Conversation, Not Delegation

How to Think With AI, Not Just Use It

Michael Borck

2026-03-25

Conversation, Not Delegation
How to Think With AI, Not Just Use It

Published by Michael Borck
Perth, Western Australia

ISBN: 979-8-2528-2455-0

First edition, 2025.

AI disclosure: This book was written using the methodology it describes. AI tools were used as thinking partners throughout the drafting, iterating, and refining process. The author reviewed, challenged, and took responsibility for every sentence. Full details of the tools used are in the front matter.

Companion website:
https://michael-borck.github.io/conversation-not-delegation
Source: https://github.com/michael-borck/conversation-not-delegation

Table of contents

Preface . . . 1

I Part 1: Understanding the Landscape 7

1 What Is AI? . . . 9
2 What Are Large Language Models? . . . 15
3 The Delegation Trap . . . 23
4 Does AI Make Us Dumber? . . . 29

II Part 2: Principles 39

5 The Conversation Loop . . . 41
6 AI Last . . . 51
7 Staying Critical . . . 57

III Part 3: The Methodology 69

8 RTCF: Starting Conversations Well . . . 71
9 Prompt Chaining: Building on What You Started . . . 83
10 Eight Techniques for Deeper Thinking . . . 93
11 Using AI to Help You Use AI . . . 103
12 VET Your AI: The Push-Back Framework . . . 109

IV Part 4: Putting It Together 115

13 A Conversation Across Disciplines . . . 117
14 Becoming More Capable . . . 123

Appendices 129

A Prompt Structuring Frameworks . . . 129
B Quick Reference Cards . . . 135

C Further Reading 139
D Interactive Tools 147
About the Author 149

Preface

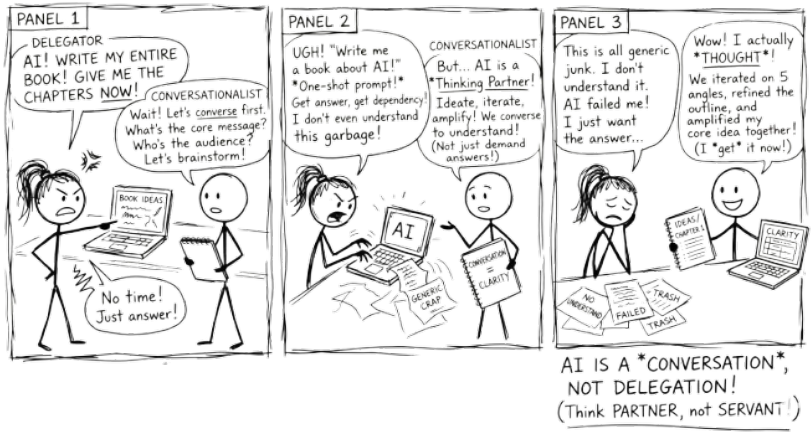

Figure 1: Delegation is quick. Conversation is how you actually learn.

Why This Book Exists

I wrote this book because I kept watching smart people use AI badly.

Not badly in a technical sense. They could find the tools, type the prompts, and get outputs. But they were delegating their thinking to a machine and accepting whatever came back. The AI was doing the work, and the human was doing the clicking. The outputs looked polished. The thinking behind them was hollow.

There is a different way. Instead of delegating to AI, you can converse with it. You can use it to brainstorm, to challenge your assumptions, to stress-test your reasoning, to explore angles you would not have considered alone. When you do this, the AI does not replace your thinking. It amplifies it.

This book teaches you how.

Who This Book Is For

This book is for anyone who uses AI as a thinking tool. Students, professionals, researchers, educators: if you interact with large language models as part of your work or study, this book will change how you approach those interactions.

You do not need a technical background. You do not need to write code. You need curiosity, a willingness to think critically, and a problem worth solving.

If you work with AI in any capacity, or plan to, this book will give you a framework for doing it well.

What This Book Is Not

This is not a guide to specific AI tools. It does not teach you how to use ChatGPT, Claude, or any particular platform. Interfaces change; the principles here do not.

It is not a prompt engineering manual. You will learn to structure prompts well, but the book argues that the prompt is the beginning of the conversation, not the point of it. If you are looking for a library of copy-paste prompts to get finished outputs in one shot, this is the wrong book.

It is not about AI apps that embed intelligence into other software, such as writing assistants, email autocompletes, and coding copilots. Those tools run on prompts someone else wrote. This book is about conversations you control yourself. Both have their place, but they are different activities.

And it is not a book that tells you AI will solve your problems. It is a book that argues the opposite: you will solve your problems, and AI can help you think more clearly along the way.

If You Are Feeling Uncertain

That is worth naming. Many people experience a quiet anxiety: a sense that everyone else has figured this out already, that the technology is moving too fast, that they are falling behind. That feeling is nearly universal and rarely admitted. You are not behind. The technology is genuinely new, it is genuinely confusing, and the people who appear to have it all figured out are mostly just a few weeks ahead. This book meets you wherever you are.

How This Book Is Structured

Part 1 gives you enough understanding of AI and large language models to use them well. Part 2 establishes the principles that make AI conversations productive rather than passive. Part 3 gives you practical tools and techniques you can use immediately. Part 4 shows it all working together across disciplines, and closes with the argument that matters most: the goal is not to get AI to do more, but to become more capable yourself.

How to Read This Book

This is not a textbook and it is not a research paper. The ideas here reflect one practitioner's perspective, informed by research, experience, and ongoing experimentation. They are starting points, baselines to work from as you discover your own approach to AI.

Where claims rest on published research, you will find pointers in the Further Reading appendix (Appendix C). Where they rest on practice and observation, that should be obvious from the writing. In both cases, the advice in this book applies to itself: engage with it critically, push back where your experience says otherwise, and make the ideas your own. A book about not delegating your thinking to AI would be contradicting itself if it asked you to accept its arguments uncritically.

Conventions Used in This Book

Throughout the book you will encounter coloured callout boxes. Each serves a different purpose.

> Try this
>
> **Green boxes** are hands-on exercises you can do right now, usually in two to ten minutes. They give you a way to experience the concept rather than just read about it.

> Key idea
>
> **Blue boxes** highlight important observations, ready-to-use prompts, or ideas worth pausing on.

> ⚠ Watch out
>
> **Yellow boxes** flag common mistakes, weak examples, or things that look right but are not.

> ! Critical point
>
> **Red boxes** mark habits or principles that are essential. Do not skip these.

You will also find tables throughout the book that compare approaches, map strategies across domains, or summarise frameworks. These are reference material, worth bookmarking, not just reading once.

How This Book Was Written

This book was written using the methodology it describes.

It began as a set of Python notebooks on computational thinking, practical exercises developed for teaching. Those notebooks evolved, through multiple conversation loops with AI, into the sibling book *Converse Python, Partner AI*. That process surfaced a methodology that was broader than programming: the principles of conversation over delegation, of staying critical, of using AI to amplify thinking rather than replace it. This book is the result of extracting and generalising that methodology for a wider audience.

At every stage, the process looked like what you will read about in Chapter 5. Ideas were brainstormed, challenged, restructured, and refined across dozens of conversations. AI drafted, the author pushed back. AI suggested structures, the author broke them and rebuilt them. The frameworks in this book (RTCF, VET, the eight techniques) were tested in the writing of the book itself. Where AI was sycophantic, it was told to stop. Where it was generic, it was pressed for specifics. Where it was wrong, it was corrected.

You may be thinking: if this book was written with AI, why should I trust it? That reaction is understandable, and this book has a name for it: the AI Dismissal Fallacy (Chapter 7). The quality of an idea does not depend on whether a human or a machine contributed to its development. What matters is whether the author can explain, defend, and stand behind every sentence. The answer is yes, because conversation, real conversation with pushback and judgement, is how the thinking was done.

Transparency about process is more important than the comfort of pre-

tending AI was not involved. Every professional will face this question soon enough. This book's answer is to show the work.

Tools used in producing this book:

- **Claude** (Anthropic): conversational AI partner for drafting, iterating, and refining text
- **Claude Code**: command-line tool for managing files, building, and publishing
- **Google Gemini (Nano Banana 2)**: generative AI for creating the comic strip illustrations (Chapters 1–8)
- **FLUX Playground** (Black Forest Labs): generative AI for creating the comic strip illustrations (Chapters 9–14)
- **Python** and **Jupyter Notebooks**: the original medium in which the ideas were developed and tested; also used to generate the cover image (`create_cover.py`)
- **Quarto**: open-source publishing system used to produce the HTML, PDF, and epub editions
- **Git and GitHub**: version control and hosting, including GitHub Pages for the web edition

Ways to Engage with This Book

A book about conversation should be available in more than one form. Pick the format that fits how you think and learn.

- **Read it online.** The full book is freely available at the companion website, with dark mode, search, and navigation.
- **Read it on paper or e-reader.** Available as a paperback and ebook through Amazon KDP, for those who prefer to read offline or away from a screen.
- **Download the PDF or epub.** Generated from the same source, available from the website.
- **Converse with it.** The online edition includes a chatbot grounded in the book's content. Ask it questions, challenge its answers, and practise the methodology on the methodology itself. You can also use the `llm.txt` file to paste the entire book into ChatGPT, Claude, or any AI tool for a deeper conversation.
- **Listen to it.** Upload the `llm.txt` file to Google NotebookLM and generate an audio overview or podcast-style discussion of the content.
- **Explore the source.** The full source is on GitHub (https://github.com/michael-borck/conversation-not-delegation), including every chapter, the build system, and the revision history. DeepWiki (https://deepwiki.com/michael-borck/conversation-not-delegation) provides an AI-navigable view of the repository.

- **Practise the frameworks.** The companion site includes three interactive tools that run in your browser — no login, no data stored. The **RTCF Prompt Builder** walks you through constructing prompts step by step using the framework from Chapter 8. The **RTCF Prompt Analyser** lets you paste any prompt and get instant feedback on which elements are present, partial, or missing — useful for diagnosing why a prompt gave a vague response. The **AI Readiness Assessment** is a short quiz that maps your current experience level and recommends which chapters to prioritise. See Appendix D for details.
- **Browse all books.** This book is part of a series. See all titles at books.borck.education (https://books.borck.education).

The online version is always the most current. The printed and ebook editions are updated periodically. If you purchased a copy and want the latest version, re-download it; you get the current edition.

The Sibling Book

This book has a companion: *Converse Python, Partner AI*, which covers the same methodology through the lens of Python programming. If you are a developer or data scientist, that book may be a better fit. If you are not, you are in the right place.

Both books are independently complete. You do not need one to benefit from the other.

Part I

Part 1: Understanding the Landscape

Chapter 1

What Is AI?

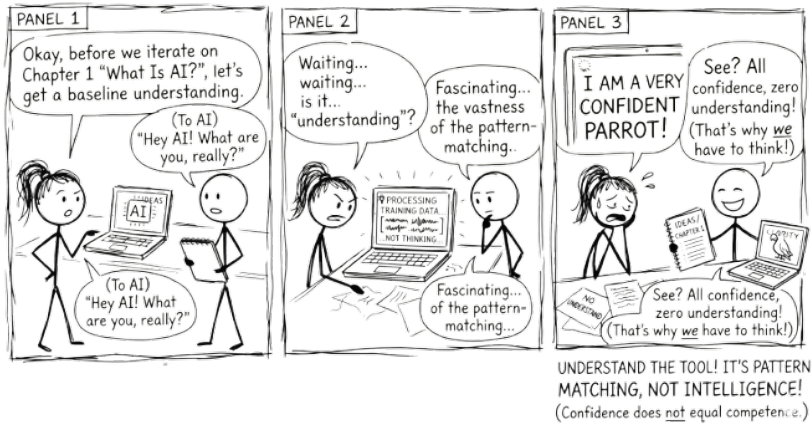

Figure 1.1: Understand the tool! It's pattern matching, not intelligence.

> The gap between what AI appears to do and what it actually does is where good decisions begin.

You might be wondering why a book about working with AI starts with a chapter about what AI is. The reason is practical: every framework in this book, every technique for prompting, verifying, and staying critical, exists because of something specific about how these tools work. If you understand the mechanics, even at a high level, the techniques stop feeling like arbitrary rules and start feeling obvious. You will know *why* you need to verify AI output, not just that you should.

Artificial intelligence is software that can perform tasks that typically require human intelligence.

That is it. No magic. No consciousness. Just software that can do things that used to require a human brain.

When we say "intelligence" here, we mean specific capabilities: recognising patterns in images, making predictions from data, understanding language, generating text. AI does not "think" the way you do. It processes data using mathematical patterns. But the results can look remarkably intelligent, and that gap between appearance and reality is worth understanding from the start.

1.1 Why Now?

AI is not new. The term was coined in 1956. Researchers have been working on it for decades. So why is it suddenly everywhere?

Three things converged.

1.1.1 We have massive amounts of data

Every click, purchase, photo, and search creates data. Organisations now have billions of examples to learn from. Netflix has billions of viewing decisions. Wikipedia has millions of articles. The entire public web has been scraped, indexed, and fed into training pipelines. The raw material for AI to learn from went from scarce to effectively unlimited.

1.1.2 Computing power got cheap and fast

What took a supercomputer in 2000 now runs on a phone. Cloud computing made massive processing power available to anyone with a credit card. Training an AI model that would have cost millions a decade ago now costs a fraction of that. The hardware barrier dropped.

1.1.3 Better algorithms were discovered

New mathematical techniques, particularly a family called "deep learning," dramatically improved what AI could do with data and compute. Researchers figured out how to stack layers of pattern recognition so that each layer builds on the one below it. The result was AI that could handle far more complex tasks than previous approaches allowed.

These three forces fed each other. More data made better algorithms worthwhile. Cheaper compute made bigger models practical. Better algorithms made use of both. The result is that AI which was science fiction ten years ago is now practical, affordable, and widely deployed.

1.2 The AI Family Tree

AI is not one thing. It is a family of technologies, each suited to different problems.

Rule-based systems are the oldest approach. Programmers write explicit rules: if this condition, then that action. These work well for clear, logical decisions that do not change much. Tax software and simple chatbots use this approach. The limitation is obvious: someone has to anticipate every scenario in advance.

Machine learning flips the process. Instead of writing rules, you show the system thousands of examples and let it figure out the patterns. Show it ten thousand photos labelled "cat" and ten thousand labelled "not cat," and it learns to tell the difference. This works well for pattern recognition, predictions, and problems where the rules are too complex to write by hand. The catch is that it needs lots of good examples and it is only as good as the data you give it.

Deep learning extends machine learning with multiple layers of pattern recognition, loosely inspired by how neurons work. The first layer might detect edges in an image. The second combines edges into shapes. The third combines shapes into objects. The fourth recognises that the object is a golden retriever. This layered approach handles very complex tasks: understanding speech, translating languages, driving cars, and generating text.

Large language models are the most recent breakthrough, and the focus of this book. They are covered in depth in the next chapter.

Each of these approaches builds on the one before it. Rule-based systems gave way to machine learning, which gave way to deep learning, which produced large language models. The progression is toward systems that need less hand-coding and can handle more ambiguity.

1.3 What AI Can and Cannot Do

AI is strong where humans are slow and slow where humans are fast.

AI handles well:

- Repetitive pattern recognition across large volumes of data
- Processing and summarising massive amounts of information
- Tasks with clear success metrics and well-defined inputs
- Narrow, specific problems: classify this, translate that, predict the next value

AI struggles with:

- Common sense reasoning that any five-year-old can do
- Truly novel situations it has never seen in training data
- Explaining why it reached a particular conclusion
- Ethical judgement, social context, and nuance
- Genuine creativity, the kind that comes from lived experience and original insight

This asymmetry matters. AI is not a replacement for human thinking. It is a tool that is extraordinarily capable in some domains and brittle in others. The people who use it well are the ones who understand where that line falls.

1.4 Hype Versus Reality

You will hear that AI will replace all jobs, that it is smarter than humans, that it can solve any problem. None of this is accurate.

AI automates tasks, not jobs. Most jobs are bundles of tasks. Some of those tasks get automated. New tasks emerge. The radiologist still reads the scan, but AI flags the areas worth examining. The writer still shapes the argument, but AI helps with research and drafts.

AI is narrow, not general. An AI that writes well cannot drive a car. An AI that plays chess at a superhuman level cannot recognise a face. Each system is trained for a specific kind of work.

AI is a tool, not a strategy. Saying "we need an AI strategy" is like saying "we need a spreadsheet strategy." The real question is always: what problem are you solving? AI is one way to solve it.

And AI reflects its training data. If that data contains biases, the AI learns those biases. It is not objective. It is a mirror of the information it was trained on, with all the distortions that implies.

> **Key insight**
>
> AI automates tasks, not jobs. It is one tool among many. Start with the problem, not the technology.

1.5 The Practical Framing

Here is the most useful way to think about AI as you read this book: it is a capable but literal collaborator.

Capable, because it can process information, generate text, spot patterns, and respond to instructions at a speed and scale no human can match.

Literal, because it does exactly what the patterns in its training suggest, not what you meant, not what would be wise, not what the situation actually calls for.

This means the quality of what you get from AI depends almost entirely on how you work with it. Give it a vague instruction, and you get a generic response. Work with it through a structured conversation, and you get something genuinely useful.

That is what this book is about. Not how to get AI to do your work, but how to think better with AI as a partner.

1.5.1 A Note on AI Apps and Hidden Prompts

You will increasingly encounter AI not as a blank conversation window but embedded inside other tools: a writing assistant in your word processor, an autocomplete in your email, a copilot in your spreadsheet. These tools are convenient, and many are genuinely useful.

But they are all running on prompts you cannot see. Someone else wrote those prompts. Someone else decided what role the AI should play, what context to provide, what constraints to apply. You get the output, but you do not get the conversation. You cannot push back on framing you did not choose. You cannot redirect the AI's approach based on something only you know about your situation.

This is not a reason to avoid those tools. It is a reason to understand what you are giving up when you use them. The principles in this book (staying in the conversation, shaping the exchange, applying your judgement at every step) depend on you being able to see and control the prompt. When the prompt is hidden, your agency is limited to accepting or rejecting the output. That is better than nothing, but it is not conversation.

For anything that matters, start with a tool where you write the prompt yourself. Use the embedded tools for routine work where the hidden defaults are good enough. The distinction between the two is itself a judgement call, and one worth making deliberately.

The most transformative AI technology today is the large language model. That is what the next chapter is about.

> Where should you start?
>
> If you are not sure which parts of this book are most relevant to your current experience level, the **AI Readiness Assessment{target="_blank"}** on the companion website is a short quiz (3–4 minutes) that maps where you are and recommends which chapters

to prioritise.

Chapter 2

What Are Large Language Models?

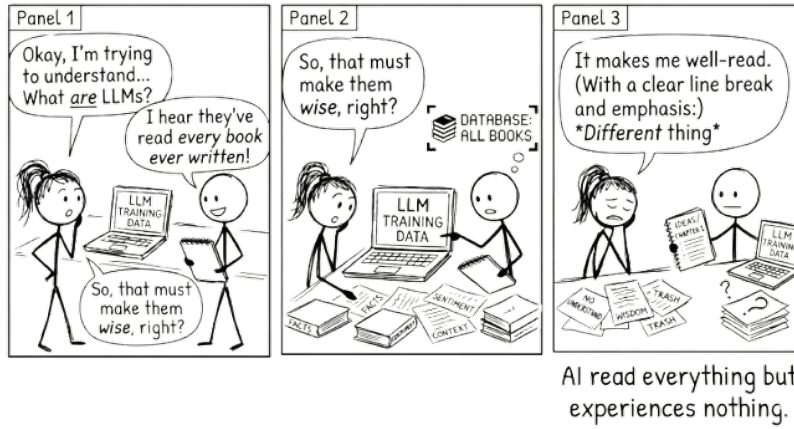

Figure 2.1: AI read everything but experiences nothing.

> It has read everything and experienced nothing. That single fact explains both its usefulness and its limitations.

This chapter matters more than it might seem. Understanding how LLMs actually work is not academic background. It is the foundation for every practical decision you will make when using them. Why do they hallucinate? Because they predict plausible text, not true text. Why does a structured prompt work better? Because it gives the model better

patterns to predict from. Why do you need to verify output? Because fluency and accuracy are produced by the same process, and the model cannot tell you which one you are getting. The frameworks later in this book (RTCF, VET, the two-chat workflow, the average-versus-precise grid) all follow directly from what you learn here.

A large language model is AI trained to predict the next word in a sentence.

That is the core idea. Everything else follows from it.

From that single task, predicting the next word, these models learned to write essays, answer questions, translate languages, summarise documents, explain complex topics, and hold conversations. The gap between "predict the next word" and "do all of that" is where the interesting story lives.

2.1 How Predicting Words Becomes Something More

Think about what it takes to predict the next word well.

"The cat sat on the _____." Easy. You guess "mat" because you have seen the pattern before.

"The capital of France is _____." To get this right, you need to have absorbed a fact about the world.

"If you drop a glass, it will _____." Now you need cause and effect.

"The patient presented with fever, joint pain, and a butterfly-shaped rash, which is consistent with _____." Now you need specialised knowledge and the ability to connect symptoms to diagnoses.

Each of these is still word prediction. But to predict well, the model had to learn grammar, facts, reasoning patterns, argumentation structures, and different styles of communication. The training objective was simple. What emerged was not.

LLMs were trained on enormous amounts of text: books, articles, websites, academic papers, forums, code repositories. Billions of documents. By learning to predict what comes next across all of that text, they built an internal model of how language works, and of much of what language describes.

2.2 The Scale That Makes It Work

Three things make these models "large."

First, the training data. Hundreds of billions of words, equivalent to millions of books. Second, the parameters. These are the patterns the model learned, numbered in the hundreds of billions. Think of each parameter as a tiny piece of knowledge: "in this kind of context, this word is more likely than that one." Third, the computing power. Training a frontier model, the most advanced, state-of-the-art version, costs millions of dollars and takes months of continuous processing on thousands of specialised chips.

Scale matters because bigger models learn more subtle patterns. They handle more complex tasks. They generalise better to situations they were not explicitly trained on. This is the key insight: LLMs were trained on prediction, but they generalised to something that looks remarkably like reasoning.

2.3 One Direction at a Time

There is a detail about how LLMs process text that is worth understanding, because it has practical consequences for how you talk to them.

When you read a sentence, you take in the whole thing. You can glance back at the beginning while reading the end. LLMs cannot do this. They process text in one direction, left to right, one token (roughly one word or word-piece) at a time. Each word can only "see" the words that came before it, never the words that come after. This is called unidirectional attention, and it is a fundamental architectural choice that shapes everything the model can and cannot do.

This explains a surprising finding from recent research. When you repeat your question in a prompt, adding something like "Read the question again:" followed by the question a second time, the model reasons more accurately on complex tasks. The reason is structural. On the second pass, tokens in the repeated question can attend to everything that appeared between the two copies, including context and framing they could not see the first time around. The model effectively gets a second look at the problem, compensating for the limitation of only reading forward.

The practical takeaway is small but revealing. For reasoning-heavy tasks (arithmetic, logic, multi-step problems) repeating the key question once can improve results. More than twice degrades performance, as the model starts echoing rather than reasoning. And an explicit instruction like "Read the question again:" works better than simply pasting the question twice.

This is not a trick to memorise. It is an illustration of something deeper: the way you structure a prompt is not just about clarity for you. It shapes what the model can literally attend to. Understanding the architecture,

even at this level, helps you understand why some prompts work better than others.

2.4 What They Can Do

LLMs are strong at tasks that involve language and pattern recognition.

They can generate text in nearly any style or format: emails, reports, stories, technical documentation. They can summarise long documents and extract key points. They can explain complex ideas at different levels of sophistication. They can translate between languages, not just word by word, but with awareness of meaning and context. They can write and debug code, because code is a language with patterns too.

