

1-pager for future reference

Getting More Value from Language Models: A Quick Reference

This one-pager summarizes key strategies and tips for effectively using language models, drawing from our workshop session and related insights.

Core Mindsets & Heuristics

- **Abundance Mindset:** Leverage the low cost of generation. Ask for many options, variations, or perspectives.
- **Verification > Generation:** It's often easier to review and select from model-generated options than to create from scratch.
- **"Laziness" as Efficiency:** Delegate liberally, including meta-tasks like improving your own prompts or structuring messy thoughts.
- **Spiky Profiles:** Recognize models excel at some tasks and struggle with others. Choose your tools and tasks wisely.
- **Lower the Barrier:** Use models to overcome friction and procrastination on tasks you might otherwise avoid.
- **Iterate Rapidly:** Embrace fast feedback loops. Start simple and refine; don't wait for the perfect prompt.

Effective Prompting Techniques

- **Explain Clearly:** Act as if instructing a bright generalist with no prior context. Would they understand? If not, clarify.
- **Be Explicit & Detailed:** Don't patronize the model, but clearly state your goal, desired output format, warnings, and relevant context. Detailed instructions often outperform few-shot examples for novel tasks.
- **Leverage Model Capabilities:**
 - Have the model interview you to extract information.
 - Use models to clean up, structure, or summarize messy inputs like transcripts or voice notes.
 - Dictate thoughts or drafts and have models structure them (see Workflow section).
- **Iterative Refinement:**
 - Review outputs critically. Don't accept first drafts.
 - Provide feedback on outputs: Explain *what's* wrong and *why*.
 - Ask the model to critique its own output and suggest prompt improvements.
 - Use qualitative feedback (e.g., "make it shorter," "be less formal," "this vibe is off") and iterate.
- **Avoid Outdated/Ineffective Tips:**
 - You aren't "conditioning internet text" in the old sense; focus on clear instructions.
 - Skip elaborate role-playing unless the analogy genuinely clarifies the task for a human.

Getting Multiple Perspectives & Variations

- **Ask Different Models:** Claude and GPT-4 often offer complementary strengths and perspectives. Paste the same prompt into multiple models.
- **Request Ranges/Scales:** Ask for variations along a specific axis (e.g., "3 versions varying in formality," "summaries of increasing technical detail," "emails from 1-5 politeness"). Use numeric scales to request intermediate points (e.g., "give me a 3.5 politeness").
- **Seek Model Feedback:** Ask models to evaluate drafts (yours or theirs) for clarity, persuasiveness, etc..

Maximizing Context & Examples

- **More is More:** Provide ample context and examples, especially for tasks requiring specific style or knowledge. Dump relevant documents, previous work, style guides, etc..
- **Use Projects (Claude/GPTs):** Organize context and instructions for recurring tasks (e.g., Email Helper, Blog Writing, Research Promoter). Use the GPT Builder or Claude Projects feature.
- **Text > Attachments:** When possible, paste context directly as text (Markdown preserves formatting) rather than attaching files, as it's often processed more efficiently.

Workflow Integration: Voice & Transcription

- **Voice Input:** Use voice dictation (e.g., via GPT-4 voice chat, SuperWhisper) to capture thoughts while multitasking (commuting, exercising). This can be easier than structured writing initially.
- **Transcription:** Use tools like Fireflies.ai to transcribe voice notes or conversations.
- **Structure from Chaos:** Feed transcripts to models to help structure content, draft sections, or create Q&A documents.

Common Pitfalls to Avoid

- **Insufficient Context:** Models need background to perform well.
- **Single-Turn Interactions:** Engage in conversation and iteration for better results.
- **Expecting Perfection/Consistency:** Models can make mistakes or forget instructions. Always review.
- **Lack of User "Ownership":** Remember models execute instructions, even flawed ones. You are responsible for defining the *right* task, providing context, evaluating outputs, and deciding next steps.

