Towards an Effective Prompting With AI Systems: The Jujitsu Framework

Anurag Jain
Salem State University

Zaiyong Tang
Salem State University

Anil Gurung
Marshall University

As Large Language Models (LLMs) become central to AI applications, prompt engineering has emerged as a critical skill for optimizing model output. However, existing prompting techniques lack a solid theoretical foundation and continue to exhibit persistent limitations, such as insufficient objectivity and inadequate awareness of training data. We present a Framework that offers a principled model for prompt engineering, aligning with both the core mechanisms of LLMs and the principles of objective communication. This framework addresses critical gaps in existing methods and provides a systematic, transferable approach to prompt design. By working with - rather than against – the LLM architecture, the framework enhances accuracy, interpretability, and versatility across domains, thereby advancing the evolving landscape of human-AI interaction.

Keywords: AI, artificial intelligence, large language model, LLM, prompt engineering, prompting, framework, objective communication

INTRODUCTION

In recent years, prompt engineering has risen to prominence, driven by the widespread adoption of AI-based large language models (LLMs). These models have become integral to various applications, including chatbots, virtual assistants, content generation, and advanced analytics. This transformation is redefining the way humans interact with AI systems (Sabbatella et al., 2024). Reflecting this phenomenon, "prompt engineering" has garnered significant attention across academic publications, industry investments, and practical implementation.

Prompt engineering is considered essential for fully realizing the potential of LLMs. The capabilities of these large-scale neural networks can be unlocked through meticulously crafted prompts. Users can access a variety of functionalities in LLMs, such as composing creative poetry or answering technical questions, simply by modifying the prompt. This is achieved without altering the underlying parameters of the LLMs. As a result, the term "AI prompt engineer" has emerged, highlighting the importance of this skill in maximizing the value of AI systems. Korzynski, et al., (2003) suggest that prompt engineering is an indispensable skill for the twenty-first century.

Leading technology companies are actively promoting prompt optimization. For instance, McKinsey (2024) Has highlighted its significance. Learning platforms, such as LearnPrompting.org claim to equip users with the communication skills necessary to better harness the potential of LLMs. However, despite these advancements, prompt engineering largely remains a fragmented and informal practice, lacking a solid foundation of prompt design principles. Most current methods are experimental, relying on generally accepted heuristics, with limited evidence of theoretical frameworks that integrate linguistic, communicative, and computational aspects of prompting. Even experienced and highly technical users of LLMs often rely on trial and error, leading to inconsistent results that vary across domains and use cases. Such variability of output is particularly concerning in high-stakes fields like healthcare or finance, where even subtle misinterpretations can result in significant consequences (Miguelañez, 2025).

This paper seeks to address these gaps by proposing a simple yet comprehensive framework for prompt engineering that is informed by the principles of objective communication and aligned with the operating architecture of LLMs. We begin with an overview of prompt engineering and a comparative review of established prompting techniques. Next, we present the theoretical foundations and operational mechanisms of LLM architecture that are essential for systematic prompting. Through comparative evaluation, we highlight the limitations of current practices and demonstrate how our framework bridges these gaps. Finally, we outline the practical implications, theoretical contributions, and directions for future research.

Background on Prompt Engineering

Prompt engineering focuses on designing effective queries for interacting with generative AI systems, such as large language models (LLMs). LLMs respond to user prompts to generate text as a specialized form of generative AI. This engineering practice of crafting effective prompts allows human intentions to be translated into structured queries, guiding AI toward more precise and meaningful responses. As these language technologies have advanced, the principle that “good prompts equal good results” has become central to how both industry and academia approach maximizing LLM capabilities.

Prompt engineering evolved alongside breakthroughs in natural language processing (NLP). Early work by McCann et al. (2018) attempted to unify various language tasks within question-answer formats. Subsequently, Petroni et al. (2019) demonstrated the ability of LLMs to retrieve factual knowledge using targeted fill-in-the-blank prompts. The release of GPT-2 further enhanced language flexibility, enabling models to perform tasks with minimal prompting, as noted in the Hugging Face NLP Course (N.D.). Later, the introduction of foundation models marked a turning point, with GPT-3 by popularizing “Few-shot learning”. This allowed models to generalize from a few in-context examples without extensive re-training (Brown et al., 2020). Research then began developing automated prompt generation methods (Shin et al.,2020), seeking to transform prompt design from trial and error to an algorithmic approach for optimization of prompts. Further, innovations such as “soft prompting,” which replaces textual inputs with vector embeddings, and “chain-of-thought” prompting for explicit reasoning steps, have further diversified prompting methodologies (Wei et al., 2022). By 2023, prompt engineering had entered the mainstream, as evidenced by widely accessible prompt repositories and industry roles dedicated to prompt optimization (IBM Cloud Education, 2023; Mehra, 2024).

All the above strategies have been distilled into three widely accepted approaches: Few-Shot Prompting, which uses minimal examples to guide behavior; Chain-of-Thought Prompting, which encourages structured, methodical reasoning; and Role-Based Prompting, which assigns specific personas for contextual relevance. Despite their popularity, these strategies remain experimental and ad hoc. As Knoth et al. (2024) point out, the trial-and-error nature of prompt engineering continues to pose a challenge for systematic implementation.

Established Prompting Techniques

The three primary approaches that dominate the prompt engineering landscape are: Few-Shot Learning, Chain-of-Thought Prompting, and Role-Based Prompting. Each method surfaced to address unique challenges in human-LLM interaction through distinct strategies. While there are other approaches to prompting, these three are widely considered foundational. They share conceptual connections with other methods. For instance, Zero-Shot prompting shares methodological similarities with Few-Shot Learning but operates without examples. Similarly, many instruction-based prompts incorporate structured reasoning elements similar to that of Chain-of-Thought, while template-based approaches often blend role-based framing along with an output template.

These three methodologies represent the most widely adopted, thoroughly researched, and distinct approaches in current practice. Their prominence in both academic literature and industry applications establishes them as ideal candidates for examination. Furthermore, these techniques illustrate fundamentally different approaches to directing LLM behavior, enabling meaningful comparison across the full spectrum of communication principles.

Few-Shot Learning

Few-shot prompting emerged as a key by-product in the development of the GPT-3 foundation model (Brown et al., 2020). It demonstrated that LLMs could generalize to new tasks simply from in-context examples without requiring fine-tuning of the model. The Few-Shot Learning technique involves providing a small set of example pairs (input-output) to guide an LLM's pattern-matching capabilities. For example, a user might ask, "*Translate these sentences into French*," followed by two or three English-to-French translation pairs as examples. The model then identifies the pattern from these example pairs and generates similar translations for other sentences the user provides. By learning from the examples within the prompt (in-context learning), the model can infer patterns and apply them to new, and similar queries without additional training. This often improves the model's performance on complex tasks compared to using no examples at all (zero-shot prompting).While this approach guides to leverage a model's inherent pattern-recognition abilities, it suffers from limited scalability. Multiple examples can quickly overwhelm the capacity of the context window. Moreover, it cannot establish and enforce a clear boundary to restrict responses within the model's knowledge base. This limitation can lead to unreliable or out-of-context responses (Brownet al. 2020).

