Course: FTEC5660 Agentic AI for Business and FinTech
Assignment: Homework 01
Name: Xie Jiangshan
Student ID: 1155173755

Multimodal Receipt Analysis using Large Language Models

1. Introduction

This report documents the development of an Agentic AI solution designed to automate the extraction and aggregation of financial data from multiple supermarket receipt images. The objective was to construct a pipeline capable of processing visual data to answer specific user queries regarding total expenditure and original prices (pre-discount), while simultaneously filtering out irrelevant inquiries. The solution leverages the multimodal capabilities of Google's Gemini model via the LangChain framework to perform optical character recognition (OCR), semantic understanding, and arithmetic reasoning in a single inference pass.

2. Methodology

The implementation is divided into three core components: Data Preprocessing, Prompt Engineering with Chain-of-Thought reasoning, and Robust Output Parsing.

Data Preprocessing and Input Formatting

The system processes a batch of seven JPEG receipt images. To interface with the Gemini API, each image is dynamically loaded from the local directory and converted into a Base64 encoded string. These strings are formatted as Data URLs (MIME type: image/jpeg), allowing them to be embedded directly into the prompt payload. Unlike traditional OCR pipelines that first convert images to text and then process the text, this solution feeds the raw visual data directly to the Multimodal LLM, allowing the model to utilize spatial layout cues (e.g., aligning "Total" labels with their corresponding numeric values) for higher accuracy.

Prompt Engineering and Chain-of-Thought

The core logic resides in the get_accountant_response function. A significant challenge in Multimodal LLMs is ensuring arithmetic accuracy across multiple data sources. Initial attempts using simple zero-shot prompting (asking for the final number directly) often resulted in calculation errors or timeouts due to the cognitive load of processing seven images simultaneously.

To mitigate this, a **Chain-of-Thought (CoT)** prompting strategy was implemented. The system prompt instructs the model to act as an "expert accountant" and enforces a step-by-step reasoning process:

1. **Individual Analysis:** The model is required to explicitly identify and list the relevant value (either "Final Total" or "Subtotal/Original Price") for *each* of the seven receipts individually.
2. **Aggregation:** Only after listing the individual values is the model permitted to sum them to calculate the grand total.
3. **Strict Output Formatting:** The model is instructed to append the final result in a specific format: "Final Answer: [NUMBER]".

This approach forces the model to generate intermediate tokens, which acts as a "scratchpad" for reasoning, significantly improving the accuracy of complex queries such as calculating the "Total without discount." Additionally, semantic guardrails were implemented to instruct the model to return "IRRELEVANT" if the user query falls outside the domain of receipt analysis (e.g., weather inquiries).

Model Configuration and Output Parsing

The solution utilizes the gemini-3-flash-preview model (via Vertex AI/Google AI Studio), selected for its high inference speed and large context window capability.

The raw textual output from the model contains the reasoning steps followed by the final answer. To ensure the system returns a strictly numerical float required for the evaluation script, a robust parsing mechanism using Python's Regular Expressions (re) was developed. The parser searches for the specific "Final Answer:" pattern. If the pattern is not found, it falls back to extracting the last numerical value in the text sequence. This decoupling of reasoning (text) and final output (float) ensures compatibility with the automated testing suite.

3. Results and Analysis

The solution was evaluated against a ground truth dataset comprising cost vectors for two specific queries and one out-of-domain test.

- **Query 1 (Total Expenditure):** The model successfully identified the "Total" line item from all seven receipts. The sum calculated by the model matched the expected ground truth of roughly **1974.30** (within the allowable

$$\pm 2$$

margin). The visual grounding was sufficient to distinguish between line items and final totals.

- **Query 2 (Total Without Discount):** This task was more complex, requiring the model to identify either the "Subtotal" or add "Savings" back to the "Total." The CoT strategy proved essential here. By listing the pre-discount value for each receipt (aggregating to approximately **2348.20**), the model avoided the common pitfall of hallucinating a sum without performing the underlying arithmetic.
- **Out-of-Domain Query:** When asked about the weather, the model correctly triggered the guardrail logic and returned "IRRELEVANT," demonstrating effective intent classification capabilities.

4. Conclusion

The implemented solution successfully demonstrates the efficacy of Multimodal LLMs in automating financial document processing. By moving beyond simple value extraction to complex arithmetic aggregation, the system proves that LLMs can function as effective agents for business analytics. The key to success was the transition from direct-answer prompting to Chain-of-Thought reasoning, which bridged the gap between visual perception and mathematical logic. The system is robust, accurate, and capable of handling both relevant financial queries and rejecting out-of-scope inputs.