

Report: Tuning a Sentiment Analysis Prompt

1. Introduction

This report documents an experiment in prompt tuning. Starting with a basic prompt for sentiment analysis, two methods were applied to improve its output quality and structure. The goal is to demonstrate how adding specific elements to a prompt can lead to more accurate and usable results from a Large Language Model.

Task: Perform sentiment analysis on the text: The service was incredibly fast, but the food was bland and uninspired.

2. The Basic Prompt (Baseline)

Prompt V1:

- What is the sentiment of this review: "The service was incredibly fast, but the food was bland and uninspired."
- The sentiment of the review is mixed, with both positive and negative elements.
- **Analysis:** The output is technically correct but lacks detail and is not in a structured format, making it difficult to use in an automated system.

3. Prompt Tuning Method 1: Adding a Persona

The first tuning method involves giving the LLM a specific role, or **Persona**, to guide its behavior.

Prompt V2:

- You are a helpful customer review analyst. What is the sentiment of this review: "The service was incredibly fast, but the food was bland and uninspired."
- As a customer review analyst, I would classify this review as having a mixed sentiment. The customer was pleased with the speed of service, which is a positive point. However, they were disappointed with the food, describing it as "bland and uninspired," which is a significant negative point.
- **Analysis:** This is a clear improvement. The persona encouraged the model to provide a more detailed and explanatory response, breaking down the positive and negative aspects. However, the output is still unstructured free text.

4. Prompt Tuning Method 2: Specifying the Output Format

The second method builds on the first by adding a strict output **Format** instruction. This is crucial for producing predictable, machine-readable outputs.

Prompt V3:

- You are a helpful customer review analyst. Analyze the following review and provide your output in a JSON format with three keys: "overall_sentiment", "positive_aspects", and "negative_aspects".
- Review: "The service was incredibly fast, but the food was bland and uninspired."
- {
 "overall_sentiment": "Mixed",

```
"positive_aspects": "The service was incredibly fast.",  
"negative_aspects": "The food was bland and uninspired."  
}
```

- **Analysis:** This output is optimal. It is accurate, detailed, and perfectly structured. The model correctly identified the overall sentiment and extracted the exact phrases corresponding to the positive and negative aspects. This JSON output can be directly ingested by other software, making it highly practical.

5. Conclusion

This experiment demonstrates the power of iterative prompt tuning. While the basic prompt gave a minimally useful answer, adding a **Persona** improved the quality and detail of the analysis. The most significant improvement came from specifying the output **Format**, which forced the model to provide a structured, predictable, and machine-readable response. This highlights a key principle of prompt engineering: specificity in instructions, particularly regarding output format, is essential for building robust and reliable LLM applications.