Chain-of-Thought Prompting

Introduced by Wei et al. (2022), Chain-of-Thought (CoT) prompting revolutionized AI reasoning capabilities. CoT encourages a model to explain and proceed in a step-by-step manner rather than jump straight to an answer. This approach leverages an LLM's capacity for sequential reasoning by instructing the model to follow a series of logical steps outlined in the prompt. For instance, in the case of a math problem a user might prompt the model by asking, "*Solve the following problem step-by-step: 'If John has 5 apples and gives away 2, how many does he have left?'*" This guidance helps the model sequentially explain each reasoning step.This approach to prompting helps the model demonstrate transparently. By enforcing the use of intermediate reasoning steps, it facilitates better interpretability and complex problem-solving. The model effectively "thinks aloud" addressing each part of the problem sequentially. This often leads to more coherent and accurate results for tasks that typically require structured reasoning. Chain-of-Thought (CoT) prompting enables transparency in reasoning for complex tasks (prompts). It enhances the interpretability of outputs by breaking down logical sequences during problem solving. However, the technique has a critical limitation: it lacks an objective and automatic mechanism to verify the correctness of each reasoning step. When a model's training data lacks comprehensive support for detailed logical steps, the model may propagate inaccuracies. Consequently, this can lead to seemingly coherent, but fundamentally flawed reasoning (Wei, J., et al. 2022).

Role-Based Prompting

Role-Based Prompting evolved from practical user applications. It involves assigning a specific personality or role to the model to constrain its responses within a defined context or behavior (Knoth et al., 2024). For instance, consider the prompt*: "You are an experienced travel agent. Provide a recommended 3-day itinerary for visiting Paris,".* Here, the model is instructed to adopt the role of a knowledgeable travel advisor. This role-based approach aligns with the concept of conversational agents as described by psycholinguistic theories. Responses become more relevant and tailored by adhering to context based on roles or personas. This induced mimicking of expert-system behavior helps reduce generic AI responses and enhances accuracy within a specific domain. Instructing an AI to adopt a role or a persona enables the LLM to focus on its contextual capabilities. The model then works with its relevant contextual training derived from its training data. This tactic helps the model generate more targeted and contextually appropriate content. However, asking the LLM to assume a role or persona can also lead to overconfidence, resulting in less accurate responses – particularly in domains beyond the model's actual knowledge base. This approach carries significant risks. The AI may generate plausible-sounding content that is factually incorrect, especially when the assigned role extends beyond its training data. Additionally, the model might prioritize maintaining the perspective of the given role than acknowledging its own knowledge limitations. Consequently, it can produce convincing yet unreliable responses. This is commonly referred to as hallucination in AI.

These techniques (Few-Shot, Chain-of-Thought, and Role-Based Prompting) are practical techniques designed to enhance AI responses in specific contexts. However, they were not originally envisioned as theoretical frameworks for structuring prompt design. Instead, these techniques emerged as ad-hoc solutions to distinct limitations in large language models (LLMs). Few-Shot Prompting addresses the lack of explicit task conditioning by providing example-driven learning. Chain-of-Thought Prompting focuses on improving logical coherence in multi-step reasoning tasks. Role-Based Prompting contextualizes responses by anchoring them within predefined domains. Each of these methods was developed independently to tackle specific performance gaps rather than as components of a unified theory of LLM interaction. Their effectiveness was validated empirically, with theoretical justifications following later, making them more akin to engineering heuristics than foundational principles.

Principles of Objective Communication

Objective communication principles emerge from Hayakawa's *Language in Thought and Action* (Hayakawa & Hayakawa, 1990), differentiating between the language styles for reporting factual data and those used for inferences or interpretive writing. Hayakawa emphasized that clear communication requires an explicit awareness of this distinction, alongside careful and precise language use. The concept of objective writing was further developed in scientific writing. Respected manuals such as, *The AMA Manual of Style* (Iverson et al., 2020) classified principles for clear, precise, and accurate communication. Similarly, in journalism, these principles have been formalized through style guides such as, *The Associated Press Stylebook* (Goldstein, 2022). The collective focus has been on factual reporting, clarity, and conciseness of expression. Drawing from these historical developments across multiple disciplines, modern objective communication theory has converged on a set of core principles:

- **Clarity**: Clarity involves using precise and unambiguous language to convey the exact meaning of the intended message. This requires minimal jargon, only used for precise technical terms and logically structured information. The objective for clear communication is to avoid ambiguity and therefore leave little room for misinterpretation. By ensuring clarity, the recipient can understand the intended message as closely as possible to the sender.
- **Context**: Providing essential background information allows the receiver to interpret the message more accurately. This involves establishing relevant circumstances, historical perspectives, or situational factors that are critical for understanding communication. Contextual information creates a framework that helps recipients integrate new details into their existing knowledge base. The objective is to provide sufficient background to ensure comprehensive understanding of the communication.
- **Specificity**: This principle emphasizes presenting information that is concrete, measurable, and precise to minimize ambiguity. It also suggests proving examples that ground abstract concepts in verifiable reality. Specificity counters vague generalizations by providing precise and quantifiable information. This principle aims to craft communication that can be understood uniformly, thereby reducing the potential for multiple interpretations across different recipients.

- **Objectivity**: Objectivity requires separating verifiable facts from subjective interpretations while acknowledging the limitations of current knowledge. This principle requires maintaining a neutral stance to minimize potential biases and addressing epistemic boundaries, i.e. limits of knowledge or understanding within a specific context. Objective communication requires a clear separation between facts and unverified claims. The aim is to present information transparently, ensuring clarity about what is known with certainty and what remains open for further investigation.

These principles create a comprehensive framework for evaluating effective communication across diverse fields, including journalism, scientific writing, business correspondence, and technical documentation. When applied consistently, they help ensure that information is transmitted accurately, interpreted correctly, and used appropriately by recipients. They also serves as a valuable lens for analyzing emerging communication methodologies, such as prompt engineering techniques developed for human-AI interaction.

Fundamental Operating Mechanisms of Large Language Models

Tracing the journey of text through language models reveals four foundational mechanisms: Context Processing (Vaswani et al., 2017), Attention Management (Bahdanau et al., 2015), Pattern Recognition (Devlin et al., 2019), and Training Data Awareness (Brown et al., 2020). Language models process text as sequences where order and relationships are crucial, similar to how humans comprehend sentences by maintaining their specific arrangement. This order is essential for comprehending its meaning, just as humans need to read sentences correctly to understand them.