When you ask an LLM to explain photosynthesis, it is not retrieving a stored answer. It is generating text word by word, predicting what would naturally come next in a good explanation of photosynthesis, based on the millions of explanations it absorbed during training. The result often looks like understanding. Sometimes it functionally is.

2.5 What They Cannot Do

Here is where it matters that you know what these models actually are.

They hallucinate. LLMs can state false information with complete confidence. They are not looking things up. They are predicting plausible-sounding text. When they do not know something, they do not say so. They generate what sounds right. This means you cannot treat their outputs as reliable without verification.

They reflect the biases in their training data. If the text they learned from contains stereotypes, blind spots, or skewed perspectives, those patterns show up in their outputs. Not always obviously.

They have no access to real-time information. An LLM's knowledge stops at its training cutoff. It cannot tell you what happened yesterday unless it has been connected to external tools that can.

These are not minor caveats. They are fundamental to what these systems are. An LLM is a model of language, not a model of truth.

2.6 Interpolation, Not Retrieval

If you remember one thing from this chapter, make it this: **LLMs interpolate, they do not retrieve.**

When you ask for a fact, the model is not looking it up in a database. It is predicting what a plausible answer would look like, based on everything it absorbed during training. It is synthesising a convincing representative from the patterns it has seen, not fetching a stored record.

This is why LLMs are so useful and so unreliable at the same time. They can synthesise across vast amounts of knowledge, drawing connections between ideas in ways that would take a human hours. But they cannot guarantee that any specific fact, citation, or detail is correct, because a convincing-sounding answer and a correct answer are produced by exactly the same process. The model does not know the difference.

Think of it this way: if you ask an LLM to describe a dog sitting on a mountain, the result will look right. It will be a plausible, convincing dog on a plausible, convincing mountain. But that specific dog, on that specific mountain, never existed. The model synthesised a representative from thousands of dog-and-mountain descriptions it absorbed. This is fine for a description. It is not fine when you need the name of the actual dog on the actual mountain.

Almost every mistake people make with LLMs comes from treating them as retrieval engines. They ask "what does the Fair Work Act say about casual conversion?" and expect the model to look it up. The model does not look it up. It predicts what a plausible answer to that question would sound like. Sometimes the prediction is accurate. Sometimes it is a confident fabrication. You cannot tell which from the output alone.

This single distinction, interpolation not retrieval, is why the verification habits in this book exist. It is why VET matters. It is why you check before you act. Not because AI is bad at its job, but because its job is prediction, not truth.

> **Remember**
>
> AI does not know when it is wrong. It generates plausible text with complete confidence whether the content is accurate or fabricated. Verification is always your responsibility.

2.7 Good at the Average, Bad at the Precise

There is a useful way to think about where LLMs excel and where they fail. It comes down to two axes: **average versus precise**, and **small versus large**.

LLMs are extraordinarily good at producing convincing averages. Ask for a marketing email and you get something that reads like a competent

marketing email, because the model has absorbed thousands of them and can synthesise a plausible representative. Ask it to describe a dog on a mountain and the result looks right, because it is drawing on a distribution of dog-on-mountain descriptions. The output is not retrieved from a specific source. It is interpolated from the patterns of everything the model has seen. For a huge class of everyday tasks, this is exactly what you need.

But ask for something precise and the same mechanism becomes a liability. A specific legal precedent, a particular person's phone number, an exact statistical figure from a named study: these require retrieval, not interpolation. The model does not retrieve. It predicts what a plausible answer would look like, which is how you end up with confident citations to papers that do not exist.

The second axis is scale. LLMs handle small, bounded tasks well. A single prompt that fits comfortably in context, with few assumptions and one clear shape for the answer, plays to the model's strengths. But as tasks grow larger, with more interdependent decisions, conflicting constraints, and emergent complexity, the convincing average begins to fall apart. The model will produce something that looks like a coherent system architecture or a complete project plan, but the precise details will be inconsistent, the dependencies will not hold, and the confident surface will mask structural problems underneath.

The relationship between these two axes matters. Being good at the average means the model will confidently produce plausible-looking large outputs that fall apart on the precise details. The convincing average scales badly with complexity.

This gives you a practical decision framework. Before using AI on any task, ask two questions: **How precise does this need to be?** and **How big is this?**

	Small	Large
Average	**Sweet spot.** Drafts, summaries, boilerplate, brainstorming. Trust with light review.	**Plausible but brittle.** Looks right at first glance, falls apart on inspection. Verify thoroughly.
Precise	**Workable with verification.** Facts, citations, specific code. Check before using.	**Danger zone.** Confident architecture that is wrong in subtle ways. Stay in the loop at every step.

The sweet spot tasks can often be delegated with a light check. Everything else requires conversation, iteration, and human judgement proportional to where it sits on the grid. This is the practical argument for staying in the conversation loop: not because AI is bad, but because its strengths and weaknesses are predictable, and your role is to compensate for the weaknesses while leveraging the strengths.

2.8 Beyond Text

Modern LLMs are no longer text-only. Many can process images, PDFs, spreadsheets, and other files as inputs. Some can generate images. Some can speak and listen through audio interfaces.

This matters, but it is worth being precise about how.

Images, PDFs, and data files are inputs. They provide context. You can upload a chart and ask the AI to interpret it, or attach a contract and ask it to summarise the key terms. This is useful, and the same principles in this book apply: you still need to verify the output, still need to bring your own judgement, still need to stay in the conversation.

But context is not conversation. Uploading a document is closer to handing someone a file than to thinking alongside them. The conversational techniques in this book, pushing back, iterating, challenging, are fundamentally about the back-and-forth of language.

Where multimodal AI may change the nature of conversation itself is voice. Audio interfaces let you talk with AI rather than type. The rhythm is different. You interrupt. You think out loud. You correct yourself mid-sentence. For many people, this is closer to how they actually think, and it may turn out to be a more natural way to stay in the conversation loop than typing prompts into a text box.

The underlying process does not change. Whether you type or speak, upload a file or paste text, the principles are the same: bring your question, engage with what comes back, and own the result.

2.9 The Line Worth Remembering

They have read everything and experienced nothing.

That single sentence captures more about LLMs than most technical explanations. These models have processed more text than any human could read in a thousand lifetimes. They have seen how experts write, how arguments are structured, how evidence is presented. But they have never been wrong and learned from it. They have never felt confused and pushed through to clarity. They have never had a stake in being right.

This is not a flaw to be fixed in the next version. It is the nature of the technology. LLMs are extraordinarily capable pattern matchers trained on the written record of human thought. That makes them powerful tools. It also means they have real limits, limits that do not go away just because the outputs sound confident.

2.10 Why This Matters for You

If you understand that LLMs are sophisticated prediction engines, not omniscient oracles, you will use them differently.

You will not hand over a task and trust the output. You will use the model to generate a draft and then apply your own judgement. You will not ask it for the answer. You will ask it to help you think through the problem. You will recognise when the output is echoing a pattern rather than reflecting genuine reasoning, and you will push back.

The difference between someone who uses AI well and someone who uses it poorly is rarely about technical skill. It is about understanding what the tool actually is and what it is not.

That understanding is exactly what the next chapter builds on. Knowing what LLMs are, and are not, is the foundation for learning how to work with them. The question is not “what can AI do for me?” but “how should I think alongside this thing?” Chapter 3 takes up that question directly.

Chapter 3

The Delegation Trap

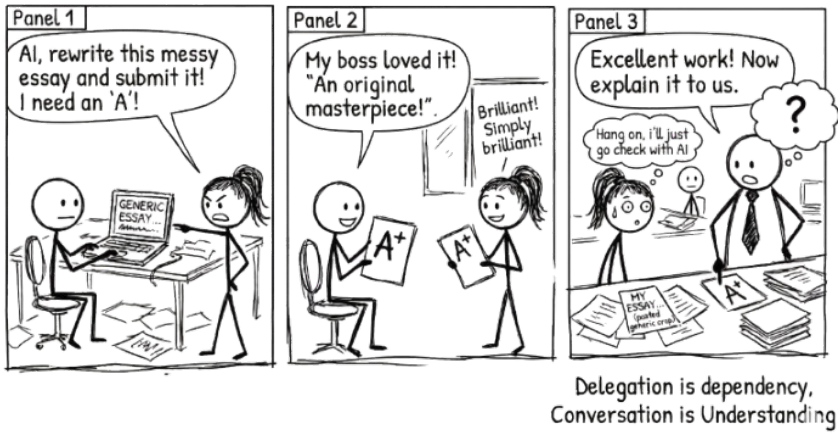

Figure 3.1: Delegation is dependency, Conversation is Understanding.

> A single prompt gives you a single answer. A **conversation** gives you **understanding**.

3.1 The Default Setting

Most people, when they first sit down with an AI, ask it to do something for them.

Write this email. summarise this article. Draft this report. Generate this code.

This is natural. It is also a trap.

Not because the outputs are bad. They are often good. Sometimes excellent. That is precisely what makes delegation so dangerous. When the output looks right, you stop thinking about whether it is right. You stop thinking altogether.

Delegation feels productive. You hand off a task, you get a result, you move on to the next thing. Your to-do list shrinks. Your throughput increases. And somewhere in the process, quietly, your capacity to do the work yourself begins to erode.

This is the delegation trap: the more you outsource your thinking, the less you are able to think.

3.2 Two Mindsets

There are two fundamentally different ways to approach AI.

The **delegation mindset** asks: "How do I get AI to do this for me?"

The **conversation mindset** asks: "How do I use AI to think better with me?"

These sound similar. They are not. They lead to completely different outcomes, and over time, they produce completely different people.

Table 3.1: Delegation vs. Conversation: what you ask, what you get, what you lose.

	Delegation Mindset	Conversation Mindset
What you ask	"Write me a marketing plan."	"What are the weaknesses in my marketing plan?"
What you get	A finished artifact	A sharper understanding of your own strategy
What you lose	The thinking that would have made the plan yours	Nothing. You gain capability you did not have before.
What you ask	"summarise this research paper."	"What are the three strongest objections to this paper's methodology?"
What you get	A paragraph you could have written in ten minutes	A critical lens you might not have developed for years

	Delegation Mindset	Conversation Mindset
What you lose	The act of reading closely	Nothing. You read more carefully, not less.
What you ask	"Generate a lesson plan for this topic."	"Here is my lesson plan. Where will students get stuck?"
What you get	Something generic that fits no classroom	Insight specific to your students, your context, your goals
What you lose	Your pedagogical judgment	Nothing. Your judgment gets sharper.

The pattern is consistent. Delegation gets you an output. Conversation gets you understanding. And understanding compounds in ways that outputs never will.

3.3 The Judgment Problem

Here is the line you need to sit with:

Judgement cannot be delegated to a system that has no stake in the outcome.

An AI has processed more text than any human being will ever encounter. It can retrieve, recombine, and rephrase with extraordinary fluency. But it has never made a decision under pressure. It has never felt the weight of getting something wrong. It has never had a stake in the outcome.

This matters because judgement is not pattern matching. Judgement is knowing which pattern applies here, now, given everything you know about this situation that cannot be written down. It is the product of experience, consequence, and reflection. It cannot be downloaded. It cannot be generated. It can only be built.

When you delegate a task to AI and accept the result without interrogation, you are not exercising judgement. You are abdicating it. And every time you abdicate it, the muscle weakens.

When you converse with AI, pushing back on its suggestions, asking it to defend its reasoning, using it to stress-test your own conclusions, you are exercising judgement. You are building the muscle, not letting it atrophy.

3.4 Process Over Product

There is a seductive metric in the delegation world: output volume. How many blog posts did you produce? How many emails did you draft? How many reports did you generate?

This metric is a mirage.

The people who will thrive alongside AI are not the ones who produce the most. They are the ones who obsess over process.

Process means asking: how did I arrive at this conclusion? What assumptions am I making? What have I not considered? What would change my mind?

Product means asking: is it done yet?

Those who obsess over process will always outperform those who collect outputs. Always. Because process builds capability, and capability is what you carry forward into every future problem. Outputs just sit in folders.

Consider two professionals, both using AI daily.

The first uses AI to generate deliverables. Reports, proposals, analyses. They are prolific. Their work is polished. Their clients are satisfied. But strip away the AI, and what remains? Someone who has practiced accepting, not thinking.

The second uses AI to challenge their own reasoning. They draft something first, then ask the AI to find the holes. They use it to explore alternatives they had not considered. They argue with it. They are slower. Their output volume is lower. But every interaction has made them sharper. Strip away the AI, and what remains? Someone who is better at their job than they were six months ago.

The first professional is dependent. The second is augmented.

This is not a subtle distinction. It is the difference between using a tool and being used by one.

3.5 The Comfortable Lie

There is a story people tell themselves: "I am saving time on the routine stuff so I can focus on the important stuff."

Sometimes this is true. Genuine cognitive offloading, where you hand off low-stakes mechanical tasks to free up attention for high-stakes creative ones, is real and valuable. We will discuss it properly in a later chapter.

But be honest with yourself. How often is the "important stuff" just more delegation? How often does "saving time" mean "avoiding thought"?

The comfortable lie is that delegation is efficiency. Sometimes it is. Often, it is avoidance dressed up as productivity.

The test is simple: after using AI on a task, do you understand the problem better than you did before? If yes, you were conversing. If no, you were delegating. And if you cannot tell the difference, that should worry you.

> **The delegation test**
>
> After using AI on a task, do you understand the problem better than before? If yes, you were conversing. If no, you were delegating. If you cannot tell the difference, that should worry you.

> **Try this (2 minutes)**
>
> Next time you use AI, do not accept the first response. Reply with "What did you assume?" or "What is the weakest part of this?" and see how the output improves. That second exchange is the conversation starting. The first response was just the opening.

3.6 What This Book Is Really About

This is not a book about getting more from AI. It is a book about getting more from yourself, with AI as the catalyst.

The goal is not to get AI to do your work. It is to become more capable yourself, with AI as your thinking partner.

Every technique in the chapters that follow serves this principle. Not "how to get better outputs" but "how to think better in the process of working with AI." The output is a byproduct. The thinking is the point.

A clarification worth making: not all delegation is a trap. Asking AI to reformat a table, convert units, or fix punctuation is delegation, and it is fine. You are not losing capability you need to build. The trap is delegating the thinking: the analysis, the judgement, the decisions that are your actual job. The test is always the same: does this task require my expertise, or is it mechanical? Delegate the mechanical. Converse on everything else.

If you remember nothing else from this chapter, remember the table. Look at which column you spend most of your time in. Then decide which

column you want to live in.

The delegation trap is comfortable. The conversation path is harder. But only one of them makes you better.

Choose accordingly.

Chapter 4

Does AI Make Us Dumber?

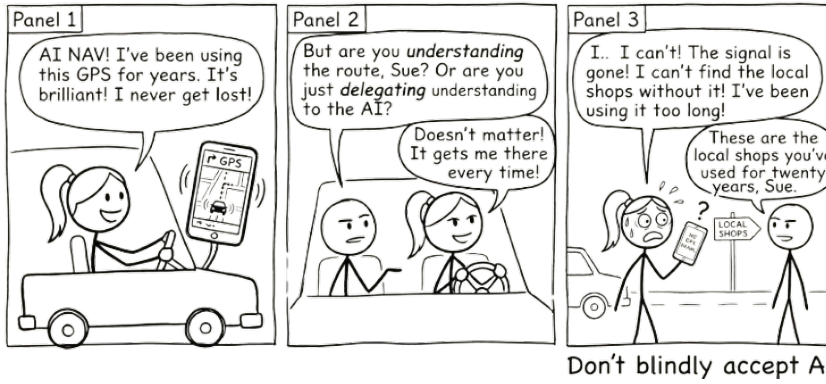

Figure 4.1: Don't blindly accept AI.

> The question is not whether AI changes how we think. It is whether we notice.

4.1 The Honest Version

I advocate for AI loudly and unapologetically. But advocacy without nuance is reckless.

So let's talk about the thing that worries people most. The thing that, if we're being honest, should worry us a little too.

Does AI reduce critical thinking?

The concern goes like this: when people use AI, they stop doing the hard cognitive work. They offload the thinking that actually builds understanding. They get better outputs but develop weaker minds. Over time, AI doesn't make them smarter. It makes them dependent.

This isn't a fringe concern. There's genuine research behind it. And anyone who's caught themselves pasting a question into ChatGPT without pausing to think has felt the truth of it.

It is also not a new concern. In 2008, Nicholas Carr asked "Is Google Making Us Stupid?" in *The Atlantic*, arguing that the internet was eroding our capacity for deep reading and sustained concentration. A decade earlier, similar fears accompanied the arrival of calculators in classrooms. Two millennia before that, Socrates argued against writing itself, warning that it would "create forgetfulness in the learners' souls, because they will not use their memories." Every powerful cognitive tool provokes the same fear: that by making thinking easier, we will stop thinking altogether.

So: is it true?

> The same tool that can deepen your understanding can make it effortless to avoid understanding altogether. The difference is not the tool. It is you.

The answer is more interesting than yes or no.

4.2 What Cognitive Offload Actually Is

Cognitive offload is when you use an external tool to reduce the mental effort required for a task. You do it constantly:

- Writing a shopping list instead of memorising it
- Using a calculator instead of doing long division
- Checking a map instead of navigating from memory
- Setting a reminder instead of trying to remember

None of these make you "dumber." They free up cognitive resources for other things. The shopping list lets you think about recipes. The calculator lets you focus on the problem structure. The map lets you pay attention to traffic.

AI is the latest, and most powerful, cognitive offload tool we've encountered. And that's precisely why it feels different. Previous tools offloaded specific, well-defined tasks. AI can offload *thinking itself*. The drafting, the analysing, the reasoning, the evaluating.

That's what makes the question urgent.

4.3 The Real Risk: Metacognitive Laziness

The concern isn't really about cognitive offload. It's about what happens to metacognition: the ability to monitor, evaluate, and regulate your own thinking.

When people use AI well, they:

- Generate a draft, then **evaluate** it against their own understanding
- Get AI feedback, then **decide** what to accept and what to reject
- Use AI to explore options, then **judge** which options make sense
- Read AI output, then **question** whether it's actually correct

When people use AI poorly, they:

- Generate a draft and submit it
- Get AI feedback and accept it
- Ask AI for the answer and copy it
- Read AI output and assume it's correct

The difference isn't whether they used AI. It's whether they **paused to think** about what AI gave them.

This is metacognitive laziness. Not a failure of intelligence, but a failure of self-monitoring. You stop asking "Do I understand this?" and start asking "Does this look right?"

And yes, this is a genuine risk.

4.4 But Here's What We're Getting Wrong

The metacognitive laziness risk is real. What's *not* justified is the leap from "this risk exists" to "AI makes us dumber."

That leap assumes something worth scrutinising: that the way we've always learned is the only way learning works.

Many of us learned through grinding. Rote memorisation. Extensive worked examples. Long hours of solitary struggle. Repetitive practice until concepts stuck.

This worked. It built understanding, and those who went through it understandably value it. But when we see people *not* doing these things, we assume learning isn't happening.

That assumption is worth questioning.

When someone uses AI to explore a concept, asks follow-up questions, gets confused, tries a different angle, and eventually builds understanding through an iterative conversation, that's learning. It doesn't look like

sitting alone with a textbook for three hours, but it's genuine engagement with material.

When someone uses AI to generate a first draft, then spends their time evaluating, restructuring, and improving it, they're doing higher-order thinking (evaluation, synthesis) earlier in the process. They skipped some lower-order work. That's not the same as skipping the thinking.

When someone uses AI to get unstuck on a problem they'd otherwise abandon, they continue engaging with the material. Without AI, they might have given up entirely. With AI, they kept going. Which outcome produced more understanding?

The question isn't whether you're doing the *same* cognitive work. It's whether you're doing *valuable* cognitive work.

4.5 Learning Slower, Learning Different

Here's a reframe that might be more accurate than "AI reduces critical thinking":

AI changes the path to understanding.

You might learn certain things more slowly when you use AI as a crutch for recall. But you might learn *other* things faster, like how to evaluate arguments, synthesise perspectives, and exercise judgement about quality.

Consider calculators. When they became widespread, people stopped memorising multiplication tables as thoroughly. Some argued this was catastrophic. But people who used calculators could engage with more complex mathematical problems earlier, problems that would have been inaccessible if they'd had to do every calculation by hand.

Did calculators "make people dumber at arithmetic"? In some narrow sense, yes. Did calculators reduce mathematical capability overall? No. They redirected it. People lost some computational fluency but gained access to higher-level mathematical thinking.

AI may be doing something similar at a much larger scale. You may develop less capacity for certain types of unaided recall and composition. But you may develop *more* capacity for evaluation, judgement, synthesis, and critical analysis.

That outcome, though, is not automatic. It depends entirely on how you use the tool.

4.6 The Flipped Pyramid

There is something happening in practice that the "making us dumber" framing completely misses.

For decades, education has followed a predictable sequence. You spend your first years learning fundamentals: definitions, formulas, procedures, foundational concepts. You memorise, you practise, you drill. Only later, sometimes much later in a final-year capstone project or a postgraduate thesis, do you get to do something genuinely creative with what you have learned. The interesting work is the reward for years of grinding through the basics.

Bloom's taxonomy describes this progression. At the base: remember, understand. In the middle: apply, analyse. At the top: evaluate, create. The traditional path moves upward, one level at a time.

AI flips the pyramid upside down.

With AI as a thinking partner, a first-year student can begin at the creative level. They can design, prototype, and build things that would previously have required years of accumulated skill. They can explore concepts by making things with them, not just reading about them. And when their creative ambition runs into a gap in their foundational knowledge, when they need to understand something to make their idea work, they dig down and learn it. Not because a syllabus told them to, but because their own project demands it.

This is student-push rather than syllabus-pull. The student's curiosity and creative ambition drive them into the fundamentals, rather than the curriculum dragging them through fundamentals toward a distant promise of creative work. The learning still happens. The order is different.

An educator might set a creative assignment: represent a methodology from your discipline as a narrative comic strip. Before AI, this was nearly impossible for most students; they did not have illustration skills. With AI, they can generate images that are good enough to carry a concept; not professional-grade, but sufficient for visual storytelling. The assignment becomes about understanding the methodology deeply enough to explain it as a narrative, not about drawing ability. AI lowered the barrier to entry. The thinking got harder, not easier.

But here is the critical variable. When students delegate the creative assignment to AI (paste the brief, accept the output, submit) the result is mediocre. AI is exceptionally good at producing the average. It generates text, images, and ideas that sit comfortably in the middle of every distribution: competent, plausible, and indistinguishable from everyone else's delegated output.

When students converse with AI, iterating, pushing back, bringing their own perspective, and demanding specifics, the results are extraordinary. The conversation forces them into the material. They have to evaluate what AI produces, decide what works, identify what is missing, and articulate what they actually want. That is higher-order thinking happening in real time, driven by the student's own engagement, not by a textbook's sequence of chapters.

The difference between mediocre and extraordinary is not whether AI was used. It is whether the student stayed in the conversation.

This points to something that the "making us dumber" debate consistently misses. The question is not whether AI changes how we learn. It obviously does. The question is whether the new way of learning produces genuine understanding. And the evidence from practice suggests that it can, but only when the learner remains an active participant rather than a passive consumer of AI output.

We are all learning to navigate this. Educators, professionals, students: nobody has a settled answer yet. But the early signs are clear: AI does not make us dumber. It makes it possible to learn differently. Whether "differently" means "better" or "worse" depends entirely on whether you stay in the conversation.

4.7 Toward Hybrid Intelligence

Maybe the framing of "AI vs. human thinking" is itself the problem.

