Integrating LightRAG and MemoRAG for Enhanced Retrieval-Augmented Generation with Knowledge Graphs

Abhinav Agarwal*
Department of Computer Science
Stanford University
abhinav4@stanford.edu

Betty Wu†
Department of Computer Science
Stanford University
bettyyw@stanford.edu

Project Category: Real-world applications of GNNs

1 Introduction

Large Language Models (LLMs) excel in natural language understanding and generation. However, they face challenges in handling complex queries with ambiguous or implicit information needs, especially when dealing with large, unstructured datasets. Retrieval-Augmented Generation (RAG) systems have been introduced to address these limitations by augmenting LLMs with external retrieval mechanisms [1].

In this project, we propose to integrate two advanced RAG models—MemoRAG [2] and LightRAG [3]—to create a hybrid system that leverages the strengths of both approaches. MemoRAG introduces a memory module that generates staging answers (clues) to guide retrieval, while LightRAG employs graph-based text indexing and a dual-level retrieval paradigm for efficient and comprehensive information retrieval.

We aim to implement this hybrid model using PyTorch Geometric (PyG) [4], extending its capabilities to support memory modules and graph-based retrieval mechanisms in RAG systems. Our work will focus on adapting LightRAG's graph-based components to utilize PyG, even though the original LightRAG implementation does not use PyG.

2 Related Work

Retrieval-Augmented Generation. RAG systems combine LLMs with external retrieval mechanisms to enhance knowledge-intensive tasks. Notable works include RAG [1], REALM [7], and DPR [8]. These models improve upon traditional LLMs by incorporating relevant context retrieved from external sources.

MemoRAG. MemoRAG [2] addresses limitations in traditional RAG systems when dealing with ambiguous queries and unstructured knowledge. It introduces a memory module that generates staging answers (clues) to guide the retrieval process, enhancing the model's ability to handle implicit information needs.

LightRAG. LightRAG [3] improves retrieval efficiency and comprehensiveness by employing a dual-level retrieval system and graph-based text indexing. It constructs knowledge graphs from text data and utilizes both low-level and high-level retrieval strategies to handle specific and abstract queries.

Graph Neural Networks and PyG. Graph Neural Networks (GNNs) have become a powerful tool for learning over graph-structured data [5, 6]. PyTorch Geometric (PyG) [4] is a library for deep learning on irregularly structured data, providing efficient implementations of various GNN models.

*SUNet ID: abhinav4
†SUNet ID: bettyyw

3 Motivation

Combining LightRAG with MemoRAG is a powerful approach that leverages the complementary strengths of both models to address their individual limitations. MemoRAG excels at managing ambiguous queries by using a memory module to generate clues or staging answers that guide the retrieval process, helping interpret complex queries and provide focused retrieval.

LightRAG, with its graph-based text indexing and dual-level retrieval, efficiently navigates large datasets by capturing entities and their relationships, enabling both detailed and broad-topic retrieval. By integrating MemoRAG's clue generation with LightRAG's graph-based retrieval, the system improves accuracy and relevance, especially for complex, ambiguous queries.

Using PyG to implement this integration allows for the use of advanced graph neural networks, further enhancing the system's ability to retrieve and reason over large-scale unstructured data. This hybrid approach is expected to significantly improve retrieval quality in RAG systems.

4 Dataset

We plan to use the **ULTRADOMAIN benchmark dataset** introduced in the MemoRAG paper [2]. This dataset includes long-context documents from various specialized domains such as finance, law, and academia.

The documents range from 50,000 to over 1,000,000 tokens and span various domains, including financial reports, legal contracts, academic papers, and college textbooks across 18 fields such as natural sciences, humanities, and social sciences.

We chose this dataset for its diversity and unstructured knowledge, which makes it ideal for testing our hybrid RAG model's ability to handle complex, ambiguous queries across a wide range of disciplines.

Task Definition

- **Objective**: Develop a hybrid RAG system that enhances LLM performance on complex tasks involving ambiguous queries and extensive contexts.
- **Specific Tasks**:
  1. Question Answering (QA) with implicit information needs.
  2. Summarization of long documents.
  3. Multi-hop reasoning over extensive contexts.

