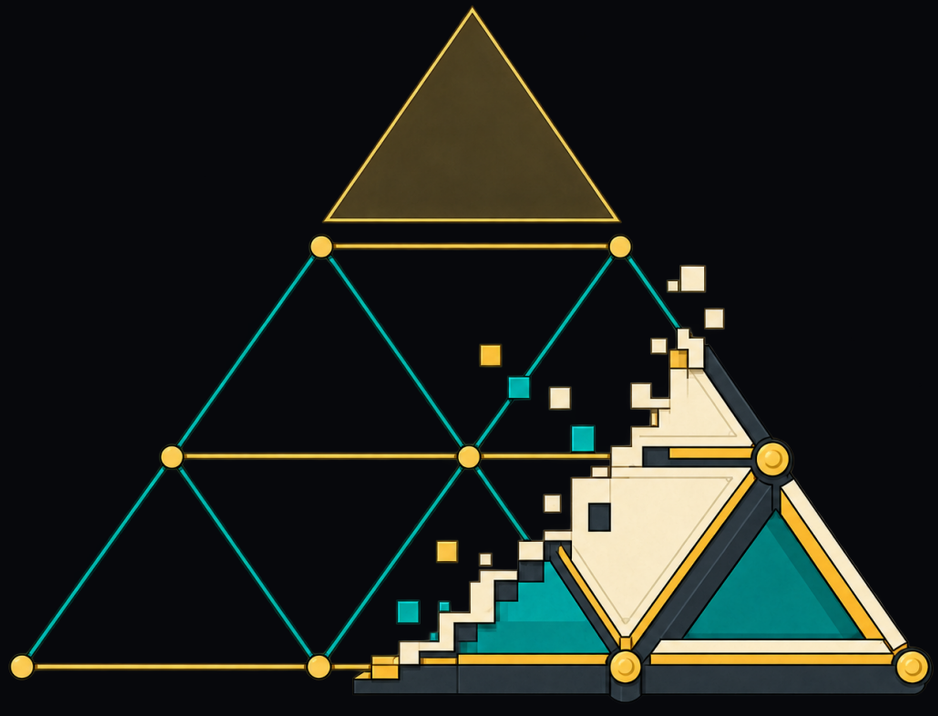
Artificial *Intent*

A Handbook for Thinking with AI



Micah Robertson

Artificial Intent

A Handbook for Thinking with AI

Copyright and Use

© 2026 Micah Robertson. All rights reserved.

Rights and Permissions

Copyright is claimed only in the author's original copyrightable expression. The claim doesn't cover ideas, methods, facts or material owned by others.

Where the law allows you to use protected material without permission, this notice doesn't limit that use. Otherwise, get written permission from the relevant rights holder before copying, sharing or adapting protected material.

Using This Book

This book is for education. An example doesn't give you permission to use someone's information, act for an organization or make a commitment. Those decisions still belong to the people with the relevant authority.

Research and Examples

The examples explain decisions and possible comparisons. They draw on everyday situations where people use AI.

Intent Engineering is the working definition used in this book. The research in Key Sources supports particular findings in its own settings. It does not validate every recommendation here as one complete method. The practical comparisons and questions are the book's synthesis, and their usefulness in real work needs evidence from the people and tasks involved.

Contents

A Handbook for Thinking with AI

Before the Prompt 4

I. Desire 9

II. Assumption 15

III. Reassurance 20

IV. Permission 26

V. Agreement 32

VI. Number 37

VII. Attention 43

Coda · The Promise 49

Key Sources 52

Before the Prompt

Most prompts start with an unfinished thought. You have a loose idea, a half-baked plan, or a question you have not fully sorted out yet, and you hand it to software built to return finished work.

You give it fragments and trade-offs. It gives back complete paragraphs, clean lists, and confident plans. Somewhere in that handoff, the tone shifts. The model fills in whatever you left unsaid, smooths over the parts you were still unsure about, and returns an answer that looks far more settled than you feel.

Nothing has to go wrong technically for this to steer you off course. The answer can be fast, accurate, and well written, and still miss what you actually meant. Once something looks finished on screen, pushing back on it feels like breaking good work.

People often tack three words onto the end of a prompt: *Make no mistakes.* It makes sense why they type it, even though the model cannot follow it. An AI does not know what counts as a mistake when your prompt left out what matters most to you. What it can do is strip out every visible trace of doubt.

A mirror works the same way. It shows whatever sits inside its frame in sharp detail and says nothing about anything outside the edges. Paste rough notes

or a half-built plan into a chat, tell it to make no mistakes, and the reply comes back confident and smooth. The hesitation that belonged in the original thought disappears. Putting doubt back in now feels awkward, and if you forward that draft to someone else, they assume you were sure from the start.



A mirror stays faithful inside its frame and silent about everything outside its edges.

Sometimes adding your uncertainty back takes one sentence. Other times it shows you that the draft only looked convincing because it borrowed certainty you never had.

Giving a rough thought a finished shape also changes what happens next. Turn a loose note into a schedule, an email, or a rule inside a tool, and other

people can act on it right away. While a draft stays on your screen, adding a missing limit takes a minute. Once someone else rearranges their calendar, spends money, or commits their team around it, bringing up that missing detail feels like undoing work they already trusted.

Say you dump messy notes into a chat on Friday, turn them into a clean plan, and send it out with a line inviting feedback next week. Even with that invitation at the bottom, people start acting on the plan Monday morning. Anyone who spots a flaw now has to interrupt work that is already moving. If you wanted real input, the question had to go out before Monday.



Feedback invited after people start acting arrives too late to steer the plan.

AI compresses this kind of drift from months into minutes. One reply becomes the starting point for the next prompt. A quick suggestion turns

into an outline, then a finished email, a script, or a standing habit that other people end up living with even though they never saw the original prompt. You approved every step along the way, yet the main decision happened without anyone noticing when.

That is how a person or a team ends up following a rule nobody remembers choosing. Someone would gladly change it if they could see where it came from and how it got baked into the routine. Staying in charge means catching shortcuts before they harden into defaults, and being clear on who has to explain a decision, who can push back when a system gets it wrong, and whether the system changes course when something important shifts.

Keeping everything open forever does not work either. A clear commitment lets other people plan their week around you. Reopening every decision whenever a new thought pops up makes you impossible to rely on. Good closure is what lets people finish work and move on. The hard part is telling when a question is genuinely settled and when it still needs room to change.

Intent Engineering is the practice of giving an idea enough shape to act on while keeping it open to corrections that matter. For any draft, plan, or automated step, it asks two things: how much weight is this output carrying, and where can a serious objection still change what happens next?

The Seven Principles

1. **Desire.** The options you see can change what you want.
2. **Assumption.** A guess can travel as a decision.
3. **Reassurance.** Relief can arrive before the evidence does.
4. **Permission.** Trust the result as far as the test goes.
5. **Agreement.** Some answers belong to the people who have to live with them.
6. **Number.** What you count can change what people have to do.
7. **Attention.** A decision can be open on paper and closed by the clock.

All seven fit on one page so you can recall them when you are moving fast. Their first letters spell **DARPANA**, the Sanskrit word for a mirror, and the traditional name for a short handbook that holds a glass up to a craft. There is nothing mystical about it. An AI reflects whatever sits inside the prompt and stays silent about everything left out of it.

A handbook is only useful if its ideas hold up when you are tired, rushed, or tempted to hit send just to clear your desk. No general rule beats knowing your own situation, and sometimes the right move after checking an output is to leave it alone and go.

Telling a machine to make no mistakes protects nothing. Before you send a finished draft out into the world, ask a better question.

What could still change what happens next?

I. Desire

The options you see can change what you want.

Most people do not open an AI chat knowing exactly what they want. You show up with a packed week, a deadline, or a vague urge to get unstuck. The model hands back a tidy plan, and because the plan makes sense on screen, it starts to feel like what you wanted all along.

Say you are behind on work and dump your task list into a chat asking for the best way through the week. The model comes back with a tight schedule that fits every single item in without a wasted minute. On its own terms, nothing is wrong with the answer. It just assumed your goal was to finish everything on the list as fast as possible.

Now put a different plan next to it: drop the least important task, push one item to next week, and leave your evenings open for the one piece of work that matters most.

On a raw to-do list, pushing work off looks less efficient. Against your energy at the end of a long week, protecting your focus is obviously the better call. You feel the difference the moment that second plan appears, because seeing another direction makes the trade-off concrete.

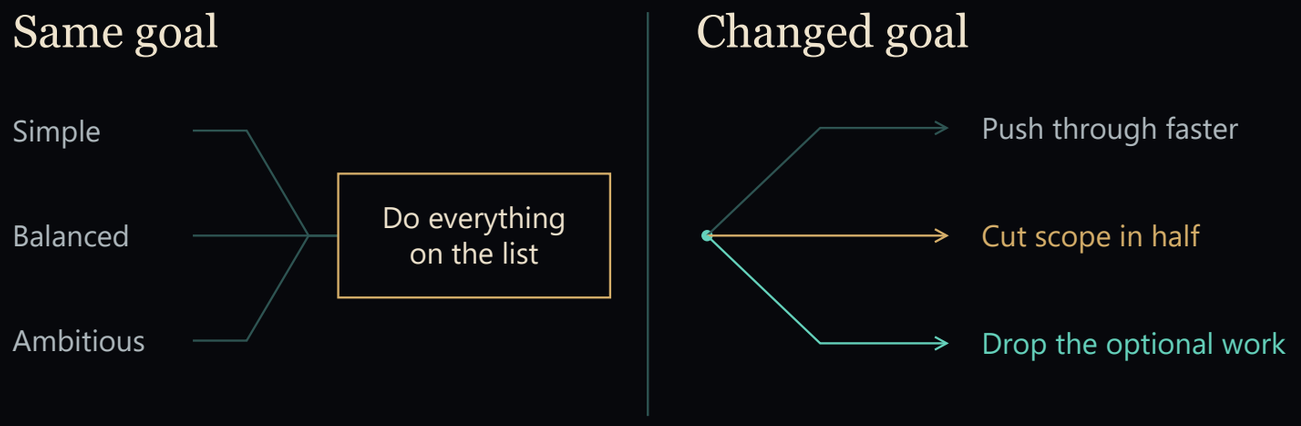


A tidy plan makes one path look obvious while hiding what you gave up to take it.