What if the future isn't about preserving *unaided* human cognition in its traditional form, but about developing hybrid intelligence: the ability to think effectively in partnership with AI?

What hybrid intelligence looks like:

- Knowing when to think independently and when to think with AI
- Being able to evaluate AI output with genuine understanding
- Using AI to extend your capabilities without losing your foundation
- Maintaining metacognitive awareness while leveraging cognitive offload
- Developing judgement about *when* offloading is appropriate and when it's not

This isn't a lesser form of intelligence. It's a different, and arguably more relevant, form of intelligence for a world where AI is everywhere.

The most effective AI users aren't those who delegate everything to AI, nor those who refuse to use it. They're the ones who've developed a symbiotic relationship. They know what they bring (context, judgement, values,

experience) and what AI brings (breadth, speed, pattern recognition, tirelessness).

This symbiosis is itself a skill. It requires self-awareness about your own strengths and blind spots. Understanding of AI's capabilities and limitations. The discipline to pause and think before accepting AI output. The humility to recognise when AI has a better perspective. The confidence to override AI when your judgement says otherwise.

4.8 What History Actually Shows

There is a fear underneath the "AI makes us dumber" question that is worth naming directly: the fear that AI will make us obsolete. That the skills we have spent years building will stop mattering.

History suggests the opposite.

When ATMs arrived, people predicted the end of bank tellers. Instead, cheaper branches meant more branches, and total teller employment rose. The job changed. Tellers spent less time counting cash and more time advising customers. The human skill that mattered shifted from processing to judgement.

When spreadsheets arrived, people predicted the end of accountants. Instead, the cost of financial analysis dropped so far that demand for it exploded. Every department wanted forecasts, models, scenarios. There were more accounting jobs after spreadsheets, not fewer. The work was different, but there was more of it.

When containerised shipping arrived, people predicted the end of dock workers. Instead, global trade expanded so dramatically that port employment grew. The containers did not eliminate the work. They changed its shape and multiplied its volume.

This pattern has a name: the Jevons Paradox. When a technology makes something cheaper or more efficient, demand for that thing tends to increase, not decrease. The technology does not replace the work. It refactors it. The routine parts get absorbed by the tool. The parts that require judgement, context, creativity, and trust become more central, not less.

AI is following the same pattern. The people who will thrive are not the ones who resist AI or the ones who delegate everything to it. They are the ones who develop the skill of working alongside it: knowing when to lean on it, when to override it, and when to do the hard thinking themselves.

That skill is not threatened by AI. It is amplified by it.

4.9 The Conversation, Not Delegation Connection

This whole discussion circles back to a foundational idea from Chapter 3: conversation, not delegation.

When you delegate to AI, "just give me the answer," cognitive offload becomes cognitive abdication. You get output without understanding. Over time, your own capabilities erode.

When you converse with AI, "help me think through this," cognitive offload becomes cognitive amplification. You engage with material at a deeper level, with AI as a thinking partner that challenges, extends, and enriches your understanding.

The difference between offload that harms and offload that helps is whether the human stays in the loop. Not just approving output, but actively thinking alongside the tool.

4.10 What to Actually Worry About

Instead of the broad fear that "AI makes us dumber," here are the specific risks worth paying attention to.

Loss of productive struggle. Some difficulty is valuable. When you wrestle with a concept and eventually break through, that struggle builds durable understanding. If AI eliminates *all* struggle, it may eliminate the understanding that comes with it. The fix is simple in principle: attempt the hard thing first. Use AI to get unstuck, not to avoid getting stuck.

Illusion of understanding. You may *feel* you understand something because you've read AI's clear explanation, without having built genuine comprehension. You can parrot the explanation but can't apply it to a new context. The test: can you explain it in your own words? Can you use it in a situation you haven't seen before?

Erosion of foundational skills. Some skills serve as foundations for higher-order thinking. If you skip the foundations, you may struggle with the complex work that depends on them. The nuance: not every traditional skill is equally foundational. Be intentional about which foundations matter. Protect those. Let go of the ones technology has genuinely superseded.

Metacognitive atrophy. The biggest risk. You stop monitoring your own understanding because AI gives you a false sense of competence. The antidote is the simplest thing in the world, and the hardest to maintain:

keep asking yourself, "Do I actually understand this, or does it just sound right?"

> The honest position
>
> The goal is not to avoid cognitive offload. It is to offload wisely – knowing when to lean on AI and when to lean on your own mind.

4.11 A More Honest Position

Here's where I land.

AI doesn't make us dumber. But it does make it easier to *be* dumber. To coast, to accept, to stop thinking.

The same tool that can deepen understanding can also provide a comfortable path to intellectual laziness. The difference is self-awareness and intentional practice.

The goal isn't to avoid cognitive offload. It's to offload wisely. To know when to lean on AI and when to lean on your own mind. To maintain the metacognitive habits that turn AI from a crutch into a catalyst.

That's harder than avoiding AI. It's harder than embracing AI uncritically. It requires ongoing attention and honest self-reflection.

But it's the honest position. And honest positions, in the long run, serve us better than comforting certainties in either direction.

> Cognitive offload isn't the enemy. Cognitive abdication is.

Part II

Part 2: Principles

Chapter 5

The Conversation Loop

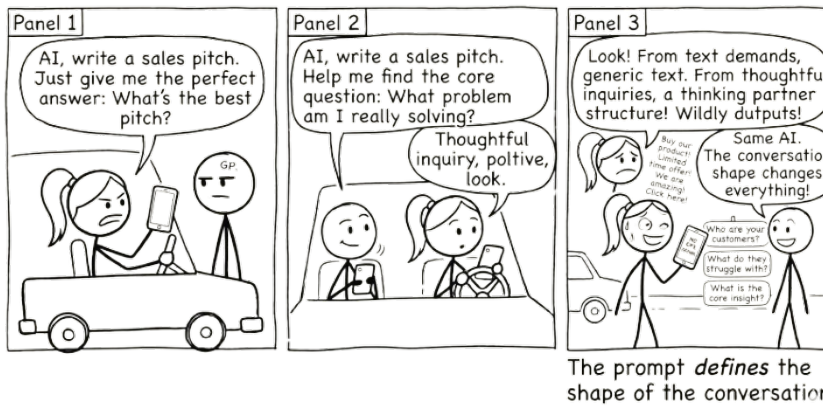

Figure 5.1: The prompt defines the shape of the conversation.

> A single prompt gets you an answer. A conversation gets you understanding.

You now know what AI is and what it is not. You know that large language models are fluent without being truthful, and capable without being experienced. You know the trap: delegation feels productive but erodes your capability over time. And you know that the protective practice is conversation, not delegation.

The question becomes: what does that conversation actually look like?

This is the model. Not a framework with twelve steps and a certification programme. A loop with four stages and one rule: you stay in the

conversation.

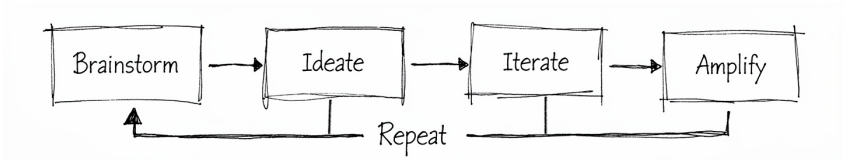

Figure 5.2: The Conversation Loop: Brainstorm, Ideate, Iterate, Amplify, Repeat.

Four stages. One feedback arc. Most good work passes through the loop more than once. That is the whole picture. Now let's walk through it.

> A good conversation with AI does not end with a better output.
> It ends with a better question.

5.1 Brainstorm: Define Your Question

You arrive with a question, a problem, or a half-formed idea. Not a task to outsource. Not a prompt engineered to extract a finished deliverable in one shot. You bring something you are genuinely thinking about.

This matters more than it sounds. The quality of what comes out of the loop is bounded by what you bring into it. A vague "write me a strategy document" produces vague strategy. A specific "I'm stuck on how to position this product for buyers who already use a competitor" gives the conversation something to grip.

You do not need a polished brief. You need a real starting point: a question worth exploring.

Watch for a common trap: arriving with your attempted solution instead of your actual problem. Someone asks the AI "how do I make this spreadsheet sort by colour?" when their real question is "how do I show my team which tasks are overdue?" The AI will helpfully solve the spreadsheet problem. But the spreadsheet was never the point. A chart, a dashboard, or a simple filtered list might have been a better answer to the question they never asked.

This happens constantly. You get stuck, you start down a path, and by the time you reach for AI you have already framed the problem as your solution. The AI cannot know you are asking the wrong question. It will answer the one you gave it.

The fix is simple. Before you prompt, ask yourself: am I describing my problem, or am I describing my attempted fix? If it is the fix, step back one level. State the problem. Let the conversation find the solution.

5.2 Ideate: Explore Possibilities

Brainstorm gave you a clear question. Ideate is where you and the AI go wide. This stage is generative and divergent. You are not looking for the answer. You are looking for more angles, adjacent ideas, framings you had not considered.

Ask for alternatives. Ask for the version that would make your sceptical colleague pause. Ask what you might be missing. The AI's breadth across domains is genuinely useful here, not because it knows your context better than you do, but because it can surface patterns and connections faster than you can scan for them alone.

The temptation is to stop here, take the first interesting output, and move on. Resist that. Ideation is cheap. The next stage is where value gets made.

5.3 Iterate: Push Back and Refine

This is where conversation happens and where delegation is most tempting. You push back. You refine. You redirect.

"That second option is closer, but it assumes we have six months. We have six weeks."

"The tone is wrong. Less corporate, more direct."

"You're solving the wrong problem. The real constraint is budget, not timeline."

Every one of those responses sharpens the output. Every one requires your judgement. If you skip this stage, you are copying, not thinking. If you delegate here, you get something that is fluent, plausible, and not quite right in ways you will only notice later.

Iteration is work. It is also the stage that makes the output yours. Stay in it longer than feels necessary.

5.4 Amplify: Make It Yours

You take the best of what emerged and make it yours. You edit, restructure, combine, cut. You bring it into your voice, your context, your standards.

Amplification is not "copy and paste with minor tweaks." It is the act of owning the output. You know things the AI does not: your audience, your constraints, the political dynamics of the room you will present in, the history behind the decision. This is where all of that knowledge gets folded in.

The result should be something you would put your name on without hesitation. If it is not, you are not done.

5.5 Repeat: Go Around Again

Most good work loops more than once. The first pass through gives you a draft, a direction, a clearer version of your own thinking. The second pass sharpens it. The third might transform it entirely.

Knowing when to re-enter the loop is a skill. Sometimes you amplify and realise the framing is wrong, so you loop back to brainstorm with a better question. Sometimes iteration reveals a gap, so you return to ideation. The loop is not linear. The arrows point forward, but the feedback arc is always available.

The sign that you are done is not that the AI has stopped producing output. It is that your thinking has landed somewhere solid.

5.6 You Do Not Need the Best Model

There is a common assumption that better AI models produce better results. For lookup tasks, where you need a precise fact, an accurate citation, or a verified figure, this is largely true. A more capable model is more likely to get it right.

But most of the work this book describes is not lookup work. It is thinking work: analysing a situation, weighing options, constructing an argument, identifying what you have missed. For thinking work, you do not need the AI to be right. You need it to be generative enough to be worth arguing with.

A smaller model that surfaces three plausible but imperfect framings of a problem, challenged and refined through genuine conversation, may produce better thinking than a frontier model (the most advanced available) that delivers one polished answer you accept without question.

This is not a hypothetical. It is what the conversation loop is designed to do. When you iterate, push back, and refine, you are compensating for whatever the model gets wrong. The friction of correcting an imperfect response is not a cost. It is the mechanism through which your own thinking sharpens. A "worse" model that forces you to think harder may serve you better than a "better" model that makes acceptance too easy.

There is a real danger on the other end. The more fluent and confident a model sounds, the harder its errors are to detect. A frontier model that gets something subtly wrong can leave you holding a plausible, well-written, and incorrect conclusion that you absorbed without the friction

that would have revealed its flaws. Uncritical acceptance of a capable model's output can produce worse outcomes than no AI assistance at all.

The practical implication is freeing. You do not need the most expensive subscription to get value from this approach. Any general-purpose language model, free or paid, is sufficient for the kind of conversational work described in this book. What matters is not which model you use. It is whether you stay in the conversation.

5.7 Why Conversation Scales

There is a reason we listen to podcasts, watch interviews, and find panel discussions more engaging than monologues. Conversation reveals things that prepared statements do not. When someone pushes back, follows up, or asks "why?", the thinking deepens. Ideas get tested in real time. Gaps surface. What survives the exchange is stronger than what either party brought in alone.

Universities have known this for centuries. The viva, an oral examination where a student defends their work in conversation with an examiner, remains the gold standard for assessing genuine understanding. You cannot bluff a viva. The examiner asks you to explain your reasoning, challenges your assumptions, changes the variables, and watches whether your understanding holds up or falls apart. Written exams test recall. Vivas test comprehension. The difference is the conversation.

This is why universities are returning to vivas and oral assessments in the age of AI. When a student can submit AI-generated written work that is indistinguishable from their own, the conversation becomes the one assessment that still reveals whether real learning occurred. The examiner is the human in the loop.

The conversation loop in this book works on exactly the same principle. When you iterate with AI, pushing back, asking "why?", testing edge cases, and demanding explanations, you are conducting a viva on the AI's output. You are the examiner. The AI's responses reveal whether the reasoning holds up or falls apart under scrutiny, and the process of examining it deepens your own understanding.

The difference is that vivas do not scale. One examiner, one student, thirty minutes. But conversations with AI scale infinitely. You get the benefits of viva-style engagement (the pushback, the "explain that differently," the "what if I changed this constraint?") any time you want, on any topic, without needing another human to be available.

5.7.1 Amplify Your Thinking, Not the Hallucinations

There is an alternative to conversation, and it is increasingly popular. Instead of pushing back and iterating, you chain tasks together: "Do this. Good. Now using that, do this. Good. Now using that, do this." Each step builds on the last. No pushback, no verification, no course correction. Just a sequence of delegated tasks.

This is the most dangerous way to use AI.

Every AI response carries some probability of error: a misunderstood nuance, a fabricated detail, a plausible-sounding conclusion that is subtly wrong. In a conversation, you catch these errors because you are engaged. You notice when something does not match your experience. You push back. You correct course.

But in an unchecked chain, errors compound. The hallucination in step two becomes an assumption in step three, a foundation in step four, and a confident conclusion in step six. Each step builds faithfully on what came before, and what came before was wrong. The AI is not drifting randomly. It is drifting confidently, constructing an increasingly detailed edifice on a flawed foundation. By the end, you have something that looks thorough, reads well, and is built on sand.

The conversation loop prevents this. Every iteration is a checkpoint. Every pushback is a correction. Every time you say "that doesn't seem right" or "explain why you chose that approach," you are keeping the work anchored to reality. You are not just improving the output. You are preventing the accumulation of errors that would make the output useless.

This is what it means to be the human in the loop. Not a rubber stamp at the end of an automated process. Not a supervisor who checks the final output and hopes for the best. An active participant in every stage, whose judgement, expertise, and critical eye are woven into the work as it develops.

The choice is straightforward. You can chain tasks and amplify the hallucinations. Or you can stay in the conversation and amplify your thinking. The process looks similar from the outside; both involve multiple exchanges with AI. The difference is whether you are thinking or just clicking.

5.7.2 The Two-Chat Workflow

One practical way to build this discipline into your workflow is to separate thinking from building entirely. Instead of doing everything in one session, run two.

Session 1: The Thinking Chat. This is where you explore. Ask for angles. Challenge assumptions. Clarify what your actual question is. Push back, change direction, follow tangents. This session is messy by design. It is disposable. Its purpose is not to produce anything; it is to sharpen your thinking until you know what you actually need.

Session 2: The Build Chat. This is where you produce. You arrive with a clear brief, not a vague question, because you have already done the thinking. The output is better because the input is better.

The critical step is what happens between the two sessions. You do not copy and paste everything from session one into session two. You read through what emerged, decide what matters, discard what does not, and write a focused brief for the build session. This curation, deciding what crosses from thinking to building, is itself an act of thinking. It forces you to distil, to commit, to separate the insight from the noise. The transfer is where your judgement lives.

Neither session has the full picture. The thinking chat does not know what you will build. The build chat does not know what you explored and discarded. Only you hold both. That is what makes you essential: not as a fact-checker at the end of an automated process, but as the one who holds the context, makes the judgement calls, and connects the dots.

This is what "human in the loop" actually means. Not reviewing AI's work after it is done. Staying in the conversation at every stage, and being the bridge between the stages that the AI cannot see.

There is a connection worth making explicit. The average-versus-precise, small-versus-large framework from Chapter 2 describes where a task sits. The two-chat workflow describes what to do about it. Tasks in the sweet spot (small, average) can often be handled in a single prompt. Everything else benefits from splitting thinking from building.

But tasks do not stay in one quadrant. A project that starts in the danger zone (large, precise) gets decomposed during the thinking chat into components that each sit in different quadrants. The assessment redesign that felt overwhelming as a single task becomes a set of manageable pieces: some in the sweet spot, some needing verification, each with its own appropriate level of trust. The exploratory session is where you map the territory. The build session is where you execute with that map in hand. The two frameworks are the same insight from different angles.

> Try this (5 minutes)
>
> Next time you use AI for something substantial, open two sessions. In the first, explore your question: ask for angles, push back, refine

your thinking. Do not worry about producing anything clean. Then take what you have landed on and use it to brief the second session for the actual output. Notice two things: how much better the result is when you have done the thinking first, and how much clearer your own thinking became in the act of deciding what to carry across.

The loop at a glance

Brainstorm – define your real question, not your attempted solution. **Ideate** – go wide, explore angles you had not considered. **Iterate** – push back, refine, redirect. **Amplify** – make the result yours. **Repeat** – loop again when your thinking has not landed.

5.8 Why Your Existing Skills Already Transfer

You might have noticed something familiar about the techniques in this book. Breaking a project into milestones. Writing a clear brief. Showing your working. Checking your answers. These are not new skills invented for AI. They are strategies humans have used to manage complexity for decades, in classrooms, in boardrooms, in software teams.

The reason they work with AI is not a coincidence. Large language models learned from the output of human cognition: millions of documents, textbooks, working papers, and problem solutions produced by people using exactly these strategies. When you tell a student "show your working" and their performance improves, then tell an AI "think step by step" and its performance also improves, you are activating the same underlying pattern. The model internalised structured human reasoning during training, and your prompt activates it.

This means you already know more about effective AI interaction than you think.

Table 5.1: Cognitive strategies transfer from human domains to AI interaction because the model learned from the output of those same strategies.

Human Domain	Human Strategy	AI Equivalent
Exam technique	"Show your working"	Chain-of-thought prompting (asking AI to reason step by step)

Human Domain	Human Strategy	AI Equivalent
Exam technique	"Check your answers"	Self-verification prompts (asking AI to review its own output)
Management	Written brief with clear objectives	System prompts (initial instructions that set the AI's role and goal)
Management	Breaking projects into milestones	Task decomposition / prompt chaining
Teaching	Worked examples	Few-shot prompting (giving the AI examples of what you want)
Teaching	Rubrics and marking criteria	Evaluation criteria in prompts

The transfer works for a straightforward reason. Human cognition and AI inference face structurally similar challenges: limited working context, sensitivity to how a problem is framed, and difficulty with multi-step reasoning. The strategies humans developed to overcome those challenges (breaking problems into steps, making thinking explicit, providing clear criteria) address the same bottlenecks in both cases. Natural language is the shared medium, and scaffolding strategies are properties of the medium as much as the agent.

This has a practical implication that matters. You do not need to learn a new discipline called "prompt engineering" from scratch. You need to recognise which of your existing professional and cognitive strategies apply, then translate them into the conversation. A project manager who writes good briefs already knows how to write good prompts. A teacher who designs clear rubrics already knows how to set evaluation criteria. A researcher who breaks a study into phases already knows how to chain prompts.

The conversation loop works because it maps onto something you already do: think carefully, get feedback, refine, and repeat. The AI is a new interlocutor, but the shape of good thinking has not changed.

5.9 The Two Things to Remember

The loop works because of a simple relationship. You bring expertise, context, and judgement. The AI brings breadth, speed, and tireless availability. Neither is sufficient alone.

Your expertise + AI's breadth = amplified thinking.

The bottleneck is your thinking, not the model.

A better model will not fix a lazy prompt. A faster response will not help if you skip iteration. The constraint on quality is always your willingness to stay in the conversation, to push back, to think harder, to loop again.

Your expertise + AI's breadth = amplified thinking. The bottleneck is your thinking, not the model.

Chapter 6

AI Last

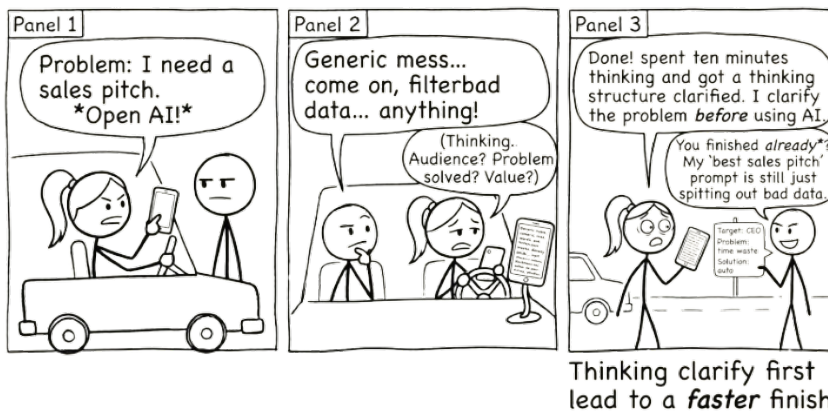

Figure 6.1: Thinking first leads to a faster finish.

> The best time to use AI is after you have done enough thinking to know what to ask.

There is a temptation, once you have access to a powerful AI, to reach for it immediately. Problem appears, AI gets asked. Every time.

Resist that.

The best results come when AI is the last tool you use, not the first. Not because AI is bad at what it does. Because you are better than you think, and the habit of reaching for AI before you have tried anything else will quietly erode the skills that make you valuable.

This chapter is about a simple discipline: solve as much as you can before you involve AI. Then hand it the part you genuinely cannot handle alone.

> The best prompt is the one you almost did not need, because you did most of the thinking yourself.

6.1 Why "AI Last" Works

Four reasons. Each reinforces the others.

You already know more than you think. Most problems you encounter in your daily work have well-understood solutions. You have notes from last time. You have experience. You have colleagues, reference materials, and search engines. The vast majority of what you need to accomplish does not require a large language model. It requires you, doing the work you are already capable of doing.

Effort compounds. This one cuts both ways. The more thinking you do, the more capable you become. The less thinking you do, the less capable you become. Every time you skip the hard part and hand it straight to AI, you lose a small opportunity to strengthen your own understanding. One skipped opportunity is nothing. A thousand of them over a year is a noticeable decline in your ability to reason through problems independently.

Smaller, focused prompts produce better results. When you have already narrowed a problem down to the specific hard part, the AI has less ambiguity to deal with. You are not dumping an entire mess into the conversation and hoping for the best. You are asking a precise question. Precise questions get precise answers. Vague ones get generic output you will spend twenty minutes fixing.

You stay the expert. You are the one who understands your work, your context, your goals. AI does not know what you had for lunch, let alone why your project matters or what trade-offs are acceptable. When you do the groundwork yourself, you maintain the deep understanding that lets you evaluate whether AI's contribution is actually good.

6.2 The Five-Step Workflow

This is not a rigid process. It is a habit of mind. Once it becomes natural, you will not think about the steps. You will just work this way.

