535106 LLM Team Project: Personal Research Notes Assistant

李志宇　彭眞人　黃昱文
Department of Electrical Engineering
Institute of Computer Science and Engineering
National Yang Ming Chiao Tung University
{leezhiyu.ee14, jenrenpong.ee14, wenyellow.cs13}@nycu.edu.tw

1 Project Overview

1.1 Profile

Track: 3. Application This project focuses on building a practical LLM application system to solve specific pain points for researchers in knowledge management.

Abstract and project goal: Researchers generate numerous scattered notes when reading multiple papers, which are often difficult to organize, retrieve, and review. This project aims to build a "Personal Research Notes Assistant." This system will automatically process and summarize the papers and construct them into a vectorized knowledge base. We will leverage Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG) [Lewis et al., 2020] to provide users with semantic retrieval and Contextual Question Answering (QA) capabilities based on the papers given. The ultimate goal is to create an intelligent assistant with personalized knowledge memory to help researchers organize knowledge more efficiently and inspire new research ideas.

TL;DR ("Too Long; Didn't Read"): An LLM assistant that understands the papers given and answers your questions on the basis of the content.

1.2 Motivation

Every researcher suffers from "knowledge silos"—notes are scattered everywhere, and human memory is limited. If an LLM can act as an intelligent index to our "second brain," allowing our "past self" to converse with our "present self," it would maximize the compounding effect of knowledge. This is not just a tool innovation but a fundamental optimization of the personal research workflow.

Most critical challenges:

- **Challenge 1: Semantic Understanding and Integration** There is abundant information in the papers, leading to the difficulty of integrating well-structured information from an extremely long context. Accurately segmenting these fragments from different topics into meaningful semantic units is fundamental to the accuracy of RAG.
- **Challenge 2: Lack of Personalized Contextual Memory** Standard RAG systems are stateless. However, a "personal" assistant must "remember" the user's recent research topics and conversations. Designing an effective memory mechanism that allows the assistant's responses to align with the user's current research context is key to improving its utility.

- **Challenge 3: Evaluation Challenges for Private Knowledge Bases**: The " quality" of the system's answers is highly dependent on the specific papers. We lack public benchmarks to evaluate "whether the answer is faithful to the original notes" and "whether the retrieved notes are the best evidence," which makes evaluation very difficult.

Current note-taking software (like Notion, Obsidian) is powerful, but its retrieval is still mostly keyword-based, lacking deep semantic understanding and automated summarization capabilities. While RAG (Retrieval-Augmented Generation) technology has matured, most research focuses on "public domain" QA (like Wikipedia), not on "private, scattered, multi-format" personal knowledge bases. How to apply RAG to highly personalized scenarios and combine it with long-term memory remains an open application problem.

Related Work:

- **Recent Work (Advanced RAG):** *Self-RAG* [Asai et al., 2023] explores letting the model decide when to retrieve and reflect on the retrieved content, which is insightful for improving the accuracy of note-based QA.
- **Recent Work (Memory):** *MEMGPT* [Packer et al., 2023]proposed a virtual context management system, enabling LLMs to handle long-term memory beyond their context window, which is crucial for implementing the "personalized memory function" of this project.

State-of-the-art method: In our opinion, the SOTA method is a hybrid architecture combining RAG with a long-term memory module. Specifically, it involves a retrieval layer using powerful embeddings with Hybrid Search (keyword + semantic), and then feeding the retrieval results along with a "session summary memory" into an LLM for generation.

2 Problem Formulation

This project addresses the task of Contextual Question Answering (QA) and summarization for personalized knowledge bases. We define this problem as a Retrieval-Augmented Generation (RAG) challenge.

Specifically, the system input $(\mathcal{D}, Q)$ consists of two parts: a corpus of unstructured personal notes $\mathcal{D} = \{d_1, d_2, ..., d_n\}$ provided by the user (where d_i is a paper), and a natural language query Q from the user. The system's goal is to generate an answer A, which must (1) accurately answer Q, and (2) be fully supported by the information within $\mathcal{D}$.

Our optimization objective $f(\cdot)$ is to maximize the "Faithfulness" and "Relevance" of the answer A.

$$A^* = \arg\max_A f(A|Q, \mathcal{D})$$

Here, "Faithfulness" means all claims in A can be supported by corresponding evidence in $\mathcal{D}$; "Relevance" means A fully and directly addresses Q. Since $\mathcal{D}$ is private, we cannot rely on traditional labeled datasets. Therefore, we will utilize an LLM-as-a-judge and user feedback as our primary supervision signals.

