

LongRAG và LightRAG – Hai bước tiến mới trong hệ thống RAG

Trong lĩnh vực Xử lý ngôn ngữ tự nhiên (NLP), mô hình **Retrieval-Augmented Generation (RAG)** đã trở thành một phương pháp nổi bật để tăng cường khả năng trả lời câu hỏi bằng cách kết hợp giữa mô hình ngôn ngữ lớn (LLM) và truy xuất thông tin từ dữ liệu bên ngoài. Tuy nhiên, quá trình **retrieval (truy xuất)** vẫn còn nhiều hạn chế. Gần đây, hai framework mới là **LongRAG** và **LightRAG** đã được đề xuất nhằm khắc phục những nhược điểm này, giúp cải thiện cả độ chính xác và hiệu quả.

Tổng quan về RAG

Hệ thống RAG hoạt động theo một pipeline cơ bản gồm:

- **Retriever:** Truy xuất các đoạn văn bản liên quan từ một kho dữ liệu lớn.
- **(Optional) Ranker:** Xếp hạng các đoạn được truy xuất (có thể có hoặc không).
- **Reader / Generator:** Mô hình ngôn ngữ sử dụng các đoạn truy xuất để sinh câu trả lời.

Hạn chế lớn nhất của RAG truyền thống là việc chia nhỏ dữ liệu thành các đoạn ngắn (~100 từ), khiến quá trình truy xuất như “tìm kim đáy bể” – thiếu ngữ cảnh, dễ gây nhiễu và không đầy đủ.

LongRAG – Tối ưu ngữ cảnh qua đoạn văn dài

Ý tưởng chính:

LongRAG, được giới thiệu trong một nghiên cứu năm 2024 của **Xian Zhang**, giải quyết hạn chế của RAG truyền thống bằng cách sử dụng các đoạn văn **dài hơn nhiều** (ví dụ: 4000 token ~ 3000 từ) để truy xuất.

Ưu điểm:

- **Giữ nguyên ngữ cảnh đầy đủ** hơn so với các đoạn ngắn.
- **Giảm số lượng đoạn cần truy xuất** → tiết kiệm chi phí tính toán.
- **Đơn giản nhưng hiệu quả cao:** Không cần fine-tune thêm mô hình.

Kết quả thực nghiệm:

Dataset	LongRAG (EM Score)
NQ (Natural Questions)	62.7
HotpotQA	64.3

→ So sánh ngang với **Atlas**, một mô hình RAG state-of-the-art nhưng không mã nguồn mở.

Nhận xét:

- LongRAG hoạt động tốt trên cả **truy vấn đơn** lẫn **multi-hop QA** nhờ khả năng duy trì ngữ cảnh dài.
- Đây là một giải pháp đơn giản, dễ triển khai, nhưng đem lại hiệu quả ấn tượng.

LightRAG - Truy xuất theo đồ thị ngữ nghĩa

Ý tưởng chính:

LightRAG, do **Xirui Guo, Lianghao Xia** và các cộng sự đề xuất (2024), tập trung vào cấu trúc hóa kiến thức truy xuất bằng cách xây dựng **đồ thị thực thể - quan hệ (graph-based retrieval)**.

Vấn đề mà LightRAG giải quyết:

- Dữ liệu phẳng, thiếu liên kết giữa các đoạn.
- Không thể hiện được **cấu trúc liên kết tri thức** giữa các thực thể.

Các bước chính:

- Deduplication**: Loại bỏ thông tin trùng lặp (VD: đoạn "beekeeper" xuất hiện nhiều lần).
- LM Profiling**: Dùng LLM tạo cặp khóa - giá trị mô tả thông tin cốt lõi của đoạn.
- Entity & Relationship Extraction**: Trích xuất các thực thể và quan hệ → xây đồ thị.
- Index Graph**: Biểu diễn tri thức dạng đồ thị kết nối các thực thể.
- Dual-Level Retrieval**:
 - Low-level**: Truy xuất chi tiết (VD: "beekeeper").
 - High-level**: Truy xuất theo chủ đề rộng (VD: "nông nghiệp").

So sánh với các baseline:

Dataset	Win Rate của LightRAG
Legal	80.95%
Common Sense	Vượt trội
Mixed	Cao nhất về diversity

→ LightRAG vượt qua mọi baseline bao gồm: **NaiveRAG**, **RQ-RAG**, **HypeRAG**, và **GraphRAG**.

So sánh LongRAG và LightRAG

Tiêu chí	LongRAG	LightRAG
Cách tiếp cận	Tăng độ dài đoạn truy xuất	Biểu diễn kiến thức dạng đồ thị
Triển khai	Rất đơn giản	Phức tạp hơn, cần xử lý NLP nâng cao
Hiệu quả	Cải thiện ngữ cảnh và giảm lỗi	Cung cấp truy xuất tầng sâu – đa tầng
Tối ưu cho	Truy vấn yêu cầu ngữ cảnh đầy đủ	Câu hỏi nhiều bước, chủ đề phức tạp
Độ mới lạ	Kỹ thuật cải tiến truyền thống	Đột phá trong cách biểu diễn tri thức

LongRAG và **LightRAG** đều đóng vai trò quan trọng trong việc cải tiến hệ thống RAG:

- **LongRAG** phù hợp với các ứng dụng yêu cầu đơn giản, dễ triển khai nhưng cần độ chính xác cao nhờ ngữ cảnh đầy đủ.
- **LightRAG** thích hợp với các hệ thống tri thức chuyên sâu, yêu cầu mô hình hiểu rõ mối liên hệ giữa các thực thể.

Cả hai đều cho thấy tiềm năng lớn trong việc giúp các mô hình LLM trả lời câu hỏi **chính xác hơn, đa dạng hơn và có chiều sâu tri thức hơn**.

Tác giả: **Đỗ Ngọc Tú**
Công Ty Phần Mềm **VHTSoft**