Working doc

Language Models Workshop

Instructions

This is a living document where we'll collect use cases for language models. Feel free to:

- Add your own use cases to any section
- Comment on existing use cases with tips or variations
- Ask questions about implementation or prompt strategies

Use Cases

Information processing

- **Non-deep search:** "I would previously have got the answer by googling and clicking on a few links"
 - Claude or ChatGPT
 - You may need a VPN if you're not in the US
- **Deep Research:** "It would be worth spending a few hours of RA time on this if I had an RA"
 - Detailed advice [here](#)
- **Reading papers:**
 - Summaries at multiple levels of detail
 - Answering qualitative questions about someone's research interests
 - Compare/contrast multiple papers on the same topic
- **NotebookLM for podcast analysis:**
 - Detailed advice [here](#)

Self-dev

- **Getting feedback on drafts:**
 - Ask for specific types of feedback (clarity, structure, tone)
 - Have models identify potential areas of confusion
 - Request both overall impressions and specific suggestions
 - Ask multiple models, and create different experts
 - Do you like Kelsey Piper's writing? Why not have a Claude project set up to give feedback with 20 examples of her best work?
- **Meeting notes and transcripts:**
 - Give me generalised feedback on my management
 - Rich/contextual search
 - Summarise/rephrase

- Analyze what impression I'm giving to my reports of how they're doing
- **Improving writing:**
 - Ask for rephrasing along specific dimensions (formality, conciseness, warmth)
 - Gemini 2.5 canvas
 - Get multiple alternatives for problematic sentences
 - Combine feedback from different models for better results
 - Lots of my [blog](#) has more detail

Writing (quickly)

- **Drafting memos or messages:**
 - Convert voice notes or transcripts into structured writing
 - Transform bullet points into coherent paragraphs
 - Create Q&A format documents from conversations
 - Filling in 1on1 and perf review templates from transcripts or notes
- **Blog writing workflow:**
 - Record voice notes while exercising/commuting/walking
 - Transcribe and have models help structure content
 - Then fill in the structure verbally
 - Get editing suggestions from multiple models
 - Lots of my [blog](#) has more detail
- **Formatting and visualization:**
 - Create conceptual/visual diagrams from text w/ NotebookLM
 - Format content for different platforms
 - Blogpost -> Tweet thread
 - Clean up messy text (e.g., Slack threads, meeting transcripts)
 - I scraped all of my blogposts into markdown using code that Claude wrote.

Coding/scripting as a non-programmer

- Models can walk you through the setup
- If you get an error message, just paste the error message straight back into the chat
- Read the CoT
- If google sheets 'should' be able to do something, it probably can with appscript written by a LLM

Personal/Misc Use Cases

- Translation and image recognition while traveling
- Medical research and information organization
- Task planning and prioritization
- Therapy session preparation/journaling

Your Use Cases (Add here!)

[Name]: Use case: Prompt example: Value gained:

[Name]: Use case: Prompt example: Value gained:

Q&A

[My current Claude Projects - by Alex Lawsen](#)

Workshop Activities

Project Setup Activity

Task: Create a Claude project for a recurring task

1. Identify your task (3 minutes)

- Choose something you do repeatedly that would benefit from AI assistance
- Example tasks: Summarizing research papers, drafting emails, preparing meeting notes

2. Set up your project (7 minutes)

- Go to claude.ai and click "Create Project"
- Name your project based on its purpose
- Write clear instructions in the "Project Instructions" section
- Add relevant examples or reference materials to the "Project Knowledge" section

3. Test your project (5 minutes)

- Start a conversation within your new project
- Try a simple task to see if Claude follows your instructions
- Make note of any adjustments needed

Advanced prompting (if you're all set for projects)

Task: Find a challenging task at the edge of what language models can do

1. **Identify a challenging task** (3 minutes)
 - Choose something you think might be just beyond what language models can do well
2. **Initial attempt** (3 minutes)
 - Try a straightforward prompt to see what the model produces
 - Note specific areas where the response falls short
3. **Iterative improvement** (9 minutes)
 - Refine your prompt using techniques discussed in the workshop
 - Try adding more context, examples, or specific instructions
 - Experiment with breaking the task into smaller steps
 - Test different approaches and compare results
 - Tell the model what's going wrong and ask it to improve the prompt

How I put this together

Sorry, my Claude subscription is a business one, so I can't share chats externally, and I've removed the link that I refer to in the transcript.