This project assumes we have sufficient computational resources to run open-source LLMs (e.g., Llama 3 8B [Touvron et al., 2023]) and a vector database (e.g., ChromaDB). We also assume the content is primarily text-based; multi-modal (e.g., figures, tables) integration will be considered future work.

3 Empirical Evaluation

We will adopt a combination of automated metrics and human evaluation to assess the effectiveness of our system.

3.1 Performance Metrics

We plan to adopt the following four Key Performance Indicators (KPIs):

- **Answer Accuracy**: We will manually create a "Golden QA Set" based on a sample note library. This metric evaluates whether the content of the model's answer is factually correct.
- **Retrieval Relevance**: This assesses the semantic relevance of the retrieved note snippets to the user's query. We will use LLM-based automatic scoring (e.g., using GPT-4 [OpenAI, 2023] or E5-large-v2 [Wang et al., 2022] for similarity) to evaluate the average relevance score of the Top-K retrieved texts.
- **Generation Coherence**: This evaluates whether the LLM-generated answer is fluent, readable, and successfully synthesizes multiple retrieved snippets into a coherent text. This metric will primarily be scored by an LLM-as-a-judge (e.g., GPT-4 self-evaluation) or human rating.
- **Latency**: We will measure the average time (in seconds) from when the user issues a query to when the system returns a complete answer. This will serve as a key metric for system optimization.

3.2 Baseline Methods

We will use the following two methods as baselines to highlight the superiority of our proposed method:

- **Baseline 1: Keyword-based Search**: This represents the retrieval capability of traditional note-taking software (like Obsidian). We will use TF-IDF or BM25 algorithms for keyword retrieval and present the raw text snippets directly to the user. This baseline is expected to perform poorly on "Relevance" and "Coherence."
- **Baseline 2: Vanilla RAG**: This will use the standard LangChain framework with a general-purpose embedding model (e.g., 'sentence-transformers/all-MiniLM-L6-v2') and a FAISS vector index. This method will not include a personalized memory module or Hybrid Search. It will serve as a strong supervised baseline.

Our proposed method will build upon Baseline 2, integrating Hybrid Search, a personalized memory module, and potentially stronger embeddings (e.g., 'E5-large-v2') for optimization.

3.3 Benchmark Tasks or Datasets

Due to the "personalized" nature of this project, we will construct our own benchmark dataset for evaluation.

- **Task**: Contextual QA on the papers given.
- **Dataset Construction Pipeline**:
  1. **Data Collection**: We will collect papers in a specific domain (e.g., "Diffusion Models" or "LLM Agents").
  2. **Annotation Process**: Based on this "note library," we will manually write some questions (Q). Each question will be annotated with a "golden answer".
  3. **Quality Control**: We will use cross-validation. Questions written by one member will be reviewed by another to ensure clarity, answer accuracy, and source traceability.

4 Methodology (Optional)

Our solution is an intelligent assistant system based on four core modules. The high-level idea is illustrated in Figure [FIGURE] (We will draw an architecture diagram):

(1) **Data Preprocessing Module**: It uses 'RecursiveCharacterTextSplitter' to segment the papers into semantically coherent chunks and extracts metadata like titles and dates.

(2) **Vector Database Module**: Uses the 'E5-large-v2' embedding model to vectorize the text chunks and stores them in ChromaDB. This module supports both Semantic Retrieval and Hybrid Search. (3) **LLM Core Module**: This is the system's brain. When a user asks a question, it first "Retrieves" the Top-K relevant note snippets from the vector store. Then, it reads the "Long-term Memory" (a summary of the user's recent research topics). It combines the "retrieved snippets," "long-term memory," and the "user query" into a carefully designed prompt template. (4) **Generation & Interaction**: Finally, the LLM (e.g., Llama 3 8B) "Generates" a coherent, accurate answer that cites the paper sources, based on the prompt. This answer will be presented to the user.

5 Experimental Results

5.1 Evaluation of Baseline Methods

6 Expected Contributions of Each Team Member

Please describe how you would collaborate with each other and specifically what each member would contribute to this project.

- Name of Member 1:
- Name of Member 2:
- Name of Member 3:
- Name of Member 4:

As you just kick off the project, it is very likely that you may need to reallocate the tasks among yourselves as you proceed. Please make sure that you reach a consensus on the expected contributions.

References

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems, 33:9459–9474, 2020.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. arXiv preprint arXiv:2310.11511, 2023.

Charles Packer, Vivian Fang, Shishir G. Patil, Kevin Lin, Sarah Wooders, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems. arXiv preprint arXiv:2310.08560, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971, 2023.

OpenAI. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. arXiv preprint arXiv:2212.03533, 2022.