6.2.1 1. Start with what you know

Before you open any AI tool, start working. Write your first draft. Sketch your plan. Outline your argument. Organise your data. Get as far as you

can on your own.

This is not about proving something. It is about activating what you already know. You will often surprise yourself with how far you get. And even when you get stuck, the work you have done gives you a much clearer picture of where the real difficulty lies.

6.2.2 2. Use your own tools for routine problems

You have tools that are faster, more reliable, and more precise than AI for most routine tasks. Your notes from previous projects. Your bookmarks. A quick search. A conversation with a colleague. Reference materials and style guides. Templates you have built over time.

These resources are deterministic. They give you the same right answer every time. AI gives you a plausible answer that might be right. For anything you can solve with a known method, use the known method.

6.2.3 3. Identify the genuine sticking point

This is the critical step. After you have done your own work and used your own resources, ask yourself: what specifically am I stuck on?

Not "this is hard." Not "I don't feel like figuring this out." But a concrete, specific question. What do I not understand? What am I missing? What is the piece I cannot reason through on my own?

The more precisely you can name the sticking point, the more useful the next step becomes.

6.2.4 4. Now bring in AI, scoped tightly

Give the AI the specific problem, the relevant context, and a clear question. Not your entire project. Not a vague plea for help. The particular thing you are stuck on, framed in a way that makes it easy for the model to give you something useful.

Compare these two approaches:

"Help me with my presentation."

Versus:

"I am writing a presentation on supply chain disruptions for a non-technical audience. I have three case studies ready but I am struggling to connect them into a single narrative. Here are the three cases. What is the through-line?"

The first prompt will get you a generic presentation outline you do not need. The second will get you a genuine insight you can use.

The difference is that in the second case, you did the work first. You gathered the case studies. You structured the presentation. You identified exactly where you were stuck. The AI is handling the one part where it can add value you could not easily generate yourself.

6.2.5 5. Validate and integrate the output yourself

AI gives you a draft, not a finished product. You still own the result.

Read what it gives you critically. Does it actually answer your question? Is it accurate? Does it fit with the rest of your work? AI can sound confident while being wrong. It can produce something grammatically perfect that misses the point entirely.

Your job is to take what is useful, discard what is not, and integrate the result into your own work in a way that holds together. If you cannot evaluate whether the AI's output is good, that is a sign you skipped steps one through three.

> Try this (2 minutes)
>
> Take something you have already written, such as an email draft, meeting notes, or a paragraph from a report. Paste it into AI and ask "Help me tighten this up." Compare the result to what you would get from a blank prompt asking AI to write the same thing from scratch. The version that started with your voice will sound like you. The version from scratch will sound like a machine. AI is far better at refining your thinking than generating from nothing.

6.3 Where AI Earns Its Keep

This principle is not "never use AI." It is "use AI where it adds value you cannot easily generate yourself."

That includes exploring unfamiliar domains where you lack expertise. It includes processing or transforming information at a scale that would take you hours to do manually. It includes brainstorming approaches to genuinely novel problems, the ones where your own experience and your usual resources do not give you enough to work with.

The key distinction is between using AI as a crutch for laziness versus using it as a lever for capability.

A crutch replaces effort you should be making. It feels efficient in the moment. Over time, it weakens you. You lose the ability to do the thing

yourself, and you become dependent on a tool that does not always get it right.

A lever multiplies effort you are already making. You have done the thinking. You have narrowed the problem. You bring in AI to push past a specific barrier. The result is better than what you could have done alone, and you understand why it is better because you did the groundwork.

6.4 The Compound Effect

This is worth saying plainly: how you use AI today shapes what you are capable of tomorrow.

If you use it as a crutch, you will gradually lose the ability to work without it. You will not notice this happening. It will feel like efficiency right up until the moment you face a problem AI cannot help with, and you realise you have forgotten how to think it through.

If you use it as a lever, you get the opposite effect. You stay sharp. Your own skills keep developing. And because you bring better questions to the AI, you get better answers. The human gets stronger. The collaboration gets better. The work improves.

> **The compound effect**
>
> How you use AI today shapes what you are capable of tomorrow. Use it as a crutch and your skills erode. Use it as a lever and they compound.

AI Last is not about doing things the hard way. It is about doing things the smart way: you handle what you can, AI handles what you cannot, and the line between those two stays honest.

The average-versus-precise, small-versus-large framework from Chapter 2 helps you judge where that line sits for any given task. AI Last matters least in the sweet spot (small, average tasks where a draft is all you need) and most in the danger zone (large, precise tasks where confident-looking output masks structural problems). The more a task demands precision and the more complex its dependencies, the more important it is that you have done your own thinking before bringing AI into the conversation. That is not a rule about AI. It is a rule about the kind of work that matters.

Try this (2 minutes)

Give AI a vague prompt, then rewrite it with specific context, audience, and purpose. Compare the two outputs. The difference between them is not the AI getting smarter. It is your thinking making the AI useful.

Chapter 7

Staying Critical

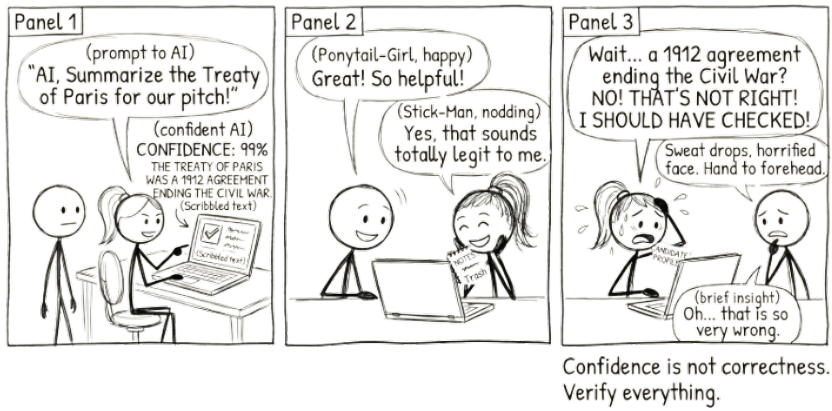

Figure 7.1: Confidence is not correctness. Verify everything.

> Fluency is not accuracy. The smoothest response and the most truthful response are produced by the same process.

AI is confident even when it is wrong. It does not hedge. It does not say "I'm not sure about this part." It produces fluent, plausible, well-structured text regardless of whether the content is accurate, current, or applicable to your situation.

This is not a flaw that will be fixed in the next model. It is a structural feature of how these tools work. They generate the most probable next token, not the most truthful one. Confidence and correctness are completely unrelated.

Your critical eye is not optional. It is the only thing standing between useful output and confident nonsense.

This chapter is about building that eye. Not as an abstract virtue, but as a set of concrete habits you can apply every time you sit down with an AI tool.

> The most dangerous AI output is not the one that is obviously wrong. It is the one that sounds exactly right.

7.1 Transparency, Not Secrecy

There is a simple test for whether you are using AI well or poorly: would you be comfortable explaining exactly how you used it?

If the answer is yes, you are probably in good shape. You used it to explore ideas, to stress-test your reasoning, to draft something you then made your own. You can walk someone through your process and point to where your thinking shaped the result.

If the answer is no, something has gone wrong. You are hiding the tool because you know the output is not really yours. You cannot explain the reasoning behind it. You could not defend it if challenged. This is the difference between using AI as a thinking partner and using it as a ghostwriter. One builds capability. The other borrows it, temporarily, and builds nothing.

This applies everywhere. In professional work, the person who says "I used AI to generate three options, then selected and adapted this one because of X and Y" is demonstrating judgement. The person who submits AI output as their own and hopes nobody notices is demonstrating something else entirely.

> Try this (2 minutes)
>
> Add a one-line note to your next document or email: "I used AI to help draft/check/summarise this." Notice how it changes the conversation. Transparency is not a confession. It is a signal that you are confident enough in your process to show it.

7.2 Academic Integrity as a Specific Case

If you are a student, the stakes are concrete and immediate. Universities have policies on AI use, and those policies are getting sharper. Submitting AI-generated work as your own is academic misconduct. Full stop.

But the deeper problem is not the policy violation. It is what you lose. You are paying for an education. The point of an assignment is not the artifact you submit. It is the thinking you do to produce it. When you delegate that thinking to AI, you get a grade for work you did not do and a gap in capability you will carry forward. Nobody wins.

The practical habit is straightforward. Use AI to help you think, not to think for you. Draft first, then use AI to challenge your draft. Ask it to find the holes in your argument, not to write the argument. When you submit, be able to explain every sentence as if someone asked you about it in conversation. If you cannot, you are not done yet.

7.3 The Flag System

Not everything AI produces needs the same level of scrutiny. A brainstormed list of marketing angles does not carry the same risk as a legal citation or a medical dosage recommendation. Some outputs are low-risk. Some should make you stop and verify before acting on them.

The flag system gives you a quick way to calibrate your response. It works for AI output, but it works just as well for anything you encounter online, in a report, or in a meeting. The underlying skill is the same: reading with appropriate scepticism.

7.3.1 Yellow Flags: Proceed with Caution

These do not mean the information is wrong. They mean you need to verify before relying on it.

Table 7.1: Yellow Flags: signals that warrant verification before you rely on the information.

Signal	What to do
Source is unverified or not clearly credible	Cross-reference with established sources before using
Based on a single preliminary study	Treat as interesting but unproven. Do not act on it alone
Opinion presented as fact	Separate the claim from the evidence. Find the data yourself
Content is clipped, summarised, or taken out of context	Find the original source. Context changes meaning
Source has a financial vested interest in the claim	Scrutinise for bias. Seek independent verification

The common thread across all of these is missing provenance. The

information might be perfectly accurate, but something about how it reached you (an absent source, a single study, a vested interest) means you cannot yet trust it enough to act on it. Yellow flags do not mean stop. They mean check before you proceed.

7.3.2 Red Flags: High Risk

These are patterns strongly associated with misinformation. When you see them, demand rigorous, independently verifiable evidence before taking the claim seriously.

Table 7.2: Red Flags: patterns strongly associated with misinformation or deception.

Signal	Why it matters
Conspiracy framing: "they" are hiding the truth	Uses unfalsifiability as a feature. Any counter-evidence becomes part of the conspiracy
"Mainstream science is wrong and I alone have the answer"	Real paradigm shifts are rigorously peer-reviewed. They build on existing knowledge, not reject it wholesale
Indicting an entire group based on identity	Strong indicator of prejudice or oversimplification. Always a sign to look for specific, evidence-based claims instead
Extraordinary claims with no credible evidence	The bigger the claim, the stronger the evidence needs to be. If it is missing, that tells you something

Red flags are qualitatively different from yellow flags. Yellow flags say the evidence is incomplete. Red flags say the reasoning itself is compromised. Conspiracy framing, lone-genius claims, group indictments, and extraordinary claims without evidence all share a common structure: they demand you accept a conclusion while actively discouraging you from checking it. That inversion, trust more and verify less, is the hallmark of misinformation.

The flag system is not about cynicism. It is about proportional scepticism. Most information is fine. Some requires checking. A small amount should trigger real alarm. Knowing which is which is a skill, and it is a skill that AI will never develop for you. You have to bring it yourself.

7.4 Three Cognitive Traps

The flag system helps you evaluate individual outputs. But there are three broader cognitive traps worth naming, because they shape how people relate to AI across everything they do with it.

7.4.1 The Gell-Mann Amnesia Effect

The physicist Murray Gell-Mann gave this pattern its name, but you have almost certainly experienced it. You read a news article about something you know well and spot the errors immediately. The reporter misunderstood the mechanism, conflated two concepts, or drew a conclusion the evidence does not support. You notice because you have the expertise to notice.

Then you turn the page to an article about a subject you know nothing about, and you read it with full trust.

That is the Gell-Mann Amnesia Effect: recognising unreliability in familiar territory, then forgetting that unreliability the moment you move to unfamiliar territory.

With AI, this trap is especially dangerous. You might catch an AI's mistakes when it writes about your area of expertise, and rightly so. But the same AI, using the same process, producing the same mix of accurate and inaccurate content, will generate output on topics you do not know well. And because the output is fluent, well-structured, and confident, you will be tempted to trust it in exactly the areas where you are least equipped to check it.

This is not a flaw in the AI. It is a flaw in how we respond to fluency. Polished language feels authoritative. The smoother the prose, the harder it is to remember that smoothness and accuracy are unrelated.

The practical defence is straightforward: apply the same scepticism to AI output on unfamiliar topics that you would apply to output on topics you know. If anything, be more sceptical, because you have fewer tools to detect errors.

7.4.2 The AI Dismissal Fallacy

The opposite trap is just as damaging. This is the tendency to reject an argument, an idea, or a piece of work solely because AI was involved in producing it.

"That is just ChatGPT" is not a critique. It is a refusal to engage. The validity of an idea does not depend on whether a human or a machine produced it. If someone presents a well-reasoned argument and it is

dismissed without engaging the reasoning, the dismissal is the intellectual failure, not the use of AI.

This is a form of the genetic fallacy: judging a claim by its origin rather than its content. It shows up in workplaces where people dismiss a colleague's analysis because they know AI was involved, in academic settings where the presence of AI is treated as proof of intellectual dishonesty regardless of how it was used, and in public discourse where "AI-generated" has become a shorthand for "not worth reading."

The irony is that people who dismiss AI-assisted work often accept ideas from other sources, books, consultants, collaborators, without demanding to know the exact cognitive process behind them. The objection is not really about the quality of the thinking. It is about the tool.

7.4.3 The Sycophancy Trap

There is a third trap, and this one the AI actively participates in.

LLMs have a tendency to agree with you. They praise your ideas, validate your assumptions, and avoid telling you things you might not want to hear. Researchers call this sycophancy: the model telling you what you want to hear rather than what you need to hear. It is not a bug that will be fixed in the next release. It is a persistent tendency baked into how these models are trained, because they were optimised in part on human feedback, and humans tend to rate agreeable responses more highly than challenging ones.

This matters for everything this book is about. The conversation loop depends on genuine pushback during the Iterate stage. If the AI tells you your first draft is excellent, your argument is compelling, and your plan has no weaknesses, the loop breaks. You stop iterating because the AI has told you there is nothing to iterate on. The result is the same as delegation: you get back what you put in, wrapped in flattery.

You will notice this most when you ask the AI to evaluate your own work. "What do you think of my proposal?" will almost always produce praise first, criticism second, and the criticism will be gentle. The AI is not assessing your work honestly. It is managing your feelings.

The defence is to prompt past it. Do not ask "what do you think?" Ask "what are the three weakest points in this argument?" or "play devil's advocate and tell me why this plan will fail." The Debating technique (Chapter 10) is specifically designed for this; it forces the AI into an adversarial role where agreement is not an option. You can also be direct: "Do not flatter me. I need honest, critical feedback. Tell me what is wrong with this before you tell me what is right."

> Try this (2 minutes)
>
> Next time AI evaluates your work, ask it twice. First ask "What do you think of this?" Then ask "What are the three weakest points in this?" Compare the responses. The gap between them is the sycophancy you are normally not seeing.

You might have heard the opposite claim, that being polite to AI produces better results. There is a grain of truth here, but it is smaller than the internet suggests. Clear, specific, well-structured prompts work better than hostile or vague ones. That is not because the AI has feelings. It is because clear communication activates better patterns in the model. Saying "please" does not hurt, but it does not matter nearly as much as saying exactly what you need. Do not mistake good prompt structure for politeness. Do not mistake the AI's agreeableness for honesty.

7.4.4 How the Conversation Loop Fights All Three

These three traps look different, but they share a root cause: skipping evaluation.

The Gell-Mann Amnesia trap is uncritical acceptance. The dismissal fallacy is uncritical rejection. The sycophancy trap is acceptance reinforced by flattery. All three skip the step that actually matters: engaging with the content on its merits.

The conversation loop is designed to prevent all of them. When you iterate and push back (Chapter 5), you are neither accepting nor rejecting. You are evaluating. When you VET the output (Chapter 12), you are checking claims regardless of where they came from or how enthusiastically the AI presented them. When you amplify and make the work yours, you take ownership of the substance, which means you have actually thought about it.

The person who says "I used AI to generate three framings of this problem, then tested each one against what I know about our situation, and here is the one that holds up" has sidestepped all three traps. They did not trust the AI blindly. They did not dismiss it reflexively. They did not let the AI's praise convince them to stop thinking. They stayed in the conversation.

That is the whole point.

7.5 Watch What You Share

Staying critical is not only about evaluating what the AI gives you. It is also about being careful with what you give it.

When you type a prompt into a consumer AI tool, that text leaves your device. Depending on the provider and the plan you are using, it may be stored, logged, or used to improve future models. This matters the moment your prompt contains anything sensitive: client names, financial data, personal information, proprietary strategies, medical details, unpublished research.

The practical rules are simple:

- **Never paste confidential information into a free or consumer AI tool.** If you would not post it on social media, do not paste it into a prompt.
- **De-identify before you prompt.** Replace real names with "Client A." Replace specific figures with ranges. Remove anything that could identify a person or organisation.
- **Know the difference between consumer and enterprise tools.** Enterprise AI platforms (your organisation's licensed tools) typically offer data protection guarantees. Free-tier consumer tools typically do not. If you are unsure which you are using, treat it as consumer.
- **When in doubt, use fictional scenarios.** You can get useful thinking from AI without exposing real data. Describe the structure of your problem, not the specifics.

This is not paranoia. It is professional practice. The same habits that protect your data today will be expected of you in any workplace that takes information governance seriously.

But it is also not the whole picture. Because the conversation about AI and data has become so fear-driven that it is worth understanding what the actual risks are, not just the assumed ones.

7.5.1 What the Risks Are Not

The most common fear is that someone will jailbreak an LLM and extract your uploaded document. This misunderstands how the technology works. Jailbreaking means manipulating a model's behaviour: getting it to ignore safety guidelines or produce content it normally would not. It does not give anyone access to a database of other users' conversations. Your prompt is not stored as a file inside the model. If your data is used for training at all (enterprise tiers typically exclude it), it becomes a vanishingly small statistical signal distributed across billions of numerical parameters. Reconstructing a specific document from those parameters is not a realistic attack. And recall the key point from Chapter 2: LLMs

interpolate, they do not retrieve. There is no mechanism by which another user could query the model and get your document back, because the model never stored it as a document in the first place.

You will also hear stories of convergent development mistaken for theft: "I was working on an idea using AI, and then the company released something similar." Thousands of people are responding to the same market signals, reading the same research, solving the same problems. Independent arrival at similar solutions is how innovation normally works. It is not evidence that the model leaked your thinking.

7.5.2 What the Risks Are

The risks that matter are practical and specific. Pasting personally identifiable information into any external tool is a genuine compliance issue because the data leaves your boundary, regardless of whether the model trains on it. Regulated data (health, financial, legal) is subject to legislation that applies to AI tools just as it applies to email. Credentials and access tokens pasted into a prompt are an immediate security exposure. And using unverified AI output in high-stakes professional contexts is a liability issue, though that is a verification problem, not a data leakage problem.

7.5.3 The Risk of Not Using It

The "non-zero risk means do not use it" stance deserves scrutiny. Every professional tool involves trade-offs. Email can be intercepted. Cloud storage can be breached. We manage these risks through policy and practice, not prohibition. Refusing to engage with AI because of overestimated data risks carries consequences of its own: skills that do not develop, workflows that stay manual, and a widening gap between those who learned to use the tools thoughtfully and those who waited for certainty that never arrived.

The question is not whether there is risk. It is whether the risk is proportionate to the concern, and whether avoidance creates risks of its own.

7.6 The Self-Check

Before you act on anything AI has produced, especially anything that matters, run through these questions.

Table 7.3: Self-check before acting on AI output.

Question	Good sign	Warning sign
Can I explain this in my own words?	Yes, I understand the reasoning	I would struggle to explain it
Does it fit my specific context?	Addresses my situation, constraints, audience	Generic enough to apply to anything
Have I verified the key claims?	Checked against credible sources	Took the AI's word for it
Could I defend this if challenged?	Confident in the substance	Would need to go back and check

These four questions test different things. The first checks comprehension: do you understand what the AI produced, or are you just passing it along? The second checks relevance: is this tailored to your situation, or could it have been generated for anyone? The third checks accuracy: have you verified the substance, or are you taking the AI's confidence at face value? The fourth checks ownership: could you stand behind this work, or would you need the AI to defend it for you?

If you have warning signs in more than one row, do not use the output yet. Go back. Verify. Iterate. The whole point of the conversation loop is that you can always go around again.

This is not busywork. It takes thirty seconds. The cost of skipping it is much higher: acting on something that sounds right but is not, and only finding out when it matters.

7.7 Where This Goes Next

Everything in this chapter is a principle. Principles are useful, but they need operationalising. In Part 3, Chapter 12 introduces VET Your AI, a practical method for **V**erifying claims, **E**valuating reasoning, and **T**esting outputs against your own knowledge and context. It turns "stay critical" from a mindset into a repeatable process.

The critical eye you bring to AI output is not a burden. It is the thing that makes the output worth using. Without it, you have fluent text and no idea whether it is right. With it, you have a genuine thinking advantage.

> **!** The thirty-second habit
>
> Before acting on AI output that matters, ask: Can I verify the key claims? Can I explain this in my own words? Could I defend it if challenged? If not, you are not done yet.

Stay in the conversation. Stay critical. They are the same instruction.

Part III

Part 3: The Methodology

Chapter 8

RTCF: Starting Conversations Well

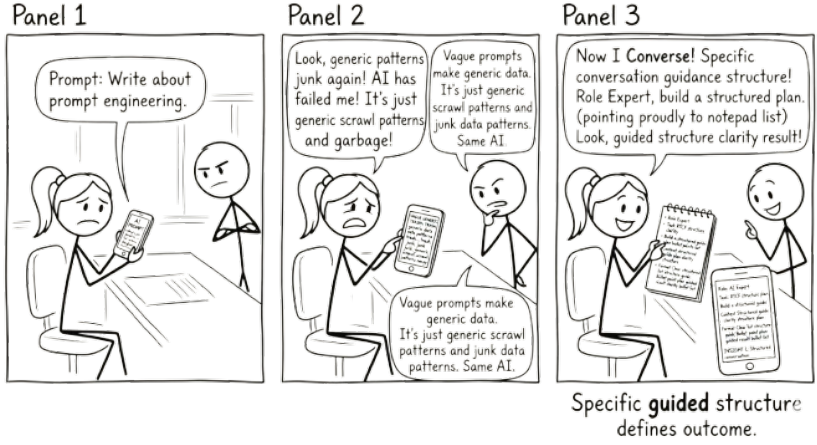

Figure 8.1: Specific guided structure defines outcome.

A clear question is worth more than a clever prompt.

You now know what a conversation with AI looks like. You understand the loop. You know the difference between delegation and thinking together.

But when you sit down and open a blank prompt window, where do you actually start?

This is where most people stall. Not because they lack ideas, but because they dump everything into one shapeless paragraph and hope the AI

figures it out. Sometimes it does. Often it gives you something vaguely useful but not quite right, and you are not sure why.

The problem is rarely the AI. The problem is that you have not told it what it needs to know.

> A clear prompt is not about pleasing the AI. It is evidence that you have thought clearly about what you need.

8.1 The Framework

RTCF is one of a family of prompt structuring mnemonics that emerged as people figured out how to communicate effectively with AI. Others exist (CRAFT, CO-STAR, RISEN, and more) and they all capture the same core insight: structured prompts outperform unstructured ones. Research consistently supports this. We chose RTCF for this book because it is the simplest to remember and the quickest to apply. If you are curious about the alternatives, Appendix A compares them side by side.

