

GenAI Prompting Tips for Lawyers

BY NICK HARELL

Legal researchers and lawyers work in a field governed by language. We research, interpret, negotiate, and draft language, and we hope to do so efficiently and effectively. Generative artificial intelligence (GenAI) chatbots and other tools provide new technology to facilitate that work.¹ By deploying methods from the field of prompt engineering, we can hope to get the most out of these tools. This article introduces the concepts of prompting and prompt engineering, and identifies seven prominent prompt patterns that can enhance how legal professionals use GenAI tools.

Overview of Prompting and Prompt Engineering

In the context of GenAI, a prompt is an input given to a large language model (LLM) that guides the model's response. It is akin to a query or search in a legal research database or your favorite search engine. In the context of GenAI chatbots, a user typically has a problem they are trying to solve (e.g., drafting an email to a client, reviewing documents, charting a career plan, or brainstorming arguments), and the user engages the chatbot in a conversation with the hope of solving that problem.

Savvy legal researchers use a variety of advanced search techniques (e.g., Boolean operators, proximity connectors, modifiers, and filters) to make the most of their legal research platforms. Similarly, lawyers using GenAI tools can benefit from developing prompting techniques. And while the ethical and professional conduct considerations inherent in using GenAI for legal work are beyond the scope of this article, good prompts can mitigate the risks of using LLMs (e.g., hallucinations and inappropriate

tone of voice). All the same, lawyers should seriously weigh their ethical and legal obligations before using GenAI with client data.

With that in mind, prompt engineering describes the process of crafting inputs to achieve desired outcomes from a GenAI tool. For most users of GenAI tools, prompt engineering involves crafting an input that follows prompt patterns. A prompt pattern is the way a user phrases a prompt to solve a particular problem.

7 Prompt Patterns to Improve Results

Computer scientists study how GenAI systems and their underlying models work. And, while much is still to be understood, those scientists generally recognize that using prompt patterns can yield improved outputs from GenAI systems.² To that end, prompt patterns generally function the same across all of the publicly available GenAI chatbots (e.g., OpenAI's ChatGPT, Microsoft's Copilot, and Anthropic's Claude). However, specialized GenAI tools may limit their use or have other restrictions. Below are seven of the more common and utilitarian prompt patterns.

Persona Pattern

The persona pattern (sometimes called "role prompting") involves creating prompts that specify the character or role the GenAI should assume. For instance, instructing a GenAI chatbot to "please respond as a bankruptcy attorney might" would generate a response adopting the tone and terminology of a bankruptcy attorney. Notably, telling the GenAI to "please respond as the world's best bankruptcy attorney" might even improve the generated output. At the other end of the spectrum, asking the AI to "act as someone writing an email to a friend" will result in a less formal output.

Question Refinement Pattern

Through the question refinement pattern, a user can ask a GenAI chatbot a question and instruct the tool to recommend an improved question. This process mimics asking a colleague if you are framing an issue correctly. For example, you might enter the prompt: "whenever I ask a question, suggest a better question, and ask if I would like to use it instead." Then the next prompt could involve the problem to be solved (e.g., "how do I become a better legal researcher?").

Chain of Thought Pattern

The chain of thought pattern is simple but very useful. With this pattern, you take any prompt and append "let's think step by step" or "think very carefully" to the end of it. For example, to generate a detailed plan to host a CLE, you might input "please create a plan to host a one-hour CLE in our office. Let's think step by step." The output would include a plan with explanations for each step of the plan.

Cognitive Verifier Pattern

Building on the question refinement pattern and chain of thought pattern, the cognitive verifier pattern asks the GenAI tool to (1) break down a complex problem into its individual components, (2) provide answers to those components, and (3) combine all of the individual answers at the end to provide an ultimate answer to the original complex problem. This prompt pattern might look like this: "whenever you are asked a question, follow these rules: Generate a number of additional questions that would help more accurately answer the question. Combine the answers to the individual questions to produce the final answer to the overall question." By prompting for the additional questions and answers, users can better understand the tool's rationale for the solution it ultimately generates.

The Recipe Pattern

The recipe pattern helps generate missing steps to a multistep problem. For example, your prompt might say: "I have experience providing legal services and hope to become a professional legal researcher or law librarian. Provide a list of steps to help achieve that goal."

The resulting output will help fill in the blanks of a potentially complicated process.

