AI Systems Assignment Report

Task 1: Rating Prediction via Prompting
Task 2: Two-Dashboard AI Feedback System

1. Introduction

This report presents the implementation and evaluation of two complementary AI systems developed as part of the assignment:

- **Task 1** focuses on evaluating different prompting strategies for predicting Yelp review star ratings using Large Language Models (LLMs).
- **Task 2** extends the use of LLMs into a production-style web application that collects user feedback and generates AI-assisted insights through two distinct dashboards.

Together, these tasks demonstrate an understanding of prompt engineering, structured LLM outputs, system reliability, data persistence, and real-world deployment considerations.

2. Overall Approach

The overall approach was to explore the use of Large Language Models (LLMs) at two levels:
first, as a **standalone reasoning system** evaluated through prompt engineering (Task 1), and second, as a **core component of a production-style web application** (Task 2).

Task 1 focused on understanding how different prompt structures affect prediction accuracy and output reliability. **Task 2** extended these learnings into a real-world system that integrates LLMs with backend services, persistent storage, and web-based dashboards.

3. Design and Architecture Decisions

For Task 1, the system was designed as an offline evaluation pipeline using Python notebooks, where multiple prompt strategies could be tested on the same dataset in a controlled manner.

For Task 2, a **single FastAPI backend** was chosen to serve both the user-facing and admin-facing dashboards. This ensured:

- A shared persistent data source
- Server-side LLM usage
- Clear separation of concerns via API endpoints

Plain HTML, JavaScript, and CSS were used for the frontend to comply with the constraint of avoiding notebook-based or framework-heavy solutions.

4. Prompt Iterations and Improvements (Task 1)

Three prompting strategies were iteratively designed:

1. A baseline prompt with minimal instructions
2. A rubric-based prompt with explicit rating definitions
3. A reason-then-classify prompt encouraging structured reasoning

Each iteration aimed to improve either prediction accuracy or consistency. The final approach demonstrated that guiding the model to reason before classifying significantly improves performance on subjective tasks such as review rating prediction.

5. Evaluation Methodology and Results (Task 1)

All prompt variants were evaluated on the same sampled subset of Yelp reviews. Performance was measured using:

- Exact-match accuracy (actual vs predicted stars)
- JSON validity rate
- Reliability across multiple calls

The reason-then-classify prompt achieved the highest accuracy while maintaining perfect JSON validity, highlighting the importance of controlled reasoning in LLM-based classification tasks.

6. System Behaviour, Trade-offs, and Limitations (Task 2)

The deployed feedback system demonstrates reliable end-to-end functionality, including user submission, AI processing, and admin review. However, several trade-offs were acknowledged:

- Free-tier LLM usage introduces rate limits and occasional failures
- AI outputs are probabilistic and may vary across runs
- UI design was kept intentionally minimal to prioritize correctness and system reliability

To address these limitations, fallback logic was implemented so that the system continues to function even when LLM calls fail.

Conclusion

Together, Task 1 and Task 2 demonstrate both analytical and practical aspects of working with LLMs from prompt evaluation and structured output enforcement to deploying a robust, production-style AI-powered web system.