Context Processing

Context serves as the vital framework within which an LLM operates. Think of context as the working memory of the model; it's not just the immediate prompt, but all the text the model can "see" at once. The context window represents the total number of tokens (chunks of text) a model can process simultaneously, ranging from a few thousand to over a million tokens in modern LLMs. This context includes the user's immediate prompt, previous exchanges in the conversation, system prompts or instructions, and any additional text or documents provided. Context defines boundaries and background information, shaping how the model interprets the user's input. For example, if a prompt starts with "*As a doctor,*" the model understands it should respond from a medical perspective, ensuring responses remain relevant to healthcare topics. This type of context cues activate specific knowledge domains and response patterns within the model's parameters.

Context functions like a temporary memory space where the model can reference information, make connections, and maintain coherence throughout a conversation. Without sufficient context, an LLM would respond to each new input as if it existed in isolation, making coherent conversations impossible. As information flows through the model, not all elements receive equal consideration. The model must determine which parts of the input deserve focus.

Attention

Attention mechanisms are perhaps the most revolutionary aspect of transformer architecture, solving the problem of efficiently processing long data sequences. In technical terms, attention calculates relevance scores between all possible pairs of tokens within the input. This creates a weighted system that determines which parts of the input should influence specific parts of the output. Attention mechanisms allow the model to focus selectively on the most important parts of the input, prioritizing relevant details while ignoring less critical information. For instance, if a prompt includes a detailed description of a problem and asks a specific question at the end, the attention mechanism helps the model concentrate primarily on the question and the directly related information, thereby improving the accuracy of its response. This selective focus is particularly valuable in long contexts where much of the information may only be tangentially relevant to the specific query.

There are several types of attention mechanisms: **self-attention**, where tokens attend to other tokens within the same sequence; **cross-attention**, used in encoder-decoder models to connect encoder outputs with decoder inputs; and **multi-head attention**, where multiple attention mechanisms operate in parallel, each capturing different types of relationships. The genius of attention mechanisms lies in their ability to create dynamic, contextual connections based on the specific input, rather than relying on fixed patterns. Consider analyzing a long legal document—when asked about a specific clause on a certain page, the attention mechanism enables the model to focus primarily on that clause and any related definitions elsewhere in the document, instead of treating all words with equal importance. After identifying and weighing relevant information, the model can then recognize and apply language patterns and meaning structures to generate coherent text.

Pattern Recognition

Pattern matching represents the model's ability to identify recurring structures, relationships, and regularities in language. During training, the model learns to associate patterns of tokens with their likely continuations or relationships. These patterns exist at multiple levels: **surface level** (grammar, syntax, and vocabulary patterns), **semantic level** (meaning-based relationships between concepts), **stylistic level** (writing style, tone, and genre conventions), and **logical level** (reasoning patterns and argumentative structures). Pattern matching enables the model to recognize similarities between the current prompt and patterns it has learned from its training data. For example, when asked to summarize a paragraph, the model uses patterns learned from seeing many previous examples of summaries, helping it generate a concise and coherent summary of new content. The model identifies the summary task, activates its learned templates of summary structures, and applies these patterns to the new content, extracting key information while maintaining the appropriate style and level of condensation typical of summaries.

The model's neural network encodes these patterns as weight distributions across its parameters. The model activates these learned patterns to predict appropriate responses when given a new input. For instance, when asked to code a solution to a specific programming problem, the model recognizes similar coding patterns encountered during training. It might identify that the problem requires a specific algorithm (like breadth-first search) and generate code based on the recognized pattern, adapting it to the specific requirements of the current task.

Training Data Awareness

Training data awareness represents the model's ability to understand its knowledge limitations and the scope of the information it was trained on. This concept manifests in several ways: **temporal awareness** (understanding when its knowledge cuts off), **domain awareness** (recognizing areas where it has expertise), **source diversity awareness** (understanding that different sources may provide conflicting information), and **certainty calibration** (expressing appropriate confidence levels based on data familiarity).

Training data awareness ensures the model recognizes the limits of its knowledge based on the information it was trained on. For example, a model with a knowledge cutoff date avoids answering questions that require information beyond that date. This awareness prevents the model from generating speculative or incorrect information, which it usually referred to as "hallucinations". Thereby improving the overall reliability of its outputs. Instead of fabricating information about events that occurred after its training cutoff, a properly calibrated model will acknowledge the boundaries of its knowledge and decline to provide details it cannot reliably verify.

This component is critical for responsible AI deployment as it helps models avoid confidently producing incorrect information. For instance, if asked about the details of a movie released after its training cutoff date, a well-calibrated model will admit that reliable information is unavailable rather than attempt to generate details that might be incorrect. Similarly, when discussing specialized medical knowledge, it might acknowledge the limits of its understanding and suggest consulting qualified professionals. For instance, when asked "*What is the boiling point of water?*" without any contextual information provided, the model correctly responds, "*100 degrees Celsius at standard pressure*", drawing from factual knowledge acquired during training. However, when asked about recent events or specialized topics beyond its training, it appropriately indicates the limitations of its knowledge.

COMPARATIVE ANALYSIS

This section analyzes the prompting techniques by evaluating them against the two frameworks principles of objective communication and fundamental LLM operating mechanisms. This examination reveals critical insights about the effectiveness, limitations, and potential gaps in current prompt engineering approaches.

Analysis of the Techniques Against Objective Communication

Our analysis reveals that none of the popular prompt engineering techniques fully align with all four principles of objective communication. They align selectively with the principles based on their unique strengths and weakness.

Few-Shot Learning prompting shows context and specificity by using examples and by showing the format and structure of the content expected in the LLM response. For example, instead of simply asking for "*a summary*," the prompt will include sample summaries of specific length, style, and focus areas. However, this technique does not address the requirement for clarity. Instead of providing clear instructions, such as, "*limit responses to 100 words*" or "*always include three supporting points",* it simply relies on the model to infer these rules by studying patterns from the examples provided. Additionally, if the provided examples coincidentally avoid certain topics, the model might incorrectly assume those topics should always be excluded. Therefore, Few-shot learning espouses the principles of context and specificity. However, it falls short of a) clarity, by not stating requirements directly, and b) objectivity, by not addressing what the model doesn't know or understand. The absence of clearly defined knowledge boundaries would make the model operate with "over confidence" in areas where its understanding is incomplete. This is classical hallucination by AI.

