Using LLM And Prompt Engineering to Build Institution Profiles

Group:
Jihad Awad – 1210088
Mohammed Abed Alkareem – 1210708
Hamzah Alqam - 121173
Abdulghafer Qeel - 1210408

Description Given:
Given an institution (in fields of Finance, Medicine, & Education) and search results search one needs to build a profile using LLMs, prompt engineering with preference to recent or quality data.

Abstract:
The incredible and recent advancement in LLMs makes it possible for us to extract and use data that isn't structured nor predictable, due to their ability to analyse and extract information from it in a human-like way (Eric Anderson, 2024), and their ability to further go an extra step by providing relevant information or performing complex tasks on that data, meaning where we used to rely on classification, general heuristics and ranking and hoping the returned documents or text contain the relevant data, which we process ourselves, we can more reliably use LLMs to do so, which are more adaptable and can capture, dismiss, analyse and summarise information with more accuracy and better understanding of the context and semantics (Eric Anderson, 2024).

Our goal is to make use of existing LLMs known to be reliable, consistent and cheap, found through public performance benchmarks (Gemini 2.0 Flash: Analysis, 2025), for two tasks: extracting all information from a website or a document, and then analysing and transforming that information in order to build a full profile of the institutions mentioned before.

To that end we have to make use of web scrapers or build our own, to search and retrieve the raw data from websites, and craft prompts that ensure the LLM adheres and understands our needs and exact output or input structure (BAML Language, n.d.) so that we can make use of it without errors, through prompt engineering (Prompt engineering, 2025). Since LLMs are sensitive to prompts and context, as well as length and data complexity, the process might involve many steps where specific information is extracted or summarised, and then used or fed back into the LLM, so we also need to test some workflows.

References

BAML Language. (n.d.). Retrieved from BoundaryML: https://www.boundaryml.com/

Eric Anderson, J. F. (2024, September 4). *The Design of an LLM-powered Unstructured Analytics System*. Retrieved from arxiv.org: https://arxiv.org/html/2409.00847v2

Gemini 2.0 Flash: Analysis. (2025, February 25). Retrieved from Artificial Analysis: https://artificialanalysis.ai/models/gemini-2-0-flash

Prompt engineering. (2025, March 24). Retrieved from platform.openai.com: https://platform.openai.com/docs/guides/prompt-engineering