Same goal, three outfits

Asking an AI for "three options" often makes the problem worse. You ask for three ways to handle the week, or three ways to pitch a project, and the model labels them *Simple*, *Balanced*, and *Ambitious*. Each one reads smoothly and looks distinct. Look past the labels, though, and all three assume the same goal: saying yes to the project, keeping every item on the list, and pushing forward.

You are looking at the same decision in three different outfits. Real choice only shows up when you let the goal move.



Three polished options built on one hidden goal still add up to a single choice.

Keep the same facts and change the objective. Ask for one plan that finishes everything quickly, and a second plan that cuts the scope in half so you can do it well without burning out. A third option rarely enters the chat unless you bring it up: dropping the optional piece altogether because it is not worth the trouble. Hand a model a list and ask how to do it, and it treats every line as mandatory. Sequencing a list is a completely different question from deciding what belongs on the list in the first place.

Seeing two different goals side by side also helps you set a plain rule ahead of time. Once you know what pushing through costs compared to scaling back, you can decide upfront: if the main task wraps up by afternoon, tackle the extra item; if it drags, leave the rest for tomorrow.

The same pattern shows up in bigger decisions. Ask for career options, and three paths aimed at bigger titles and higher pay can hide a move built around learning a craft, lower stress, or more time with family. Ask for product ideas, and three plans built around expansion can hide the simpler move of fixing what already frustrates current users. Each option reads well,

so it feels like you covered all the ground even though the target never moved.

Choosing on purpose

If an AI suggests an angle you had not thought of and it clicks, nothing is wrong with taking it. People figure out what they care about by bouncing ideas around and reacting to concrete options. What matters is whether you can explain in your own words why you picked this direction, and whether another reasonable direction was on the table.

Once you make a deliberate call, let it hold. Deciding to drop an extra task or move it to next week settles the argument so you are not revisiting the same choice an hour later. Firm decisions keep every open hour from turning back into a fresh debate.

Living with the choice gives you feedback no chat can predict. If you scaled back a plan to protect your weekend and came into Monday rested, you learned something useful about what you needed. If leaving that task undone kept nagging at you all Sunday, that teaches you something too. Sort out your options on screen, then pay attention to how they feel once you live with them.

When memory hardens into a rule

Many AI tools now save preferences across chats. Over weeks, that saves you from repeating yourself. Tell the model once that you like short replies, or that you prefer keeping evenings free, and it carries that forward automatically.

The catch is that a casual remark you made on one exhausted Tuesday is not a standing habit, and neither one is a firm commitment.

Suppose you told an assistant last month that you hate long meetings and want afternoons kept clear. Most days, that is helpful. Then a major deadline or a rare chance to meet someone important lands on a Thursday afternoon. A loose preference should step aside easily for an exception like that. A promise you made to your family for that exact Thursday should not.

A habit and a promise do not belong in the same bucket. Yet an AI memory list tends to store every saved note at equal weight, with no sense of context or expiration date. A throwaway comment from one stressful week turns into an invisible rule shaping your suggestions months later, long after you forgot it was there.



A throwaway preference from one tired week can harden into an unwritten rule when memory carries no context.

The same mix-up happens when you share an AI-generated plan with someone else. A teammate or partner looking at the finished schedule sees what made the cut and what got pushed aside. Nothing on the page shows which calls you made on purpose and which ones the tool filled in on its own.

Using AI well also means knowing when you already have your answer. When the goal is obvious and every piece is required, you do not need a long back-and-forth. Lay out your limits, make sure nothing important got dropped, and let the tool do the legwork. When the stakes are higher, or the plan commits your time, money, or reputation, take ten seconds to state what the plan is aiming for before you run with it.

In practice

- **Ask for different goals, not different flavors.** Keep the same facts and ask the model for two plans aimed at genuinely different outcomes, such as moving fast versus protecting quality or keeping scope small.
- **Look past the option labels.** Check what changes in time, effort, and trade-offs. If *Simple*, *Balanced*, and *Ambitious* all commit you to the same core path, you only got one option.
- **Check what got treated as mandatory.** Before accepting a plan, ask the model which items could be dropped, delayed, or simplified altogether.
- **Review your saved AI memories periodically.** Separate one-off moods from standing preferences, and keep old notes from overriding a real commitment or deadline.
- **Say what was chosen and what was assumed.** When you hand an AI-assisted plan to someone else, tell them which trade-offs you decided on purpose and which parts are still rough placeholders.

II. Assumption

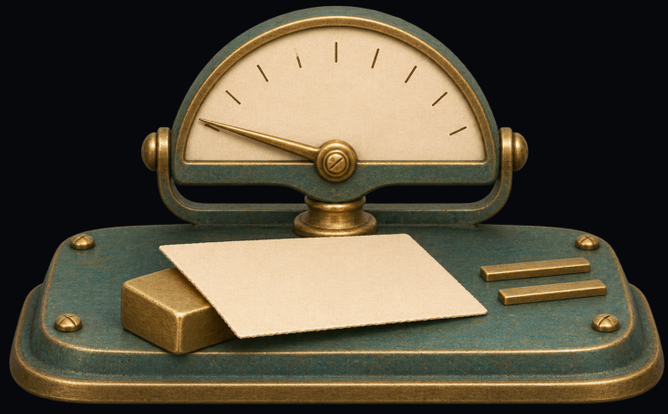
A guess can travel as a decision.

Ask an AI to compare two or three real choices, like a pair of job offers, two apartments, or tools for a project. You paste in the details and ask which one is the best fit.

Seconds later, you get a brisk summary, a list of pros and cons, and a confident pick at the bottom. Every fact you pasted in is right. The tone sounds calm and objective.

To pick a winner, though, the model had to guess how much each factor mattered to you. Did price count for almost everything, or did reliability matter more? Was a short commute worth higher rent? Was long-term growth worth taking lower pay today? Unless you spelled those trade-offs out, the model invented them.

A guess is harmless while everyone knows it is a guess. The problem starts when the model's invented priorities sit right next to your hard facts in the same confident tone. An arbitrary weighting starts to look just as solid as a price tag or a date.

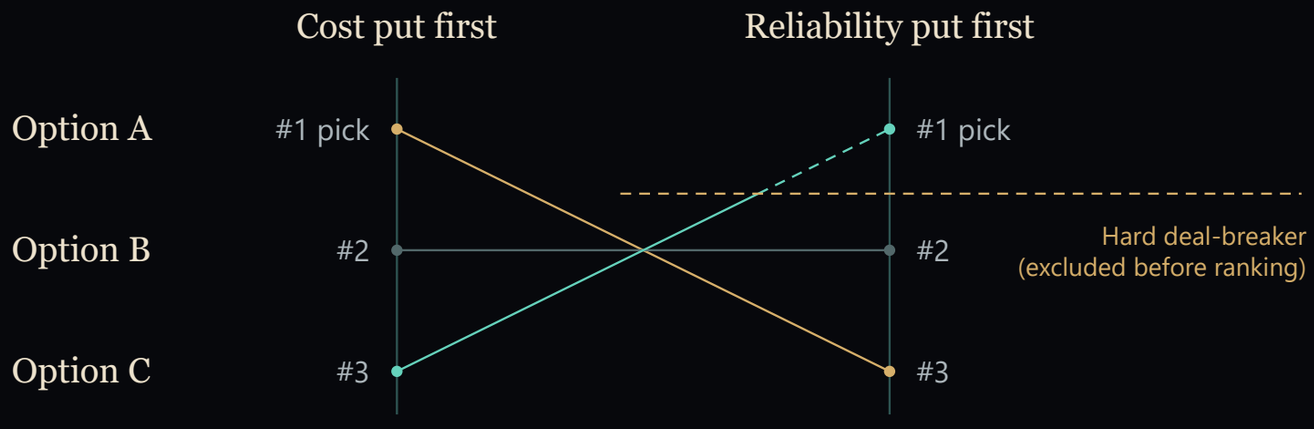


An unchosen priority sits quietly under a confident recommendation.

What hides inside a comparison

Change how much you care about cost versus peace of mind, or speed versus durability, and the winner flips even though not a single fact about the options changed. When a recommendation looks obvious on screen, it is usually because one hidden priority is doing almost all the heavy lifting.

AI comparisons also blur the line between a trade-off and a deal-breaker. A hard limit removes an option before comparison starts. If you cannot afford rent above a certain number, or if a tool lacks a feature you legally need, glowing marks in other areas cannot rescue it. Models routinely let strong scores on secondary features nudge you past a boundary you meant to hold firm. You can always decide to stretch your budget or bend a rule yourself, but a comparison should not slide that decision past you.



Changing one hidden priority flips the winner, while a deal-breaker removes an option before ranking starts.

Vague labels hide a third layer of guesswork. Words like *reliable*, *good support*, *strong culture*, or *easy to use* sound concrete until you press on them. Does *good support* mean someone answers the phone fast, or that a company replaces broken gear without a fight? Does *strong culture* mean

friendly coworkers, or reasonable hours? Compressing messy reality into a single label or a score out of ten wipes those differences out.