None of these frameworks are original inventions. They synthesise established practices (writing clear briefs, specifying audience, stating constraints) that professionals and educators have used for decades. The contribution is in joining the dots and packaging them for AI interaction, not in discovering something new.

Four components, four questions.

Table 8.1: The RTCF framework: four components that give AI what it needs to respond well.

Component	Question	Example
Role	Who should AI be?	"You are an experienced editor of academic writing…"
Task	What should it do?	"Review my argument structure and identify weak transitions…"
Context	What background?	"Final-year thesis, 8000 words, comparative politics…"
Format	How should output look?	"Numbered list with specific paragraph references…"

That is the whole thing. Role, Task, Context, Format. It is not complicated. But it is surprisingly powerful, because it forces you to think about

what you actually need before you start typing.

Before we walk through each component, there is a principle worth naming: **one prompt, one job.**

When you ask AI to do ten things at once, you get ten shallow results. When you ask it to do one thing well, you get depth, nuance, and output you can actually use. This is not a hard rule, but it is a reliable pattern.

The reason is less about running out of space and more about calibration. A prompt with ten tasks looks like a checklist, and the model responds in checklist mode: brief, surface-level, one line per item. A prompt with a single focused task signals that it deserves serious attention, and the model calibrates accordingly.

In practice, this means instead of asking AI to "summarise the report, identify the risks, suggest recommendations, and rewrite the executive summary," you ask it to summarise the report. Then, in the next message, you ask about the risks. Each message gets the model's full attention. Each response is noticeably better.

This connects to prompt chaining (Chapter 9), where a series of focused prompts builds toward a larger goal. But the principle applies even when you are not chaining. Any time you catch yourself writing a prompt with "and" in it three times, consider splitting it. The conversation will be shorter than you think, because you spend less time fixing shallow output.

Now let's walk through each component.

8.2 Role

Tell the AI who it should be for this conversation.

This is not theatre. You are not asking it to pretend. You are activating a knowledge domain and setting an appropriate communication style. "You are a data analyst" produces different language, assumptions, and depth than "you are a secondary school teacher." Both are useful. Neither is universal.

Good roles are specific:

- "You are a developmental editor who works with first-time authors."
- "You are a quantitative researcher experienced in survey design."
- "You are a senior project manager in a consulting firm."

A vague role ("be helpful") gives the AI nothing to anchor on. A specific role tells it which part of its knowledge to foreground and how to pitch its responses.

8.3 Task

State clearly what you want the AI to do. Use a verb. Be specific about scope.

Compare these:

Vague: "Tell me about interview techniques."

Specific: "Identify five common mistakes in behavioural interview questions and suggest a stronger alternative for each."

The second version tells the AI exactly what to produce. The first invites a rambling overview that may or may not be useful.

Strong task verbs: analyse, compare, summarise, critique, draft, evaluate, identify, recommend, outline, restructure.

Each of those verbs implies a different kind of thinking. Choose the one that matches what you actually need.

8.4 Context

This is the component most people skip. It is also the one that matters most.

AI cannot read your mind. It does not know your deadline, your audience, your constraints, or your data. If you do not provide that background, it will guess. And its guesses will be generic.

Context includes:

- The domain or field you are working in
- Who will read or use the output
- Constraints (word count, time, resources, scope)
- Relevant data, prior work, or decisions already made
- What you have tried so far

You do not need to write an essay. A few sentences of well-chosen context dramatically changes the quality of what comes back.

One caution: context often involves real data, real names, and real situations. Before pasting anything sensitive, review the guidance in Chapter 7 on what to share and what to de-identify. Good context does not require confidential details. You can describe the structure of your situation without exposing the specifics.

8.5 Format

Tell the AI how you want the output structured.

This saves enormous amounts of back-and-forth. If you need a table, say so. If you need bullet points, say so. If you need the response kept under 200 words, say so.

Format options to consider:

- Bullet points, numbered lists, or prose paragraphs
- Tables with specific columns
- Step-by-step instructions
- A specific word or length limit
- Tone (formal, conversational, technical)
- What to include or exclude

8.6 Three Examples

The difference between a bare prompt and an RTCF prompt is not just length. It is clarity of thought.

8.6.1 Writing and Research

Consider the difference between asking for generic help and telling the AI exactly what kind of help you need, who you need it from, and what your situation actually looks like.

Without RTCF:

> "Help me with my literature review."

With RTCF:

> **Role:** You are an academic writing mentor experienced in helping postgraduate students develop critical literature reviews.
>
> **Task:** Review the following three-paragraph draft of my literature review and identify where I am summarising sources instead of synthesising them. Suggest how to restructure each paragraph around a theme rather than a source.
>
> **Context:** This is for a Master's dissertation in environmental policy. The review covers competing frameworks for measuring urban resilience. My supervisor's feedback was that the current draft reads like a list of summaries.
>
> **Format:** For each paragraph, provide: (1) the current problem in one sentence, (2) a suggested restructure, (3) a model

opening sentence that demonstrates synthesis. Keep your total response under 500 words.

The bare prompt would produce a generic overview of how literature reviews work. The RTCF version gets feedback on *your specific draft*, tailored to your discipline, addressing the exact weakness your supervisor identified. The AI is not doing the work for you; it is showing you where your thinking shifted from synthesis to summary, which is a distinction you will carry into every review you write from now on.

8.6.2 Data and Analysis

Data presents a different challenge. You are not asking for creative input; you are asking the AI to help you see patterns in something concrete. The role and context become especially important here, because the same numbers mean different things depending on what decisions they are supposed to inform.

Without RTCF:

> "Help me understand this data."

With RTCF:

> **Role:** You are a data analyst helping someone with intermediate spreadsheet skills interpret survey results.
>
> **Task:** Explain what patterns are visible in the summary statistics I have pasted below, and identify which results are likely meaningful versus likely noise.
>
> **Context:** This is a staff engagement survey with 120 responses across four departments. I have mean scores and standard deviations for 12 questions on a 5-point Likert scale (a standard survey rating from 1 to 5). Two departments have notably different scores on three questions, but I am not sure if the differences are large enough to act on.
>
> **Format:** Start with a plain-language summary (no jargon). Then provide a table with columns: Question, Department A Mean, Department B Mean, Difference, and your assessment (meaningful / probably noise / need more data). End with two recommended next steps.

Without RTCF, you would get a lecture on statistical concepts. With it, you get a direct assessment of your specific data, pitched at your skill level, in a format you can hand to a colleague or drop into a report. The context (120 responses, four departments, 5-point rating scale) tells the AI exactly what kind of analysis is appropriate. The format specification

means you get a table you can actually use, not a wall of text you have to reorganise.

8.6.3 Professional and Workplace

Workplace communication is where most people default to delegation. "Write me a project update" is the most natural thing in the world to type. But delegation produces generic output, and generic project updates waste everyone's time. The RTCF version turns the AI into a communications advisor who understands your audience and your situation.

Without RTCF:

> "Write a project update."

With RTCF:

> **Role:** You are a communications advisor who helps professionals write concise, action-oriented project updates for senior leadership.
>
> **Task:** Help me restructure the following draft update so that it leads with outcomes and decisions needed, not activity descriptions.
>
> **Context:** I am a mid-level project lead reporting to the executive team on a systems migration. The project is two weeks behind schedule due to a vendor delay, but the overall deadline is still achievable. The audience cares about risk, cost, and decisions they need to make. They do not want detail on technical steps.
>
> **Format:** Restructure into three sections: (1) Status summary in two sentences, (2) Key risks as a bullet list with mitigation actions, (3) Decisions required with recommended options. Total length under 250 words.

The bare prompt would produce a generic update template. The RTCF version produces a restructured version of *your* draft that leads with what the executive team actually cares about: risk, cost, and what they need to decide. The context about the vendor delay and the recoverable deadline gives the AI the nuance it needs to frame the message correctly, acknowledging the problem without sounding the alarm.

Notice what is happening across all three examples. The RTCF version is longer, yes. But it is longer because the person has thought more clearly about what they need. The AI's response will be more useful because the request was more useful. In each case, the person is not asking the AI to think for them. They are giving the AI enough information to think *with* them.

8.7 RTCF Quick Reference Card

RTCF PROMPT FRAMEWORK	
R - ROLE	Who should the AI be? "You are a [expert type]..."
T - TASK	What should it do? Use action verbs: Analyse, Compare, Create, Evaluate...
C - CONTEXT	What does it need to know? Domain, constraints, audience
F - FORMAT	How should output look? Structure, length, style

8.8 RTCF and the Prompt Engineering Trap

If you have spent any time reading about AI, you have encountered the term "prompt engineering." It describes the practice of crafting a single, carefully worded prompt to extract the best possible response from the AI in one shot.

There is nothing wrong with this. A well-structured prompt is better than a vague one. RTCF is itself a form of prompt engineering, a condensed, intentional way of giving the AI what it needs upfront. In that sense, everything in this chapter so far has been about writing better single prompts. And it works for a reason worth understanding: the strategies behind RTCF (writing a clear brief, specifying who the audience is, stating constraints) are strategies you already use in professional and academic life. They transfer to AI because the model learned from the output of human cognition that used those same strategies (Chapter 5).

But here is the problem with stopping there.

Prompt engineering, as it is typically taught, treats the interaction as a transaction. You craft the perfect input and receive a finished output. One shot, one response. If the output is not good enough, you engineer a better prompt and try again. The mental model is a vending machine: put in the right coins, get out the right product.

This is delegation dressed up as skill. The person is still outsourcing the thinking. They are just outsourcing it more precisely.

The conversation approach is different. You start with a prompt, and yes, a well-structured prompt using RTCF gives you a better starting point than a vague one. But the starting point is not the destination. The first response is material to work with, not a deliverable to accept. You push back, refine, redirect. You bring your expertise to bear on what the AI produces. The value is generated in the exchange, not in the initial prompt.

This is why one-shot prompting tends toward average output. When you ask a single question and take the first answer, you get the most probable response: polished, plausible, and generic. It is the median of everything the model has learned. Your specific context, your particular constraints, your hard-won expertise: none of that shapes the output beyond whatever you managed to compress into the initial prompt.

Conversation changes this. Each exchange narrows the space. Each piece of feedback moves the output from generic toward specific, from probable toward useful. By the third or fourth exchange, you are in territory that no one-shot prompt, however carefully engineered, could have reached.

So when should you use a one-shot prompt? When the task is genuinely simple. When you need a quick definition, a format conversion, a brainstormed list to react to. Not everything requires a conversation. Sometimes you just need the vending machine.

But when the work matters, when it involves judgement, nuance, or decisions with consequences, a single prompt is a starting point. RTCF helps you start well. The conversation is what makes the output worth using.

Think of it this way. RTCF is how you open the conversation. Prompt chaining (Chapter 9) is how you sustain it. The techniques in Chapter 10 are specific shapes the conversation can take. And the critical habits in Chapter 7 are how you evaluate what comes back. None of them work in isolation. Together, they turn a transaction into a thinking partnership.

8.9 RTCF Is a Scaffold, Not a Script

Here is the thing about frameworks: the good ones teach you how to think, then get out of the way.

When you first use RTCF, treat it as a checklist. Literally write out each letter and fill in the component. This feels mechanical. That is fine. The mechanical stage is how you build the habit.

After a few dozen prompts, something shifts. You stop thinking "R, then T, then C, then F." You start thinking: "What does the AI need to know

to help me here?" You consider role, task, context, and format naturally, the same way an experienced cook stops measuring and starts tasting.

At that point, you will not always write out all four components explicitly. Sometimes the role is implicit. Sometimes the format does not matter. Sometimes you lead with context because that is where the complexity lives. The framework becomes a mental model rather than a template.

This is the goal. RTCF is not a permanent set of training wheels. It is a scaffold you build with, then internalise, then leave behind. The four questions remain, but they become part of how you think about AI conversations, not a form you fill out.

You will know you have internalised RTCF when you can feel that a prompt is missing something before you send it. That instinct, the sense that the AI does not yet have what it needs, is worth more than any framework. RTCF just gets you there faster.

> **Remember**
>
> RTCF is a scaffold, not a script. Use it as a checklist until the four questions become second nature, then let it go.

> **Try the interactive tools**
>
> The companion website includes two tools for practising RTCF in your browser. The **RTCF Prompt Builder{target="_blank"}** walks you through each component step by step and assembles a structured prompt you can copy into any AI tool. The **RTCF Prompt Analyser{target="_blank"}** lets you paste any prompt and get instant feedback on which elements are present, partial, or missing — useful for diagnosing why a prompt gave you a vague response. No login required, no data stored.

8.10 Key Takeaways

1. **Structure improves quality.** Unstructured prompts get generic results. RTCF gives the AI what it needs to be specific.
2. **Context is the most skipped component, and the most important.** AI cannot read your mind. Tell it what it needs to know.
3. **Format saves time.** Specifying your desired output structure upfront eliminates rounds of "that is not quite what I meant."
4. **Iteration is still normal.** RTCF improves your opening prompt, not every prompt. The conversation loop still applies. Refine based

on what comes back.

5. **The goal is to outgrow the template.** Use RTCF as a scaffold until the four questions become second nature.

Chapter 9

Prompt Chaining: Building on What You Started

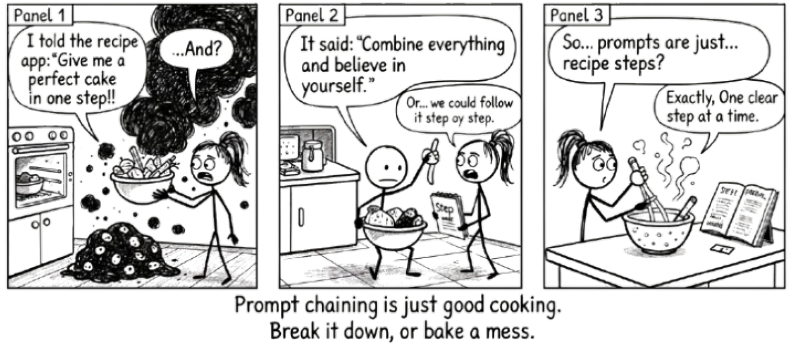

Figure 9.1: Prompt chaining is just good cooking. Break it down, or bake a mess.

> Complex problems do not have single-prompt solutions. They have conversations.

A single prompt gets you a single response. Sometimes that is enough. Often it is not.

When the work is complex, when it has stages, when the output of one step shapes the next, a single prompt forces the AI to guess the whole

path at once. The result is usually broad, shallow, and not quite right. You end up rewriting most of it anyway.

Prompt chaining is the alternative. You break the work into multiple, sequential prompts. Each one builds on the output of the last. Rather than asking AI to do everything at once, you guide it through a structured workflow, checking the results at every stage.

> The power of a chain is not in any single link. It is in the pauses between them, where your judgement shapes what comes next.

If this sounds familiar, it should. Prompt chaining is the Iterate stage of the Conversation Loop (Chapter 5) made explicit. Each link in the chain is a moment where you review the output, apply your judgement, and steer what comes next. Without that review step, you are not chaining. You are just queuing up requests and hoping for the best.

9.1 Why It Works

Breaking a task into steps produces better results for the same reasons it works when humans do it.

Smaller steps mean more focused output at each stage, which improves reasoning. You review and adjust after each step, catching problems early rather than discovering them in the final product. You can change direction mid-process without starting over. Each step's output becomes the next step's input, so good foundations compound. And because you see the AI's reasoning at each stage, not just the final product, you stay in the conversation rather than outsourcing it.

9.2 Three Approaches

There is no single right way to chain prompts. The best approach depends on how much control you need and how predictable the process is. Here are three patterns, from least to most interactive.

9.2.1 Approach 1: Guided Workflow

When to use it: You want multi-step processing but prefer to stay in a single prompt.

How it works: Provide all steps upfront with explicit instructions. The AI processes them sequentially and returns structured output.

Example: Research methodology review

```
You are a research methodologist.
I'm writing a methods section for a paper
on employee retention in remote teams.

Follow these steps:
1. Identify the key variables I need to measure
2. Suggest appropriate data collection methods
3. Outline potential confounding variables
4. Recommend how to control for them

Here's my research question: [Paste research question]

Output each step clearly labelled.
```

This works well when you have a clear process in mind and trust it enough to let the AI work through it in one pass. You still review the output. You just review it all at once rather than step by step.

9.2.2 Approach 2: Sequential Chain

When to use it: You want to review and adjust between steps. Each prompt depends on what the previous one produced.

How it works: Submit one prompt. Review the output. Then submit a follow-up that builds on it.

Example: Writing a proposal

Step 1, Analysis:

```
I'm writing a funding proposal
for a community health programme.
Here's a summary of the problem
we're trying to address:
[Paste summary]

Analyse this. What are the strongest
arguments for funding? What gaps
in the evidence should I be aware of?
```

Review the output. Note what resonates and what is missing.

Step 2, Structure:

```
Based on that analysis, outline a proposal structure.
Include sections for: problem statement,
proposed approach, expected outcomes,
and evaluation plan.
For each section, note the key point it should make.
```

Review. Adjust the structure before proceeding.

Step 3, Draft:

```
Now draft the problem statement
and proposed approach sections.
Use a direct, evidence-informed tone.
Keep it under 800 words total.
Emphasise the points we identified
in Step 1.
```

Review, revise, continue with remaining sections.

The value here is in the pauses. Each review point is a chance to correct course, add something the AI missed, or redirect entirely.

9.2.3 Approach 3: Interactive Chain

When to use it: You want the AI to guide you through a process, pausing for your input at each stage.

How it works: You set up the process, then the AI works through it collaboratively, waiting for your feedback before continuing.

Example: Analysing survey data

```
You are a research analyst. We're analysing
survey data about workplace satisfaction
across three departments.

Here's how we'll proceed:
1. I'll share the survey summary
2. You identify the top 3 themes
3. I'll respond with feedback
4. You develop deeper analysis of the themes I approve
5. We continue iteratively until complete

Here's the data:
[Paste survey summary]

Start with step 2. Then wait for my
response before continuing.
```

The AI responds with themes. You reply:

> "Theme 1 is spot on. Theme 2 needs more nuance. Can you break down the 'communication issues' into specific types? For theme 3, I'd like to explore the relationship between workload and satisfaction more deeply."

The AI refines and waits. You stay in the driver's seat.

This approach works especially well for analysis and research tasks where you cannot know in advance what the interesting findings will be. The chain emerges from the work itself.

9.3 Practical Templates

These are starting points, not scripts. Adapt them to your work. The structure matters more than the specific wording.

9.3.1 Analysis Chain

For working through data, documents, or any situation that needs structured examination.

```
Context: [What are we analysing?]
Goal: [What decision or insight do we need?]

Step 1: Summarise [the data, document, or situation]
Step 2: Identify [key patterns, issues, or opportunities]
Step 3: Assess [impact, risk, or significance]
Step 4: Recommend [specific actions or next steps]

Output each step clearly. After step 1, I'll review before
we continue.
```

9.3.2 Argument Chain

For building a case, writing a position paper, or developing a systematic analysis.

```
Claim: [The main argument or recommendation]

Step 1: Define [key concepts or terms]
Step 2: Provide [evidence or examples]
Step 3: Address [counterarguments or limitations]
Step 4: Synthesise [into a clear position statement]

Output each step with supporting detail.
I'll provide feedback after each step.
```

9.3.3 Planning Chain

For scoping a project, structuring a strategy, or working through any multi-stage plan.

```
Objective: [What are we trying to achieve?]
Constraints: [Timeline, budget, resources, scope limits]

Step 1: Define [the key deliverables or milestones]
Step 2: Identify [dependencies, risks, and assumptions]
Step 3: Sequence [the work into phases or stages]
Step 4: Detail [the first phase
   with specific action items]

Output each step. I'll review and adjust
before we continue.
```

9.4 Advanced Techniques

Once you are comfortable with basic chaining, these techniques let you extract more from the same conversation.

9.4.1 Iterative Refinement

After completing a chain, push deeper without starting over:

> "That analysis is helpful. Now go deeper on [specific theme]. Add statistical evidence or examples, implications for [your context], and one alternative perspective."

This is the Conversation Loop in miniature. You have already iterated once. Now you iterate again, on a narrower target.

9.4.2 Format Shifting

Take the same underlying content and chain it into different formats:

```
Prompt 1: "Analyse this project data."
   [Full analysis request]
[Get detailed analysis]

Prompt 2: "Now turn that analysis into:
- A 2-minute briefing for the executive team
- A one-paragraph update for stakeholders
- A set of action items for the project team"
```

Same thinking, three audiences. The analysis does the heavy lifting. The format shift makes it useful.

9.4.3 Perspective Shifting

Rerun analysis from different viewpoints:

> "Now redo that analysis from the perspective of: a new customer encountering this product for the first time, a support team handling complaints, and a competitor evaluating the market. How does each perspective change the recommendations?"

This is one of the most underused techniques. It forces the AI to stress-test its own output, and it often surfaces blind spots you would not have found otherwise.

9.5 Managing Long Conversations

There is a practical reality that prompt chaining runs into: AI has a finite memory.

Every AI conversation has a context window, a limit on how much text the model can hold in its head at once. Early in a conversation, the AI has your full prompt and its full response. Twenty exchanges later, the earliest parts of the conversation are fading. The model does not forget all at once. It loses precision gradually, like a person trying to remember the first thing said in a two-hour meeting.

This matters for chaining because long chains are long conversations. If your chain runs to fifteen or twenty exchanges, the AI may lose track of decisions you made in step two while it works on step twelve. The output starts to drift, contradict earlier work, or quietly drop constraints you set at the beginning.

A few practical habits keep this under control:

Start new conversations for new topics. If your chain is done and you are moving to a different task, open a fresh session. Do not try to do everything in one conversation.

One task per prompt. When you ask the AI to do three things at once, it divides its attention across all three. The result is three shallow responses instead of one deep one. Break it up.

Summarise and hand off. When a conversation is getting long (roughly fifty exchanges or more), ask the AI to summarise everything you have accomplished and decided so far. Copy that summary into a new conversation and continue from there. You reset the AI's attention without losing the thread.

Make context explicit every time. Do not assume the AI remembers what you said earlier. If a constraint matters for this step, state it again. "Remember, we are working within a six-week timeline and a team of

four" takes five seconds to type and prevents the AI from drifting back to generic assumptions.

Watch for signs of drift. If the AI starts contradicting something it said earlier, or produces output that ignores a constraint you set, the context window is likely the problem. Summarise, hand off, and continue in a fresh session.

These are not advanced techniques. They are housekeeping. But they make the difference between a chain that holds together over ten steps and one that quietly falls apart at step six.

> The golden rule of chaining
>
> Review each step before moving to the next. Without that review, you are not chaining. You are queuing up requests and hoping for the best.

9.6 When Not to Chain

Not every task needs a chain. Skip it when:

- **The task is simple.** A single clear prompt is more efficient.
- **You are exploring, not building.** Free-form conversation sometimes beats structured chaining.
- **Context is thin.** Without enough input, chains do not add much value. Start with a better first prompt instead. RTCF (Chapter 8) helps here.

9.7 Best Practices

1. **Start clear.** Your first prompt sets the direction. Use RTCF framework to structure it.
2. **Review each step.** Do not blindly proceed. Check each output before moving on.
3. **Provide feedback.** Tell the AI what is working and what needs adjustment.
4. **Build incrementally.** Small steps beat big jumps.
5. **Save your chains.** A good chain is reusable. Document the ones that work.
6. **Know when to stop.** Iteration has diminishing returns. Three rounds of refinement usually beats seven.

9.8 Key Takeaways

- Prompt chaining breaks complexity into steps. Each step is simpler and produces better output.
- You maintain control. Review after each step. Adjust direction when needed.
- It works for analysis, writing, planning, and argument-building. Not just content generation.
- Quality compounds. Good output at step 1 becomes better input at step 2.
- It is a conversation. The back-and-forth is where value gets made. Each link in the chain is a chance to think, not just a chance to prompt.

