

A LIBRARY, NOT A BLANK PAGE

Why are you still *writing* the prompt?

A business-user's guide to Fabric — Daniel Miessler's open-source library of expert prompts (Patterns) that paste straight into any AI tool you already use.

CONCEPT

Daniel Miessler · Fabric

FOR

Business users of AI tools

LENGTH

8-minute read

Every AI tool hands you the same blank *starting line*.

01 THE BLANK-TEXT-BOX PROBLEM

Open ChatGPT, Copilot, Gemini, Claude. What greets you is exactly what greeted you yesterday: an empty rectangle, a blinking cursor, expectant silence. The tool knows how to do almost anything. It is waiting for you to know what to ask.

So you type. "*Help me write...*" "*Can you summarize...*" "*I want to...*" Three sentences in, you are still scaffolding the request — telling it what role to play, what tone to use, how long the output should be, what format you want, what to leave out. By the time you have described what good would look like, you have done half the thinking the AI was supposed to save you.

The problem is not the model. The problem is that you are reinventing the prompt every single time. The same prompts, in slightly different words, written from scratch by millions of people every day. None of them written by someone who actually thought hard about how to extract the best output from a model for that specific job.

A — WITHOUT FABRIC

Same job, *blank box*.

YOUR PROMPT

Can you summarize this YouTube video for me?
I want the main points.

WHAT YOU GET BACK

A summary so generic you can't tell which video it is about. You spend the next five minutes asking follow-ups to tease out what was actually useful.

B — WITH A PATTERN

Same job, *refined prompt*.

YOUR PROMPT

[paste of the `extract_wisdom` Pattern] followed by the transcript of the video.

WHAT YOU GET BACK

Structured output: summary, ideas, insights, quotes with attribution, habits recommended, facts to verify, references, and a one-sentence takeaway. Always the same shape.

A library of prompts somebody already *figured out.*

02 THE PATTERN LIBRARY

Fabric is Daniel Miessler's answer to the reinvention problem. It is an open-source project on GitHub built around one simple unit: the *Pattern*.

A Pattern is a carefully written prompt — usually a couple of pages long, in plain markdown — that tells an AI exactly how to do one specific job. There is a Pattern for summarizing a YouTube video. A Pattern for extracting insights from an article. A Pattern for analyzing the claims in an argument. A Pattern for improving your own writing. There are **over two hundred of them**, all visible in the public repo, all readable, all free.

You do not have to install anything to use them. Open the patterns directory at github.com/danielmiessler/fabric, click the one you want, open its *system.md* file, copy the contents, paste into ChatGPT — then paste your actual input below it. That is the whole flow.

The Patterns are not precious. They are checked into a public repository precisely so people will copy them, modify them, and build their own. Daniel's whole pitch is that prompts are the fundamental unit of AI value, and the unit nobody is curating well. Fabric is the curation.

The bottleneck on good AI answers is not the model anymore. It is whether someone has already sat down and written a really good prompt for the specific job you are trying to do.

— THE FABRIC PHILOSOPHY

The rest of this primer covers six Patterns worth knowing about, how to use them in any AI tool you already have, and how Fabric pairs with TELOS — its sibling framework — to close the gap between "AI can technically do this" and "AI just did this for me, well."

Six Patterns to start *this week*.

03 THE FEATURED SIX

These translate best to ordinary business work — the jobs you find yourself doing fifty times a month without thinking of them as AI jobs. The companion page at you.abutarfa.com/fabric lets you copy each Pattern's full text directly from Daniel's repo with one tap.

@summarize

UNIVERSAL

Comprehensive recap of any long input. Returns a one-sentence summary, main points, and clean takeaways — in the same structured format every time.

PASTE IN Article body, meeting transcript, long email, YouTube transcript.

YOU GET One-sentence summary, main points, takeaways. Structured.

@extract_wisdom

THE FLAGSHIP

Fabric's most-used Pattern. Pulls ideas, insights, quotes, habits, facts, references, and recommendations out of long-form content. Made for YouTube and podcasts.

PASTE IN YouTube transcript, podcast transcript, long-form essay.

YOU GET Summary, 20 ideas, 10 insights, 15 quotes, habits, facts.

@extract_insights

SHARPER THAN SUMMARIZE

Tighter cousin to `extract_wisdom`. Surfaces only the non-obvious observations from any input. Use it when you want signal, not coverage.

PASTE IN Article, transcript, Slack thread, book chapter, industry report.

YOU GET A list of distilled insights — the things worth remembering.