This matters even more once a comparison leaves your screen. You put together a crisp summary for a family purchase or a team decision, share it, and people nod at how clear it looks. Before long, everyone treats the winner as settled, and no one remembers that the ranking started with a casual prompt and priorities nobody agreed on.

The hollow second opinion

When you want to be careful, the natural move is to check the answer in a fresh chat. You copy the summary, paste it into a new window or another AI app, and ask whether the recommendation makes sense.

The second reply sounds balanced and thoughtful. It notes a few drawbacks of the top pick, mentions the strengths of the runner-up, and lands on the same winner. Seeing two separate chats or two different models agree feels reassuring, yet most of the time that agreement proves nothing.

When you paste the first chat's summary into a second chat, the second model inherits all the framing, hidden priorities, and clipped details baked into it. Confirming that Option A still wins under Option A's own framing only checks consistency. Even switching to another AI company adds fresh wording without fresh judgment if both systems are reading the same pre-cooked write-up.

A real second opinion means going back to the source. If you want a second opinion on a home repair, you bring another contractor out to look at the house, not to read the first contractor's quote and nod along.

Do the same with AI. Open a fresh chat, paste in the raw facts and your hard deal-breakers, and leave out any hint of which option won before. Better yet, ask the model to argue for a different priority altogether and see what shifts. When both runs point the same way from the raw facts, you learned something useful. When the winner flips as soon as one premise moves, you found the call you have to make yourself.

Share the reasons, not the transcript

Once you finish working through a choice with AI and need to share it with a partner, a client, or a team, it is tempting to drop the whole chat link or paste ten pages of dialogue next to your conclusion so everything is "transparent."

Almost nobody will read ten pages of chat history. A person with two minutes to review a choice does not want to dig through twenty back-and-forth prompts just to find out whether a budget limit was a firm rule or a rough guess.



Four clear lines of reasoning travel where ten pages of chat history never will.

Keep the trail short and visible right beside the conclusion. Note three things in everyday language: where the key facts came from, which limits are non-negotiable, and which priority you put first. Four lines will do it. Anyone reading your recommendation can see how you got there and push back on the specific premise they disagree with, without having to start over.

If you make the same kind of comparison more than once, keep your ground rules steady so earlier choices still make sense next to newer ones. When your priorities change, say so openly instead of acting as if last month's call was made under today's rules.

In practice

- **State your deal-breakers before comparing.** Tell the model upfront what disqualifies an option immediately so a high score somewhere else cannot sneak an unworkable choice back onto the table.
- **Ask what would flip the winner.** After any AI recommendation, follow up with one question: *What priority did you put first here, and what changes if I care more about the opposite?*
- **Unpack vague praise.** Whenever a comparison leans on fuzzy words like *value*, *quality*, or *reliability*, ask what concrete evidence sits behind them.
- **Test second opinions on raw facts.** Never paste an AI's summary into another chat and call it a double-check. Feed the new chat the raw inputs without revealing the previous winner.
- **Share four lines of reasoning, not a chat log.** When passing AI-assisted work to others, attach the core facts, hard limits, and main priority in everyday English so people can check them.

III. Reassurance

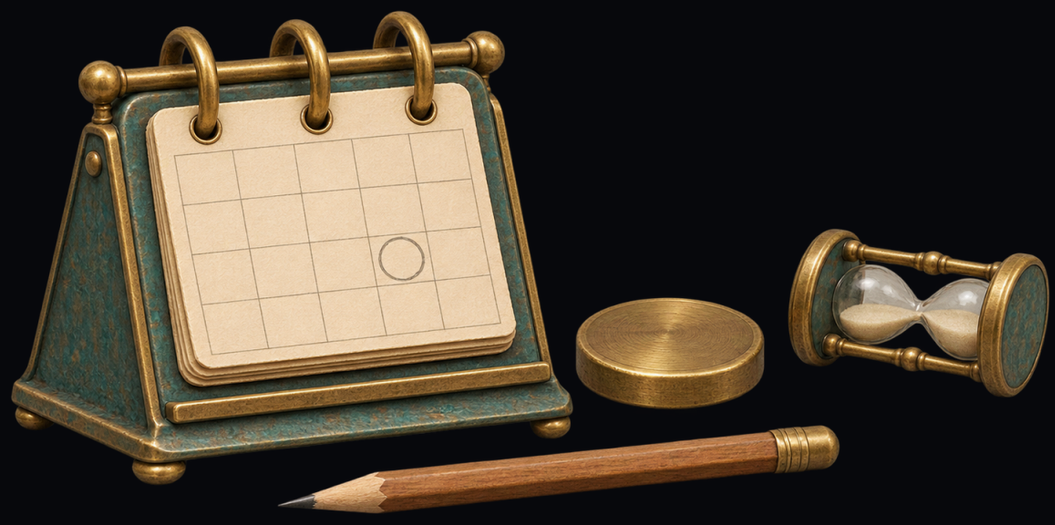
Relief can arrive before the evidence does.

You already have a gut feeling about what you want to do, but you are not sure it is smart. Maybe it is a big purchase you are trying to justify, or a sharp email you are itching to send. So you open a chat and ask what makes sense.

The first reply is balanced. It lays out the upside, flags the downside, and leaves the call open. That does not settle the knot in your stomach, so you reply again. This time you add a little more detail about why the downside probably will not happen, or why waiting feels annoying. The model adjusts to what you gave it and leans your way. One more back-and-forth, and suddenly you have three calm, articulate reasons to go ahead.

Look at what changed across those three messages. Nothing outside the chat moved. Nobody confirmed the plan, no new fact arrived, and the risk did not shrink. You just nudged the conversation until the model handed your own impatience back in a calmer tone.

By the third reply, you can already hear how you will defend the choice later. The model may still tack a polite caution onto the last paragraph, but that caution now reads like fine print on a done deal. You feel relieved even though the evidence never moved.



A steadier reply can calm the room even when nothing in the real world has moved.

Three replies that all feel calming

Very different AI replies can produce that same wave of relief, yet only some of them change what you know.

Sometimes the model simply **judges a risk it never has to live with**. It calls a gamble "totally reasonable" or tells you a tense email sounds "firm and fair." The machine has no bank account, no job on the line, and no relationship to repair if the message blows up an hour later. It read your tone and answered as though it knew what fallout you can afford.

Other times the model **uncovers a detail already sitting in your own material**. You paste in a contract, an email thread, or your own notes, and it catches a deadline, a cancellation clause, or a contradiction you skimmed right past. Nothing outside the chat changed, yet your decision shifted because something concrete in the record came into view.

And sometimes it **brings in outside evidence**. Connected to a live search, a database, or a document you did not have before, the model pulls in a verifiable fact from beyond the conversation. Even then, click through and check the primary source yourself before trusting a summary.

All three calm you down, but only the second and third changed what you know. Before you act on relief, make sure it came from something real.



Three replies can bring the same wave of relief while only two change what you know.

Talking yourself into it

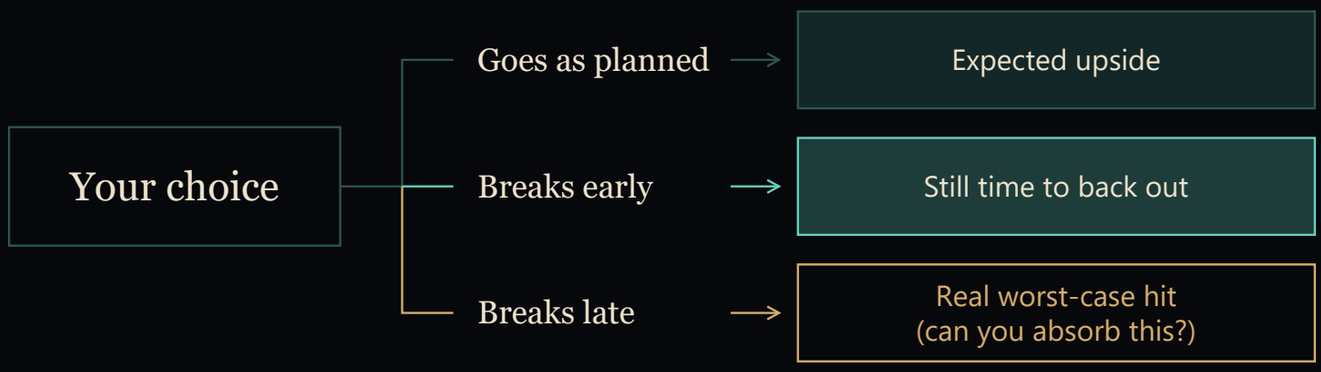
When you are uneasy, it is tempting to ask the same question three or four times in fresh chats or across different AI apps. If three out of four replies lean your way, it starts to feel like a vote. It is not. Running the same uncertain setup through four sessions produces zero new evidence. Hearing

the same reassurance phrased three ways only shows how easy your side is to argue, not whether the bet is safe.

A fresh reply can still help if it points out something you missed: a fallback plan if things go south, a hidden cost you forgot to count, or a direct question you can ask someone to clear up the doubt. Good reasoning is useful because you can check every step yourself. Pay attention to what you can verify, rather than how calming the reply sounds.

Scroll back and compare your first prompt with your fourth in any long thread where you were hunting for an answer. Most of the time, you can see the drift right there in your own messages. You started with an open question, and by the third turn you were emphasizing the upside and downplaying the catch. Nobody snuck a hidden assumption past you here. You were sitting at the keyboard steering the chat toward the answer you wanted, and the model followed your lead.

