

# LLM-Based Rating Prediction System

Fynd AI Intern Assessment - Task 1 | Harshal Ahire | 16<sup>th</sup> December 2025

## Overview

Developed and evaluated three distinct prompting strategies for automated Yelp review rating classification (1-5 stars) using LLM technology. Implemented structured JSON output validation and conducted comparative analysis across 200 stratified samples to identify optimal approach for production deployment.

**Dataset** - Yelp Reviews dataset from Kaggle

**LLM Configuration** - Google Gemini 2.0 Flash with Pydantic schema validation enforcing strict JSON output format: `predicted_stars` (1-5) and `explanation` (reasoning).

**Evaluation Metrics** - Accuracy, Precision, Recall, F1-Score (Macro), JSON validity rate, and processing time per strategy.

## Prompting Strategies

**Prompt 1: Zero-Shot** - Direct classification with minimal instruction. System prompt defines task clearly: "Analyze the review text and assign an overall star rating from 1 to 5." Relies on model's inherent understanding without examples.

**Prompt 2: Few-Shot** - Example-based learning with 3 representative samples (5-star, 3-star, 1-star) demonstrating rating criteria and explanation format. Guides model through concrete reference points.

**Prompt 3: Chain-of-Verification (CoV)** - Multi-step reasoning approach: (1) Analyze sentiment, (2) Assign provisional rating, (3) Self-verify consistency, (4) Output final verified rating. Implements internal error-correction mechanism.

## Technical Implementation

- **Stack:** Python, Google Gemini API, Pydantic, scikit-learn, pandas
- **Architecture:** Modular design with `classify_review()` function handling API calls, error management, and schema validation
- **Output Validation:** Pydantic BaseModel enforcing `Literal[1, 2, 3, 4, 5]` type constraint for `predicted_stars` field
- **Error Handling:** Comprehensive try-catch blocks with detailed error logging for API failures and quota limits

## Results

The 'Zero-Shot approach' emerged as the optimal strategy

# AI-Powered Feedback System

Fynd AI Intern Assessment | Harshal Ahire | December 2025

## Overview

Developed a full-stack AI feedback management system with dual interfaces - a user-facing feedback form and an administrative analytics dashboard. The system generates intelligent, context-aware responses to user submissions in real-time using LLM technology.

## System Architecture

**User Dashboard** - Streamlit-based feedback form collecting star ratings and text reviews. Integrates Groq API (Claude Sonnet) to generate personalized AI responses based on sentiment analysis.

**Admin Dashboard** - Analytics interface displaying KPIs (average rating, submission count, positive reviews), filterable data tables, and AI-generated internal summaries.

**Backend** - Modular Python architecture with `data_handler.py` managing Google Sheets integration via OAuth2 service account authentication, and `llm_service.py` orchestrating structured AI response generation.

**Data Layer** - Google Sheets API storing submissions with schema: timestamp, user\_rating, user\_review, ai\_user\_response, ai\_summary, ai\_actions.

## Key Features

- Real-time AI response generation with sentiment-aware tone adaptation
- Persistent cloud storage with instant data synchronization
- Interactive analytics with dynamic filtering and cache management
- Comprehensive error handling and logging system
- Responsive design optimized for desktop and mobile

## Technical Implementation

- **Stack:** Python, Streamlit, Groq API, Google Sheets API, gspread
- **Deployment:** Render.com with automated CI/CD from GitHub
- **Architecture:** Modular design with separation of concerns (UI, business logic, data layer)
- **Security:** Environment-based secrets management for API keys and service credentials

## Challenges Solved

1. Configured service account authentication for cloud deployment with single-line JSON environment variables
2. Implemented cache invalidation strategy for real-time data refresh in admin dashboard
3. Built robust error handling with detailed logging for production debugging

## Deployed

Two production-ready web applications deployed on Render.com capable of handling requests with sub-2-second AI response times and persistent data storage.