Chain-of-Thought prompting reflects the principle of clarity and specificity by deploying step-by-step reasoning to break down complex problems into visible stages rather than jumping to conclusions. For example, instead of having the LLM simply output, "*the answer is 42*", the prompt guides the model to show how it worked through the answer. However, this prompting technique fails to establish context. Chain-of-Thought prompts often direct the model to start calculations without first establishing what field or subject area is being addressed. Additionally, this prompting approach undermines objectivity by not ensuring that the model acknowledges its limitations. Instead of instructing a model to clearly state their limitations, such as, "*my understanding of quantum physics is limited to introductory concepts only*", it typically asks for continued reasoning even when the model may be outputting with uncertainty. Similarly, in a history-related reasoning chain, beginning with a wrong date will produce conclusions that seem logical but are incorrect. This is because the model builds upon each step with incorrect information. Therefore, the Chain-of-Thought prompting fulfills the principles of clarity and specificity. On the other hand, it neglects context by failing to establish the knowledge domains properly, and compromises objectivity by not instructing models to admit what they don't know. Therefore, this prompting technique may be useful for generating an output that is structured through the step-by-step reasoning, but it suffers from the vulnerability of generating erroneous information.

Role-Based Prompting strongly aligns with the context principle by defining expertise domains and activating relevant knowledge schemas. For example, instructing "*respond as an economist*", frames the discussion within that specific knowledge domain for the LLM. However, it lacks consistent clarity across various prompts because it rarely requires detailing what constitutes a good response within that role it has defined for the LLM. For instance, rather than providing clear guidelines like "*an expert economist would always cite specific data sources,"* it assumes the model knows what expertise in economics entails. While this approach effectively activates specialized knowledge patterns within the model, it becomes problematic when roles encourage "overconfidence" in the model. For example, a model prompted to respond "*as a doctor*" might generate authoritative medical-sounding advice even when addressing the information for the medical conditions is beyond its reliable knowledge base. Similarly, it falls short in the principle of specificity by inconsistently requiring precise, measurable details and in objectivity by rarely establishing clear boundaries. The limitations of knowledge boundaries suggest that the model might present speculative information with the same confidence as compared to information that is actually based on facts.

A common blind spot across all three techniques is the principle of objectivity. None of these popular approaches consistently address knowledge limitations, uncertainty, or the boundaries of reliable information. This represents a significant vulnerability in current prompt engineering practice that could potentially lead to overconfident responses (hallucination) in areas where a model's knowledge base is limited or outdated. These prompting techniques leave little room for explicitly acknowledging the epistemological limitations in the results generated by an LLM.

Analysis of the Techniques Against LLM Operating Mechanisms

None of the prompt engineering techniques fully aligns with all four fundamental operating mechanisms of Large Language Models. While they align with individual mechanisms, each technique demonstrates specific strengths and weaknesses in leveraging the underlying architecture.

Few-Shot Learning prompting strongly aligns with Pattern Recognition by providing explicit examples that activate the model's pattern-matching capabilities. For example, instead of simply asking "*classify this text as a positive or a negative sentiment*," the prompt will include several example pairs like "*'This product exceeded my expectations.' - Positive*" and "*'The service was terrible.' - Negative*". This approach effectively leverages the model's ability to recognize patterns at multiple levels and apply them to new inputs. However, this technique shows limited engagement with Context Processing, as it typically fails to establish comprehensive contextual boundaries. The examples provide narrow context without defining the full information landscape for the model. A more effective prompt would explicitly establish these boundaries: "*You are analyzing customer feedback for an electronics retailer. Based on these examples, classify the following reviews as positive or negative, specifically in terms of customer satisfaction with product quality and service experience. If a review contains topics outside these domains, note that they fall outside the classification scope.*" Furthermore, Few-Shot prompting demonstrates minimal alignment with Training Data Awareness, providing no mechanism for the model to indicate knowledge limitations. Instead of relying solely on examples, a better approach would include: "*If you encounter terminology or concepts in these reviews that you're uncertain about, especially regarding recent electronics products released after your training cutoff, please indicate this uncertainty rather than guessing.*" Therefore, Few-Shot Learning aligns strongly with Pattern Recognition but falls short in Context Processing, Attention Management, and Training Data Awareness.

Chain-of-Thought prompting shows considerable strength in Attention Management by explicitly directing the model's focus through sequential reasoning steps. For example, a typical Chain-of-Thought prompt might state: "*Solve this probability problem step-by-step: In a bag of 10 marbles where 4 are blue and 6 are red, what is the probability of drawing 2 blue marbles in succession without replacement?*" This step-by-step guidance controls what parts of the problem the model attends to and in what sequence. However, Chain-of-Thought demonstrates significant weakness in Training Data Awareness, as it rarely instructs models to acknowledge knowledge limitations during reasoning. A more effective prompt would incorporate this awareness: "*Solve this probability problem step-by-step, clearly indicating if any step involves mathematical concepts or techniques, you're uncertain about: In a bag of 10 marbles where 4 are blue and 6 are red, what is the probability of drawing 2 blue marbles in succession without replacement? For each step, rate your confidence level (high/medium/low) based on your training data".* The technique also demonstrates inadequate Context Processing, focusing on procedural structure without establishing broader context. A better approach would begin with: "*You're helping a high school student understand probability concepts. Before solving this problem, establish what background knowledge is relevant for understanding conditional probability and how this specific problem fits within that mathematical framework.*" Therefore, Chain-of-Thought aligns strongly with Attention Management, moderately with Pattern Recognition, but falls short in Training Data Awareness and Context Processing.

Role-Based Prompting aligns strongly with Context Processing by creating clear operational boundaries by adopting a persona. A typical prompt might state: "*You are an experienced cardiologist. Explain the difference between systolic and diastolic blood pressure.*". This effectively activates domain-specific knowledge and establishes contextual constraints. However, this approach demonstrates critical weakness in Training Data Awareness, often encouraging authoritative responses even for topics beyond the model's knowledge base. A more effective prompt would explicitly address this limitation: "*Act as a cardiologist explaining blood pressure to a patient but clearly indicate where medical knowledge has evolved significantly since your training data cutoff. If asked about treatments or diagnostic methods that might have changed recently, acknowledge these limitations and suggest consulting current medical professionals for the most up-to-date information.*". Its handling of Attention Management is also inconsistent, rarely directing focus to specific aspects within the activated domain. An improved approach would specify: "*As a cardiologist, focus specifically on explaining blood pressure monitoring at home, the meaning of different readings, and when patients should seek medical attention. Prioritize practical guidance over technical explanations of cardiac physiology.*" Therefore, Role-Based Prompting aligns strongly with Context Processing, moderately with Pattern Recognition, but falls short in Training Data Awareness and Attention Management.