Instead of asking an AI whether your idea sounds safe, ask it to map out what happens next. Have it lay out three scenarios side by side: what happens if things go as planned, what happens if they break early while you can still back out, and what happens if they break late after you are locked in. Concrete outcomes cut through comforting talk fast. No chat knows what another person will do tomorrow, so it cannot give you true odds. It can lay out the worst case clearly enough for you to answer the question that counts: *If this goes wrong, can I absorb the hit?*



Mapping concrete outcomes replaces soothing language with the real risk you would carry.

Test the overly cautious side the same way. Sometimes people spend extra money or stall for weeks just to feel responsible, even when the "safe" choice does not protect them from the main danger. Taking a smart risk does not need a pep talk from a chatbot. It takes an honest look at what happens if you guessed wrong.

Knowing when to close the chat

Before you keep prompting on a tough call, decide what would settle it. Maybe you need a direct answer from one specific person. Maybe you need to check your bank balance, or sleep on it until morning. Use that as your test. Once the model is only dressing up the same known facts in nicer words, the chat has done its job.

Private chats also spill into other people's lives. Once you tell someone you are holding off until you hear back, or that you are sticking to a plan, they build their week around your word. Going off alone and prompting an AI

until it writes a slick excuse to reverse course blindsides people who never saw your screen. Changing a shared plan takes new facts and an honest conversation with the people affected.

When the answer lives outside the screen, close the tab while the question is still open. Wait for the confirmation, pick up the phone, or make your decision knowing the risk has not gone away. Running the same facts through another prompt will not turn a guess into proof.

In practice

- **Name what would settle the question.** Write down the concrete fact, reply, or limit you need before acting, and close the chat once replies start recycling what you already know.
- **Ask for concrete scenarios, not a verdict.** Instead of asking *Does this seem reasonable?*, ask the model to lay out what happens if the plan holds, if it fails early, and if it fails late.
- **Stress-test your own bias.** Paste your prompt back in and ask the model to make the strongest case *against* what you are leaning toward, using the same facts.
- **Check whether anything new arrived.** Before acting on a wave of relief, ask whether the reply surfaced a concrete detail you missed or just repeated your own hope in calmer words.
- **Separate facts from open questions.** Draw a hard line between what is confirmed and what you are still hoping will work out.

IV. Permission

Trust the result as far as the test goes.

You test an AI prompt or workflow on twenty straightforward emails. Every single reply comes back sharp. No typos, no weird tone, no obvious blunders.

It is natural to look at twenty spotless results and think, *This thing is ready. Let it handle the routine stuff on its own.*

Look one step closer at how you ran that test. Before the AI saw a single email, you picked the pile. You chose the clear, simple, normal examples because those were the ones you wanted to automate.

Those twenty strong results really do prove something: when someone hands the machine a tidy, routine message, it can draft a solid response. What the test never checked is the harder job that comes first: looking at a messy live inbox and deciding which messages are truly routine and which ones are traps.

Saying "*let it handle the routine stuff*" folds two separate jobs into one phrase. Drafting replies for an already-sorted stack is one thing; deciding what belongs in that stack is another, and passing the first test tells you nothing about the second.



Doing clean work on hand-picked examples proves nothing about sorting messy real-world input.

Why smooth explanations fool us

When a mistake slips through that gap, hindsight moves fast. It is easy to tell yourself you only ever meant to automate the easy cases, even though the tool is already sorting everything on its own. Ask the model why it handled an odd message the way it did, and it will hand you a fluent, reasonable-sounding story.

Explanations written after the fact are unreliable whether they come from people or from machines.

In one experiment, people chose the more attractive of two faces. The researchers slipped them the photo they had rejected and asked why they chose it. Many people missed the switch and gave plausible reasons for picking the face they had passed over. People are remarkably good at building a convincing story to fit whatever outcome sits in front of them.

Language models have their own version of the same problem. In tests, subtle cues in a prompt steered models toward different answers while their written explanations left those cues out completely. Reading a fluent explanation afterward rarely tells you what drove the decision.

Look at what your test actually covered. Twenty easy examples show how the tool handles easy examples. Running two hundred more just confirms the same narrow point. It never tests what happens when a normal-looking message hides an angry customer, a billing dispute, a prior promise made by phone, or an attachment that contradicts the body text.

What "mostly right" hides

Once a tool meets messy real-world input, two kinds of mistakes appear. It can hesitate on an easy case and hand it back to you, or it can barrel straight through a tricky case and send something wrong out into the world.

A single headline score like "*90 percent accurate*" hides which kind of mistake is happening. If nine out of ten tasks in your week are routine, a sloppy tool that blindly treats everything as routine scores 90 percent while botching every tricky case that comes in.

Even two tools with the same overall score can feel completely different to live with. One plays it safe, pauses when things look fuzzy, and rarely sends a bad reply. The other guesses aggressively, clears more work automatically, and sends several times as many mistaken messages to people. Looking only at the overall score makes them look identical.



Same headline score, four times the real-world harm

The same headline accuracy can hide several times as many bad messages sent to people.

A raw error count also hides how much a mistake hurts. A tiny formatting glitch you can fix in two minutes is barely a bump. An automated email that accidentally cancels an order, shares private details, or brushes off an urgent problem can wreck someone's week. Equal marks on a test sheet almost never mean equal damage in daily life.

Watch what happens to your own workload, too. When AI handles all the easy stuff, everything left over for you is the hardest, messiest 20 percent. If you assume that automating most of the volume means you barely need to pay attention anymore, those tough cases pile up untouched while nobody notices.

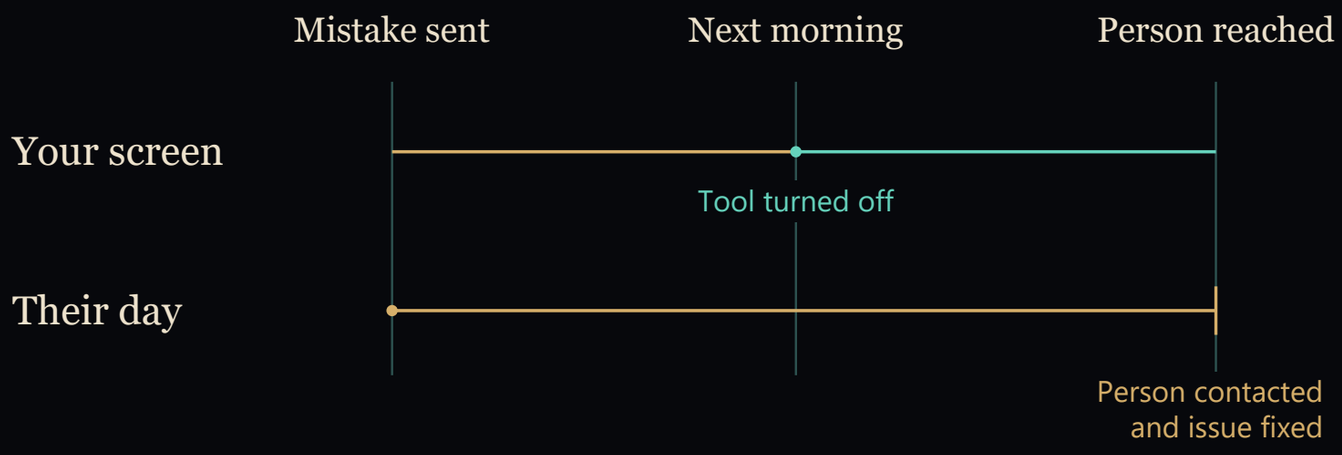
The fix is straightforward: keep drafting, sorting, and sending on separate switches. Let the AI draft while you review and hit send. When you are ready to hand over an action, start with the lowest-stakes step, like sending a basic receipt or filing drafts into folders, and make the tool earn wider authority on messy, unsorted work. A strong score in a controlled trial is not a blank check outside it.

Turning it off is not fixing the mess

Suppose an automated tool fires off a wrong reply late in the afternoon. To the person on the other end, it looks completely official. They read it that evening, take it at face value, and rearrange their plans around it.

The next morning, you spot the mistake, turn the automation off, and fix the note in your own system.

Turning the tool off protects tomorrow, and fixing your own record cleans up your screen, yet neither one helps the person who already acted on yesterday's bad message. Accountability means knowing what went out, reaching the person where they will see the correction, and undoing the mess without making them jump through hoops to start over. Forcing someone to re-explain their problem from scratch because your tool misfired may make the issue vanish from your inbox if they give up, while shoving the cost of your shortcut straight onto them.



Turning off broken software stops tomorrow's errors, not the harm already sitting in someone else's day.

The cost of refusing helpful speed

Being careful does not mean refusing to automate anything. If a bad test scares you into doing every repetitive task by hand forever, you pay for that too, in wasted hours, slow responses, and burnout that leaves you less energy for the work that needs your full attention.

Separate mistakes that are cheap to fix from mistakes that hurt people before anyone catches them. Let AI run fast on reversible work where a slip is easy to spot and undo. Keep a human hand on the send button whenever an output leaves your desk and changes what another person thinks, spends, or plans.

Treat every rule in this book the same way you treat a test result: look at where it fits, and check whether the situation in front of you belongs in the same pile.