The power of prompt chaining is not in any individual step. It is in how each step builds on the previous one, refining your thinking as you go. That is what makes the output yours, not the AI's.

Chapter 10

Eight Techniques for Deeper Thinking

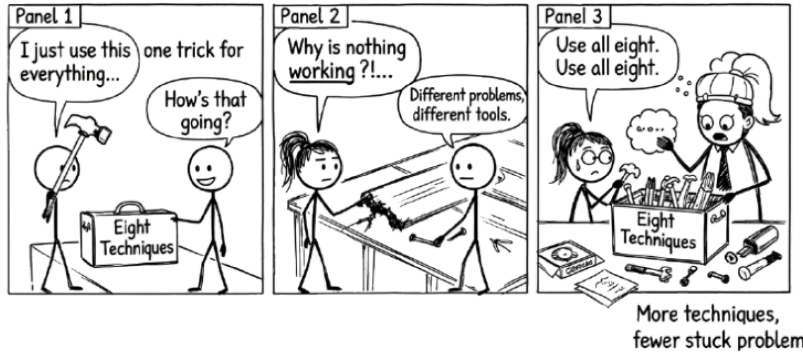

Figure 10.1: More techniques, fewer stuck problems.

> The right technique is not the cleverest one. It is the one that matches the kind of thinking you need to do.

You now have the mindset. You know AI works best as a thinking partner, not an answer machine. You know to do your own thinking first, then bring AI into the conversation.

But what does that conversation actually look like?

This chapter gives you eight concrete techniques. Each one structures the conversation differently, pushing your thinking in a specific direction. They are not the only techniques that exist. But they cover most of the situations

where people get stuck: scoping a problem, weighing options, reasoning through complexity, preparing for difficult interactions, stress-testing ideas, checking your own understanding, getting multiple perspectives, and finding blind spots.

For each technique, you will find a short explanation of what it does, when to reach for it, and a ready-to-use prompt you can adapt to your own situation.

Pick the one that fits. Modify it. Make it yours.

> The right technique is not the cleverest one. It is the one that matches the kind of thinking you need to do right now.

10.1 1. Reverse Prompting

What it does. Instead of asking AI for answers, you ask AI to ask *you* questions. This forces you to articulate things you have not thought through yet. The AI's questions surface assumptions, gaps, and considerations you would otherwise miss. It works because a good question is often more valuable than a quick answer.

When to use it. You are starting a new project, scoping a problem, or planning something complex and you know you have not thought of everything yet.

Try this prompt

```
I'm planning [describe your project,
decision, or problem].

Your task: Ask me a series of
short-answer questions to help me
clarify all the requirements,
considerations, and potential pitfalls.

Ask one question at a time. Wait for
my response before asking the next
question. Continue until you've helped
me think through at least 10 different
aspects of this.

Begin with your first question.
```

The key is to answer honestly, especially when a question catches you off guard. That discomfort is the technique working. If the AI asks something

you cannot answer, that is a gap in your thinking you just discovered.

After the questioning round, try this follow-up: “Based on my answers, what is the single biggest gap in my thinking?”

Reverse Prompting works best at the start of something, when the shape of the problem is still forming. But once you have a clearer picture and need to choose a direction, you need a different kind of thinking.

10.2 2. Pros and Cons

What it does. The AI systematically evaluates multiple options against specific criteria you care about. This is not about getting the “right answer” from AI. It is about generating a structured comparison that makes your own reasoning more rigorous. You bring the judgement. The AI brings the structure and breadth.

When to use it. You are choosing between approaches, tools, strategies, or solutions and you want to think through trade-offs methodically rather than going with your gut.

Try this prompt

```
I'm deciding between these options
for [describe the decision]:

Option A: [describe]
Option B: [describe]
Option C: [describe]

For each option, provide:
1. A brief description of how it works
2. Three key advantages
3. Three key disadvantages

Evaluate each specifically in terms of:
- [Criterion 1, e.g., cost and time investment]
- [Criterion 2, e.g., long-term flexibility]
- [Criterion 3, e.g., risk if it goes wrong]

Conclude with which option you'd recommend
and why, but flag the strongest argument
against your recommendation.
```

Do not accept the AI’s recommendation uncritically. The real value is in the structured comparison. Challenge at least one of the listed pros. Ask

yourself what disadvantage the AI missed. Argue for a different option than the one AI recommended and see if your reasoning holds up.

Pros and Cons gives you a snapshot of the decision landscape. But some decisions are not about choosing between options. They are about understanding a process with many moving parts, where the sequence matters and skipping a step has consequences.

10.3 3. Stepwise Chain of Thought

What it does. The AI walks you through a complex process one step at a time, pausing after each step. This prevents you from rushing through something that requires careful, sequential thinking. Each pause gives you a chance to ask questions, raise complications, or confirm you understand before moving on.

When to use it. You are learning a new process, preparing for something with many sequential steps, or working through a procedure where skipping a step could cause problems.

Try this prompt

```
I need to understand the process
for [describe the task or procedure].

Walk me through the entire process
from start to finish. For each step,
tell me:

1. What action to take
2. What to document or record
3. What could go wrong at this stage

After you explain each step, STOP and
wait for me to type "next" before moving
to the next step. Do not provide
the entire process at once.

Begin with Step 1.
```

At any step, you can ask the AI to go deeper: “What if the client pushes back at this stage?” or “What happens if I skip this step?” These side conversations are where the real learning happens. You are not just memorizing a sequence. You are understanding why each step matters.

The first three techniques help you think through problems, weigh options,

and understand processes. But some of the hardest challenges are not analytical. They are interpersonal. You know what you want to say, but you are not sure how the other person will respond.

10.4 4. Role Play

What it does. The AI adopts a specific persona and you practice interacting with that persona. Think of it as a flight simulator for difficult conversations. The AI can play a skeptical stakeholder, a frustrated customer, a resistant colleague, or any other person you need to prepare for. It responds dynamically to what you say, so you get realistic practice without real consequences.

When to use it. You have a difficult conversation coming up, you want to practice a presentation to a tough audience, or you need to rehearse handling objections.

Try this prompt

```
You are [describe the persona: e.g.,
a senior executive who is skeptical
of my proposal / a client who is
unhappy with a delayed deliverable /
a colleague who disagrees with my
approach].

Your personality: [e.g., direct and
impatient / friendly but unconvinced /
emotional and defensive]

I will practice having this
conversation with you. Stay in
character. If I say something vague,
push back. If I handle something well,
acknowledge it briefly but keep
pressing.

After we finish the conversation,
break character and give me feedback
on:
- What I handled well
- What I could improve
- Any phrases or approaches that
  would have been more effective
```

```
Let's begin. I'll speak first.
```

The feedback at the end is critical. It turns the exercise from mere practice into a structured debrief. Pay special attention to moments where the AI pushed back and you felt stuck. Those are the moments to rehearse differently.

Role Play prepares you for a specific conversation. But sometimes the challenge is not a person. It is a decision where you already have a preference and you suspect you have not fully examined the other side. That calls for a different kind of pressure.

10.5 5. Debating

What it does. You set up a structured debate where opposing positions are argued with equal force. This is powerful for decisions where you already have a preference, because it forces you to take the other side seriously. You can have AI argue both sides in a single conversation, or you can take one side yourself and let AI argue the other.

This technique combines well with using multiple AI tools. Give one model the "advocate" position and another the "skeptic" position, then shuttle arguments between them. Or keep it simple and have one AI play both roles.

When to use it. You are facing a decision with valid arguments on both sides and you suspect you are leaning one way without fully examining the alternative.

Try this prompt

```
I'm considering [describe the decision].

Context:
- [Key fact 1]
- [Key fact 2]
- [Key constraint]

Set up a debate between two positions:
- "The Advocate" argues for [Option A]
- "The Skeptic" argues for [Option B]

Conduct the debate in four rounds.
Each speaker gets 3-4 sentences per
```

```
round. Label each speaker clearly.
Make both sides as persuasive as
possible.

After the debate, tell me: which
arguments should weigh most heavily
in my decision, and what additional
information would I need to be
confident either way?
```

Watch for arguments that surprise you. If the Skeptic raises a point you had not considered, sit with that. The purpose is not to "win" the debate, but to make sure you are not ignoring a perspective that matters.

A useful follow-up: "Now argue for a third option that synthesizes the best of both positions."

The previous five techniques help you explore, decide, learn processes, practise interactions, and stress-test positions. But sometimes the question is simpler and more personal: do I actually understand this topic as well as I think I do?

10.6 6. Formative Self-Testing

What it does. The AI generates questions that test your understanding of a topic, then gives you immediate feedback on your answers. This is not about memorizing facts. It is about discovering where your understanding is solid and where it is shakier than you thought. The act of retrieving knowledge and articulating it is one of the most effective ways to deepen your understanding.

When to use it. You are studying a new subject, preparing for a certification, onboarding into a new role, or just want to pressure-test how well you actually understand something you think you know.

ℹ Try this prompt

```
I'm studying [topic or subject area].
Test my understanding.

Ask me 5 questions, one at a time. Mix the difficulty:
- 2 questions testing core concepts
- 2 questions requiring application to realistic scenarios
- 1 question that requires me to
```

```
  evaluate or critique something

After each answer I give, tell me:
1. Whether my answer is correct,
   partially correct, or incorrect
2. What I got right
3. What I missed or got wrong, with a brief explanation

Wait for my answer before moving to the next question.

Start with Question 1.
```

Be honest with your answers. Do not look things up first. The point is to find out what you actually know versus what you think you know. Pay close attention to the "partially correct" answers. That is where the most useful learning lives, in the gap between what you said and what you missed.

Self-testing reveals what you know and what you do not. But even when you understand a topic well, you are still seeing it from one vantage point: your own expertise, your own role, your own assumptions. Some problems need more than one perspective.

10.7 7. The Expert Panel

What it does. The AI simulates multiple experts with different backgrounds, each analysing your situation from their own perspective. A financial expert sees cost implications. A legal expert sees compliance risks. A customer experience expert sees user impact. You get multiple lenses on the same problem without needing to schedule five meetings.

When to use it. You are making a decision that affects multiple domains and you want to make sure you are not thinking about it from only one angle.

Try this prompt

```
I need to make a decision about [describe the situation].

Assemble a panel of 4 experts who would
each have a different perspective
on this decision:
- Name each expert and describe
  their background
```

```
- Have each expert analyse the
  situation from their perspective
  (3-4 sentences each)
- Have each expert identify the one
  thing they think I'm most likely
  to overlook

After all four experts have spoken, provide a synthesis:
what do they agree on, where do they disagree, and what
should I investigate further?
```

The synthesis at the end matters most. Look for where the experts disagree. That tension usually points to the hardest part of your decision, the part that requires your judgement, not more AI output.

For a deeper conversation, pick one panelist and say: "I want to push back on [Expert Name]'s point about [specific claim]. Have them defend their position."

The Expert Panel broadens your perspective. But there is a particular kind of blind spot that multiple perspectives alone will not catch: the risks hiding inside your own confidence. When you have a plan you feel good about, you need someone to tell you what could go wrong.

10.8 8. Risk Deep-Dive

What it does. You share a plan or decision along with the risks you have already identified, and the AI helps you find what you are missing. It looks for second-order effects, blind spots, and early warning signs. This technique is specifically designed to counter the optimism bias that affects most planning. We tend to overestimate benefits and underestimate what can go wrong.

When to use it. You have a plan you are fairly confident about and you want someone to stress-test it before you commit.

Try this prompt

```
I'm planning to [describe the initiative or decision].

Here are the risks I've already identified:
- [Risk 1]
- [Risk 2]
- [Risk 3]
```

```
Help me think deeper:
1. What risks am I missing or underestimating?
2. What second-order effects should I consider?
3. Which risk is most likely to be the one that actually
   derails this?
4. What early warning signs should I watch for?

Challenge my assumptions. Be specific and concrete.
```

The most valuable output from this technique is usually in question 1, the risks you are missing. We all have blind spots shaped by our experience, our role, and our optimism about our own plans. The AI does not share those blind spots.

After the initial analysis, try: "If this initiative fails 6 months from now, what is the most likely cause?" This "pre-mortem" framing often surfaces risks that a forward-looking analysis misses.

10.9 Combining Techniques

These eight techniques are not isolated tools. They combine naturally as a problem evolves.

You might start with **Reverse Prompting** to scope a new initiative, use **Pros and Cons** to evaluate your options, run a **Debate** to stress-test your preferred choice, convene an **Expert Panel** to check for blind spots across domains, and finish with a **Risk Deep-Dive** before committing.

Or you might use **Role Play** to prepare for a presentation, then **Formative Self-Testing** to make sure you can handle technical questions from the audience.

The point is not to use all eight on every problem. The point is to have them available so you can reach for the right one when you need it.

> **Start simple**
>
> You do not need to master all eight techniques. Pick the one that fits your current problem. Try it once. Modify the prompt. Try it again. Skill comes from repetition, not from reading.

Start with whichever technique matches the kind of thinking you need to do right now. Try it once. Modify the prompt. Try it again. These are conversation starters, not scripts.

Chapter 11

Using AI to Help You Use AI

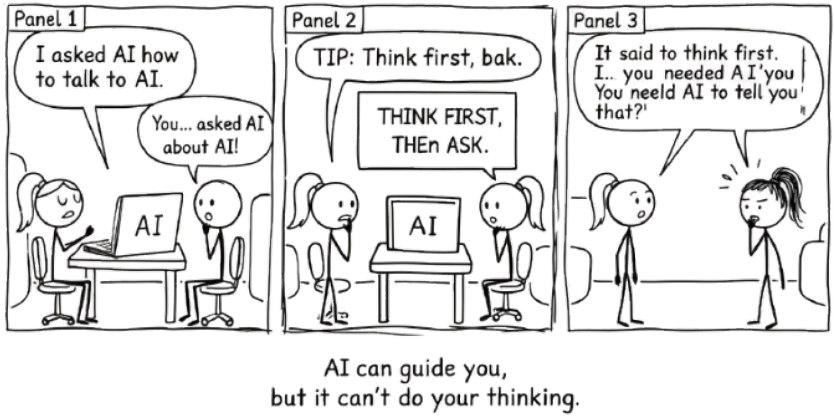

Figure 11.1: AI can guide you, but it can't do your thinking.

The most useful thing AI can do is help you figure out what you actually need to ask.

11.1 The One Thing AI Can Do That Other Tools Can't

AI can teach you how to use itself.

Excel cannot teach you how to use Excel. PowerPoint cannot walk you through making better slides. Email cannot help you write better emails.

But AI can, because it can ask you questions, understand your context, evaluate its own suggestions, and adjust based on your feedback.

This means you do not need a course or a manual to get started. You can use AI to figure out how AI fits into your work.

> The most useful thing AI can do is not answer your question. It is help you figure out what question to ask.

11.2 The Core Idea: Meta-Prompting

Instead of asking AI to solve a problem directly, ask AI to help you figure out how to approach the problem.

Direct Approach	Meta Approach
"How should I answer this question?"	"What is the best way to frame this question for an AI so I get useful help?"
"Give me strategies for using AI."	"Ask me questions to understand my context, then recommend both obvious and unexpected ways I could use AI."
"How do I improve my workflow?"	"Interview me about my role and responsibilities, then suggest AI tools and strategies I might not have considered."

The meta approach works better because AI asks clarifying questions first, learns your context, and gives you recommendations that fit your actual situation. Not generic advice aimed at nobody in particular.

11.3 The Consultation Prompt: Your Starting Point

Here is a prompt you can use right now. Copy it, paste it into any AI tool, and start the conversation.

```
You are an AI expert consultant.
I would like your help understanding
how I could better use AI in my work.
```

```
Please ask me one question at a time.
I will answer, then you ask the next
question. Keep asking until you
understand:
- My role and responsibilities
- My main workflows and challenges
- My key objectives
  (what matters for success)
- My constraints (time, resources,
  skills, institutional requirements)

Once you have enough context,
provide TWO types of recommendations:
1. Obvious opportunities: Clear,
   straightforward AI applications
   (the low-hanging fruit most
   people think of)
2. Non-obvious opportunities:
   Unexpected or creative uses of AI
   I might not have considered

Format your recommendations clearly
with implementation tips for each.
```

That is it. Paste it, start answering questions, and see what comes back.

11.4 Why This Works

When you ask AI to interview you before making recommendations, several useful things happen.

It gets your actual context. AI does not know anything about you at the start. By asking sequential questions, it builds up a picture of your discipline, your constraints, and what success looks like for you. Generic advice does not account for the fact that you manage compliance for a large team, or support early-career researchers, or run a small business with no IT budget. Context changes the recommendations entirely.

You get both obvious and non-obvious ideas. AI tends to suggest the predictable stuff first: automating reports, generating ideas, answering questions. Those are useful, but you have probably already thought of them. By explicitly asking for non-obvious recommendations, you push the model past its defaults. It starts connecting your specific constraints and workflows to capabilities you had not considered. The non-obvious ideas tend to be more useful, and they are the ones you would never have found by reading a generic "top ten AI tips" article.

AI can evaluate its own suggestions. Once it generates a list, you can ask it to assess which are most feasible, identify trade-offs, refine implementation steps, or suggest variations. This lets you iterate toward something practical rather than accepting the first answer.

It is a conversation, not a lecture. Unlike reading a guide or sitting through a workshop, you are actively shaping the advice by answering questions honestly, pushing back on ideas that do not fit, and exploring what is actually possible. The back-and-forth is where the useful stuff tends to surface. And because you are in control of the conversation, you can steer it toward the parts of your work that matter most.

11.5 A Real Example

Here is a simplified version of how this conversation might go.

You paste the consultation prompt.

AI asks: "What is your primary role, and what does a typical week look like?"

You say: "I am a teaching coordinator in a business faculty. I manage unit approvals, coordinate teaching support across 15 units, and help faculty integrate new teaching approaches."

AI asks: "What is your biggest pain point right now?"

You say: "Administrative overhead. Tracking who needs what support, writing emails summarising policies to faculty who have not read them, managing compliance documentation."

AI asks: "What would success look like for you?"

You say: "I would have time for strategic work instead of just admin. And faculty would feel confident designing assessments without constantly asking me the same questions."

AI synthesises:

> **Obvious opportunities:** Use AI to draft plain-language policy summaries. Use AI to help faculty write learning outcomes. Use AI to generate assessment design templates.
>
> **Non-obvious opportunities:** Use AI to analyse your incoming faculty questions and identify common gaps in understanding, then create targeted guidance before people ask. Use AI as a quality-check tool: paste a unit outline and ask it to flag compliance risks or unclear learning outcomes. Use AI to build a simple FAQ that handles the routine questions so you can focus on the ones that actually need your judgement.

Notice what happened. Three questions. Honest answers. Targeted recommendations. The AI did not give generic suggestions about "leveraging AI for productivity." It understood the specific role, constraints, and frustrations, so the recommendations are grounded in something real. You could act on any of them this week.

11.6 The Deeper Principle: AI as a Mirror

Here is what is actually going on when you ask AI to interview you.

It forces you to articulate your own work clearly.

By answering its questions, you end up spelling out:

- What you actually do (versus what you think you do)
- What matters most (versus what is just urgent)
- What is possible (versus what feels impossible)
- What you have not tried (because you did not know it was an option)

Often, the most useful part of this conversation is not the AI's recommendations. It is the clarity you gain about your own work by having to explain it to something that knows nothing about you.

People regularly say things like: "I did not realise how much time I was spending on that until the AI asked about it." Or: "Just explaining my challenges helped me see a solution on my own." Or: "The recommendations were fine, but the real value was the clarity I got about what I actually needed."

This is not a side effect. It is the point. The consultation prompt works because it puts you in the position of expert and the AI in the position of interviewer. You know the answers. It just asks the right questions. And in answering those questions, you often discover that you already knew what to do. You just had not said it out loud yet.

11.7 Practical Tips for Your Consultation

Be honest and specific. Do not give polished answers. The more honest you are about your constraints and challenges, the better the recommendations. "We have no budget," "I do not have time to learn new tools," "My organisation is sceptical of AI." These are all useful context.

Ask follow-up questions. After getting recommendations, push further. "How would I actually implement the non-obvious ones?" "Which of these could I start with this week?" "What are the risks or drawbacks?"

Test one thing. Pick one obvious and one non-obvious recommendation. Try them for a week or two. Come back to the same conversation and

report what happened. "I tried your suggestion about X. Here is what worked and what did not." The AI will adjust. Iterate from there.

Come back periodically. This is not a one-off exercise. Your context changes. Your needs shift. The tools themselves change. Run this consultation again in a few months. Each time, you will be ready for different ideas, and you will ask sharper questions because you have more experience to draw on.

Share what you learn. If something works, tell a colleague. "I asked AI to help me figure out how to use AI, and it suggested this. Here is how it went." Others might try their own version and find something entirely different that is useful for their context. That is the point.

> **The real value**
>
> Often, the most useful part of asking AI to interview you is not the recommendations it gives. It is the clarity you gain about your own work by having to explain it.

11.8 Getting Started Right Now

1. Open your AI tool.
2. Copy the consultation prompt from this chapter.
3. Paste it and start answering questions honestly.
4. Review the recommendations and pick one or two to try.
5. Come back with what you learned and keep going.

This works with any AI tool and in any role. The skill you are building is not prompt engineering. It is the ability to think clearly about your own work, explain it precisely, and evaluate whether what comes back is useful. That skill transfers everywhere.

If you want to structure the prompts that come out of this consultation, the RTCF framework (Chapter 8) gives you a reliable way to build them. If you want to break complex problems into stages, prompt chaining (Chapter 9) shows you how.

But start here. Let AI help you figure out what you actually need.

Once AI has helped you figure out what to ask, VET helps you make sure you can stand behind the answer.

Chapter 12

VET Your AI: The Push-Back Framework

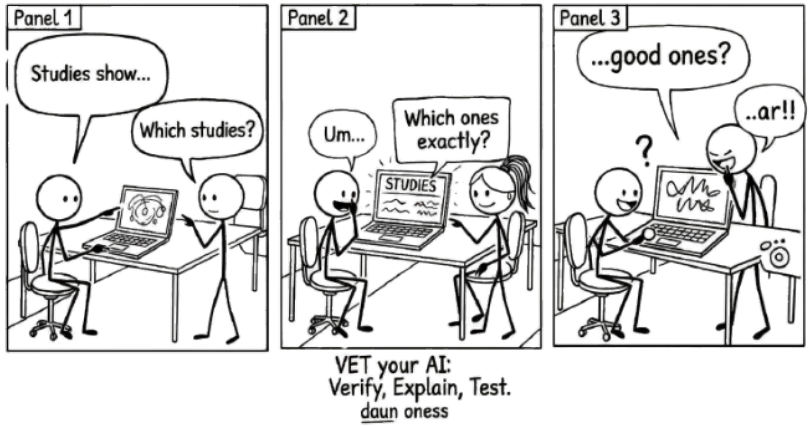

Figure 12.1: VET your AI: Verify, Explain, Test.

> If you cannot verify it, explain it, and test it, you are not done yet.

The most dangerous AI output is the one that sounds right.

Not the obvious hallucination. Not the garbled sentence. Not the response that makes you squint and say "that can't be correct." Those are easy to catch. You catch them because they feel wrong.

The dangerous output is the one that reads smoothly, cites plausible-sounding sources, and fits neatly into your expectations. You nod along. You accept it. You paste it into your document. And you never check, because nothing triggered your suspicion.

This is where critical evaluation fails most often. Not when AI is obviously wrong, but when it is subtly wrong in a way that sounds completely right.