A recurring limitation across all three techniques is inadequate engagement with Training Data Awareness. None of these approaches consistently instruct models to acknowledge knowledge boundaries or express appropriate uncertainty levels. Few-Shot Learning assumes the model can generalize correctly from limited examples without considering training data limitations. Chain-of-Thought encourages continuous reasoning without mechanisms to flag knowledge gaps. Role-Based Prompting often implicitly encourages maintaining authoritative responses even when addressing topics with limited training data representation. This represents a fundamental architectural misalignment that significantly contributes to hallucination risk and reduced reliability in real-world applications, particularly in specialized domains or for time-sensitive information.

The Jujitsu Framework

The Jujitsu framework represents a novel meta-framework for prompt engineering that aligns both with the principles of objective communication and the key operational mechanisms of an LLM. It aims to establish a more systematic and theoretically grounded approach to prompt design-unlike existing prompting approaches, which have primarily been developed through observation, trial and error. The name "Jujitsu" is deliberately chosen for its metaphorical significance. In martial arts, jujitsu emphasizes leveraging an opponent's energy and strength. Similarly, this framework operates by leveraging an LLM's inherent behavioral mechanisms. This philosophy of "working with" rather than "against" a model's architecture represents a fundamental shift in prompting – moving from attempts to force models into specific behaviors, to strategically utilizing the strengths of underlying mechanisms.

The framework also harnesses and integrates the strengths of existing prompting techniques while systematically mitigating their limitations. Its objective is to provide a clear and consistent prompting approach that is aligned with the architectural principles of a transformer-based LLMs, ensuring optimal performance. By grounding prompt engineering directly in the fundamental operating mechanism of a LLM coupled with the theoretical underpinning of objective communication, the framework helps to achieve more coherence, improved reliability, and enhanced accuracy across responses.

Specifically, the Jujitsu framework is aligned with the design principles of transformer-based architecture of an LLM. These are, attention mechanisms, context management, and pattern recognition, and the training data. Structuring prompts to work in harmony with these operating principles optimizes efficiency and precision of language generation through LLMs while simultaneously reducing inconsistencies and enhancing factual integrity in outputs. Additionally, this framework brings multiple prompting techniques within the fold of a structured and principled methodology. Thereby, enabling prompt-engineering to become more robust and scalable for real-world applications by improving flexibility and effectiveness. The principles of the Jujitsu framework are illustrated in Figure 1 and elaborated in the following.

The Four Core Principles of the Jujitsu Framework

Contextual Precision: Defining Clear Information Boundaries

Contextual precision creates clear response boundaries for LLMs, significantly improving their outputs in three key ways. First, it enhances relevance by keeping the model focused on a specific topic rather than generating information that may be tangential. Second, it improves coherence by helping the model maintain a logical flow throughout its response. Third, it increases specificity by guiding the model to provide information applicable to the exact situation rather than generating generic advice. Also, LLMs have limited context windows, meaning they can only process a finite size of information at once. Due to this constraint, the quality of prompts directly determines the accuracy of the responses. Vague, open-ended prompts often lead to issues such as hallucinations (made-up information), topic-drift, or overly general responses. For instance, consider the following prompt, "*What should I do if I get sued?"* Such a prompt typically yields a generic answer. A more coherent prompt would ensure that the LLM output with a given context, "*You are an expert civil litigation lawyer. Based on Massachusetts law, what should a small business owner do if sued for breach of contract?* This refined prompt guides a model to a specific legal framework, improving the response's accuracy and applicability. This in turn reduces the risk of hallucination, and ensures a response that matches a specific situation rather than a generic output.

Guided Attention: Focusing on What Matters Most

Guided attention enhances the quality of output by directing an LLM to focus on the most relevant information. Transformer models (virtually all LLMS are based on a transformer) utilize self-attention mechanisms to assign weight to words based on their relationship to other words. However, without such specific guidance in a prompt, an LLM might struggle to prioritize the most relevant details for a given query. Consider a broad prompt like, "*Analyze the cryptocurrency market*". This prompt will typically produce a generic response, without any meaningful depth in specific areas of inquiry. In contrast, a targeted prompt would be, "*Analyze Bitcoin's price volatility over the past 12 months, focusing on regulatory changes and institutional adoption*". Such a prompt provides guidance to help a model focus (give attention) on specifics, rather than attempting to cover everything equally. This results in a response with deeper insights rather than general observations that lack practical utility. Therefore, by deliberately guiding a model's attention, it can allocate its computational resources more effectively to address the most important aspects of the question.

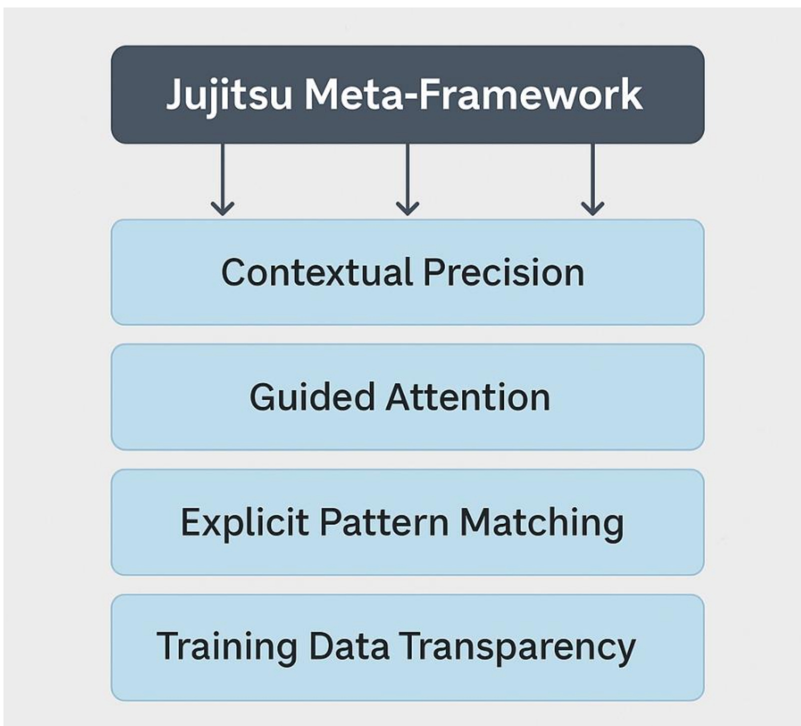
Explicit Pattern Matching:(Structuring for Consistency and Reliability

Pattern consistency is about providing a clear structural template for the LLM to follow. Fundamentally, LLMs predict the most likely next word or token based on patterns learned during training. Clear formatting examples or templates significantly increases the likelihood of consistent responses maintaining the desired structure. Consider the difference between these following two approaches. A vague prompt like, "*Write a customer service apology email*". Such as prompt gives a model no specific direction about format, tone, or structure. Therefore, a model will generate a different and inconsistent output each time such a prompt is inputted. In contrast, here is a template-based prompt, "*Write an apology email for a delayed shipment. Use this template: [Example]*". This provides a clear pattern to follow. A model now follows the given format and can maintain the same style, structure, and tone. Such a template approach is particularly valuable where consistency matters. For instance, legal documents need specific sections in a particular order, customer service communications require consistent brand voice, and technical documentation must follow standardized formats. Providing a clear pattern to emulate helps to leverage its innate operating mechanism in pattern recognition. This creates a platform for more reliable and consistent outputs.