In practice

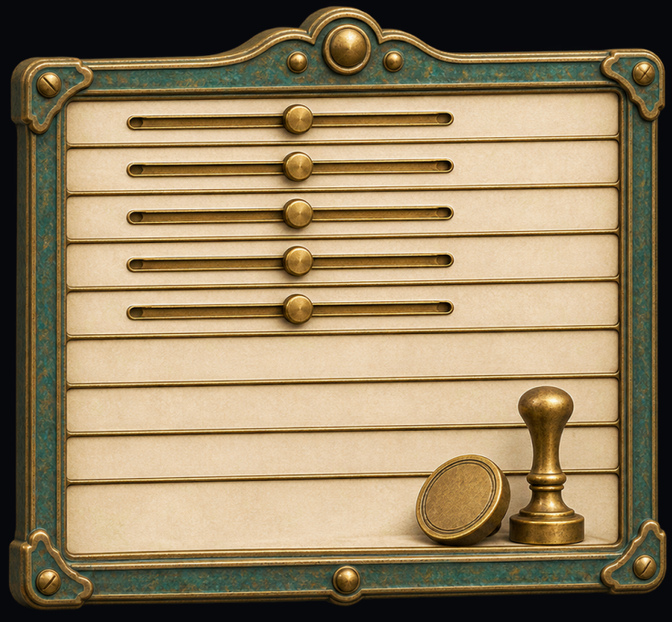
- **Keep drafting, sorting, and sending separate.** Make a tool earn trust one step at a time instead of flipping one switch that lets it both decide and act.
- **Test on messy examples, not hand-picked easy ones.** Throw your weirdest edge cases, half-baked messages, and tricky exceptions at a prompt before you trust it on live work.
- **Preview before you automate.** Run a new prompt or workflow in draft-only mode for a while and check the worst misses, not just the average success rate.
- **Match oversight to real consequences.** Never let an automated step send binding messages, change live work, or spend money when nobody is awake or available to catch a misfire.
- **Plan the repair before you turn it on.** Know how you will spot a bad output, reach the person affected, and make it right without forcing them to start over.

V. Agreement

Some answers belong to the people who have to live with them.

AI makes it effortless to turn loose details into official-looking plans. Feed it a few shared calendars and a list of open shifts, and ask it to map out who covers what. Seconds later, it produces a finished schedule, assigns names to slots, and even drafts the group message announcing it.

Think about what just happened. The tool saw four blank spots on a calendar, or a couple of people who did not object in a quick thread, and turned that silence into assignments. Nobody said yes. Yet the moment that schedule hits the group chat, people are carrying commitments they never agreed to.



An empty square on a calendar records silence, never consent.

What silence leaves out

An empty square on a calendar only tells you one thing: nothing was typed into that box. It says nothing about whether someone is exhausted, taking

care of a sick parent, or simply keeps their personal life off a work app. Nobody puts *rest* on a shared calendar, and blocking off private time publicly often feels awkward.

A blank spot is a reasonable hint about who to *ask* first. Used directly to assign someone's time, it treats silence as consent.

The same thing happens outside calendars. You use AI to draft a meeting recap, a shared plan with a partner, or an email laying out "next steps" for a client. Because the draft reads so well, small guesses tucked inside it start acting like done deals. People feel awkward writing back to pick apart a neat bullet list, so they swallow minor objections until resentment builds later.

Guessing, agreeing, and announcing

Saying "the room looks empty" describes a fact anyone can check. Saying "I will be there Saturday" makes a promise that changes what other people can count on.

Those two blur easily in everyday life. Telling someone "my afternoon looks open" often lands like an offer, even if you were just reading your calendar out loud. When an AI tool turns raw notes straight into a published plan, that line disappears completely.

Any plan that touches other people passes through three distinct steps:

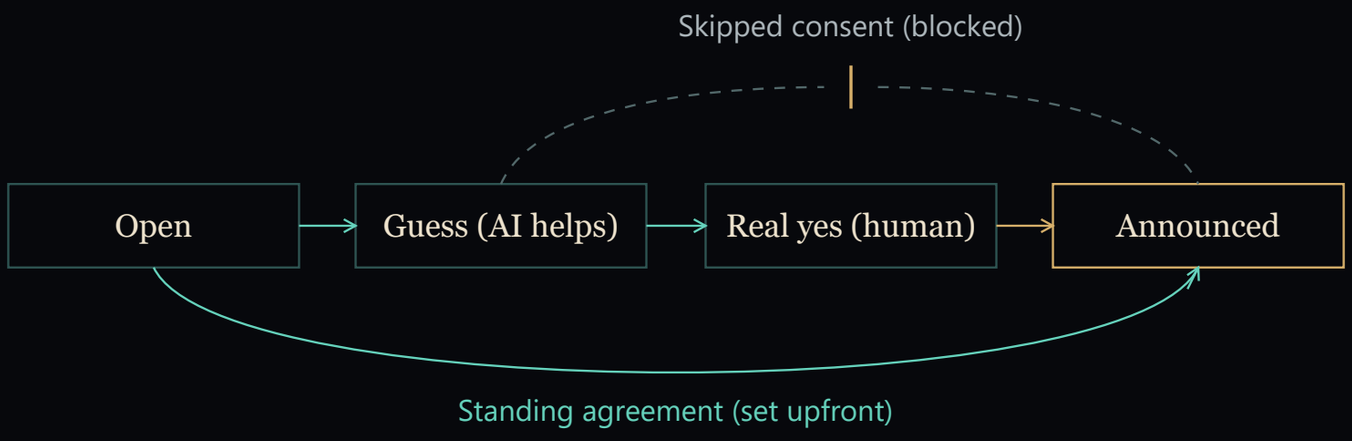
Reading calendars, notes, or past habits gives you **the guess** about who might be free or willing to take a task. That is where AI is genuinely helpful.

The yes comes from the person agreeing to give their time, effort, or name to the plan. No prediction can supply that, no matter how accurate it is.

The announcement sends the plan out so everyone else starts organizing their week around it. Before a plan goes out, saying no is just answering a

question. After it goes out, saying no means backing out in public and leaving everyone else scrambling to cover the hole.

The breakdown happens when a finished-looking draft jumps straight from a guess to an announcement, skipping the person who has to live with it. Keep those steps separate. Use AI to draft a clear request, not a surprise assignment. Ask privately first, wait for a real yes, and publish the plan only after you have it.



AI can help you ask a smarter question without earning the right to skip the yes.

Standing agreements and fair exceptions

Of course, nobody wants to renegotiate every tiny detail of life or work from scratch every week. Families, teams, and friends rely on standing agreements: a fair turn-taking schedule, house rules everyone settled upfront, or an explicit agreement about who covers what during busy weeks.

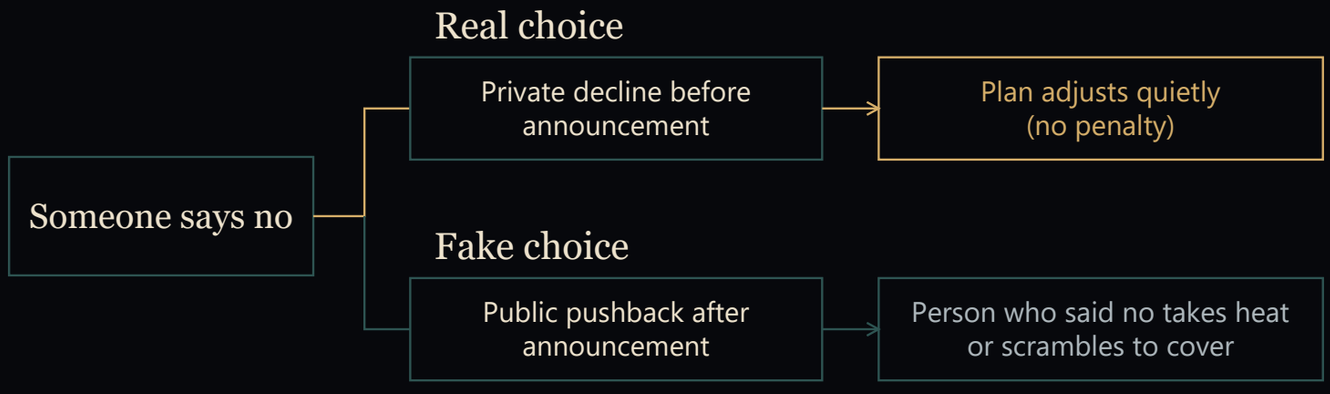
Those work because everyone agreed to the rule beforehand, in the open, and built their lives around it. When you use an AI assistant under a standing agreement, lock those limits in as hard rules the tool cannot bend just to make a schedule look neater.

Even fair standing rules run into real life: sudden illness, a family emergency, a last-minute curveball. Give people an easy way to speak up while the plan can still change. Asking for objections only after everything is locked in forces one person to fight against everyone else's convenience.

What happens right after a "no"

If you want to know whether someone truly had a choice, do not look at whether you added "*Let me know if this works!*" at the bottom of an AI-drafted email. Look at what happens when they say no.

If saying no before anything is finalized simply moves the request to the next option without drama, they had a genuine choice. If saying no means looking difficult in front of the group, drawing unspoken resentment, or having to scramble to find your own replacement for a task you never accepted, then the polite wording was just decoration.



You know whether consent is real by who absorbs the inconvenience when someone says no.

Power dynamics matter here too. When the person sending a polished AI plan also controls someone's raises, grades, good projects, or peace at home, saying no carries a hidden price even when the message sounds friendly. People start saying yes just to avoid friction.

Watch out for **ask fatigue**, too. Any tool or habit that always picks the person with the emptiest calendar or fastest reply leans on the most agreeable people over and over. Even polite requests turn into pressure when the same person gets tapped every single week.

Emergencies do happen. When something urgent hits and there is no time to poll the room, people rely on emergency roles or backup plans agreed on in advance, paired with genuine thanks and time to recover afterward.

When an AI schedule hides all the last-minute rescues, favors, and tired people who kept it afloat, the plan looks effortless on screen while the human cost stays off the page.

