Document Search and Summarization Using Large Language Models (LLM)

Objective

The main objective of this assignment is to design a system that can search and summarize vast amounts of textual data efficiently. The candidate should be able to utilize a Large Language Model (LLM) like GPT-4 or its successors to accomplish this task.

Background

With the advent of LLMs, numerous possibilities in the realm of Natural Language Processing have emerged. In this assignment, the candidate is expected to harness the capabilities of these models to perform accurate document searches and succinct summarizations.

Task Details

1. **Data Preparation:**
    - Choose a sizable corpus for the task.
    - Pre-process and clean the dataset, making it suitable for search and summarization.
2. **Document Search:**
    - Implement a mechanism to ingest user queries and search the corpus to return relevant documents.
    - Use a combination of traditional information retrieval methods and LLM embeddings for enhanced search accuracy.
    - Return the top N most relevant documents or excerpts for a given query.
3. **Document Summarization:**
    - Once the relevant documents are retrieved, summarize them using an LLM.
    - The summary should be coherent and capture the essence of the returned documents.
    - The user should have the option to specify the desired length of the summary.
4. **Evaluation:**
    - Create a subset of the corpus as a test set. For each document in the test set, generate a query that ideally returns that document as a relevant result.

- Measure the accuracy of the search mechanism based on the relevance of returned documents.
- Evaluate the summaries' quality using automated metrics (e.g., ROUGE scores) and human evaluation.

5. **Interface (Bonus):**
    - Develop a user-friendly interface where users can input their queries and get the summarized results.
    - Implement features like auto-suggestion for queries, pagination for search results, and adjustable summary lengths.

Deliverables

1. A detailed report documenting:
    - Data pre-processing steps and justification.
    - The methodology adopted for document search and summarization.
    - The evaluation procedure and results.
    - Any challenges faced and their solutions.
2. Codebase:
    - The entire code is used for data preparation, examination, summary, and evaluation.
    - A README file detailing how to set up and run the solution.
3. (Bonus) A hosted version of the interface or detailed instructions on how to set it up locally.

Evaluation Criteria

1. **Functionality:** The solution should effectively search and summarize the corpus based on user queries.
2. **Accuracy:** The relevance of search results and the quality of summaries will be paramount.
3. **Scalability:** The solution should be scalable for even larger datasets.
4. **Efficiency:** The time taken for search and summarization should be reasonable.
5. **Report Quality:** The clarity, structure, and depth of the report.
6. **Code Quality:** Cleanliness, modularity, and documentation of the code.

Hints and Suggestions

- Consider fine-tuning the LLM on the chosen corpus to improve domain-specific accuracy.
- Efficient indexing methods like TF-IDF, BM25, or even embeddings (e.g., sentence embeddings, document embeddings) can be used for the search.

- LLMs can be very resource-intensive. Consider solutions that optimize computational resources.
- Regularly test the system with real-world queries to gauge its effectiveness.