My input was almost entirely dictated with [superwhisper](#)

Gemini rewrote the 1-pager once I was finished editing the rest.

Alex's session notes/talking points

Language Models Workshop - Facilitator Guide

Workshop Structure (60 minutes)

1. Introduction & Your Experience (7 minutes)

- Welcome everyone and explain the session's purpose
- Share your journey with language models and how your approach evolved
- Set expectations: "The goal is for everyone to start actively using these tools"

Talking Points:

- Started using LMs strategically - saw they were improving rapidly, wanted to develop skills early
- Initial experience wasn't immediately productive - early blog posts might have been quicker without LMs
- Clear now that I'm getting significant value, partly from models improving and partly from practice
- Developed tacit knowledge about when/how to use models effectively - this is what I want to share
- Bottom line message: "Just actually use them - here are ideas for how, get inspired and iterate"
- Note the range of familiarity in the room (some power users, some beginners)

2. Key Heuristics for Language Model Use (10 minutes)

High-level approaches to LM use:

- **Adopt an abundance mindset:** LMs can do things extremely cheaply - use this to your advantage
- **Verification is easier than generation:** Get models to generate many options, then pick the best ones
- **Embrace "laziness" as efficiency:** Delegate liberally, even for meta-tasks like prompt improvement
- **Recognize LMs have "spiky" skill profiles:** Exceptionally good at some things, terrible at others
- **Ask for multiple perspectives:** Different models often have complementary strengths
- **Lower the barrier to starting:** Models make it easier to overcome procrastination
- **Iterate quickly:** Fast feedback loops enable rapid improvement

How to prompt well:

1. **Explain as if to a bright generalist**
 - Would someone with no context understand your instructions?
2. **Leverage model capabilities**
 - Get models to interview you, then refine your answers
 - Have them clean up your messy thinking
3. **Be explicit but not condescending**
 - Write out exactly what you want
 - Don't patronize or oversimplify
 - Think "educated layperson" standard
4. **Use iterative improvement**
 - Read outputs and refine prompts
 - Ask the model to suggest prompt improvements
 - Tell models what's wrong and get them to fix it
5. **Avoid outdated tips**
 - You're not "conditioning internet text"
 - Detailed instructions > examples (few-shot) for most tasks
 - Skip unnecessary role-playing
 - Only use analogies if genuinely helpful

3. Use Cases Showcase (12 minutes)

- Walk through select use cases from the collaborative document (highlighted)
- Highlight when to use one model versus another (Claude vs GPT-4)
- Encourage participants to add their own use cases to the shared doc

4. Q&A Session (8 minutes)

- Open the floor for questions about specific use cases or prompting techniques
- Address common concerns/misconceptions
- Surface pain points people are experiencing with their current usage

5. Projects Setup & Hands-on Workshop (23 minutes)

- **Explain Claude projects** (5 min)
 - What they are and why they're useful
 - Quick demo of setting one up
 - Share examples from your personal projects (email helper, blog writing, etc.)
- **Activity instructions** (3 min)
 - Option 1: Set up a Claude project for a recurring task
 - Option 2: Try the "boundary pushing" exercise

- Emphasize that both activities can be done remotely or in person
- **Hands-on time** (15 min)
 - Circulate and help troubleshoot
 - For remote participants, be available in chat
 - End recording before this section