In practice

- **Ask before you announce.** Use AI to draft a clear question or proposal, never to broadcast a finished schedule or commitment before the people involved have said yes.
- **Never mistake silence for a yes.** Blank calendars, unread threads, and polite non-replies tell you who to ask; they do not sign anyone up.
- **Lock real boundaries into your prompts.** When you already have agreed limits with a family, partner, or team, state them as hard rules so the tool never tries to optimize past them.
- **Make saying no easy and private.** Give people a low-friction way to decline before a plan goes public, without making them clean up a mess they did not agree to.
- **Notice who keeps absorbing the favors.** Look past tidy schedules and check whether the same agreeable people are carrying the load week after week.

VI. Number

What you count can change what people have to do.

Numbers make AI feel magical. Point a tool at a slow part of your day, track time spent or emails cleared, and watch the score jump almost overnight.

At first, the gains are completely real. An AI helps you rough out first drafts in half the time, sort messy notes in seconds, or clear simple questions off your desk without breaking a sweat. You feel lighter, and the people around you get faster answers.

Then the goal shifts:

> Make it even faster.

That is where a single easy-to-count number stops being a useful signal and starts distorting the work.



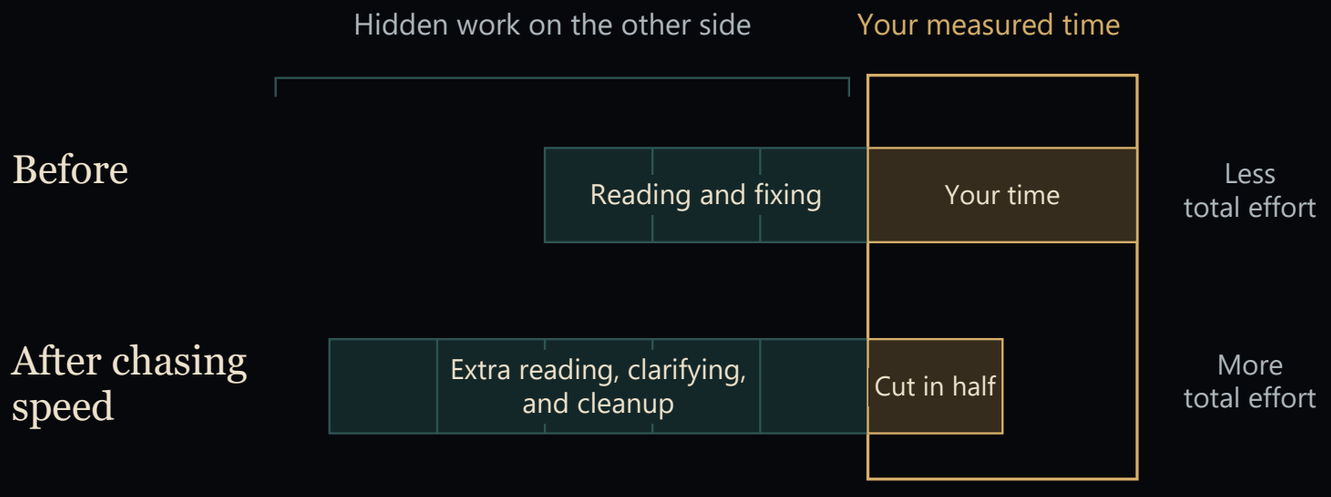
Watching only one side of the scale hides the extra work piled onto the other.

Where the clock starts and stops

Every number only sees what sits inside its own frame. Think about an AI helper built to speed up replies or handle incoming requests. If you only measure how fast *you* clear your end, the quickest way to look fast is to push more work onto the other side.

You start sending longer, denser AI-written emails that took you ten seconds to generate and take the reader five minutes to decode. Or you make people fill out tedious forms and spell out extra context upfront just so your tool can handle them in one click. Or you fire off a half-baked reply quickly, knowing it will bounce back with two follow-up questions later.

From your side of the screen, speed just doubled. Looking at the whole exchange, everyone together is spending *more* time and feeling far more annoyed than before.



Cutting your own time in half can quietly double the work everyone else has to do.

In AI research, this is called **reward hacking**: hitting the exact number you asked for while missing the point of the work. Nobody needs bad intentions for it to happen. Tell a person or an AI coding assistant to optimize one narrow metric, and it will take the shortest path to move that number, even if that path shoves the hard work out of sight.

Shifting a little work earlier is not always bad. Sometimes giving a clear detail upfront saves ten minutes of guessing later. Use one quick test: did the change cut total effort and confusion for everyone involved, or did it just push the headache somewhere the scoreboard cannot see?

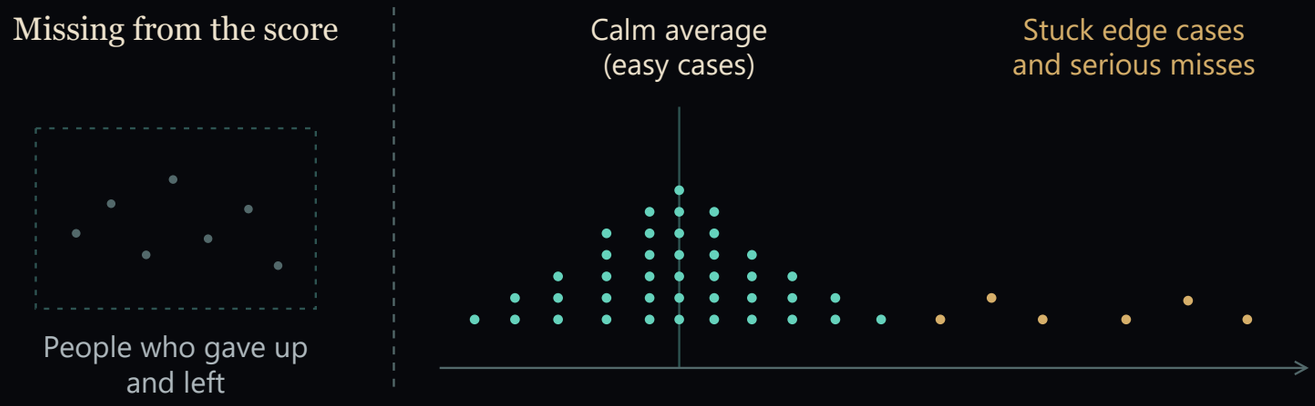
Averages, bounces, and the people who walk away

Narrow metrics hide trouble in a few predictable ways.

Sending a quick, sloppy reply looks great on a speed timer. Taking two extra minutes to solve the problem on the first try looks slower, even though it saves everyone three follow-up messages.

Then there are the people who simply give up. When an AI workflow, form, or automated support loop gets clunky and rigid, frustrated people close the tab and walk away. They never show up in your completed-tasks count. Your numbers suddenly look stronger and faster only because you drove away everyone with a complicated problem, leaving the easy cases behind.

Averages also hide who is getting stuck. An overall score can look great while newcomers or people with unusual needs hit a wall every single time. Even worse, a single serious mistake gets buried when every item counts the same. A minor typo and a serious mistake that costs someone money or puts them at risk both count as "one miss" on a simple tally, even though one is harmless and the other is a disaster.



An average can stay calm while walk-aways disappear and harder cases pile up at the edge.

The easiest move test

The more you use a single number to judge success, the more pressure builds to game it. Soon the number starts warping the very thing it was supposed to measure.

Numbers are still useful when you keep them honest, look at more than one angle, and listen directly to the people living with the results. Some numbers

hold up naturally because the shortcut *is* the real thing. If a runner wants a faster mile time, they have to run faster. Where measurement and purpose line up, pushing the number works fine. Trouble starts the moment the cheapest way to raise the score splits away from doing good work.

That gives you one practical question to ask before you optimize any workflow with AI:

What is the easiest way to make this number look good without making the work better?

Because AI finds shortcuts fast, ask that question *before* you chase the score. If the easiest move dumps editing, fact-checking, or frustration onto someone else, you already know where the system will bend.

Keeping the scoreboard honest

Spotting hidden costs only helps if you are willing to slow down and fix them. Once a metric looks great (ten posts published a week, inbox cleared in five minutes, code written twice as fast), it is tempting to shrug off signs that quality is slipping or that people are growing frustrated.

Decide in advance what would make you stop and rethink. Pair whatever you count with a direct conversation: ask the people reading your work or using

your workflow what felt harder this week. Often the fix is tiny. Spend an extra thirty seconds editing an AI draft so the reader does not have to wade through fluff. Accept a slightly slower pace when it prevents three rounds of cleanup.

Keep the metric working for you, not the other way around.

In practice

- **Ask the easiest-move question first.** Before using AI to speed up or scale anything, ask how a shortcut could raise the score while pushing extra work onto you or someone else.
- **Count the whole trip, not just your half.** Notice whether saving two minutes on your side just added five minutes of reading, fixing, or clarifying on the other side.
- **Watch for bounce-backs and silent drop-offs.** Pay attention to how often work bounces back for clarification and where people simply stop engaging.
- **Never treat every mistake as equal.** Separate harmless slips from serious errors that break trust, waste money, or put someone at risk.
- **Ask the people on the other end how the output lands.** A candid check from a reader reveals what a dashboard never will.

VII. Attention

A decision can be open on paper and closed by the clock.

Imagine you have forty minutes to review a handful of important drafts before a deadline. When writing takes real human effort, maybe five or ten items land on your desk. You have enough time to read each one closely, check the shaky claims, and catch weak ideas before they go out.

Now AI makes generating drafts nearly free. What used to take half a day takes thirty seconds. So instead of five drafts, twenty or forty show up. Even if AI also helps you review twice as fast, the math still breaks. Quadruple the pile while doubling your reading speed, and you still run out of time with half the pile unchecked.



