Adaptive and Self-Correcting Prompt Engineering for Robust and Human-Centric LLM Interactions

Academic Researcher

August 20, 2025

Abstract

Current Large Language Model (LLM) applications, despite their advancements, suffer from significant limitations including lack of robustness, frequent failures, and difficulty in handling nuanced human factors. Traditional prompt engineering often relies on static, pre-defined prompts, leading to generalization issues and hindering self-correction. This proposal outlines a novel research direction focusing on adaptive and self-correcting prompt engineering. We propose a framework for dynamically generating and modifying prompts based on real-time interaction, user feedback, and LLM's inferred internal state, enabling proactive error identification and recovery. Furthermore, we aim to develop techniques for eliciting and integrating "soft" human factors like emotions and values into LLM reasoning. Our contributions include a systematic methodology for dynamic prompt optimization, enhanced evaluation frameworks, and the development of more reliable and human-aware LLM systems, pushing the boundaries of human-AI collaboration.

1 Introduction

Large Language Models (LLMs) have revolutionized various fields, demonstrating remarkable capabilities in natural language understanding and generation. However, their deployment in critical applications is often hampered by inherent limitations, including a notable lack of robustness, susceptibility to hallucinations, and inconsistent reliability [**?** **?**]. A significant contributing factor to these challenges lies in the static and often brittle nature of current prompt engineering practices. While sophisticated agentic loops can improve task completion, as observed in autonomous agents [**?**], the underlying prompt structures remain fragile, leading to failures stemming from planning, execution, and response generation.

Furthermore, the evaluation methodologies for LLM outputs frequently overrely on basic success rates, neglecting a systematic analysis of internal interactions or root causes of failures [**?**]. This deficiency extends to prompt engineering, where understanding *why* a prompt succeeds or fails is crucial for

advancement. The sensitivity of LLMs to context and domain specificity also means that prompts often lack generalization, necessitating extensive customisation for new applications [**?**]. Moreover, current prompt engineering largely overlooks the critical role of "soft" human factors such as emotions, values, and communication nuances, which are paramount in complex human-centric processes like requirements elicitation [**?**].

This research proposal addresses these fundamental gaps by proposing a paradigm shift from static to adaptive and self-correcting prompt engineering. We aim to develop a comprehensive framework that enables LLMs to dynamically adjust their prompts, diagnose their own errors, and proactively recover, while simultaneously enhancing their capacity to understand and integrate human-centric elements into their reasoning and responses.

1.1 Related Work

Existing research in prompt engineering has explored various techniques to enhance LLM performance, including few-shot learning, chain-of-thought (CoT) prompting, and role-playing. Works like [**?**] demonstrate the potential of LLMs in autonomous agent systems, highlighting the need for improved reliability and systematic failure analysis. While they propose a taxonomy for agent failures, a similar, in-depth taxonomy for prompt engineering failures remains largely unexplored.

The challenge of LLM hallucinations and the necessity for grounding responses, particularly in sensitive domains, is well-documented [**?**]. This work introduces a "context-aware query translator" that adapts retrieval depth and response style, representing a crucial step towards dynamic prompting. However, the broader concept of dynamically generating or modifying prompts based on real-time interaction or LLM's internal state is still nascent.

The importance of human-centric factors in complex processes is underscored by research in areas like requirements elicitation [**?**], which emphasizes the critical role of emotions, values, and communication noise. Current prompt engineering largely lacks effective strategies to elicit, interpret, and incorporate these subjective human elements into LLM reasoning or output generation, limiting their utility in nuanced human-human or human-AI interactions.

While advancements have been made in specific prompt optimization techniques, a holistic approach that integrates dynamic adaptation, self-diagnosis, human-centric considerations, and automated optimization within a unified framework is missing. This proposal seeks to bridge these identified gaps, moving prompt engineering beyond empirical trial-and-error towards a more systematic, robust, and human-centric discipline.

1.2 Research Objectives and Contributions

This research aims to achieve the following objectives:

1. To develop a novel framework for **adaptive and dynamic prompt gen-**

eration that allows LLMs to modify prompts based on real-time interaction, task progress, and inferred internal state.

2. To design and implement **prompting strategies for proactive self-diagnosis and recovery**, enabling LLMs to identify their own errors, inconsistencies, and propose corrective actions during multi-step processes.
3. To explore and formalize **prompting techniques for eliciting and integrating human-centric and "soft" factors** (e.g., emotions, values, communication nuances) into LLM reasoning and output generation.
4. To propose and validate **systematic, multi-faceted evaluation methodologies** for prompt engineering, moving beyond simple success rates to include granular failure analysis and qualitative assessment of nuanced outputs.
5. To contribute towards **automated prompt optimization and generation** techniques that reduce reliance on manual trial-and-error, fostering more efficient and effective prompt design.

The primary contributions of this research will be:

- A conceptual and practical framework for dynamic prompt engineering.
- A taxonomy of prompt-induced failures, complementing existing agent failure taxonomies.
- Novel prompt patterns and strategies for self-correction and human-centric interaction.
- Standardized benchmarks and comprehensive evaluation metrics for adaptive prompting.
- Proof-of-concept implementations demonstrating enhanced robustness, reliability, and human-awareness in LLM applications.

2 Methodology

Our proposed methodology integrates several innovative approaches to address the identified gaps in prompt engineering. We envision a multi-layered system that moves beyond static prompt templates to a dynamic, reflective, and human-aware prompting paradigm.

2.1 Proposed Approach

2.1.1 Dynamic Prompt Generation Framework

We will develop a meta-prompting agent or a prompt orchestration layer that dynamically constructs and modifies prompts. This framework will operate based on:

- **Interaction History and User Feedback:** Analyzing previous turns in a conversation or explicit user feedback to refine subsequent prompts.
- **Task Progress and State:** Adapting prompts based on the current stage of a multi-step task (e.g., planning, execution, verification) and the success/failure of previous steps.
- **Inferred LLM Internal State:** While direct access to LLM internal states is limited, we will leverage techniques like Chain-of-Thought (CoT) or Tree-of-Thought (ToT) to elicit the LLM's reasoning process. This reasoning output can then be analyzed by a meta-prompting module to infer potential ambiguities, uncertainties, or logical inconsistencies, which in turn inform the next prompt.
- **Contextual Adaptation:** Building upon concepts like the "context-aware query translator" [**?**], our system will dynamically adjust prompt components (e.g., level of detail, tone, required reasoning steps) based on the domain, user expertise, and specific sub-task.

2.1.2 Prompting for Proactive Self-Diagnosis and Recovery

This core component focuses on enabling LLMs to identify and correct their own errors. We will design specific prompt patterns for:

- **Reflection Prompts:** After an LLM generates an output or completes a step, a subsequent prompt will ask it to critically evaluate its own work. For example, "Review your previous answer for logical consistency and factual accuracy. Identify any potential errors or ambiguities."
- **Critique Prompts:** These prompts will ask the LLM to act as a critic of its own output, identifying weaknesses or alternative approaches. "If you were to improve your previous plan, what would be the top three changes you would make and why?"
- **Error Categorization and Correction Prompts:** Based on a newly developed taxonomy of prompt-induced failures (see Technical Details), prompts will guide the LLM to categorize its own errors and then generate corrective actions or alternative strategies. "You made a planning error (Type X). Propose three alternative steps to overcome this."
- **Ambiguity Resolution Prompts:** When an LLM expresses uncertainty or asks for clarification, the system will dynamically generate prompts to help it resolve ambiguities or gather more information.

2.1.3 Prompting for Human-Centric and "Soft" Factors

Addressing the gap highlighted by [**?**], we will develop strategies to prompt LLMs to:

- **Elicit Emotional Cues:** Design prompts that encourage users to express emotions or that guide the LLM to infer sentiment from user input (e.g., "How does this situation make you feel?").
- **Incorporate Values and Beliefs:** Prompts that ask the LLM to consider ethical implications, user preferences, or cultural contexts in its responses. This could involve providing explicit value frameworks in the prompt or asking the LLM to infer them.
- **Manage Communication Noise:** Prompts designed to help the LLM identify misunderstandings, clarify ambiguous statements, or rephrase information for better comprehension.
- **Facilitate Human-Human Interaction:** Prompting LLMs to act as mediators, summarize differing viewpoints, identify common ground, or suggest conflict resolution strategies, considering the emotional and relational dynamics.

2.1.4 Automated Prompt Optimization and Generation

To reduce the manual effort in prompt engineering, we will explore automated techniques:

- **Reinforcement Learning (RL):** Using RL agents to search the prompt space, where rewards are based on desired output characteristics (e.g., accuracy, coherence, robustness to errors).
- **Evolutionary Algorithms:** Applying genetic algorithms to evolve prompt structures and content based on performance metrics.
- **Meta-Learning for Prompts:** Training a meta-model to learn how to generate effective prompts for new tasks or domains, leveraging insights from successful prompts in related areas.

2.2 Technical Details and Innovations

- **Prompt Orchestration Layer:** A dedicated software module that manages the dynamic generation, modification, and chaining of prompts. This layer will analyze LLM outputs, user inputs, and task state to decide the next optimal prompt.
- **Prompt-Induced Failure Taxonomy:** Building upon the agent failure taxonomy in [**?**], we will develop a detailed taxonomy specifically for prompt engineering failures. This will categorize errors (e.g., misinterpretation, logical fallacy, hallucination due to prompt ambiguity, insufficient conte