing Liu and colleagues, University of Maryland. The Effects of Course-Integrated AI Tutoring on Student Performance and Engagement: A Randomized University Trial. EdWorkingPaper 26-1598, October 2026. Working paper.

[36] Barry J. Zimmerman. Becoming a Self-Regulated Learner: An Overview. Theory Into Practice 41(2), 64–70, 2002. Journal article.
[37] Alex Bell, Raj Chetty, Xavier Jaravel, Neviana Petkova and John Van Reenen. Who Becomes an Inventor in America? The Importance of Exposure to Innovation. Quarterly Journal of Economics 134(2), 647–713, 2019. Peer-reviewed study.
[38] Mark Zuckerberg. Mark Zuckerberg's Message to Meta Employees. Meta, 9 November 2022. Company announcement.
[39] Andy Jassy. Message from CEO Andy Jassy: Some thoughts on Generative AI. Amazon, 17 June 2025. Company announcement.
[40] Erik Brynjolfsson, Bharat Chandar and Ruyu Chen. Canaries in the Coal Mine? Six Facts about the Recent Employment Effects of Artificial Intelligence. Stanford Digital Economy Lab working paper, revised, 12 August 2026. Working paper.
[41] Anders Humlum and Emilie Vestergaard. Still Waters, Rapid Currents: Early Labor Market Transformation under Generative AI (first circulated as "Large Language Models, Small Labor Market Effects"). NBER Working Paper 33777, revised March 2026. Working paper.
[42] Wilmers, Catipon, Massenkoff and McCrory, Anthropic. How employment is changing at Anthropic. Anthropic, 16 October 2026. Company report.
[43] Aruna Ranganathan and Xingqi Maggie Ye. AI Doesn't Reduce Work—It Intensifies It. Harvard Business Review, 9 February 2026. Magazine article (qualitative study).
[44] Julie Bedard, Matthew Kropp, Megan Hsu, Olivia T. Karaman, Jason Hawes and Gabriella Rosen Kellerman. When Using AI Leads to "Brain Fry". Harvard Business Review (a BCG survey of 1,488 US workers), 5 March 2026. Magazine article (survey).
[45] Christophe Pennetier. When the Engineering Loop Closes. Generative Engineering, Volume II, Essay 03, October 2026. Volume II essay.
[46] Daniel Kahneman and Gary Klein. Conditions for intuitive expertise: a failure to disagree. American Psychologist 64(6), 515–526, 2009. Peer-reviewed study.
[47] Mark Russinovich and Scott Hanselman. Redefining the Software Engineering Profession for AI. Communications of the ACM, April 2026, doi:10.1145/3779312 (as reported by InfoQ, 27 April 2026), April 2026. Journal article, read through press coverage.
[48] Charity Majors. Generative AI is not going to build your engineering team for you. Stack Overflow blog, 10 June 2024. Blog post.
[49] The Next Web, after Yonhap. World champion Go player retires: AI 'cannot be defeated'. The Next Web, 27 November 2019. Press report.
[50] Minkyu Shin, Jin Kim, Bas van Opheusden and Thomas L. Griffiths. Superhuman artificial intelligence can improve human decision-making by increasing novelty. PNAS 120 (12), e2214840120, 2023. Peer-reviewed study.
[51] Choi, Kang, Kim and Kim. How does artificial intelligence improve human decision-making? Evidence from the AI-powered Go program. Strategic Management Journal 46(6), 1523–1554, 2025. Peer-reviewed study.
[52] Choi, Kang, Kim and Kim. The dual edges of AI: Advancing knowledge while reducing diversity. PNAS Nexus 4(5), pgaf138, 3 May 2025. Peer-reviewed study.