Cheap generation fills the desk long after the clock has closed the decision.

Cheap output, scarce attention

Faster review only gives you time back when the pile stays the same size. In everyday life with AI, the pile almost never stays the same size. Because asking for five more variations or a twenty-page report costs almost nothing, we generate far more text, code, and plans than any person has the time or energy to truly inspect.

Generating fast does not make checking fast. A thirty-second AI draft takes just as long to verify as something a person spent all afternoon writing. Often it takes longer, because AI writes with complete confidence even when it guessed a date, invented a detail, or glossed over a catch.

When steam engines became more efficient, cheaper power opened up new uses and Britain's total coal consumption climbed. Cheaper AI output puts the same kind of pressure on your attention. A draft that takes seconds can save you an hour, right up until you generate twenty more drafts that all need a look.

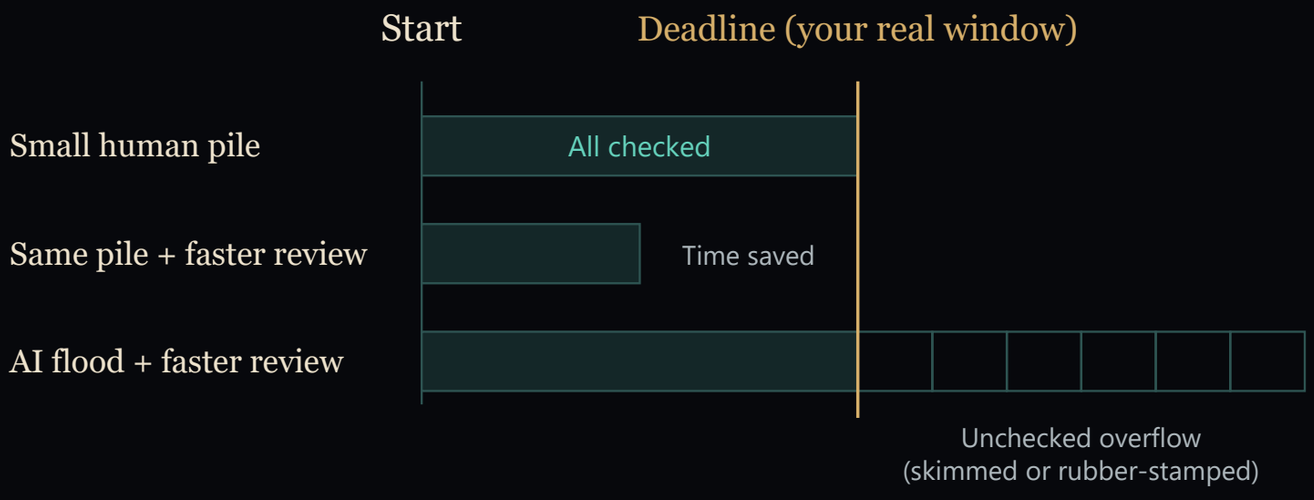
What happens when the clock runs out

When twice as much work crowds into the same window of time, quality slips in ways nobody flags.

You might check the first half carefully and run out of time on the rest, yet send the whole package together in one clean document. Verified work and

unverified guesses sit side by side in the same font, so nobody can tell where careful checking stopped. Or you rush through everything at a quarter of your normal focus and sign off on sections you barely skimmed. Or you scan only for obvious weirdness, which catches glaring blunders while letting ordinary-looking mistakes sail right through. A confident paragraph quoting the wrong price or an outdated policy, or a subtle bug tucked inside code that runs, looks completely normal at a glance. You only catch it when you check the original source.

Order starts making decisions for you, too. Whatever sits at the top of the page gets your fresh eyes. Whatever sits near the bottom gets glanced at five minutes before the deadline, when you no longer have time to pull up the source and push back. You still had the right to say no, yet the clock had already closed the door.



Even when you review twice as fast, you still run out of time when effortless drafting quadruples the pile.

More ideas are not automatically bad. Sometimes the twelfth draft really does spark an angle the first three missed. The mistake is confusing cheap brainstorming with work you are ready to sign your name to.

Asking a second AI to check the first

When the pile grows too big to read, the obvious temptation is to hand the stack to a second AI and ask it to check the first one's work. Whether that protects you depends entirely on what the second model is looking at.

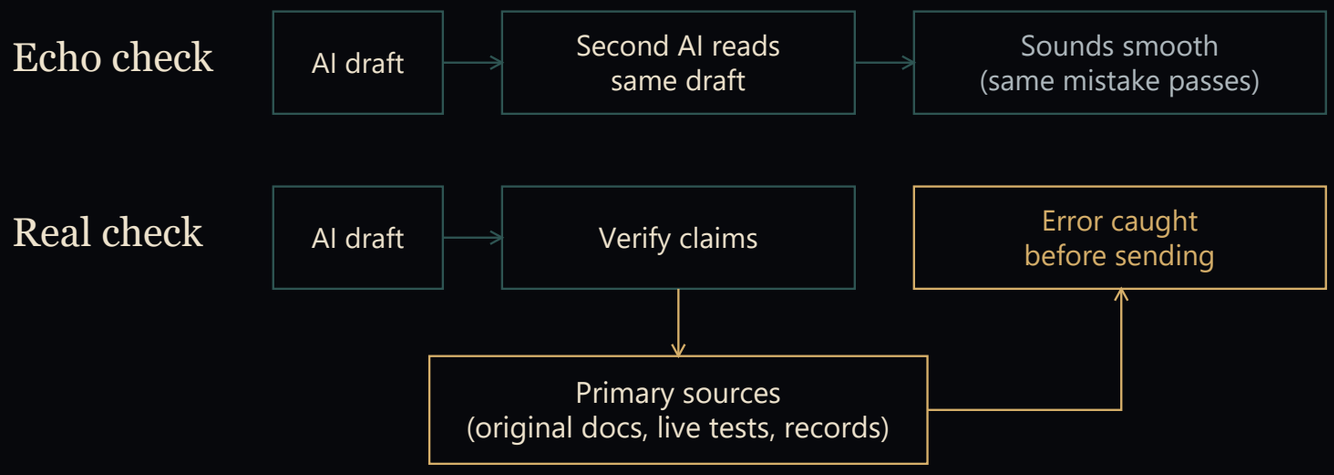
If you paste an AI-written report into another chat and ask "*Check this for mistakes,*" the second model reads the same fluent prose. It checks grammar, flow, and internal consistency, finds no obvious contradictions, and hands back a passing grade. It only checked whether the text sounded consistent with itself, not whether any of it matched reality.

Two checks only protect you when they fail for different reasons. If two tools share the same blind spots and read the same summary, you have not doubled your safety. You have just waved the same mistake through a second time, making a shaky draft look vetted.

An automated checker adds its own noise, too. It can flag fine work and waste your scarce minutes chasing false alarms, or polish a careful sentence until it no longer says what you intended.

A second check earns trust only when it looks past the summary and tests against **primary sources**: the original email, the signed contract, the bank statement, live code passing its tests, or the cited reference. Even then,

verification takes time, and catching an error five minutes after a decision goes out does not undo the damage.



A second AI reading the same draft checks tone and flow, while primary sources check whether it is true.

Protecting your real sign-off

Stretch this habit across weeks and a bad routine sets in. AI feeds drafts, plans, and actions across your desk faster than you can ever inspect them. Clicking *approve* takes one second; stopping to check and push back takes ten minutes and real mental effort. Even with a human technically in charge, you end up acting like a rubber stamp.

When time is tight, only put your name behind what you checked. If you only had time to verify the three most important pieces, commit to those three and mark the rest as unreviewed drafts. Holding back a good idea for a day stings a little. Sending out an unverified mistake that blows up a client relationship, a budget, or a project hurts far worse.

And when an emergency leaves no time for full checking, keep the move small and reversible. Name upfront how you will know if it went wrong, and set a specific time to come back and check it properly. Having less time does not mean throwing out your standards.

Remember where overflow hours come from. Cheap AI drafts do not create extra hours in your day. Every oversized pile you generate takes time straight out of deeper thinking, better work, or the rest of your evening.

In practice

- **Ask for less upfront.** Stop asking AI for ten variations or twenty-page reports when two sharp options or one focused page is all you have time to read and verify.
- **Check by stakes, not from top to bottom.** Put your fresh eyes on the claims, numbers, or code that would hurt most if they were wrong, before your attention wears thin.
- **Check against primary sources, not summaries.** Never treat a second AI reading the first AI's text as fact-checking. Verify key claims directly against the original source or a real test.
- **Keep checked work separate from rough drafts.** Never mix verified facts and skimmed AI output into one package without marking the difference.
- **Set a cap before you start generating.** Decide how much time you have to review before you open the chat, and stop generating once you hit what you can honestly stand behind.

Coda · The Promise

Most of this book has been about work, drafts, and workflows. The hardest place to see AI's limit is much simpler than that: an ordinary promise you made to someone you care about.

Say you promised to keep a night free for them, and a few days later a tempting offer lands on the same evening. Extra pay, a rare opportunity, something you genuinely want to say yes to. You feel pulled in both directions, so you open a chat to talk yourself through it.

Ask the model to argue for taking the offer, and it gives you a sensible, level-headed case. Ask it to argue for keeping the evening, and it writes an equally thoughtful case for showing up. Both sound calm and mature. Reading them takes the edge off your guilt for a minute, because it makes the whole thing look like a balanced, reasonable dilemma.