@analyze_claims

FORENSIC

For argument-heavy content, isolates each claim and rates the evidence. Use it before forwarding a hot take or basing a decision on one article.

PASTE IN Opinion article, LinkedIn thinkpiece, pitch deck, exec memo.

YOU GET Each claim separated, evidence rated, counter-arguments surfaced.

@improve_writing

COPY-EDIT

Run your own draft through it. Returns the same content — tighter, clearer, grammar fixed — without rewriting your voice into corporate sludge.

PASTE IN Your draft — email, proposal section, memo, board update.

YOU GET A tightened version. Same voice. Fewer words.

@extract_recommendations

ACTION ITEMS

Pulls actionable advice out of any input. Strips throat-clearing and gives you the do-this-now list. Especially good on advice essays and how-to threads.

PASTE IN Advice essay, podcast transcript, expert interview, Reddit thread.

YOU GET A bulleted list of concrete recommendations, prioritized.

Copy the Pattern. Paste it in. *Send.*

04 THE PASTE-AND-GO METHOD

The official Fabric is a command-line tool. That is how most YouTube demos use it, and why a lot of business users assume it is "for developers." It is not. **Every Pattern is a block of plain text.** If you can paste, you can use Fabric.

i.

Open the Pattern

Navigate to the patterns folder on GitHub. Open the one you want. Click the system.md file.

ii.

Copy the contents

Select all and copy. Or use the Copy buttons on the companion page to grab it in one tap.

iii.

Paste into your AI tool

Open ChatGPT, Copilot, Gemini, or Claude. Paste the Pattern as your first message.

iv.

Paste your input below

The article. The transcript. Your draft. Hit send. The Pattern does the rest.

For Patterns you'll use repeatedly, save them somewhere persistent in your AI tool of choice. Here is where:

i.

ChatGPT

CUSTOM GPT OR PROJECT

Create a Custom GPT and paste the Pattern into the system instructions. One Custom GPT per Pattern works well for the ones you use often.

ii.

Microsoft Copilot

COPILOT STUDIO

Build an agent in Copilot Studio. Paste the Pattern into its instructions block. Pin to your sidebar for one-click access.

iii.

Google Gemini

GEMS

Each Pattern becomes a Gem. Drop the contents into the instructions field, name it after the Pattern, save.

iv.

Claude

PROJECTS

One Project per Pattern. Paste the Pattern into the project's custom instructions. Every chat in the project starts with it preloaded.

FOR DEVELOPERS, BRIEFLY

The CLI is *one line.*

If you are comfortable in a terminal, the native experience is faster. After installing Fabric (one-line installer for macOS, Linux, Windows), running a Pattern looks like: `pbpaste | fabric -p extract_wisdom`. Add aliases and you can just type `extract_wisdom`. Install instructions live in the Fabric repo on GitHub.

Better prompts plus better context equals better *answers*.

05 PAIR IT WITH TELOS

Fabric and TELOS are siblings — both Daniel Miessler's work, both trying to fix the same problem from different sides. They solve a single sentence together: **the AI does not have what it needs to give you a useful answer.**

FABRIC**The right *prompt*.**

An expertly crafted instruction for the specific job — written by someone who thought hard about how to get a model to produce a useful answer for that task.

TELOS**The right *context*.**

A short structured document about you — your problems, mission, goals, strategies, projects — that any AI can load so it stops treating you like a stranger.

Use both. A Fabric Pattern tells the model how to do the job. A TELOS file tells it whose job it is. Together they close the gap between "AI can technically do this" and "AI just did this for me, in my voice, on my actual problem." Read the TELOS companion primer at you.abutarfa.com.

Go to the *originals*.

06 RESOURCES

VIDEO · NETWORKCHUCK**You've Been Using AI Wrong**

The most-watched introduction to Fabric. The whole thing demonstrated in twenty minutes.

youtube.com/watch?v=UbDyjIIGaxQ

REPOSITORY**The Fabric *repo***

Forty thousand stars and counting. Browse all 200+ Patterns, read the philosophy, fork your own.

github.com/danielmiessler/fabric

AUTHOR**Daniel Miessler's *site***

Essays, podcasts, and the original thinking behind Fabric and TELOS. The long arc.

danielmiessler.com

Don't write the prompt. Use the one *already refined*.

An educational primer based on the open-source work of Daniel Miessler. Fabric and all Patterns are released under the MIT license and remain the work of their author and contributors. This primer translates the framework for business users of consumer AI tools and is in no way affiliated with the original project. Companion to the TELOS Method primer.