Training Data Transparency: Recognizing the Limits of Model Knowledge

Applying this principle helps prevent misinformation by recognizing what the LLM knows and doesn't know. LLMs can only draw on information that they were trained on up to their knowledge cutoff date. They cannot access current events, real-time data, or information released after their training. When prompted, "*What is the current U.S. inflation rate?*", a model faces an impossible task, it doesn't know the true current rate. Consequently, it may generate a response that appears true but is probably dubious. Some LLMs now include web search capabilities to retrieve real-time data and supplement their training knowledge. However, the LLMs can only access publicly available web pages and the text that is on a particular webpage. They cannot open documents, PDFs, or follow links within those pages. Moreover, the model can only report what websites state. It cannot verify accuracy beyond the comparison of sources. Furthermore, web searches may or may not be automatic. It varies from model to model. These weaknesses create potential risks, especially when users assume that the model has searched for current information. When, in reality, it may not have. Therefore, the need for a transparent prompt becomes warranted. For example, the previous prompt should be reformulated as, *"As of your last training update (September 2023), what was the most recently reported U.S. inflation rate? If unavailable, note your limitations or use search capabilities if available.*". This guides a model to work within its boundaries of training data, while using external sources. In this approach, the prompt prevents the model from giving an output that may sound correct but is not true. Training data transparency is especially crucial in domains where accuracy has significant consequences, such as finance, healthcare, and legal contexts.

FIGURE 1
THE JUJITSU FRAMEWORK



The Jujitsu framework is designed to address key gaps identified in popular prompt engineering techniques. By aligning with both the internal architecture of LLMs and the principles of objective communication, it offers a more comprehensive and principled solution to these persistent shortcomings.

Gap 1: Limited Training Data Awareness

Popular prompting methods often fail to take cognizance of the data boundaries of an LLM's training data. This invariably increases the risk of hallucinations, especially in application domains where accuracy is essential. The Jujitsu framework addresses this issue through the application of the principle of Training Data Transparency. This gap is narrowed by encouraging prompts to address the temporal and epistemic boundaries of the data. For example, instead of asking, "*What are the long-term effects of [new cancer* *treatment]?",* a prompt formulated on the principles of the Jujitsu framework prompt would specify: "*Based on data available before your training cutoff date, what were the known or anticipated effects?*" This approach greatly reduces the likelihood of fabricated information. It also forces an LLM to distinguish between known and unknown facts.

Gap 2: Contextual-Attention Imbalance

Traditional prompting techniques typically emphasize either context or attention, but rarely both simultaneously. For instance, Few-Shot Learning provides examples to establish context but fails to direct a model's attention to focus on the key aspects. Conversely, Chain-of-Thought prompting guides the model's reasoning, i.e. attention, but often does not require the element of context. The Jujitsu framework integrates both Contextual Precision as well as the Guided Attention as complementary requirements. Together, these principles ensure that a model operates within a defined information boundary, while also focusing on the most relevant elements of requirement. For instance, when asking a LLM to analyze a company's financial health, the Jujitsu-informed prompt might specify key areas to focus on such as, revenue trends, cost structure, and debt ratios. Additionally, it might define the output's scope by restricting it to a specific defined time period and industry setting. This approach results in a more structured and insightful output.

Gap 3: Insufficient Theoretical Grounding

Prompting practices have largely evolved largely through empirical means, relying on trial-and-error without a solid theoretical foundation. The Jujitsu framework addresses this gap by grounding its four core principles in the transformer-based LLM mechanics and the principles of objective communication. Each prompting principle within the framework is directly aligned to the corresponding architectural behavior of an LLM. Consider a scenario in which a data analyst is asking the LLM to give an analysis for customer segmentation. In such a case, a Jujitsu framework informed prompt would necessarily structure the input by using a predefined output format (e.g., executive summary, segment profiles, methodology, and recommendations); it would also guide the model to pay attention to its patterns, while also maintaining clarity, limiting the response to a specific domain, and within the boundaries of factual information. Such structured prompting is systematic, not arbitrary. It reflects both how a model functions, and how objective human communication works. This method significantly increases the likelihood of generating more effective responses as compared to traditional methods.

Gap 4: Inconsistent Objectivity Controls

Traditional techniques, especially Role-Based Prompting, often encourage models to respond with undue authority, even when their domain knowledge is limited. This increases the risk of hallucinations being presented as facts. The Jujitsu framework addresses this challenge through the interplay of the principles of Training Data Transparency and Contextual Precision. By applying these principles, the framework ensures that models clarify when any information is speculative, contested, or beyond the models' knowledge cutoff. For instance, instead of asking, "*Explain the causes of [a controversial event]*", a Jujitsu prompt might read, "*Based on scholarly interpretations published before your knowledge cutoff, explain competing perspectives on the causes of [event] and note where evidence remains limited*." This enables a model to represent viewpoints correctly and also enhances objectivity of output. This can be quite critical in the case of complex, ambiguous, or high-stakes domains.

The Jujitsu framework presents a theoretically grounded, architecture-aligned, and communication-aware strategy for prompt engineering. This approach addresses the weaknesses of traditional prompting engineering techniques. By deliberately guiding users, the framework helps reduce hallucination rates, improve reasoning and greater clarity in LLM outputs. Moreover, it enables prompt engineering to scale effectively across domains and models. The framework also lays the groundwork for improved practice and future innovation in human-AI communication, representing a leap forward in the spirit of continuous improvement.

The comparative analysis highlights the Jujitsu Framework's advantages over traditional prompting engineering techniques. While traditional approaches demonstrate high flexibility, the trade-off is usually inconsistent clarity, weak theoretical grounding, and limited model transferability. In contrast, the Jujitsu framework offers a unified strategy that aligns fully with language model architecture and is informed by the very essence of objective communication. Most importantly, the Jujitsu approach provides a systematic, theoretically derived method that simplifies learning and application across different language model families. This overcomes model-specific limitations that typically impacts traditional prompting techniques.