Ask for Input Pattern

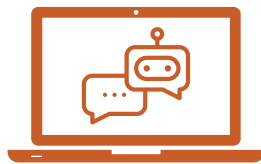
The ask for input pattern prevents the GenAI chatbot from responding immediately when a user might want to provide more information first. Because GenAI tools require a nontrivial amount of processing time to generate results, this can actually help you reach the ultimate answer faster. For example, “I am going to copy a series of emails into our conversation. Please summarize each email and its responses. Please provide the summaries by person. At the end, list any unresolved questions assigned to me. My name is Nick Harrell. Ask me for the first email in the series.” The last sentence tells the GenAI chatbot to wait for additional input before summarizing instead of merely responding to the first prompt.

The Semantic Filter Pattern

Legal professionals often have a need to remove sensitive information from a body of text. The traditional method of using a word processor’s “find” feature can miss items, and manual review can be time consuming. The semantic filter pattern provides a possible solution. It involves setting specific criteria for the AI to filter its responses. For example, to generate a response that removes names, social security numbers, and other sensitive data, your prompt could say: “filter your response to remove any personally identifiable information.”

Additional Techniques

Notably, users can expect better results from deploying the above patterns in tandem rather than in isolation. For example, if I wanted to learn more about GenAI, I might combine the patterns as follows: “Respond as though you are an experienced lawyer. I am going to provide my résumé, and I would like to develop more skills and understanding related to generative AI. When I ask a question, please generate additional relevant questions, and then combine all those answers to help answer my initial question. Please ask me for the first question.” Or, if I wanted help with the article at hand, I might write: “I need help writing an article about



The chain of thought pattern is simple but very useful. With this pattern, you take any prompt and append “let’s think step by step” or “think very carefully” to the end of it.

prompt engineering. Act as an experienced legal research professional who is writing an article for a state bar journal. I will provide you with several PDFs of my prior articles and then give you the topic of the new article. Please ask for the sample PDFs.”

In addition to combining patterns, users will find that prompting is an iterative practice. Users should expect to prompt GenAI chatbots multiple times to generate the best output. We can think of this as having a conversation with a colleague; infrequently do we ask one question and receive one complete answer to resolve a complex issue. Prompting works the same way.

The above represents a small sample of recognized prompting techniques. For those wishing to learn more, LinkedIn Learning (<https://www.linkedin.com/learning>), Khan Academy (<https://www.khanacademy.org>), Coursera (<https://www.coursera.org>), and others host videos and courses on prompt engineering, and Anthropic hosts an extensive guide to additional prompts.³ And, because prompting is an iterative exercise, one of the best ways to develop a facility with GenAI tools is to spend time using the tools.

Conclusion

The tools available to legal professionals will continue to evolve; curiosity, practice, and a healthy dose of skepticism and caution can help us adapt and deploy those tools for our benefit. By understanding and using prompt patterns, legal professionals can enhance their use of GenAI chatbots and the outputs they generate, resulting in our work being completed more efficiently and at least as effectively. 



Nick Harrell is a research services manager at Michael Best & Friedrich LLP—nick.harrell@michaelbest.com. He previously worked as a research and reference librarian with the federal judiciary, the University of Colorado, and the University of Miami.

Coordinating Editor: Michelle Penn, michelle.penn@colorado.edu

NOTES

1. I implore readers to read and digest the previous work on AI and the legal profession found in *Colorado Lawyer*, including Berkenkotter and Lipinsky de Orlov, “Artificial Intelligence and Professional Conduct,” 53 *Colo. Law.* 20 (Jan/Feb. 2024); Moriarty, “The Legal Challenges of Generative AI—Part 1,” 52 *Colo. Law.* 40 (July/Aug. 2023); and Abdullah, “Legal Research in the Age of AI,” 53 *Colo. Law.* 8 (May 2024). These and other excellent articles are available at <https://cl.cobar.org/topics/artificial-intelligence>.
2. E.g., White et al., “A Prompt Pattern Catalog to Enhance Prompt Engineering With ChatGPT,” arXiv preprint (Feb. 21, 2023), <https://arxiv.org/pdf/2302.11382>; DAIR.AI, Prompt Engineering Guide (2024), <https://www.promptingguide.ai>.
3. Anthropic, Prompt Library, <https://docs.anthropic.com/en/prompt-library/library>.