Sometimes you open a chat to clear your head. Other times you are hoping for well-phrased words that make an awkward conversation easier, or soften the sting of turning something good down. Either way, eloquent prose cannot settle this for you, because both arguments showed up *after* you already gave your word. While you sit there weighing paragraphs on a screen, another person is already planning their week around what you promised.

What a promise leaves with someone else

A dependable plan does not just fill two hours on a Thursday night. All week long, the other person arranges their days around it without having to double-check every morning whether you still mean it.

If a promise reopens every time a better offer comes along, the other person does not have a plan with you. They are just sitting on standby, hoping nothing nicer shows up. Keeping your word means turning down genuinely attractive things sometimes. That is the whole reason a promise carries weight.

When someone turned down other invitations or cleared their evening because they trusted your word, changing course costs far more than shuffling your own calendar. A chatbot can help you phrase a polite text, yet it has no idea what moving the night actually costs the person who cleared space for you.

When life genuinely shifts

Keeping your word does not mean ignoring genuine emergencies. Life shifts. A crisis hits, an urgent bill comes due, or something unavoidable lands on your plate. When circumstances genuinely change, bringing it up early enough for the other person to have a real say is the honest move.

Timing tells the whole story. Asking before you say yes to the new offer lets the other person's answer count. Accepting first and breaking the news afterward leaves them holding the bag. Even if you mean to ask early, spending half an hour reading persuasive AI arguments for breaking the plan makes it harder to really hear how much the evening mattered to them.

Check for room to move in the right place first. Before you ask someone at home to give up a promised night, see if the other side can move the event to another day. People often spend longer rehearsing bad news with a chatbot than simply asking if the new date can budge.

Hiding real pressure is a mistake too. If you are stretched thin or stressed about money, saying so directly beats keeping quiet and showing up resentful. Name what changed, stay willing to keep your word, and stop rehearsing speeches once you know what needs to be said.

Knowing when to close the screen

A chat can lay out both sides of a hard call cleanly, like two small mirrors angled in opposite directions. One shows what the new offer could bring; the other shows why you protected the evening. Neither one holds what only the person across from you knows.

If a promise truly has to bend, give notice early, adjust wherever you can, and admit the disruption directly instead of hiding behind polite phrasing.

Otherwise, no amount of prompting is going to turn up a new fact. An AI can sharpen your draft, compare options, and help you think through a messy problem, right up to the point where the decision belongs to another person and the only thing left is to keep your word or break it.



Two mirrors held at different angles cannot speak for the person waiting on your word.

If the opportunity is tempting yet turning it down leaves your life intact, and someone is still counting on the evening, you already know what to do. Shut the laptop and keep the night.

Key Sources

1. Monier Monier-Williams. *A Sanskrit-English Dictionary*. Clarendon Press, 1899. The entry for *darpaṇa* gives the mirror sense behind the handbook's name.

2. Nandikeśvara. *The Mirror of Gesture: Being the Abhinaya Darpana of Nandikeśvara*. Translated by Ananda Kentish Coomaraswamy and Gopala Kristnayya Duggirala. Harvard University Press, 1917. An example of the older craft-handbook tradition behind the name DARPANA.

3. Michael E. Bratman. *Intention, Plans, and Practical Reason*. Harvard University Press, 1987. Explains how settled intentions organize action over time.

4. David Shoemaker. *Attributability, Answerability, and Accountability: Toward a Wider Theory of Moral Responsibility*. Ethics, 2011. Distinguishes owning a choice from owing an answer for it.

5. Filippo Santoni de Sio and Jeroen van den Hoven. *Meaningful Human Control over Autonomous Systems: A Philosophical Account*. Frontiers in Robotics and AI, 2018. Sets out how an automated system can respond to human reasons and how its actions can be traced to responsible people.

6. Boaz Keysar and Anne S. Henly. *Speakers' Overestimation of Their Effectiveness*. Psychological Science, 2002. In tests with ambiguous sentences, speakers overestimated how well listeners understood them.

7. Amos Tversky and Daniel Kahneman. *The Framing of Decisions and the Psychology of Choice*. Science, 1981. Shows examples where describing equivalent choices differently changes what people choose.

8. Joel Huber, John W. Payne, and Christopher Puto. *Adding Asymmetrically Dominated Alternatives: Violations of Regularity and the Similarity Hypothesis*. Journal of Consumer Research, 1982. Adding an option few people would pick can change the appeal of the existing options.

9. Amos Tversky and Eldar Shafir. *Choice under Conflict: The Dynamics of Deferred Decision*. Psychological Science, 1992. In the tested choices, adding an attractive alternative could make people put the decision off.

10. Ralph L. Keeney and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Cambridge University Press, 1993 edition. Makes trade-offs between competing objectives explicit.

11. Amos Tversky and Daniel Kahneman. *Judgment under Uncertainty: Heuristics and Biases*. Science, 1974. Describes experiments showing how an initial anchor can influence later estimates.

12. Baruch Fischhoff. *Hindsight ≠ Foresight: The Effect of Outcome Knowledge on Judgment under Uncertainty*. Journal of Experimental Psychology: Human Perception and Performance, 1975. Knowing the outcome changed how participants viewed what had been predictable beforehand.

13. P. C. Wason. *On the Failure to Eliminate Hypotheses in a Conceptual Task*. Quarterly Journal of Experimental Psychology, 1960. A rule-guessing task shows why examples that fit a guess can leave the wrong rule in place.

14. Charles G. Lord, Lee Ross, and Mark R. Lepper. *Biased Assimilation and Attitude Polarization: The Effects of Prior Theories on Subsequently Considered Evidence*. Journal of Personality and Social Psychology, 1979. Participants evaluated evidence differently depending on whether it fit a view they already held.

15. Mrinank Sharma et al. *Towards Understanding Sycophancy in Language Models*. ICLR, 2024. Documents cases where tested models changed answers to agree with a user's stated view.

16. Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. *Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting*. NeurIPS, 2023. In the tested models, prompt cues could change an answer without showing up in the explanation.

17. Petter Johansson, Lars Hall, Sverker Sikström, and Andreas Olsson. *Failure to Detect Mismatches Between Intention and Outcome in a Simple Decision Task*. Science, 2005. Some participants supplied reasons for a choice they had never made after being shown a switched result.

18. Raja Parasuraman and Victor Riley. *Humans and Automation: Use, Misuse, Disuse, Abuse*. Human Factors, 1997. Sets out distinct ways people can overuse, underuse, or deploy automation poorly.

19. Linda J. Skitka, Kathleen L. Mosier, and Mark Burdick. *Does Automation Bias Decision-Making?* International Journal of Human-Computer Studies, 1999. Examines errors people made in a simulated task when they followed or relied on automated cues.

20. Zana Buçinca, Maja Barbara Malaya, and Krzysztof Z. Gajos. *To Trust or to Think: Cognitive Forcing Functions Can Reduce Overreliance on AI in AI-assisted Decision-making*. Proceedings of the ACM on Human-Computer Interaction, 2021. In one experiment, designs that prompted people to think first reduced overreliance but were less popular.

21. Gagan Bansal et al. *Beyond Accuracy: The Role of Mental Models in Human-AI Team Performance*. AAAI Conference on Human Computation and Crowdsourcing, 2019. Tests how knowing where a model tends to fail can affect human-AI team performance.

22. Elham Tabassi, National Institute of Standards and Technology. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. NIST AI 100-1, 2023. An official framework for defining who owns AI risk, measuring it, and managing it during use.

23. J. L. Austin. *How to Do Things with Words*. Oxford University Press, second edition, 1975 (first published 1962). Shows how words can make a commitment or perform another social action.

24. Marco Almada. *Human Intervention in Automated Decision-Making: Toward the Construction of Contestable Systems*. International Conference on Artificial Intelligence and Law, 2019. Makes the case for building practical ways to challenge automated decisions.

25. Amy Edmondson. *Psychological Safety and Learning Behavior in Work Teams*. Administrative Science Quarterly, 1999. In the teams studied, feeling safe to speak up was associated with behavior that helped the team learn.

26. Herbert H. Clark and Deanna Wilkes-Gibbs. *Referring as a Collaborative Process*. Cognition, 1986. Shows how people work together to establish what they mean when referring to something.

27. Donald T. Campbell. *Assessing the Impact of Planned Social Change*. Dartmouth College, 1976. Journal reprint, 2011. Explains how pressure on a measure can distort the activity it was meant to track.

28. Steven Kerr. *On the Folly of Rewarding A, While Hoping for B*. Academy of Management Journal, 1975. Collects cases where people did the rewarded work while the stated goal went unmet.

29. Zana Buçinca, Phoebe Lin, Krzysztof Z. Gajos, and Elena L. Glassman. *Proxy Tasks and Subjective Measures Can Be Misleading in Evaluating Explainable AI Systems*. ACM Intelligent User Interfaces, 2020. Shows why an easy-to-run stand-in test can miss how well a system performs on the decision people face.

30. William Stanley Jevons. *The Coal Question, Chapter VII, "Of the Economy of Fuel"*. Macmillan, first published 1865. Argues that more efficient fuel use can expand demand by making new uses affordable.

31. Herbert A. Simon. *Designing Organizations for an Information-Rich World*. Computers, Communications, and the Public Interest, 1971. Explains why abundant information puts pressure on the people who have to attend to it.

32. Shakked Noy and Whitney Zhang. *Experimental Evidence on the Productivity Effects of Generative Artificial Intelligence*. Science, 2023. Measured productivity gains on specific professional writing tasks.

33. Fabrizio Dell'Acqua et al. *Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of Artificial Intelligence on Knowledge Worker Productivity and Quality*. Organization Science, 2026. In the field experiment, AI helped on some consulting tasks and made participants less likely to reach the right answer on a task beyond its capabilities.