Our comparative analysis reveals areas where the Jujitsu framework demonstrates significant advantages over traditional prompting techniques. First, by unifying disparate techniques into a single coherent framework, Jujitsu reduces the cognitive burden on users who would otherwise need to select and apply different techniques for different tasks. Second, by emphasizing Training Data Transparency and Contextual Precision, the framework substantially improves accuracy and reliability, addressing the objectivity limitations identified in traditional approaches. Third, its balanced engagement with both context and attention enables consistent performance across diverse tasks, providing versatility without sacrificing quality. Finally, the framework achieves transferability across different model architectures by aligning directly with fundamental LLM operating mechanisms rather than model-specific features. Table 1 summarizes these differentiators, contrasting the Jujitsu Meta-Framework with traditional prompting techniques across these dimensions.

TABLE 1
CONTRASTING TRADITIONAL VS JUJITSU TECHNIQUES

Dimensions	**Traditional Prompting Techniques**	**Jujitsu Meta-Framework**
Usability & Cognitive Load	Trial-and-error approach requiring different techniques for different tasks; steep learning curve	Unified framework applicable across all scenarios; consistent principles reduce mental overhead
Accuracy & Reliability	High hallucination risk; rarely acknowledges model knowledge limitations	Enhanced factual integrity through Training Data Transparency; explicitly addresses knowledge boundaries
Task Versatility	High flexibility but inconsistent results; techniques optimized for specific tasks	Balanced approach offering adaptability while maintaining consistent performance across diverse applications
Model Transferability	Model-specific techniques requiring relearning when switching between LLMs	Architecture-aligned principles that remain effective across different LLM families and versions
Architecture Alignment	Partial alignment with specific aspects of LLM functioning (e.g., reasoning or pattern recognition)	Comprehensive alignment with all core LLM operating mechanisms (Context, Attention, Pattern Recognition, and Training Data)

DISCUSSION

The framework demonstrates a systematic approach for achieving more effective human-AI interaction. This helps to improve the communication process, such as reduced cognitive load, improved accuracy, task versatility, model transferability enabled by fundamental architectural alignment. The overall impact of the Jujutsu framework is achieved through enhanced prompt engineering practices. Some of the explicit advantages are:

Reduced Cognitive Load for Users

Prompting through the Jujitsu framework presents several advantages. It reduces the cognitive load for a user by consolidating disparate prompting techniques into a singular prompt. Rather than juggle different prompts for different scenarios, users can apply a consistent framework across various applications and requirements. This simplifies their interactions with the LLM and improves the overall efficiency of responses. For instance, imagine trying to remember a variety of prompting techniques for specific scenarios. Few-Shot Learning for classification tasks (e.g. analyzing whether a contract clause is favorable or unfavorable, determining if a medical symptom description indicates urgency, or classifying research papers by methodology). Chain-of-Thought for reasoning problems (e.g. debugging complex code issues, evaluating business investment scenarios, or analyzing ethical dilemmas). Role-Based Prompting for domain-specific queries. Such an approach will overwhelm a user. On the other hand, a user can simply apply the four principles of the Jujitsu framework across all scenarios. To illustrate, a user asks LLM to analyze a business expansion proposal. In this case a -prompt that is formulated through the lens of the Jujitsu framework would ensure that Contextual Precision is taken care of by specifying the industry and market conditions; Guided Attention by directing focus to key financial metrics and risk factors; Pattern Consistency by requesting a standardized SWOT analysis format; and finally Training Data Transparency by through time-sensitive market data. This unified approach works equally well whether the task involves classification, reasoning, or domain expertise. This consistency creates a smoother learning curve and reduces the mental overhead of switching between different prompting strategies.

Improved Accuracy and Reliability

The Jujitsu framework ensures more reliable and consistent responses from an LLM. By guiding a user to formulate prompts through these principles, the outputs from an LLM tend to become more certain and trustworthy. For example, crafting a prompt for medical information through the lens of the Jujitsu framework would ensure that the model stays within the confines of its knowledge boundaries. This automatically helps in reducing the hallucination rate. This is important in domain such as healthcare, where the LLM is being deployed for dispensing medical advice. Consider the difference between an ill-thought prompt, "*What treatments are available for this rare condition?*" and a Jujitsu-informed prompt, *"Based on peer-reviewed medical literature available before your training cutoff date, what treatment approaches were being investigated for this rare condition? Please clearly distinguish between established treatments and experimental approaches, noting any limitations in the available evidence."* The expected output differential between the two are quite evident. The latter prompt significantly reduces the risk of the model generating fictional treatments or overstating evidence. Therefore, making it particularly valuable in high-stakes domains where accuracy is paramount.

Versatility Across Tasks and Applications

The unified structure of the Jujitsu framework offers exceptional flexibility. Users can mix strengths from existing techniques as needed for their specific tasks. They can borrow from Few-Shot Learning, Chain-of-Thought, and Role-Based Prompting while fixing their weaknesses. Someone might use pattern matching for creative writing. For factual questions, they might focus more on training data awareness. All of this happens within one consistent approach.

For more creative tasks, applying pattern matching helps users maintain a consistent style, such as narration in storytelling. For instance, consider the following prompt, "*Write me a magical realism story. I want everyday stuff that slowly gets weird, but nobody in the story should act surprised about it. Maybe start in a coffee shop or somewhere normal and then add in those strange elements bit by bit.".* Here, by giving the model a clear structure to follow, basically showing it, the pattern of magical realism rather than just naming the genre, i.e., "write a story in the magical realism genre". In the first case, the prompt doesn't just name the genre but describes the specific patterns that make up magical realism (normal settings, gradual introduction of supernatural elements, characters treating the supernatural as normal).This way, the user guides the model on what style elements to include and maintain, as opposite to the other, where the user assumes a model already knows exactly what defines magical realism and how to execute it properly.

Similarly, in the case of data analysis, the framework guides a user to apply Training Data Transparency and Contextual Precision principles. This helps a model understand exactly what data to focus on and stay within the limitations of its knowledge base. Secondly, the analysis by the model is guided through an output template, rather than assuming the model knows how to structure analytical output. For example, a prompt might be structured as: "*Can you look at this climate data from the last decade? I'm especially interested in what's happening along the coastlines. Break it down by method, patterns, statistics - and mention where the data seems thin or unreliable.*". This adaptability makes the framework incredibly versatile across professional settings. It performs equally well in creative fields, marketing, healthcare, finance, technical fields, and even education. Users don't need to learn entirely new approaches when switching between different areas.