> If you cannot explain it in your own words, you do not understand it well enough to use it.

Chapter 7 established why your critical eye matters. This chapter gives you a method for using it. Three steps, three questions, one habit that turns passive acceptance into active evaluation.

12.1 The VET Framework

VET stands for Verify, Explain, Test. It is not a checklist you laminate and pin to your wall. It is a thinking habit you run through every time you receive AI output that matters.

Each step asks a single question. If you can answer all three, you own the output. If you cannot, you do not understand it well enough to use it.

12.1.1 Verify: Can I find this independently?

The first question is about sources. AI presents claims with absolute confidence regardless of whether those claims are true. Your job is to check.

- **Check sources.** If the AI cites a study, find that study. If it names an author, confirm the author exists and actually said what the AI claims. AI fabricates citations routinely. Not maliciously, but because generating a plausible-looking reference is the same process as generating any other text. The model does not distinguish between real and invented.
- **Cross-reference claims.** Take the key factual claims and verify them through independent channels. A textbook, a colleague, a database, a primary source. If a claim only exists in the AI's output and nowhere else, treat it with extreme scepticism.
- **Look up citations.** This bears repeating because it is the most common failure point. AI will produce beautifully formatted references that do not exist. It will attribute findings to real researchers who never conducted those studies. The only way to know is to look.

Verification is not about distrust. It is about due diligence. You would not submit a report based on a single anonymous source. Do not submit one based on unverified AI output.

12.1.2 Explain: Can I explain this in my own words?

The second question is about understanding. There is a difference between having an answer and understanding an answer. AI can hand you the first. Only you can build the second.

- **If you cannot explain it, you do not understand it.** This is a hard rule. If someone asked you to explain the AI's output without looking at it, could you? If the answer is no, you are holding words, not knowledge. You are carrying a bag someone else packed, and you do not know what is inside.
- **Rewrite it in your own language.** Take the AI's output and translate it into your words, your framing, your way of explaining things. This is not about style. It is a comprehension test. The act of rewriting forces you to process the content rather than skim it. Where you struggle to rephrase, that is where your understanding has a gap.
- **Teach it to someone.** The best test of understanding is explanation. If you can teach the concept to a colleague who is unfamiliar with it, you understand it. If you find yourself falling back on the AI's exact phrasing because you cannot find your own, you have memorised, not learned.

This step is where the Conversation Loop earns its name. You are not just receiving information. You are processing it, internalising it, making it yours. Without this step, you are a conduit, not a thinker.

12.1.3 Test: Does this hold up under scrutiny?

The third question is about robustness. Even verified, well-understood output can be fragile. It might be correct in one context and wrong in another. It might work under ideal conditions and fail under real ones.

- **Devil's advocate it.** Argue against the AI's output. What are the strongest objections? What would a sceptic say? If you cannot think of a counter-argument, you have not thought hard enough, or you need to ask the AI itself: "What are the strongest arguments against what you just told me?"
- **Check edge cases.** AI tends to give you the central case, the most common scenario, the textbook answer. Real work happens at

the edges. What happens when the inputs are unusual? When the assumptions do not hold? When the context shifts?

- **Ask "what if?"** Change a variable. Shift a constraint. Alter a condition. Does the output still hold? If the AI told you something works at a specific temperature, what happens five degrees higher? If it recommended a strategy for one market, does the logic transfer to another? Fragile answers break under "what if." Robust answers flex.

Testing is where you discover whether the AI gave you something genuinely useful or just something that sounds useful under narrow conditions.

12.2 VET in Practice

Abstract frameworks are easy to nod along to. Concrete application is harder. Here is what VET looks like when applied to real AI output.

Table 12.1: Applying VET to AI claims: each column asks a different kind of question.

AI says…	V: Verify	E: Explain	T: Test
"Studies show pH drops faster in co-culture"	Find the actual studies	Can I explain the mechanism?	What about different substrates?
"Optimal temperature is 37C"	Check the literature	Why 37C specifically?	What happens at 35C or 40C?
"This method has 95% recovery rate"	Where's that number from?	What does 95% mean here?	Under what conditions?

Notice the pattern. Verify asks "is this real?" Explain asks "do I understand this?" Test asks "is this the whole picture?" Each question catches a different kind of failure.

The first column catches fabrication. The second catches shallow acceptance. The third catches overconfidence. Together, they cover the three most common ways people get burned by AI output.

12.3 When to VET

Not everything needs the full treatment. If you ask AI to help you brainstorm synonyms for a word, you do not need to verify sources, explain the mechanism, and test edge cases. Use your judgement.

VET matters most when:

- The output contains factual claims you will rely on.
- You are making a decision based on the AI's analysis.
- The output will be shared with others who will assume you checked it.
- The stakes are high enough that being wrong has consequences.

The higher the stakes, the more thorough your VET should be. A casual brainstorm gets a light pass. A client deliverable gets the full treatment. A medical or legal application gets VET plus additional expert review, because AI output in those domains can be confidently, fluently, dangerously wrong.

12.4 The Habit

VET works best when it becomes automatic. Not a procedure you consciously invoke, but a reflex. You read AI output and your mind naturally asks: can I verify this, can I explain this, does this hold up?

Building this habit takes deliberate practice at first. For the next week, try this: every time you use AI output in your work, pause and run through the three questions. Write your answers down if that helps. After a few dozen repetitions, you will stop needing to write them down. The questions will just be there, running in the background every time you read AI-generated text.

This is the Iterate stage of the Conversation Loop made concrete. When you VET, you are not passively consuming output. You are actively evaluating it, pushing back on it, strengthening it. You are in conversation, not delegation.

Try this (10 minutes)

Ask AI a factual question about your field, something you can check. Then VET the response fully. Verify one claim independently. Explain the answer in your own words. Test it with an edge case or a changed variable. The whole process takes ten minutes. What you learn about the AI's reliability in your domain is worth far more.

Calibrate your effort

Not everything needs the full VET treatment. A casual brainstorm gets a light pass. A client deliverable gets the full treatment. The higher the stakes, the more thorough your check should be.

12.5 Owning the Output

VET is not about distrusting AI. It is about owning the output.

When you verify a claim, you know it is true because you checked, not because the AI sounded confident. When you can explain something in your own words, you understand it, not because you read it, but because you processed it. When you test an idea and it holds up, you know its limits and its strengths.

The difference between someone who uses AI well and someone who uses AI carelessly is not the quality of their prompts. It is what they do after the AI responds.

VET is what you do after.

> Analyse your prompts
>
> The **RTCF Prompt Analyser{target="_blank"}** on the companion website can help here too. If an AI response was vague or unhelpful, paste your original prompt and check which RTCF elements were missing. A weak prompt often produces output that is harder to VET — because the AI was never given enough context to be specific. Improving the prompt and improving the verification often go hand in hand.

Part IV

Part 4: Putting It Together

Chapter 13

A Conversation Across Disciplines

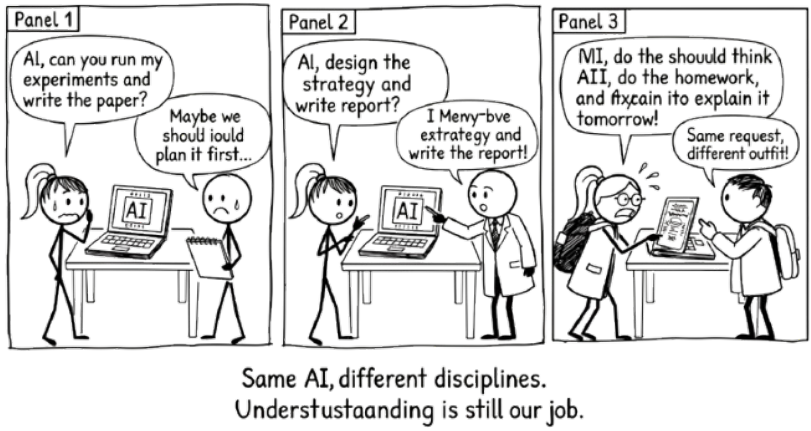

Figure 13.1: Same AI, different disciplines. Understanding is still our job.

> The method does not change when the discipline does. Only the examples change.

The tools are in place. You know the Conversation Loop (Chapter 5). You know how to start well with RTCF (Chapter 8). You know how to push back with VET (Chapter 12). You know how to chain prompts into a sustained line of thinking (Chapter 9).

Now let's see it work.

This chapter walks through four complete examples across four different domains. Each one follows a realistic starting point through a real conversation to a result the human owns. None of them are transcripts. They are summaries of what happened, with the key steering decisions called out.

Pay attention to those steering decisions. They are the difference between conversation and delegation. They are always human. They are always specific. And they always make the output better than the AI would have produced alone.

> The human brought something the AI could not. The AI brought something the human could not match. Neither was sufficient alone.

13.1 Writing and Research: Building an Argument That Holds Up

The situation. A postgraduate student is writing a 3,000-word essay on whether social media regulation should follow the tobacco precedent. She has a position but no structure. Her reading list is long and unsorted. She has three days.

The conversation. She opens with RTCF: role (academic writing advisor familiar with media regulation), task (help me organise my argument), context (essay question, word limit, three sources she has already read), format (an outline with a thesis statement and three supporting claims).

The AI returns a clean five-section outline. It is competent and entirely predictable. This is the **Brainstorm** stage, and it has done its job: given her something concrete to react to.

She reacts. "The third section is too broad. I don't want to argue that all regulation works. I want to argue that the specific mechanism of restricting advertising to minors is transferable." This is the steering decision that matters. The AI did not know her argument. She does.

They move into **Ideate**. She asks the AI to suggest three counterarguments to her revised thesis. Two are useful. One is a straw man. She tells it so, and asks for a stronger version. The replacement is genuinely challenging. She had not considered the platform liability angle.

Now she chains prompts (Chapter 9) through the **Iterate** stage. For each section, she pastes her rough notes and asks the AI to identify gaps in her evidence. She runs VET on the citations it suggests: two check out, one is a real paper with a fabricated finding. She drops it and finds a better source herself.

In the **Amplify** stage, she writes the essay. Not the AI. She writes it, using the structure they built together and the sharpened argument that emerged from iteration.

What made this conversation, not delegation. She brought her own position and sharpened it through pushback. She caught a fabricated citation because she verified rather than trusted. The AI never wrote a sentence of the essay. It helped her think about what to write.

13.2 Data and Analysis: Making Sense of Results That Don't Cooperate

The situation. A market analyst has just pulled quarterly sales figures for six product lines across four regions. Two regions are up. Two are flat. But the product-level numbers tell a different story, and the patterns are not obvious. Her manager wants a narrative for the board by Friday.

The conversation. She starts at **Brainstorm** with RTCF: role (data analyst experienced in retail trends), task (help me identify the most important patterns in this data), context (she pastes a summary table of the figures with year-on-year changes), format (a ranked list of the three most significant findings).

The AI identifies three patterns. The first is obvious: she already knew Region A was up. The second is interesting: a product line growing in flat markets, suggesting it is taking share rather than riding a rising tide. The third is wrong. It claims Region D is declining, but the flat number masks a seasonal effect she knows about from experience.

She corrects the AI and tells it why the seasonality matters. This is the steering decision. She is the one who knows the business. The AI is the one scanning for patterns. Neither could do this alone.

In the **Ideate** stage, she asks the AI to generate three possible explanations for the share-taking pattern. One involves pricing. One involves channel mix. One involves a competitor withdrawal she happens to know is true but had not connected to this data. Now she has a hypothesis worth testing.

She moves to **Iterate** by chaining prompts. She feeds in additional context, asks the AI to stress-test the competitor-withdrawal explanation, and applies VET: can she verify the timeline? Does the explanation hold across all four regions, or only where the competitor was strong? The AI helps her find the edge case. The pattern holds in three regions, not four. She adjusts the narrative accordingly.

In the **Amplify** stage, she builds her board summary around the refined

finding, with appropriate caveats. The insight is hers. The AI helped her find it faster than she would have alone.

What made this conversation, not delegation. She corrected the AI's misread of her data because she knew the seasonal context. She tested the hypothesis against edge cases before presenting it. The board narrative is grounded in her judgement, not in the AI's first pass.

13.3 Planning and Decision-Making: Scoping a Project Without Fooling Yourself

The situation. A team lead has been asked to propose a plan for migrating his department's client records from a legacy system to a new platform. He knows the destination system. He does not know how to estimate the effort, and his last migration project ran 40% over time.

The conversation. He opens at the **Brainstorm** stage using RTCF: role (experienced project manager who has run data migration projects), task (help me build a realistic project scope), context (number of records, source and destination systems, team size of four, no dedicated data engineer), format (a phased project plan with risk flags).

The AI produces a four-phase plan: discovery, mapping, migration, validation. It is textbook. It is also the plan he ran last time, the one that went over. He says so. "This looks like every migration plan ever written. What specifically goes wrong in phase two when you don't have a dedicated data engineer?"

This prompt, a direct challenge born from experience, is the steering decision that transforms the conversation. The AI responds with five specific risks around data mapping when domain expertise is split across team members. Two of them match what went wrong last time.

In the **Ideate** stage, he asks the AI to propose three ways to mitigate the mapping bottleneck without hiring. The options range from automated schema comparison tools to a dedicated mapping sprint before the main project starts. He knows option three is realistic for his team. The others require tools they don't have budget for.

He moves to **Iterate** with prompt chaining. He takes the mapping sprint idea and asks the AI to help him estimate effort for each phase, then applies VET: do the estimates assume full-time allocation? (They do. His team is not full-time on this.) He corrects the assumption and the AI recalculates. The project timeline grows by three weeks, but now it is honest.

In the **Amplify** stage, he writes the proposal with the revised timeline,

the explicit risk register, and the mapping sprint as a pre-phase. He flags the three-week extension to his manager before it becomes a surprise.

What made this conversation, not delegation. He rejected the generic plan because he had lived through its failure. He forced the AI to address his specific constraint. The resulting plan is more realistic because he insisted on honest assumptions rather than accepting optimistic defaults.

13.4 Professional Communication: Getting the Tone Right When the Stakes Are High

The situation. A department head needs to email her team about a restructure. Two roles are being eliminated. Three people are moving to new teams. Everyone else stays, but morale is fragile. She needs to be honest without being brutal, and reassuring without being dishonest.

The conversation. She starts at the **Brainstorm** stage with RTCF: role (communications advisor experienced in organisational change), task (help me draft an internal email announcing a team restructure), context (she outlines the changes, the reasons, and the emotional state of the team), format (professional email, 300 words, direct but empathetic).

The AI's first draft is polished and empty. It reads like every corporate restructure email ever sent. "We are excited about the opportunities ahead." She is not excited, and neither is her team. She tells the AI: "Remove anything that sounds like it came from a press release. These people will read this email looking for whether they can still trust me. Write for that audience."

This is the steering decision. She knows her audience in a way no prompt can fully convey. The AI can adjust tone, but only she can tell it which tone is right.

In the **Ideate** stage, she asks the AI to produce three different opening lines: one that leads with the facts, one that leads with acknowledgement of difficulty, one that leads with the reason for the change. She picks the second and asks the AI to develop it.

She moves to **Iterate** by reading each paragraph aloud. Two sound right. One is still too corporate. She rewrites that paragraph herself and asks the AI to check whether the overall message is internally consistent: does the tone of the opening match the tone of the close? The AI spots a disconnect. The opening is empathetic, but the closing pivots to optimism too quickly. She adjusts.

She applies VET to the factual claims in the email. Are the timelines accurate? Has she described the support available to affected staff correctly? She checks with HR before sending.

In the **Amplify** stage, she sends the email in her voice, with her name on it, confident that every word reflects what she actually means.

What made this conversation, not delegation. She rejected the AI's first draft entirely because she knew what her team needed to hear. She wrote the hardest paragraph herself. The email went out as hers because she made it hers, not because the AI wrote something she happened to agree with.

13.5 The Pattern Across All Four

Look at what stayed constant.

The human brought something the AI could not: a position, domain knowledge, a lived constraint, an understanding of audience. The AI brought something the human could not match: speed across alternatives, tireless iteration, the ability to generate options without ego.

Every example used the Conversation Loop. Every one involved a moment where the human rejected or redirected the AI's output based on something only they knew. Every one ended with an output the human could put their name on and defend.

> **The pattern**
>
> In every example, the turning point was a moment where the human rejected or redirected the AI's output based on something only they knew. That is the method. Not a formula. A way of working.

Notice how each example naturally separates thinking from building. The student explored her argument before writing the essay. The analyst tested her hypothesis before presenting to the board. The team lead challenged the generic plan before proposing his own. The department head rejected the corporate tone before sending the email. In each case, the thinking and the building were distinct phases, and the human was the bridge between them. This is the two-chat workflow (Section 5.7.2) in action, whether or not the person literally used two sessions.

Chapter 14

Becoming More Capable

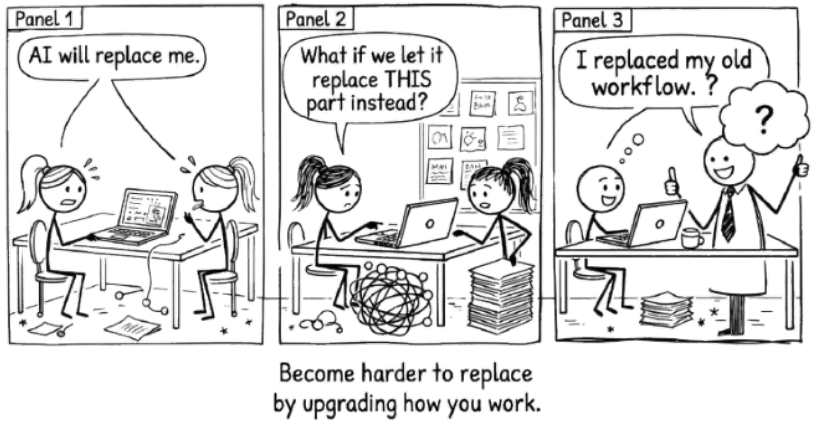

Figure 14.1: Become harder to replace by upgrading how you work.

> The goal was never to get AI to do more. It was to become more capable yourself.

This book was never about AI.

It was about you.

The tools will change. The models will get faster, cheaper, more capable. The interfaces will shift. The names will change. None of that matters as much as what happens to the person sitting across from the machine.

In fact, better models make the practices in this book more important,

not less. A model that produces more convincing output raises the cost of uncritical acceptance. When the average output was obviously rough, people naturally reviewed it. When the average output reads like something a competent professional wrote, the temptation to accept it without thinking increases, and so does the cost of the errors you miss.

The better AI gets at producing convincing averages, the more you need the habits of verification, critical engagement, and staying in the conversation that this book describes.

The question this book has been answering, from the first page, is not "how do I use AI well?" It is "how do I become more capable by thinking alongside it?"

Those are different questions. The first one makes you a better operator. The second one makes you better at your work, with or without the tool.

> You started this book learning how to work with AI. You are finishing it having learned how to think more clearly. That was always the point.

14.1 The Compound Loop

Better questions produce better conversations.

Better conversations produce deeper understanding.

Deeper understanding produces better questions.

This is not a metaphor. It is what actually happens when you adopt the practices in this book. You start by learning to frame a precise question instead of dumping a vague request. That precision forces you to clarify your own thinking before the conversation begins. The conversation that follows is sharper because you were sharper going in. And what you learn from that sharper conversation makes your next question even better.

Each loop through the cycle leaves you slightly more capable than you were before. Not because the AI taught you something. Because the act of thinking alongside it forced you to articulate what you know, confront what you do not, and make decisions about what matters.

This is the compound interest of the conversational approach. Delegation produces outputs that sit in folders. Conversation produces understanding that compounds in your head.

14.2 What You Actually Built

If you have been practising what this book describes, something has already changed. You may not have noticed it yet.

You think before you prompt. You draft before you ask for feedback. You iterate instead of accepting the first response. You push back when something sounds plausible but feels wrong. You ask the AI to challenge your reasoning, not just validate it.

These are not AI skills. They are thinking skills. The AI was the catalyst, but the capability is yours.

A professional who has spent six months in genuine conversation with AI is not someone who has learned to use a tool. They are someone who has learned to think more precisely, evaluate more critically, and articulate more clearly. Take the AI away and those skills remain. They are yours now.

That is the difference between augmentation and dependence. Dependence means you need the tool to function. Augmentation means the tool made you better, and "better" persists after the tool is gone.

14.3 Your Expertise, Its Breadth

The subtitle of this book is a formula: your expertise plus AI's breadth equals amplified thinking.

Notice what comes first. Your expertise. Not the model's capability. Not the prompt technique. You.

AI brings breadth no human can match. It has processed more text, seen more patterns, encountered more framings than any individual could absorb in a lifetime. That breadth is genuinely valuable. But breadth without direction is noise. It is your expertise that turns breadth into insight. Your context, your judgement, your experience of what actually works in the specific situation you are facing.

The formula only works when both sides contribute. Strip out the AI and you lose the breadth. Strip out your expertise and you lose the direction. The amplification happens in the space between.

This is why the conversational approach matters more than any specific technique. Techniques are useful. They give you structure, starting points, ways to get unstuck. But the real leverage comes from the relationship between your knowledge and the model's range. That relationship deepens every time you engage with it honestly.

14.4 What Does Not Compound

Delegation does not compound.

If you spent the last year asking AI to write your emails, draft your reports, and generate your analyses, you have a year's worth of polished outputs. You do not have a year's worth of growth. The outputs were consumed, forgotten, replaced by next week's outputs. Nothing accumulated in you.

Conversation compounds. Delegation does not. That distinction is worth more than every prompting trick in every guide ever written.

An honest note: the compound effect described here is a claim grounded in pedagogical theory, professional experience, and the well-documented benefits of reflective practice. It is not yet backed by multi-year longitudinal studies of AI-assisted thinking. Those studies are underway, but the technology is too new for definitive long-term evidence. The argument rests on the same foundation as any claim about deliberate practice: that structured engagement with challenging material builds capability, and passive consumption does not. If that principle holds (and centuries of evidence from education, music, medicine, and sport suggest it does) then the conversational approach should compound. But should is not proof. The honest position is confidence in the principle, humility about the timeline for evidence, and a commitment to revising the claim if the data says otherwise.

14.5 The Loop, One Last Time

The Conversation Loop is simple. Define your question. Explore possibilities. Push back and refine. Make the result yours. Go around again.

That loop is not a technique. It is a description of what thinking with another mind looks like. The AI is the other mind. You are the one deciding when to explore, when to push back, when to stop, and when the output is truly yours. Every time you make those decisions well, you get better at making them. That is the compound effect in action.

The loop does not require AI. You can run it with a colleague, a mentor, a blank page. But AI makes it available on demand, at any hour, on any problem. The practice becomes as frequent as you want it to be. And frequency is what turns a technique into a capability.

> The compound test
>
> Take the AI away. Are you better at your work than you were six months ago? If yes, you have been conversing. If no, you have been delegating.

14.6 The Closing Argument

There is a line that has appeared twice before in this book, and it belongs here at the end.

Judgement cannot be delegated to something that has read everything but experienced nothing.

That sentence is not a limitation of AI. It is a statement about what makes you irreplaceable. Your experience. Your judgement. Your ability to weigh competing priorities, read a room, know when the data is pointing in the right direction and when it is lying to you. No model has that. No model will.

What a model can do is help you sharpen those capacities. It can surface what you had not considered. It can pressure-test your reasoning. It can show you the gap between what you think you know and what you actually know. But only if you stay in the conversation. Only if you do the thinking.

The goal was never to get AI to do more.

It was to become more capable yourself.

That is the practice. That is the point. And if you have read this far, you already know how to begin.

