Enhancing Large Language Models with Agentic Retrieval-Augmented Generation

Noe Javet

June 3, 2025

Abstract

This thesis investigates whether small, locally running large language models (LLMs) can achieve comparable performance to commercial cloud-based systems like ChatGPT in evaluating medical discharge reports. The evaluation focuses on identifying incomplete or missing information across structured markdown-formatted sections of clinical documents. Several strategies were explored, including optimized prompt engineering, manual formatting of Word documents, parameter tuning (e.g., temperature settings), and agentic workflows involving tool use and multi-model collaboration. Performance was assessed using precision, recall, and qualitative criteria such as conciseness and relevance. Results show that while small LLMs improve with structured input and specialized prompting, they remain limited in reliability compared to state-of-the-art proprietary models.

1 Introduction

The ability of language models to analyze medical documents is of increasing importance in clinical decision support and medical auditing. While commercial models like ChatGPT have demonstrated strong performance in natural language understanding tasks, they pose privacy and latency concerns in sensitive domains. This work investigates whether small, open-source LLMs running on local hardware can provide a viable alternative for evaluating medical discharge reports. The task centers on identifying missing or incomplete sections in structured documents — a challenge requiring nuanced understanding of clinical content. We explore the potential of formatting, prompt engineering, and agentic workflows to enhance local model performance.

2 Methodology

To evaluate the capabilities of various language models (LLMs) in processing medical discharge reports, a multi-stage evaluation pipeline was implemented. The process consisted of document preprocessing, prompt engineering, model inference, and result evaluation using defined metrics.

Initially, raw discharge summaries in DOCX format were parsed using the `python-docx` library. This stage focused on extracting plain text content from the original documents while preserving section headers and structural elements. The extracted text was then

sent to nine different LLMs using a uniform system prompt designed to assess completeness and relevance of medical sections.

Subsequently, to reduce variability introduced by inconsistent formatting, the documents were manually reformatted to adhere to a markdown-like structure. These manually structured texts were then re-evaluated using the same baseline system prompt to measure improvements attributable solely to input clarity.

In the final stage, the same markdown-formatted texts were reprocessed using a refined system prompt with more explicit instructions aimed at improving model focus and output consistency. This allowed for a comparative assessment of prompt sensitivity across models.

The evaluation metrics comprised both structural and content-based indicators: true positives, true negatives, false positives, and false negatives regarding section detection; as well as binary assessments of section relevance and conciseness (`0/1` scale). Sections marked as missing or irrelevant were explicitly recorded to aid in qualitative analysis.

The evaluation pipeline consisted of the following stages:

Data Preparation A corpus of psychiatric discharge reports was converted from .docx to markdown using python-docx, with manual post-processing to standardize section headers and content layout.

Baseline Evaluation Each markdown-formatted document was evaluated using nine different LLMs under a shared system prompt. The models' outputs were compared against a human-annotated reference.

Prompt and Parameter Tuning Experiments with modified system prompts and various temperature settings were conducted to assess their impact on performance.

Agentic Workflow A second phase introduced tool use and inter-model collaboration. One model (planner) orchestrated evaluation plans, delegating judgment and justification tasks to others. This setup aimed to simulate a form of self-verifying reasoning.

Evaluation Metrics We measured:

True/False Positives and Negatives (per section)

Irrelevant response ratio (0/1)

Conciseness of justification (0/1)