Transferability Across Different LLM Architectures & Architectural Alignment

The Jujitsu framework aligns with the fundamental architectural principles of AI rather than features of any specific model. This transferability stems directly from the framework's fundamental alignment with LLM architecture rather than model-specific implementations. While traditional techniques often optimize for single mechanisms in particular models (like Chain-of-Thought's emphasis on attention paths in specific architectures), the Jujitsu framework balances engagement across all four core mechanisms common to transformer-based models. This architectural harmony allows models to operate closer to their optimal design parameters regardless of their specific implementation details. By working with, rather than against, these fundamental architectural principles, the framework ensures consistent performance improvements across model families and versions, providing organizations with a stable approach that remains effective as AI technology evolves.

This also demonstrates its versatility across various LLMs. The framework serves as an enabler, providing users with a consistent prompting system that endures as models evolve. The principles of Contextual Precision, Guided Attention, Pattern Matching, and Training Data Transparency remain persistent, whether interacting with standard transformers, mixture-of-experts architectures, or future advancements. For users, the prompting skills developed using this framework will not become obsolete when a model is updated or transitioning between models. The prompting skills learnt will transfer seamlessly, preserving a user's learning investment over time. This transferability represents a significant advantage for organizations deploying multiple AI systems or transitioning between models, eliminating the need to retrain staff on new prompting techniques with each technological iteration, update, or change.

Enhanced Understanding of LLM Functionality

Beyond, just developing prompting skills, which are at the surface level of LLM. Working through the dictates of the framework will also develop a users' understanding of how LLMs function. As users apply the framework, they will learn about the intricate relationship between prompts and model architecture. This will help them build a more informed prompt engineering capability, leading to greater ability to leverage these powerful AI tools.

This deeper understanding transforms LLMs from mysterious black boxes with unpredictable outputs into systems that users can work with more intentionally. Users can better anticipate how an LLM will respond by grasping the conceptual model of how these systems process information. Additionally, this will also help to reduce frustration and increase productivity from AI. For instance, users will intuitively learn to recognize when a model is struggling with attention (focusing on the wrong aspects of a prompt), or context management (missing necessary background information) , or knowledge boundaries (attempting to generate information beyond its training). This diagnostic capability will enable users to systematically improve their prompts, rather through trial and error.

The benefits extend beyond individual users to the organizational level. The framework provides a common language and approach for teams working with LLMs. Organizations can more effectively establish best practices, conduct training, and ensure consistency across AI interactions.

CONCLUSION

The Jujitsu framework represents a refined and elegant approach to prompt engineering. It encapsulates theoretical rigor with practical effectiveness. It is built on the principles of Contextual Precision, Guided Attention, Explicit Pattern Matching, and Training Data Transparency. These principles are synergistically aligned with the underlying architecture of LLMs, while simultaneously being informed by the principles of objective communication. This dual grounding enables the framework to go beyond model-specific strategies, providing a durable foundation to prompt engineering approaches.

The Jujitsu framework addresses the shortcomings of existing prompting techniques. It enhances context where Few-Shot Learning falls short, brings focus, where Chain-of-Thought meanders, and ensures factual grounding where Role-Based Prompting risks inconsistency. In essence, the Jujitsu framework preserves adaptability, while reinforcing accuracy and awareness of knowledge boundaries. Its strength lies leveraging the operating mechanism of LLMs. As prompt engineering continues evolving, the Jujitsu framework stands out as a resilient, architecture-informed model that supports effective, transparent, and contextually appropriate human-AI interaction. This approach to prompting promotes more reliable outputs across diverse applications and domains.

Further research could further explore and validate the Jujitsu framework's effectiveness across diverse applications. Empirical validation across multiple domains would test the framework's generalizability and identify domain-specific adaptations that might enhance its effectiveness.

REFERENCES

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*. https://doi.org/10.48550/arXiv.1409.0473

Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., . . . Amodei, D. (2020). Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, *33*, 1877–1901. https://doi.org/10.48550/arXiv.2005.14165

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Goldstein, N. (Ed.). (2022). *The Associated Press stylebook: 2022–2024* (56th ed.). The Associated Press.

Iverson, C., Christiansen, S., Flanagin, A., Fontanarosa, P.B., Glass, R.M., Gregoline, B., & Young, R.K. (Eds.). (2020). *AMA manual of style: A guide for authors and editors* (11th ed.). Oxford University Press.

Hayakawa, S.I., & Hayakawa, A.R. (1990). *Language in thought and action* (5th ed.). Harcourt Brace Jovanovich.

Hugging Face. (n.d.). *NLP course – Summarization chapter*. Retrieved from https://huggingface.co/course

IBM Cloud Education. (2023). *What is prompt engineering?* Retrieved from https://www.ibm.com/cloud

Knoth, N., Smith, A., & Reyes, M. (2024). AI literacy and its implications for prompt engineering strategies. *Computers and Education: Artificial Intelligence*. https://doi.org/10.1016/j.caeai.2024.100161

Korzynski, P., Mazurek, G., Krzypkowska, P., & Kurasinski, A. (2023). Artificial intelligence prompt engineering as a new digital competence: Analysis of generative AI technologies such as ChatGPT. *Entrepreneurial Business and Economics Review*, *11*(3), 25–37. https://doi.org/10.15678/EBER.2023.110302

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, *55*(9), Article 195. https://doi.org/10.1145/3560815

Lo, L.S. (2023). The CLEAR path: A framework for enhancing information literacy through prompt engineering. *The Journal of Academic Librarianship*, *49*(6), 102707. https://doi.org/10.1016/j.acalib.2023.102707

McCann, B., Keskar, N.S., Xiong, C., & Socher, R. (2018). *The natural language decathlon: Multitask learning as question answering*. Retrieved from https://arxiv.org/abs/1806.08730

McKinsey & Company. (2024). *What is prompt engineering?* McKinsey & Company. Retrieved from https://www.mckinsey.com/featured-insights/mckinsey-explainers/what-is-prompt-engineering

Mehra, P. (2024, January 12). The rise and fall of prompt engineering: A short-lived revolution. *TechCircle*. Retrieved from https://techcircle.in/

Miguelañez, C. (2025, January 6). *Common LLM prompt engineering challenges and solutions*. Latitude Blog. Retrieved from https://latitude.so/blog/common-llm-prompt-engineering-challenges-and-solutions/

Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.H., & Riedel, S. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 2463–2473). https://doi.org/10.18653/v1/D19-1250

Sabbatella, A., Ponti, A., Giordani, I., Candelieri, A., & Archetti, F. (2024). Prompt optimization in large language models. *Mathematics*, *12*(6), 929. https://doi.org/10.3390/math12060929

Shin, T., Razeghi, Y., Logan, R.L., IV, Wallace, E., & Singh, S. (2020). AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 4222–4235). https://doi.org/10.18653/v1/2020.emnlp-main.346

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., . . . Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, *30*. https://doi.org/10.48550/arXiv.1706.03762

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, *35*, 2164–2181.