Evaluation Metrics

- **Retrieval Metrics**:
  - Recall@K: Measures the proportion of relevant documents retrieved in the top K results.
  - Mean Reciprocal Rank (MRR): Evaluates the average rank position of the first relevant document.
- **Generation Metrics**:
  - F1 Score: For assessing the accuracy of answers in QA tasks.
  - ROUGE-L: For evaluating the quality of summarizations.
- **Human Evaluation Metrics**:
  - Comprehensiveness: How thoroughly does the answer address the question?
  - Relevance: How relevant is the answer to the query?
  - Coherence: Logical flow and clarity of the generated text.

5 Methodology

Our proposed approach integrates MemoRAG's memory module with LightRAG's graph-based retrieval, leveraging PyG for efficient graph operations and GNN models.

MemoRAG introduces a memory module Θ_{mem} that compresses the input sequence X into memory tokens X^m:

$$X^m = \Theta_{\text{mem}}(X).$$

It then generates clues y based on the global memory and query q:

$$y = \Theta_{\text{mem}}(q, X^m).$$

These clues guide the retrieval process, helping to address ambiguous queries.

LightRAG constructs a knowledge graph $G = (V, E)$ from text data, where V represents entities and E represents relationships. The original implementation does not utilize PyG; however, we propose to adapt LightRAG's graph-based components using PyG for the following reasons:

- **Efficient Graph Representation**: PyG provides optimized data structures (`Data`, `HeteroData`) for representing large-scale graphs.
- **Graph Neural Networks**: PyG offers a variety of GNN layers (e.g., `GCNConv`, `GATConv`) that can be used to learn representations over the knowledge graph.
- **Scalability**: PyG's efficient handling of sparse data and batching mechanisms allows for scaling to large graphs.

By integrating the clues generated by MemoRAG's memory module with LightRAG's graph-based retrieval, we aim to create a system where the memory-inspired clues guide the traversal and retrieval processes within the knowledge graph. This integration involves:

- Using the clues y as input features or query vectors in the GNN models.
- Enhancing node and edge representations in the knowledge graph using the memory module outputs.
- Implementing attention mechanisms to focus on relevant subgraphs based on the generated clues.

The final answer Y is generated using the generator LLM Θ:

$$Y = \Theta(q, C),$$

where C is the retrieved context from the knowledge graph.

The model is appropriate because it integrates MemoRAG's ability to handle ambiguous queries with LightRAG's efficient graph-based retrieval, making it well-suited for extracting relevant information from the long, unstructured documents in the ULTRADOMAIN dataset.

6 Conclusion

We propose to develop a hybrid RAG system by integrating MemoRAG and LightRAG, adapting LightRAG's components to utilize PyG. This integration aims to enhance LLM performance on complex tasks involving ambiguous queries and large contexts. Our implementation will extend PyG's capabilities and contribute to the graph machine learning community by providing new tools for combining memory modules with graph-based retrieval.

References

[1] Lewis, P., et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.* Advances in Neural Information Processing Systems, 33, 9459–9474.

[2] Qian, H., et al. (2024). *MemoRAG: Moving Towards Next-Gen RAG via Memory-Inspired Knowledge Discovery.* arXiv preprint arXiv:2309.05591.

[3] Guo, Z., et al. (2024). *LightRAG: Simple and Fast Retrieval-Augmented Generation.* arXiv preprint arXiv:2310.05779.

[4] Fey, M., & Lenssen, J. E. (2019). *Fast Graph Representation Learning with PyTorch Geometric.* In ICLR Workshop on Representation Learning on Graphs and Manifolds.

[5] Kipf, T. N., & Welling, M. (2017). *Semi-Supervised Classification with Graph Convolutional Networks.* In International Conference on Learning Representations (ICLR).

[6] Veličković, P., et al. (2018). *Graph Attention Networks.* In International Conference on Learning Representations (ICLR).

[7] Guu, K., et al. (2020). *REALM: Retrieval-Augmented Language Model Pre-Training.* In International Conference on Machine Learning (ICML).

[8] Karpukhin, V., et al. (2020). *Dense Passage Retrieval for Open-Domain Question Answering.* In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP).