Prompt Engineering Case Study

Sarah Sair — Prompt Engineer | LLM Optimization | AI Reliability
GitHub • LinkedIn • Portfolio

Problem Statement

Large Language Models (LLMs) frequently generate hallucinations, inconsistent reasoning, and unstructured outputs. This case study demonstrates how structured prompt engineering can significantly improve reliability, accuracy, and production readiness.

Approach

• Designed structured prompts using Few-Shot, Chain-of-Thought, and explicit constraints
• Tuned temperature, Top-P, and token limits for deterministic behavior
• Benchmarked prompts across GPT-4, Gemini, and Claude
• Implemented guardrails, fallback logic, and safety constraints

Evaluation & Benchmarking

Prompts were evaluated using accuracy, reasoning consistency, hallucination rate, and output structure compliance across multiple LLM providers.

Metric	Before	After
Accuracy	Inconsistent	+30–40% Improvement
Hallucinations	Frequent	Significantly Reduced
Output Consistency	Variable	Stable & Predictable

Key Takeaways

• Prompt engineering is an engineering discipline, not trial-and-error
• Small prompt changes can produce large performance gains
• Guardrails are critical for ethical and scalable AI systems
• Evaluation loops dramatically improve LLM reliability

Why This Matters

This project demonstrates my ability to translate human intent into machine-executable instructions, optimize LLM behavior using data-driven methods, and build safe, production-grade prompt systems.