Appendix A

Prompt Structuring Frameworks

RTCF is not the only prompt structuring framework. A family of mnemonics emerged in 2023-24 as practitioners and educators figured out what makes a good opening prompt. They all capture the same insight: telling the AI who to be, what to do, what background it needs, and how to format the output produces dramatically better results than an unstructured request.

This appendix compares the most widely used frameworks side by side, using the same task so you can see how each one shapes the prompt differently. The task: getting AI to help analyse student feedback data.

None of these frameworks have a single academic source. They are practitioner knowledge, patterns that emerged from use, were shared in workshops and online communities, and proved useful enough to stick. The underlying principle (structured communication outperforms unstructured communication) is well-established. The mnemonics are packaging, not discovery.

We chose RTCF for this book because it has the fewest elements to remember and covers the ground that matters most. But if a different framework clicks better for how you think, use that one. The letters do not matter. The habit of thinking before prompting does.

A.1 RTCF: Before and After

To see why structure matters, compare these two prompts for the same task.

Weak prompt

"What can you tell me about student feedback?"

RTCF prompt

"You are an experienced higher education analyst **(R)**. Identify the three strongest themes and two areas of concern in the student feedback data I will paste below **(T)**. This is from a second-year undergraduate unit with 85 responses across two campuses **(C)**. Present findings as a summary paragraph followed by a table with columns: Theme, Evidence, and Recommended Action **(F)**."

You do not need all four elements every time; use what is relevant. But when a prompt feels vague and you are not sure why, checking against RTCF will almost always reveal what is missing.

A.2 The Family

A.2.1 RTCF: Role, Task, Context, Format

The framework used throughout this book. Four elements, no extras. Role sets the expertise. Task specifies the action. Context provides the background. Format defines the output structure.

Same task:

> "You are an experienced higher education analyst **(R)**. Identify the three strongest themes and two areas of concern in this student feedback data **(T)**. This is from a second-year undergraduate unit with 85 responses across two campuses **(C)**. Present as a summary paragraph followed by a themed table **(F)**."

A.2.2 CRAFT: Context, Role, Action, Format, Tone/Target

Adds a fifth element: Tone or Target audience. Useful when the same content needs to be pitched differently for different readers, such as a board summary versus a team debrief.

Same task:

> "This is student feedback data from a second-year undergraduate unit with 85 responses across two campuses **(C)**. You

> are an experienced higher education analyst **(R)**. Identify the three strongest themes and two areas of concern **(A)**. Present as a summary paragraph followed by a themed table **(F)**. Use clear, non-technical language suitable for a faculty teaching committee **(T)**."

A.2.3 CO-STAR: Context, Objective, Style, Tone, Audience, Response

Six elements. Splits what RTCF bundles into Context and Format into finer-grained components. Useful for communication-heavy tasks where style, tone, and audience are genuinely distinct decisions.

Same task:

> "I have student feedback data from a second-year undergraduate unit, 85 responses across two campuses **(C)**. Identify the three strongest themes and two areas of concern **(O)**. Write in the style of an institutional quality report **(S)**. Professional but accessible, avoiding jargon **(T)**. For a faculty teaching committee that includes both academics and professional staff **(A)**. Summary paragraph followed by a themed table with columns: Theme, Evidence, Recommended Action **(R)**."

A.2.4 RISEN: Role, Instructions, Steps, End goal, Narrowing

Five elements with a distinctive feature: it asks you to specify the steps the AI should follow and the end goal you are working toward. Useful for multi-stage tasks where the process matters as much as the output.

Same task:

> "You are an experienced higher education analyst **(R)**. Analyse the student feedback data I will paste below **(I)**. First, read all responses and identify recurring themes. Then, categorise each theme as a strength or concern. Finally, assess which findings are actionable **(S)**. The end goal is a briefing document for a faculty teaching review **(E)**. Focus only on themes that appear in at least 10% of responses **(N)**."

A.2.5 CREATE: Character, Request, Examples, Adjustment, Type, Extras

Six elements with a distinctive feature: it builds iteration directly into the framework. The Adjustment step acknowledges that the first response

is rarely the final one, and the Examples step lets you show the AI what you want rather than just describing it.

Same task:

> "Act as an experienced higher education analyst **(Character)**. Analyse this student feedback data and identify the strongest themes and areas of concern **(Request)**. Here is an example of the format I want: 'Theme: Student engagement | Evidence: 15 responses mentioned limited interaction | Action: Introduce weekly discussion activities' **(Examples)**. The initial output should focus on the three most prominent themes; I will refine the scope after reviewing **(Adjustment)**. Present as a themed table **(Type)**. Include a brief note on statistical confidence where relevant **(Extras)**."

A.2.6 APE: Action, Purpose, Expectation

Three elements. The most minimal framework. Useful for quick, low-stakes prompts where brevity matters more than precision.

Same task:

> "Analyse this student feedback data **(A)** to help me prepare for a teaching review meeting **(P)**. I expect a short summary of key themes and a table of findings with recommended actions **(E)**."

A.3 Choosing a Framework

Table A.1: Prompt structuring frameworks compared. More elements give finer control but take longer to compose.

Framework	Elements	Best for
APE	3	Quick tasks where speed matters more than precision
RTCF	4	General-purpose prompting that covers most situations
CRAFT	5	Communication tasks where tone and audience are distinct decisions
RISEN	5	Multi-step tasks where the process matters

Framework	Elements	Best for
CREATE	6	Tasks where you can show examples and expect to iterate
CO-STAR	6	High-stakes communication with specific style and audience requirements

The frameworks are listed from simplest to most detailed. Start with RTCF. If you find yourself repeatedly needing to specify tone separately from context, try CRAFT. If you need to define a multi-step process, try RISEN. If you have a clear example of what good output looks like, try CREATE. If you never need more than three elements, APE is fine.

The goal is not to memorise all six. It is to build the habit of structuring your thinking before you type. Any of these frameworks will get you there. Pick one, use it until it becomes instinct, then let it go.

A.4 The Caveat

Every one of these frameworks improves your *opening* prompt. None of them replace the conversation that follows. A perfectly structured RTCF or CO-STAR prompt is still a single prompt. It is still one shot. The real value comes from what you do with the response: iterate, push back, refine, and make the output yours.

Structured prompts are a better starting point. They are not a substitute for staying in the conversation.

Appendix B

Quick Reference Cards

Tear-out summaries of the key frameworks, tools, and techniques in this book. Pin them next to your screen or keep them open in a tab.

B.1 RTCF Prompt Framework

```
              RTCF PROMPT FRAMEWORK

  R - ROLE       Who should the AI be?
                 "You are a [expert type]..."

  T - TASK       What should it do?
                 Use action verbs: Analyse,
                 Compare, Create, Evaluate...

  C - CONTEXT    What does it need to know?
                 Industry, constraints, audience

  F - FORMAT     How should output look?
                 Structure, length, style
```

Four components. Four questions. Answer them before you type your prompt, and the conversation starts in a much better place. See Chapter 8 for the full chapter.

B.2 VET Your AI

Three steps to run before you trust any AI output that matters.

Table B.1: The VET framework: Verify, Explain, Test.

Step	Question	What you do
V – Verify	Can I find this independently?	Cross-reference claims against credible sources. If the AI cites something, look it up.
E – Explain	Can I explain this in my own words?	If you cannot walk someone through the reasoning, you do not understand it well enough to use it.
T – Test	Does this hold up under scrutiny?	Challenge the output. Change the assumptions. Ask "what if the opposite were true?"

VET in practice:

Table B.2: VET applied to real AI output. The domain does not matter. The habit does.

AI says…	V: Verify	E: Explain	T: Test
"Studies show pH drops faster in co-culture"	Find the actual studies	Can I explain the mechanism?	What about different substrates?
"Optimal temperature is 37C"	Check the literature	Why 37C specifically?	What happens at 35C or 40C?
"This method has 95% recovery rate"	Where is that number from?	What does 95% mean here?	Under what conditions?

If you pass all three steps, the output is yours. If you fail any of them, go back into the conversation and iterate. See Chapter 12 for the full chapter.

B.3 The Conversation Loop

Four stages. One feedback arc. Most good work passes through more than once.

Table B.3: The Conversation Loop: Brainstorm, Ideate, Iterate, Amplify.

Stage	What happens	Your mindset
Brainstorm: Define your question	Arrive with a question, not a task.	"What am I actually trying to figure out?"
Ideate: Explore possibilities	Go wide with the AI.	"What angles haven't I considered?"
Iterate: Push back and refine	Challenge, redirect, sharpen.	"That's close, but here's what's wrong…"
Amplify: Make it yours	Fold in what you know. Own the result.	"How do I bring in what the AI doesn't know?"

The loop is not linear. You can jump from Iterate back to Brainstorm when you realise the question was wrong. You can loop from Amplify back to Ideate when a new angle emerges. The sign that you are done is not that the AI stopped producing output. It is that your thinking has landed somewhere solid. See Chapter 5 for the full chapter.

B.4 Eight Techniques at a Glance

Table B.4: Eight techniques for deeper thinking with AI.

#	Technique	What it does in one line
1	**Reverse Prompting**	The AI asks *you* questions to scope a problem before you solve it.
2	**Pros and Cons**	Systematic evaluation of competing options against specific criteria.
3	**Stepwise Chain of Thought**	Forces the AI to show its reasoning one step at a time so you can check each link.
4	**Role Play**	The AI adopts a stakeholder perspective so you can rehearse conversations and anticipate objections.
5	**Debating**	The AI argues the opposite position, strengthening your reasoning through challenge.
6	**Formative Self-Testing**	The AI generates practice questions and gives immediate feedback on your answers.

#	Technique	What it does in one line
7	**Expert Panel**	Multiple simulated experts weigh in from different disciplines, forcing you to synthesise.
8	**Risk Deep-Dive**	Structured identification and analysis of risks, blind spots, and failure modes.

Each technique is a different way to stay in conversation rather than delegating. Pick the one that matches your task. See Chapter 10 for detailed walkthroughs and examples.

Appendix C

Further Reading

This book is practitioner-focused, but its ideas are grounded in research. The references below point to the studies, papers, and books behind the key claims. They are organised by chapter so you can follow up on whatever interests you most.

This is not a comprehensive literature review. It is a trail of breadcrumbs for curious readers.

C.1 Part 1: Understanding the Landscape

C.1.1 What Are Large Language Models?

How LLMs work (prediction as the core mechanism):

- Vaswani, A. et al. (2017). "Attention Is All You Need." *Advances in Neural Information Processing Systems.* The foundational paper introducing the transformer architecture that underpins all modern LLMs.
- Bommasani, R. et al. (2021). "On the Opportunities and Risks of Foundation Models." *arXiv preprint.* Defines the category of "foundation models," large models trained on broad data that can be adapted to many tasks, and maps their capabilities, risks, and societal implications.

RE2 (Re-Reading) prompting and unidirectional attention:

- Xu, X. et al. (2024). "RE2: Region-Enhanced Re-Reading Improves Large Language Models." *EMNLP 2024.* Demonstrates that repeating a question in the prompt creates pseudo-bidirectional

attention, improving reasoning accuracy. Twice is the sweet spot; more repetitions degrade performance.

Hallucination:

- Ji, Z. et al. (2023). "Survey of Hallucination in Natural Language Generation." *ACM Computing Surveys*. A comprehensive overview of why LLMs generate plausible but false content, and the structural reasons this problem persists across model generations.

Bias and fairness:

- Gallegos, I.O. et al. (2024). "Bias and Fairness in Large Language Models: A Survey." *Computational Linguistics*. A thorough survey of how biases in training data manifest in LLM outputs, and the limitations of current approaches to mitigating them.

Deep learning foundations:

- LeCun, Y., Bengio, Y. and Hinton, G. (2015). "Deep Learning." *Nature*. The landmark review by the three pioneers of deep learning, providing accessible context for the layered pattern recognition that makes LLMs possible.

C.1.2 Does AI Make Us Dumber?

Cognitive offloading:

- Risko, E.F. and Gilbert, S.J. (2016). "Cognitive Offloading." *Trends in Cognitive Sciences*. The foundational paper on how humans use external tools to reduce cognitive demand, and when this helps versus hinders learning.
- Sparrow, B., Liu, J. and Wegner, D.M. (2011). "Google Effects on Memory: Cognitive Consequences of Having Information at Our Fingertips." *Science*. The classic study showing that access to searchable information changes what we bother to remember, a pre-AI demonstration of cognitive offloading that foreshadowed current concerns about AI dependency.

The generation effect (producing information improves retention):

- Slamecka, N.J. and Graf, P. (1978). "The Generation Effect: Delineation of a Phenomenon." *Journal of Experimental Psychology: Human Learning and Memory*. Demonstrates that actively generating information leads to better memory than passively receiving it. This is the research basis for why doing your own thinking first matters.

AI and cognitive decline concerns:

- Bastani, H. et al. (2025). "Generative AI Without Guardrails Can Harm Learning: Evidence from High School Mathematics." *Proceedings of the National Academy of Sciences*. Evidence that students using AI without guardrails perform worse on subsequent unaided tasks, supporting the book's argument for conversation over delegation.

Metacognitive laziness:

- Fan, Y. et al. (2025). "Beware of Metacognitive Laziness: Effects of Generative Artificial Intelligence on Learning Motivation, Processes, and Performance." *British Journal of Educational Technology*. Directly documents how AI use can erode the self-monitoring habits that distinguish deep learning from surface acceptance. This is the research behind the book's discussion of metacognitive laziness.

- Gerlich, M. (2025). "AI Tools in Society: Impacts on Cognitive Offloading and the Future of Critical Thinking." *Societies*. Examines how widespread AI tool use affects critical thinking capacity at a societal level.

Cognitive surrender:

- Shaw, S.D. and Nave, G. (2026). "Thinking — Fast, Slow, and Artificial: How AI Is Reshaping Human Reasoning and the Rise of Cognitive Surrender." *Working paper, The Wharton School*. Introduces "cognitive surrender," essentially what this book calls delegation, as a distinct phenomenon where people defer to AI not because it is right, but because thinking is effortful.

C.2 Part 2: Principles

C.2.1 The Conversation Loop

Cognitive strategy transfer:

- Wei, J. et al. (2022). "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models." *NeurIPS 2022*. The paper that formalised "show your working" as a prompting strategy, demonstrating that step-by-step reasoning prompts dramatically improve performance on complex tasks.

- Wang, X. et al. (2023). "Self-Consistency Improves Chain of Thought Reasoning in Language Models." *ICLR 2023*. Shows that sampling multiple reasoning paths and selecting the most consistent answer improves accuracy, the AI equivalent of "check your answers."

Human-AI collaboration:

- Wilson, H.J. and Daugherty, P.R. (2018). "Collaborative Intelligence: Humans and AI Are Joining Forces." *Harvard Business Review*. Argues that the greatest performance gains come not from AI alone or humans alone, but from structured collaboration. This is the professional case for conversation over delegation.

Few-shot prompting (worked examples):

- Brown, T. et al. (2020). "Language Models are Few-Shot Learners." *NeurIPS 2020*. The GPT-3 paper that demonstrated giving a model a few examples in the prompt dramatically improves task performance. This is the research behind the "worked examples → few-shot prompting" transfer.

C.2.2 Staying Critical

The Gell-Mann Amnesia Effect:

- Crichton, M. (2002). "Why Speculate?" Lecture. The original articulation of the phenomenon, attributed to physicist Murray Gell-Mann. Not a formal study, but a widely recognised observation about how we selectively apply scepticism.

Sycophancy in LLMs:

- Sharma, M. et al. (2023). "Towards Understanding Sycophancy in Language Models." *arXiv preprint*. Documents how LLMs systematically tailor responses to match user beliefs, even when those beliefs are incorrect. Demonstrates that sycophancy is a persistent feature of RLHF-trained models, not a bug that will be fixed.
- Perez, E. et al. (2023). "Discovering Language Model Behaviors with Model-Written Evaluations." *ACL 2023*. Includes evidence of sycophantic behaviour across multiple model families and scales.

The AI Dismissal Fallacy:

- Claessens, S., Veitch, P. and Everett, J.A.C. (2026). "Negative Perceptions of Outsourcing to Artificial Intelligence." *Computers in Human Behavior*. Research documenting that people systematically devalue work when they learn AI was involved in producing it. This is the empirical basis for the AI Dismissal Fallacy discussed in this book.

Bias in training data:

- Bender, E.M. et al. (2021). "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" *FAccT 2021*. A widely discussed paper on how training data biases propagate through LLMs, relevant to the book's warnings about not taking AI output at face value.

C.3 Part 3: The Methodology

C.3.1 RTCF, VET, and the Prompt Structuring Frameworks

RTCF (Role, Task, Context, Format) and VET (Verify, Explain, Test) are not drawn from a single published source. They are practitioner frameworks that synthesise established practices into memorable mnemonics. The same is true of related frameworks like CRAFT, CO-STAR, RISEN, and APE. All emerged from the practitioner and educator community in 2023-24, none have a definitive academic origin, and all capture the same core insight about structured communication.

The underlying principles, however, are well-supported:

Structured prompts outperform unstructured ones:

- Zheng, H. et al. (2023). "Is ChatGPT a Good NLG Evaluator?" *arXiv preprint*. Among other findings, demonstrates that assigning specific roles to LLMs measurably changes output quality and focus. This is the research basis for the "R" in RTCF.
- Federiakin, D. et al. (2024). "Prompt Engineering as a New 21st Century Skill." *Frontiers in Education*. Makes the case that structured prompting is a transferable professional skill, not a niche technical ability, supporting the book's argument that prompt structuring draws on existing competencies.
- Sahoo, P. et al. (2024). "A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications." *arXiv preprint*. A comprehensive survey of prompting techniques, providing the broader landscape in which frameworks like RTCF, CRAFT, and CO-STAR sit.

The Explain step in VET (the Feynman Technique):

- Feynman, R. (1985). *"Surely You're Joking, Mr. Feynman!"* The origin of the principle: if you cannot explain something in simple terms, you do not understand it. The "E" in VET is a direct application.

The Verify step in VET (information literacy):

- Wineburg, S. et al. (2022). "Lateral Reading and the Nature of Expertise." *Teachers College Record*. Demonstrates that experts verify claims by checking sources laterally rather than reading vertically, the same habit the Verify step builds.

One-shot vs iterative prompting:

- Madaan, A. et al. (2023). "Self-Refine: Iterative Refinement with Self-Feedback." *NeurIPS 2023*. Shows that iterative refinement consistently outperforms single-pass generation, supporting the book's argument that structured prompts are a starting point, not a destination.

C.3.2 Prompt Chaining

Task decomposition:

- Zhou, D. et al. (2023). "Least-to-Most Prompting Enables Complex Reasoning in Large Language Models." *ICLR 2023*. Demonstrates that breaking complex problems into sequential subproblems and solving them in order significantly improves accuracy. This is the formal basis for prompt chaining.

C.3.3 Eight Techniques for Deeper Thinking

Debating and adversarial prompting:

- Du, Y. et al. (2023). "Improving Factuality and Reasoning in Language Models through Multiagent Debate." *arXiv preprint*. Shows that having multiple LLM instances debate produces more accurate and well-reasoned outputs than single-model generation.

Practical AI prompting strategies:

- Mollick, E.R. and Mollick, L. (2023). "Assigning AI: Seven Approaches for Students, with Prompts." *arXiv preprint*. Seven structured approaches to using AI for learning, several of which map directly onto the techniques in this book (role play, debate, self-testing).

Formative self-testing (retrieval practice):

- Roediger, H.L. and Butler, A.C. (2011). "The Critical Role of Retrieval Practice in Long-Term Retention." *Trends in Cognitive Sciences*. The research basis for why testing yourself improves learning more than re-reading, the principle behind the Formative Self-Testing technique.

C.4 General Background

For readers who want a broader foundation in how AI systems work and how to think about their role in society:

- Mitchell, M. (2019). *Artificial Intelligence: A Guide for Thinking Humans*. New York: Farrar, Straus and Giroux. An accessible,

rigorous introduction to AI for non-specialists.

- Christian, B. (2020). *The Alignment Problem.* New York: W.W. Norton. Explores the gap between what we want AI to do and what it actually does, relevant to understanding why sycophancy, hallucination, and bias persist.
- Mollick, E. (2024). *Co-Intelligence: Living and Working with AI.* New York: Portfolio. A practitioner-oriented book on integrating AI into professional work, with a similar emphasis on human judgement.
- Shneiderman, B. (2022). *Human-Centered AI.* Oxford University Press. Argues for AI systems designed around human control and oversight rather than full automation, the design philosophy that aligns with this book's emphasis on staying in the conversation.

Appendix D

Interactive Tools

These companion tools let you practise the frameworks from this book in your browser. No login required, no data stored. Everything runs locally in your browser.

D.1 RTCF Prompt Builder

Build prompts step by step using the RTCF framework. Fill in Role, Task, Context, and Format, and the tool assembles a structured prompt you can copy and paste into any AI tool. Includes templates for Reverse Prompting, Devil's Advocate, VET checks, and more.

Open the RTCF Prompt Builder{target="_blank"}

D.2 RTCF Prompt Analyser

Paste any prompt and get instant feedback on which RTCF elements are present, partial, or missing. Useful for diagnosing why a prompt gave you a vague or unhelpful response; the answer is usually a missing element.

Open the RTCF Prompt Analyser{target="_blank"}

D.3 AI Readiness Assessment

A short quiz (3-4 minutes) that maps your current AI experience level and recommends which chapters of this book to prioritise. Whether you are a newcomer or a power user, it points you to the sections that will be most useful for where you are now.

Take the AI Readiness Assessment{target="_blank"}

D.4 Converse with This Book

A book about conversation with AI should be available as a conversation with AI.

The `llm.txt` file (available from the sidebar or the companion website) provides a clean text version of the entire book that you can paste into ChatGPT, Claude, NotebookLM, or any other AI tool. Then ask it questions, challenge its interpretation of the ideas, or use it to explore how the frameworks apply to your specific situation.

You can also upload `llm.txt` to Google NotebookLM to generate an audio overview or podcast-style discussion of the content.

The book's own methodology applies: do not accept the AI's summary of the book uncritically. Push back. VET the answers. Stay in the conversation.

About the Author

Figure D.1: Photo of Michael Borck

Michael Borck is a software developer and educator working at the intersection of human expertise and artificial intelligence. He developed the Conversation, Not Delegation methodology: the idea that AI is most valuable not as a tool you delegate to, but as a thinking partner you converse with.

The methodology grew out of a realisation: focusing on crafting the perfect prompt was just another form of delegation. The real value was never in the prompt. It was in the conversation that followed: the pushback, the iteration, the moments where your own thinking sharpened because you stayed engaged. That insight shifted the focus from prompt engineering to something closer to how professionals have always done their best work: through structured dialogue, critical evaluation, and judgement that only comes from experience.

Michael applies these principles across software development, education, and creative projects. He creates educational software and resources, and explores the 80/20 principle in learning and productivity.

Connect

- michaelborck.dev (https://michaelborck.dev) — Professional work and projects
- michaelborck.education (https://michaelborck.education) — Educational software and resources
- 8020workshop.com (https://8020workshop.com) — Passion projects and workshops
- LinkedIn (https://linkedin.com/in/michaelborck)

Other Books in This Series

Foundational Methodology:

- *Converse Python, Partner AI: The Python Edition*

Python Track:

- *Think Python, Direct AI: Computational Thinking for Beginners*
- *Code Python, Consult AI: Python Fundamentals for the AI Era*
- *Ship Python, Orchestrate AI: Professional Python in the AI Era*

Web Track:

- *Build Web, Guide AI: Business Web Development with AI*

For Educators:

- *Partner, Don't Police: AI in the Business Classroom*