JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 30 No. 10, pp. 23-31, October 2025
https://doi.org/10.9708/jksci.2025.30.10.023

ToR-RAG: A Tree-of-Retrieval-based Retrieval-Augmented Generation for Complex Question Processing

Hee-Kyong Yoo*, Nammee Moon**

*Student, Dept. of Convergence Engineering, Hoseo University, Seoul, Korea
**Professor, Dept. of Computer Science and Engineering, Hoseo University, Asan, Korea

[Abstract]

Large Language Models have advanced natural language understanding, yet remain limited in handling complex multi-hop queries requiring integration across multiple documents. Traditional Retrieval-Augmented Generation adopts a linear query-retrieval-generation pipeline, which often causes error propagation, incomplete evidence coverage, and reduced reliability. To overcome these issues, this study proposes a Tree-of-Retrieval based RAG (ToR-RAG). ToR-RAG decomposes queries into binary sub-queries via LLM prompting, performs retrieval and partial answer generation at each branch, and evaluates outputs using an LLM-as-a-Judge module. Branches below a quality threshold are pruned, ensuring efficiency. An MMR-based retrieval strategy (λ=0.75) with top-k=5 selection balances relevance and diversity. Experiments on the MultiHop-RAG dataset show that ToR-RAG improves Exact Match by +6.42 and F1 by +6.04 compared to Non-RAG, Native-RAG, and CoR-RAG. Performance peaked at Depth=3, while Depth=4 caused degradation from excessive branching and token usage. These results demonstrate that ToR-RAG enhances both accuracy and reliability in multi-hop reasoning, suggesting applicability in domains such as policy analysis, medical decision-making, and financial risk assessment.

▸ **Key words**: RAG, Tree of Retrieval, Multi-Hop QA, Query Decomposition, Structured Reasoning

[요 약]

대규모 언어모델(LLMs)은 자연어 이해 능력을 크게 발전시켰으나, 다수 문서에 분산된 근거를 통합해야 하는 복합 멀티홉 질의 처리에서는 한계를 보인다. 기존 검색증강생성(RAG)은 선형 파이프라인을 채택하여 오류 전파, 근거 불충분, 신뢰성 저하 문제가 발생한다. 이를 해결하기 위해 본 연구는 Tree-of-Retrieval 기반 RAG(ToR-RAG)를 제안한다. ToR-RAG는 LLM 프롬프트를 통해 질의를 이진 하위 질의로 분해하고, 각 분기에서 검색 및 부분 응답 생성을 수행한 후, LLM-as-a-Judge 모듈로 품질을 평가한다. 기준 이하의 분기는 가지치기를 통해 효율성을 확보한다. 검색 단계에서는 MMR 기반 전략(λ=0.75, top-k=5)을 적용하여 관련성과 다양성을 균형 있게 확보하였다. MultiHop-RAG 데이터셋 실험 결과, ToR-RAG는 Non-RAG, Native-RAG, CoR-RAG 대비 Exact Match +6.42, F1 +6.04의 성능 향상을 보였다. 특히 깊이(Depth)=3에서 최적 성능을 기록했으며, Depth=4에서는 과도한 분기와 토큰 사용으로 성능이 저하되었다. 이러한 결과는 ToR-RAG가 멀티홉 추론에서 정확성과 신뢰성을 동시에 강화함을 보여주며, 정책 분석, 의료 의사결정, 금융 리스크 평가 등 다양한 응용 분야에서 활용 가능성을 제시한다.

▸ **주제어**: 검색증강생성, 트리 기반 검색, 복합 질문 응답, 질의 분해, 구조적 추론

• First Author: Hee-Kyong Yoo, Corresponding Author: Nammee Moon
*Hee-Kyong Yoo (hkyoo@dataslab.co.kr), Dept. of Convergence Engineering, Hoseo University
**Nammee Moon (mnm@hoseo.edu), Dept. of Computer Science and Engineering, Hoseo University
• Received: 2025. 09. 11, Revised: 2025. 09. 23, Accepted: 2025. 10. 01.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

대규모 언어모델(Large Language Models, LLMs)은 방대한 데이터 학습을 기반으로 언어 이해와 응답 생성 능력을 크게 향상시켜 다양한 응용 분야에서 활용되고 있다[1]. 그러나 LLM은 다수의 문서에서 분산된 근거를 통합해야 하는 복합 질의 (complex queries), 특히 다문서 기반 멀티홉 추론(multi-hop reasoning) 문제에서 여전히 한계를 보인다[2][3]. 즉, LLM이 근거를 단계적으로 결합하여 추론하는 과정에서 오류 누적과 할루시네이션(hallucination)이 발생하며[1][4][5][6], 결과적으로 신뢰성이 저하된다.

이를 보완하기 위해 제안된 방법이 검색증강생성(Retrieval-Augmented Generation, RAG)이다. RAG는 외부 지식베이스에서 관련 문서를 검색하여 LLM 입력에 포함시킴으로써 파라미터 지식의 부족을 보완하고 사실성을 강화한다[1]. 그러나 기존 RAG는 대부분 단일 질의-단일 검색-단일 응답의 선형 파이프라인 구조를 따르며, 이로 인해 정보 누락과 오류 전파의 문제가 발생한다[1][4]. 이는 복합 질의에서 요구되는 다양한 증거 포착과 단계적 추론을 충분히 지원하지 못한다[5][7][8].

이러한 한계를 극복하기 위해 구조적 추론 기법이 도입되었다. Chain-of-Thought(CoT)는 단계적 사고 과정을 모사하여 추론 성능을 높였으나, 선형 구조의 특성상 오류 전파에 취약하다[9][10]. 반면 Tree-of-Thought(ToT)는 트리 구조를 기반으로 다수의 경로 탐색과 백트랙킹(backtracking)을 가능하게 하여 보다 체계적이고 견고한(robust) 추론을 지원한다[9]. 이러한 아이디어는 검색 구조에도 영향을 주어 트리 기반 RAG 연구(T-RAG, CFT-RAG, HiRMed)로 확장 되었으나, 주로 정적 구조 반영에 머물러 동적 분기 생성과 가지치기 최적화까지는 충분히 다루지 못하였다[7][11][12].

또한, 단순 정답 산출을 넘어 인사이트를 생성하는 Insight-RAG가 제안되었으며[8], Sharma[5]는 최신 RAG 서베이에서 **구조적 멀티홉 추론과 동적 검색 최적화**를 차세대 핵심 과제로 제시하였다. 이에 본 연구에서는 이러한 문제의식을 바탕으로 Tree-of-Retrieval 기반 RAG(ToR-RAG)를 제안한다. ToR-RAG는 원 질의를 이진 하위 질의(binary sub-queries)로 분해하고, 각 분기에 대해 검색과 부분 응답 생성을 수행한 후, LLM-as-a-Judge 모듈을 통해 분기 품질을 평가한다. 기준에 미달하는 분기는 가지치기(pruning)하고, 유효한 분기만을 재귀적으로 확장하여 리프 노드의 증거를 통합한다. 본 연구의 목적은 이러한 구조적 멀티홉 추론 성능을 실험을 통해 검증하는 것이다.

II. Preliminaries

1. Related works

Retrieval-Augmented Generation(RAG)은 대규모 언어모델의 지식 한계를 보완하기 위해 제안된 대표적 접근이다[1]. RAG는 외부 지식베이스에서 관련 문서를 검색하여 모델 입력에 포함시킴으로써 사실성과 최신성을 확보할 수 있으나, 기존 RAG는 선형 파이프라인 구조에 기반해 복합 질의 처리에서 정보 누락과 오류 전파 문제가 발생한다[2][5][6].

이를 개선하기 위해 Chain-of-Retrieval (CoR-RAG)이 제안되었다. CoR-RAG는 원 질의를 순차적 보조 질문으로 분해하여 단계별 검색을 수행해 커버리지를 확대하였으나, 초기 분해 단계에서 오류가 발생할 경우 최종 응답까지 전파되는 구조적 한계를 지닌다[4].

한편, Tree-of-Thought(ToT)는 LLM의 추론 능력 강화를 목적으로 제안된 기법이다. Yao et al.[9]은 ToT를 통해 다수의 추론 경로를 병렬적으로 탐색하고 백트랙킹을 통해 오류를 보완할 수 있음을 보였으며, Long[2]은 LLM을 활용한 ToT 유도 방안을 제시하여 탐색적 추론의 가능성을 확장하였다. 이러한 연구는 RAG에도 영향을 주어 트리 기반 검색 및 추론 프레임워크로 발전하였다.

구체적으로, Fatehkia et al.[11]의 **T-RAG**는 트리형 엔터티 계층 구조를 활용하여 검색 과정의 문맥적 풍부함을 확보하였고, Li et al.[12]은 **CFT-RAG**를 통해 대규모 데이터셋에서 엔터티 트리 탐색 효율성을 개선하였다. 또한, Yang과 Huang [7]의 **HiRMed**를 제안하여 의학적 검사 추천 문제에 트리 기반 RAG 구조를 적용하였다. 그러나 이들 연구는 정적 구조 반영에 머물렀으며, 복합 질의 처리에서 요구되는 동적 분기 생성 및 가지치기 최적화를 충분히 다루지 못하였다.

또한, Pezeshkpour et al.[8]의 **Insight-RAG**는 단순 정답 산출을 넘어 인사이트 생성과 해석 지원을 목표로 연구를 확장하였다. Sharma[5]는 최신 RAG 종합 서베이에서 구조적 멀티홉 추론과 동적 검색 최적화를 차세대 발전 과제로 제시하였다. 이러한 논의 속에서 본 연구는 **ToR-RAG**라는 새로운 접근을 통해 기존 한계를 극복하고자 한다.

2. Multi-Hop Question Answering and Task Definition

복합 질의응답(MultiHop QA)은 단일 문서에서 답을 찾는 기존 질의-응답과 달리, 여러 문서에서 분산된 증거를 결합하여 답을 산출해야 하는 고난도의 문제이다[2][3][5]. MultiHop QA의 주요 특징은 1) 복수 문서 간 연결 관계를 추론해야 하며, 2) 중간 추론 단계에서 핵심 단서를 포착해야 하고, 3) 증거가 불충분할 경우 오류가 쉽게 발생한다는 점이다.

본 연구에서 정의하는 문제는 다음과 같다.

- **입력(Input):** 자연어로 표현된 복합 질의 Q
- **출력(Output):** 단답형 응답 A (Yes/No, 짧은 구절 (short span answer), 숫자)
- **목표(Objective):** 질의를 적절히 분해하여 검색을 수행하고, 검색된 증거를 통합하여 정확한 응답을 도출하는 것

이를 위해 본 연구가 제안하는 ToR-RAG는 기존 선형 파이프라인의 한계를 극복하고자, 질의를 WH-형 이진 분기(WH-binary decomposition)로 분해하고 동적 탐색과 가지치기를 통해 효율적이고 정확한 멀티홉 질의응답을 지원한다.

3. Dataset: MultiHop-RAG

본 연구의 실험에는 Tang & Yang[2]이 제안한 MultiHop-RAG 데이터셋을 사용하였다. 이 데이터셋은 총 2,556개의 질의로 구성되며, 각 질의에 대한 증거는 2~4개의 문서에 분산되어 있다. 또한 MultiHop-RAG는 단순한 질의-응답 구조가 아니라 지식 베이스, 멀티홉 질의 컬렉션, 정답(Label), 그리고 이에 대응하는 증거 세트로 구성되어 있어, RAG 시스템의 복합 질의 처리 능력을 평가하는 데 최적의 자원이다. Tang & Yang[2]은 기존 RAG 시스템들이 MultiHop-RAG에서 멀티홉 질의에 대한 증거 검색 및 정답 도출 성능이 충분하지 못한 것을 실험적으로 보고하였으며, 이는 해당 벤치마크의 중요성을 더욱 부각시킨다. 이와 같은 구성 덕분에, MultiHop-RAG는 다중 문서 기반 연결 추론, 노이즈 문서 포함 환경, 정량적 평가 지표(EM 및 F1) 적용 가능성 등을 평가하기에 매우 적합하다.

실험 데이터셋의 주요 특징은 다음과 같다.

- **복합성(Complexity):** 각 질의는 최소 2개 이상의 문서 간 연결 관계 추론을 요구하며, 중간 단서를 올바르게 식별해야 최종 응답에 도달할 수 있다.
- **노이즈(Noise):** 관련 없는 문서가 검색 결과에 포함될 수 있어, 모델이 정확한 증거를 선택하고 불필요한 정보를 배제하는 능력을 평가할 수 있다.
- **평가 용이성(Evaluation):** 응답이 단답형 (Yes/No, 짧은 구절, 숫자)으로 구성되어 있어, Exact Match (EM)와 F1 Score 같은 정량적 지표를 적용하기 용이하다.

따라서 본 연구에서는 MultiHop-RAG를 활용하여 ToR-RAG의 질의 분해 및 분기 기반 탐색 효율성을 검증한다.

4. Evaluation Metrics

본 연구에서는 질의-응답(QA) 태스크에서 널리 사용되는 Exact Match(EM)와 F1 Score 두 가지 지표를 사용하여 성능을 평가하였다[13].

- **Exact Match (EM):** 모델의 최종 응답이 정답과 완전히 일치할 경우 1점을 부여하고, 그렇지 않으면 0을 부여한다. Yes/No 및 단답형 QA에서 직관적이고 엄격한 지표로 활용된다.
- **F1 Score:** 모델 응답과 정답을 토큰 단위로 분해한 뒤, 정밀도(Precision)와 재현율(Recall)의 조화 평균을 계산한다. 부분적으로 일치하는 경우까지 반영할 수 있어 EM보다 유연한 평가가 가능하다.

두 지표를 병행 사용함으로써, ToR-RAG가 단순한 정답 일치도뿐 아니라 부분적 증거 포착 능력에서도 개선을 보이는지 종합적으로 확인할 수 있다.

5. Background Theory

본 연구는 Tree-of-Thought(ToT) 기반 추론, 질의 분해(Query Decomposition), 그리고 검색 모델(Retrieval Models)의 세 가지 핵심 개념을 바탕으로 한다.

5.1 Tree-of-Thought (ToT)

Tree-of-Thought(ToT)는 대규모 언어모델의 추론 능력을 확장하기 위해 제안된 프레임워크로, 복수의 사고 경로(thought paths)를 동시에 탐색할 수 있도록 한다[9]. 기존 Chain-of-Thought(CoT)가 단일 선형 경로에 따라 추론을 진행하는 반면, ToT는 트리 구조를 활용하여 병렬

적인 추론을 수행한다. 이를 통해 오류가 발생한 분기는 국소화되어 다른 경로 탐색에 영향을 주지 않으며, 백트래킹(backtracking)을 통해 대체 경로를 선택할 수 있다. Yao et al.[9]은 ToT를 LLM 추론을 탐색(search) 문제로 일반화할 수 있음을 보였으며, 이러한 구조적 특징은 멀티홉 질의와 같이 다양한 증거 결합이 필요한 문제에서 높은 견고성을 제공한다.

5.2 Query Decomposition

복합 질의 처리를 위해서는 원 질의를 적절히 분해하는 과정이 필수적이다. Query Decomposition은 주어진 질의를 여러 하위 질의(sub-queries)로 나누어 단계적 검색과 추론을 가능하게 하는 기법이다[4][14]. Sharma[5] 또한 최신 RAG 서베이에서 효과적인 질의 분해가 멀티홉 QA 성능의 핵심임을 지적하였다.

- **Chain 기반 분해(CoR-RAG):** 원 질의를 순차적 하위 질의로 변환하여 선형적으로 처리한다. 커버리지를 확대할 수 있으나, 초기 단계 오류가 최종 응답까지 전파되는 문제가 있다.
- **Tree기반 분해(ToR-RAG):** 원 질의를 이진 혹은 다분기 구조로 분해하여 병렬 탐색을 수행한다. 커버리지와 정밀도의 균형을 확보할 수 있으며, 가지치기를 통해 비효율적 경로를 제거할 수 있다.

본 연구에서는 LLM-as-a-Judge 평가 모듈을 활용하여 생성된 분기의 유효성을 검증함으로써, 질의 분해 품질이 검색 및 응답 정확도에 미치는 영향을 최소화한다.

5.3 Retrieval Models

검색 모델(Retrieval Models)은 RAG 구조에서 핵심 역할을 수행한다. 일반적으로 문서 집합은 벡터 임베딩(embedding)으로 표현되며, 질의와의 유사도를 기반으로 관련 문서를 검색한다. 대표적인 벡터 검색 엔진에는 FAISS, Milvus, Weaviate 등이 있다. 본 연구에서는 FAISS(HNSW 기반 인덱싱)를 사용하여 대규모 문서 집합에서도 높은 검색 효율성을 확보하였다.

또한, 검색 단계에서는 단순 유사도 기반 검색 대신 Maximal Marginal Relevance(MMR) 전략을 적용하였다[1]. MMR은 질의와의 관련성을 보장하면서도 검색된 문서 간 중복을 줄여, 정보 다양성과 정확성을 동시에 확보할 수 있도록 한다.

III. The Proposed Scheme

본 장에서는 본 연구에서 제안하는 Tree-of-Retrieval 기반 RAG (ToR-RAG)의 구조와 동작 과정을 기술한다. 제안된 방식은 ①질의 분해, ②검색 및 부분 응답 생성, ③분기 평가, ④가지치기, ⑤최종 응답 통합의 다섯 모듈로 구성된다.

1. Overall Architecture

ToR-RAG의 전체 구조는 Fig.1에 나타낸 바와 같이 다섯 개의 모듈로 구성된다.

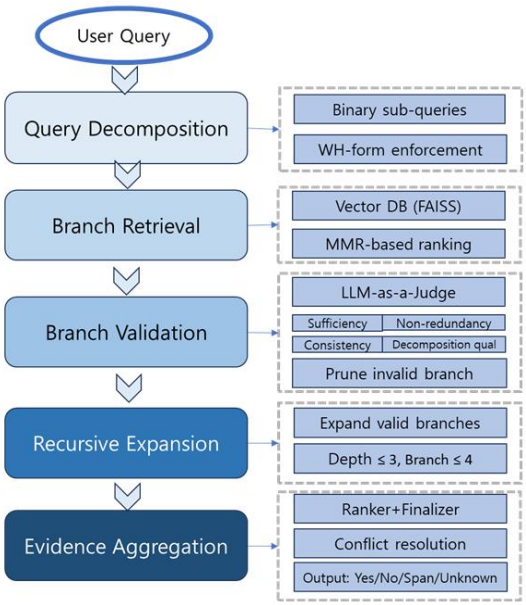

Fig. 1. ToR-RAG Workflow and Modules

1) **질의 분해 모듈(Query Decomposer)**
입력된 원 질의를 WH-형 이진 하위 질의로 분해하여 탐색 공간을 생성한다. 이는 기존 CoT의 선형 분해 방식과 달리, 병렬 탐색을 가능하게 한다[4][9].

2) **검색·응답 모듈(Retriever & Answer Generator)**
각 하위 질의에 대해 검색을 수행하고, 검색된 문서로부터 부분 응답(candidate answer)을 생성한다. 이 단계는 RAG의 핵심 단계로, 검색 품질이 최종 응답 정확도에 직접적인 영향을 미친다[1].

3) **분기 평가 모듈(LLM-as-a-Judge)**
: 분기별로 생성된 부분 응답의 신뢰도를 평가한다. 최근 연구에서는 LLM을 평가자로 활용하는 다양한 접근이 다수 제안되었으며[5], 본 연구에서도 이를 적용하여 유효하지 않은 분기를 식별한다.

4) **가지치기 및 확장 모듈(Pruner & Expander)**
평가 점수가 낮은 분기는 가지치기(pruning)하고, 점수가 높은 분기만을 재귀적으로 확장(expansion)한다. 이를 통해 탐색 효율성과 정확성을 동시에 확보한다.

5) **최종 응답 합성 모듈(Finalizer)**
최종적으로 리프 노드에서 수집된 응답과 증거를 통합하여 일관성 있는 최종 응답을 산출한다.

2. Query Decomposition

입력 질의 Q는 Query Decomposer 모듈을 통해 두 개의 독립적이고 상보적인 하위 질의(Q_1,Q_2)로 분해된다. 본 연구에서는 **LLM 기반 자동 분해 방식**을 채택하였으며, LLaMA 3.1 8B 모델을 활용하여 원 질의를 WH-question (Who, What, Where, When, Why, How, Which)형식으로 변환한다.

- 분해 기준: 생성된 하위 질의는 반드시 (1)원 질의의 의미적 범위를 충분히 포괄하고, (2) 중복되지 않고 독립적이어야 한다.
- 다음은 하위 질의 분해 결과를 평가하기 위한 판정 프롬프트 예시이다.

```
template_judge = """
You are a "decomposition judge."
TASK: Evaluate whether two sub-questions are a good Tree-of-Thought style decomposition of the original question for RAG.
RUBRIC (1 point each):
1) WH-form: Each is a WH-question (who/what/when/where/why/how/which).
2) Specific & measurable: Answerable with concrete facts/metrics (not opinion).
3) Relevant: Both directly address the original question.
4) Non-overlapping: They are semantically independent (neither presupposes the other's answer).
5) Completeness: Together they are sufficient to answer the original question.
OUTPUT RULES:
- Print exactly one line.
- Format: VERDICT=<VALID|INVALID>; SCORE=<0-5>; REASONS=<comma-separated tags or empty>
- Always respond in English.
Original question:
{question}
Sub-question 1:
{sub1}
Sub-question 2:
{sub2}
"""
prompt_judge =
 ChatPromptTemplate.from_template(template_judge)
```

Fig. 2. Query decomposition prompt (LLM-as-a-Judge)

- 프롬프트 제약: 프롬프트는 "주어진 질의를 두 개의 WH-형식 하위 질의로 나누라"는 제약이 포함되며, 질문이 충분히 복잡하지 않은 경우 불필요한 세분화를 피하고 최소 단위인 이진 분기를 유지한다.
- 재귀 분해: Depth 제한 (본 연구에서는 최대 3)을 두어, 1차 분해에서 충분히 세분화되지 않은 경우 다음 단계에서 추가 분해가 가능하도록 설계하였다.
- 수식적 정의: 분해 과정은 다음과 같이 정의된다.

$$Decompose(Q) \rightarrow \{Q_1, Q_2\}$$

이와 같이 Query Decomposition은 LLM 기반 이진 분기, WH-제약 프롬프트, 재귀적 확장을 통해 ToR-RAG의 핵심적인 구조를 뒷받침한다.

3. Branch Retrieval and Partial Answer Generation

각 하위 질의는 임베딩 벡터로 변환되어 FAISS 기반 인덱스에서 관련 문서 스니펫을 검색한다. 검색된 스니펫은 LLM 입력에 포함되어 부분 응답 (A_1, A_2)를 생성한다. 검색 단계에서는 단순 유사도 기반 검색 대신 Maximal Marginal Relevance (MMR)을 적용하여, 질의와의 관련성을 보장하면서도 검색된 문서 간 중복을 줄여 정보 다양성과 정확성을 동시에 확보하였다[15].

$$Retrieve(Qi) \rightarrow Di,\ Generate\ (Qi, Di) \rightarrow Ai$$

4. Branch Evaluation (LLM-as-a-Judge)

분기 평가는 LLaMA 3.1 8B 모델을 기반으로 수행되며, Judge 프롬프트는 "주어진 하위 질의(Q_1,Q_2)와 부분 답(A1, A2)의 조합이 원 질의(Q)를 해결하는 데 충분한가?"라는 질문을 중심으로 설계하였다.

- 평가 기준:
  1. **충분성(Sufficiency):** Q_1,Q_2의 결합이 원 질의를 충분히 커버하는가?
  2. **비중복성(Non-redundancy):** 두 하위 질의가 중복되지 않고 독립적인가?
  3. **정합성(Consistency):** 원 질의와 하위 질의 간 논리적 충돌은 없는가?
  4. **분해 품질(Decomposition Quality):** 과소/과잉 분해 없이 균형적인가?
- 점수 방식: 각 기준에 대해 0~5점의 점수를 부여하고, 종합 평균이 임계치 θ =0.4 이상일 경우 해당 분기를

유효하다고 간주한다.

- 재분기 루프: 임계치 미달 시 해당 노드는 재분해를 시도하며, 기여도가 낮은 하위 질의를 중심으로 새로운 쌍(Q_1', Q_2')를 생성하여 반복 평가한다.
  Max-Depth(3)에 도달하면 분해를 중단하고 리프 노드로 확정한다.

$$Judge\ (Q,\ Q_1,\ Q_2,\ A_1,\ A_2\ ) \rightarrow Score \in [0,5]$$

5. Pruning and Recursive Expansion

분기 점수가 임계치 θ 미만일 경우, 해당 분기는 가지치기(pruning)되어 더 이상 확장되지 않는다. 반면 점수가 임계치를 충족하면 새로운 노드로 확장(expansion)되며, 동일한 과정이 재귀적으로 반복된다. 트리 깊이(Depth)는 지수적 탐색 복잡도를 제어하기 위해 제한되며, 본 연구에서는 Depth=3일 때 가장 높은 성능을 기록하였다.

6. Evidence Aggregation

리프 노드에서 수집된 부분 답은 Retrieval MMR 점수(λ=0.75)에 따라 우선순위가 부여되며, 상위 k=5개의 문서 스니펫 (각 300 tokens, 총 결합 길이 1,500 tokens 이내)을 기반으로 최종 응답을 도출한다.

- 랭킹 기준: 임베딩 유사도와 MMR 다양성 점수를 반영하여 후보 증거를 정렬한다.
- 충돌 해결: 상충되는 증거가 존재할 경우, 1) 출처 신뢰도, 2) 다수성(majority voting), 3)문맥 일치도를 고려하여 최종 근거를 선택한다. 필요 시 “Unknown”으로 응답을 제한하여 불필요한 추측을 방지한다.
- 최종 합성: Finalizer는 단답형(Yes/No 또는 짧은 span) 출력을 강제하는 프롬프트를 사용하여 장황한 응답 생성을 억제한다.

최종 응답 A는 다음과 같이 정의된다.

$$Aggregate\ (\{A_1,\ A_2,\ ....,\ A_n\},\ \{Evidence\}) \rightarrow A$$

7. Complexity Considerations

트리 구조는 잠재적으로 지수적 복잡도를 가지므로, 본 연구에서는 ① 이진 분기 고정, ② Depth 제한 (Depth≤ 3), ③ 임계치 기반 가지치기를 통해 연산량을 제어하였다. 이러한 제약 조건은 실험 결과에서도 성능과 효율성 간의 균형을 확보하는 데 효과적임을 확인하였다.

IV. Experiments and Analysis

1. Experimental Setup

실험은 LangChain 프레임워크를 기반으로 구현되었으며, Table 1과 같이 동일한 환경에서 수행하여 제안 기법(ToR-RAG)의 성능을 검증하였다. 실험 데이터셋은 MultiHop-RAG 전체 2,556개의 질의를 사용하였으며, train/dev/test 분할 없이 Zero-shot 평가를 수행하였다. 이는 제안 기법의 구조적 효과를 직접적으로 검증하기 위한 설정이다.

공정 비교를 위해 모든 방법론은 다음과 같은 동일한 조건에서 수행하였다.

- LLM: LLaMA 3.1 8B (temperature는 0.2, short token limit 적용)
- 임베딩 모델: Qwen3-Embedding-0.6B (출력 차원 1024, 다국어 지원)
- 검색 엔진: FAISS(HNSW 인덱싱, MMR λ=0.75, k=5)
- 문서 단위: 스니펫 길이 300 tokens, 전체 결합 길이 1,500 tokens 제한
- 출력 포매팅: 최종 응답은 단답형 (Yes/No, short span)으로 강제

Table 1. Experimental Environment

category	Details
LLM	LLaMa 3.1 8B (Ollama기반, temperature=0.2)
Search Engine	FAISS (HNSW indexing)
Embedding Model	Qwen3-Embedding-0.6B (1024, multilingual support)
Pipeline Implementation	- OS: Ubuntu 22.04 LTS - HW: NVIDIA A6000 GPU - Framework: LangChain
Evaluation Metrics	Exact Match (EM), F1 score

2. Baselines

제안 기법 ToR-RAG는 다음과 같이 네 가지 방법론과 비교하였다.

1) **Non-RAG:** 외부 검색을 수행하지 않고 LLM 파라미터 지식만으로 답변을 생성
2) **Native-RAG:** 단일 질의-검색-응답의 선형 구조를 가지는 기본 RAG
3) **CoR-RAG:** 원 질의를 순차적 보조 질문으로 분해하여 단계적 검색을 수행하는 Chain-of-Retrieval 방식

4) **ToR-RAG:** 본 연구에서 제안하는 Tree-of-Retrieval 기반 검색 구조

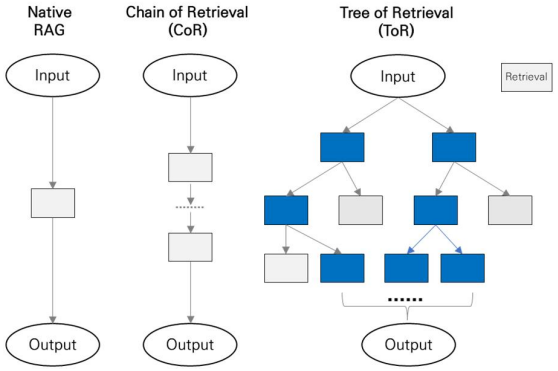

Fig. 3. Retrieval Architectures of baseline methods

3. Dataset

본 실험에서는 Tang & Yang[8]이 제안한 MultiHop-RAG 데이터셋을 사용하였다. 이 데이터셋은 다문서 기반 질의-응답을 목적으로 설계되었으며, 복수의 문서에 분산된 증거를 결합해야만 정답에 도달할 수 있는 복합질의(complex queries)를 포함한다.

기존의 HotpotQA등의 복합 질의 데이터셋이 추론 단계 중심 평가에 초점을 두었던 것과 달리, MultiHop-RAG는 검색 단계와 추론 단계의 통합 성능을 검증하는 데 중점을 둔다.

Table 2. Benchmark Dataset overview

category	Details
Name	MultiHop-RAG
Total Samples	2,556 (queries)
Document sources	Wikipedia, external knowledge based DB
Query Types	Yes/No, 단답형(숫자/엔터티)
Document Length	600 tokens
Supporting Documents per Query	2~4 (multi-hop reasoning required)

본 데이터셋의 특징은, ① 다문서 기반 질의 포함, ② 정답은 Yes/No 및 단답형 구조, ③ 노이즈 문서 포함으로 증거 선택 성능이 중요하다는 점이다.

4. Results

(1) 방법론별 성능 비교

ToR-RAG는 EM과 F1에서 각각 +6.42, +6.04의 성능 향상을 달성하여 기존 방법론 대비 우수한 성능을 보였다.

Table 3. Performance Comparison of Methods

Method	MultiHop-RAG		Remarks
	EM	F1	
Non-RAG	49.06	50.11	
Native-RAG	53.83	55.92	
CoR-RAG	54.26	55.14	
ToR-RAG	**60.25**	**61.18**	Depth=3

또한 Non-RAG 대비 ToR-RAG는 **EM +11.19, F1 +11.07**의 개선을 보여, 질문 분해 및 구조적 검색 전략이 단순한 LLM 응답보다 효과적임을 입증하였다. (Fig.3)

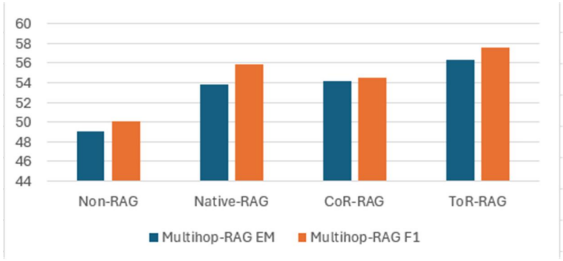

Fig. 4. Performance Comparison by Methods

(2) Depth에 따른 성능 분석

본 실험 결과, Depth=3에서 최적 성능을 기록하였으며, Depth=4에서는 입력 토큰 초과와 과도한 분기 확장으로 인해 성능이 급격히 저하되었다. 이는 트리 기반 검색에서 깊이 제한이 성능과 효율성간 균형 유지에 필수적임을 보여준다.

Table 4. Performance Comparison by Depth

Depth	Performance		Remarks
	EM	F1	
1	54.58	55.52	
2	56.30	57.61	
3	**60.25**	**61.18**	Best
4	26.13	30.13	

5. Analysis

실험 분석에서 ToR-RAG는 Non-RAG, Native-RAG, CoR-RAG 대비 일관된 성능 향상을 보였다.

- **깊이 실험(Depth study):** Depth 1~4를 비교한 결과, Depth=3에서 최적의 성능을 기록하였다.
- **Judge on/off:** 제안 기법의 핵심 모듈이므로 별도 ablation은 수행하지 않았으나, baseline (Non-RAG, Native-RAG)이 사실상 Judge off 조건에 해당한다.
- **분해 품질·임계치 스윕:** 본 연구는 방법론 제안 및 구조적 효과 검증에 중점을 두었으므로, 관련 실험은 후속 연구로 남긴다.

- **설명 가능성(Explainability):** 본 연구에서의 설명가능성은 xAI 수준의 정량 지표 검증을 의미하지 않는다. 대신, 분기된 하위 질의, 검색 증거 및 부분 답이 명시적으로 기록·저장됨으로써 응답의 신뢰도와 근거 정확도를 강화하는 효과로 정의하였다.

실험 결과를 종합하면 다음과 같다.

첫째, ToR-RAG는 트리 구조를 통해 다양한 경로를 탐색하면서도 가지치기를 통해 커버리지와 정밀도 균형을 확보하였다.

둘째, CoR-RAG 대비 오류 국소화 효과를 통해 초기 단계 오류가 최종 응답에 미치는 영향을 줄였다.

셋째, 추론 경로가 트리 구조로 명시적으로 표현되므로 응답의 신뢰도와 근거 정확도를 높였다.

따라서 ToR-RAG는 복합 질의 처리에서 기존 접근법보다 더 안정적이고 신뢰할 수 있는 성능을 제공한다.

V. Conclusions

본 연구에서는 기존 RAG의 선형 파이프라인 구조가 가지는 한계를 극복하기 위해 Tree-of-Retrieval 기반 RAG (ToR-RAG)를 제안하였다. 제안 기법은 질의를 이진 분기로 분해하고, 각 분기에서 검색과 부분 응답 생성을 수행한 뒤, LLM-as-a-Judge를 통해 품질을 평가하고 가지치기를 적용한다. 이러한 구조는 정보 커버리지를 확보하면서 오류 전파와 중복 문제를 억제할 수 있었다. MultiHop-RAG 데이터셋을 이용한 실험 결과, ToR-RAG는 Non-RAG, Native-RAG, CoR-RAG 대비 Exact Match(EM) +6.42, F1 +6.04의 성능 향상을 달성하였다. 또한 Depth=3에서 최적 성능을 기록하였으며, Depth=4에서는 성능이 저하되어 트리 기반 탐색에서 깊이 제한이 필수적임을 확인하였다.

본 연구의 기여는 다음과 같다.

1. **학문적 측면:** 트리 기반 검색 구조를 적용한 새로운 RAG 패러다임을 제시하고, 동일한 LLM·검색 환경에서 구조적 멀티홉 추론의 효과를 실험적으로 검증하였다.
2. **실무적 측면:** 정책 분석, 의료 의사결정, 금융 리스크 평가 등 고신뢰성이 요구되는 응용 분야에 ToR-RAG의 적용 가능성을 제시하였다.

그러나, 본 연구는 영어 기반 데이터셋에 한정되었으며, 법률·의료 등 특수 도메인에 대한 검증은 수행하지 않았다. 또한 트리 탐색 과정에서 계산 복잡도가 증가하는 한계가 존재한다. 설명 가능성 또한 정량 지표 기반 검증은 수행하지 않았으며, 본 논문에서의 "설명 가능성"은 분기별 증거와 응답 생성 과정을 명시적으로 기록하여 응답 신뢰도와 근거 정확도를 강화하는 수준으로 정의한다.

향후 연구 방향은 다음과 같다.

- **강화학습 기반 분기 최적화:** PPO, DPO 등 강화학습 기법을 적용하여 분기 선택 정책을 자동 최적화
- **도메인 확장:** 법률, 의료, 과학등 다양한 전문 영역에서 성능 검증
- **멀티모달 확장:** 텍스트, 이미지, 수치 데이터를 포함하는 복합 데이터 처리
- **에이전트 결합:** Agentic RAG와 통합하여 자율적 질의 분해 및 검색 실행 지원
- 추가 검증: 분해 임계치(θ) 스윕, Judge on/off, supporting-facts F1등 정량적 설명 가능성 지표 도입

이상의 결과를 통해, ToR-RAG는 복합 질의 처리에서 기존 RAG 대비 성능과 응답 신뢰도를 동시에 개선하였으며, **차세대 RAG 연구의 유효한 대안**으로서 학문적·실무적 의의를 가진다.

REFERENCES

[1] Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems 33 (NeurIPS 2020), Dec. 2020.

[2] Tang and Yang, "MultiHop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries," arXiv:2401.15391v1, Jan. 2024.

[3] Geva et al., "Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies," arXiv:2101.02235v1, Jan. 2021.

[4] Wang et al., "Chain-of-Retrieval Augmented Generation," arXiv:2501.14342v2, Jan. 2025.

[5] Sharma, "Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers," arXiv:2506.00054v1, May 2025.

[6] Han et al., "Fine-grained Knowledge Enhancement for Retrieval-Augmented Generation," Findings of the Association for Computational Linguistics (ACL 2025), Paper 10031-10044,

July 2025.

[7] Yang et al., “A Tree-based RAG-Agent Recommendation System: A Case Study in Medical Test Data,” arXiv:2501.02727v1, Jan. 2025.

[8] Pezeshkpour et al., “Insight-RAG: Enhancing LLMs with Insight-Driven Augmentation,” arXiv:2504.00187v1, Mar 2025.

[9] Yao et al., “Tree of Thoughts: Deliberate Problem Solving with Large Language Models,” arXiv:2305.10601v2, Dec 2023.

[10] Long, “Large Language Model Guided Tree-of-Thought,” arXiv:2305.08291v1, May 2023.

[11] Fatehkia et al., “T-RAG: Lessons from the LLM Trenches,” arXiv:2402.07483v2, Jun 2024.

[12] Li et al., “CFT-RAG: An Entity Tree Based Retrieval Augmented Generation Algorithm with Cuckoo Filter,” arXiv:2501.15098v1, Jan. 2025.

[13] Rajpurkar et al., “SQuAD:100,000+ Questions for Machine Comprehension of Text. In Proceedings of EMNLP, pp.2383-2392, 2016

[14] Khot et al., “Text Modular Networks: Learning to Decompose Tasks in the Language of Existing Models”, ACL, Jun 2021

[15] Gao et al., “VRSD: Rethinking Similarity and Diversity for Retrieval in Large Language Models, arXiv:2407.045735v2, Nov 2024.

[16] Lee, “Transforming Questions and Documents for Semantically Aligned Retrieval-Augmented Generation,” arXiv:2508.09755v1, Aug. 2025.

[17] Agrawal et al., “SCMRAG: Self-Corrective Multihop Retrieval Augmented Generation System for LLM Agents,” ACM Digital Library, pp. 50-58, May 2025.

[18] Chen et al., “HiQA: A Hierarchical Contextual Augmentation RAG for Multi-Documents QA,” arXiv:2402.01767v2, Sep 2024.

[19] Wang et al., “Cross-Granularity Hypergraph Retrieval-Augmented Generation for Multi-hop Question Answering,” arXiv:2508. 11247v1, Aug. 2025.

[20] Siriwardhana et al., “Improving the Domain Adaptation of Retrieval Augmented Generation(RAG) Models for Open Domain Question Answering,” arXiv:2210.02627v1, Oct. 2022.

[21] Li et al., “Tree of Reviews: A Tree-based Dynamic Iterative Retrieval Framework for Multi-hop Question Answering,” arXiv:2404.14464v1, Apr. 2024.

[22] Zhang et al., “RATT: A Thought Structure for coherent and Correct LLM Reasoning,” arXiv:2406.02746v1, June 2024.

[23] Shi et al., “Generate-then-Ground in Retrieval Augmented Generation for Multi-hop Question Answering,” Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024, Long Papers), pp. 7339-7353, August 2024.

[24] Zhang et al., “Hierarchical Retrieval-Augmented Generation Model with Rethink for Multi-hop Question Answering,” arXiv:2408.11875v1, Aug 2024.

[25] Ramlochan, “Tree of Thought Prompting – Walking the Path of Unique Approach to Problem-Solving,” Prompt Engineering and AI Institute, Jun 2023.

[26] Poliakov and Shvia, “Multi-Meta-RAG: Improving RAG for Multi-Hop Queries using Database Filtering with LLM-Extracted Metadata,” arXiv:2406.13213v2, Aug. 2024.

[27] F22Labs, “What is Multi-Step RAG (A Complete Guide),” Retrieved from: https://f22labs.com., Sep 2025.

[28] Zhang et al., “SiReRAG: Indexing Similar and Related Information for Multihop Reasoning,” arXiv:2412.06206v2, Apr 2025.

[29] Zhu et al., “Knowledge Graph-Guided Retrieval augmented Generation,” arXiv:2502.06864v1, Feb 2025

Authors

Hee-Kyong Yoo received the B.S.degree in Law from Ehwa Womans University, Korea in 1993, and M.B.A. degree in Business Administration from Korea University, Korea, in 2003.

She is a Ph.D. candidate in Convergence AI Engineering at Hoseo University, Korea. Since 2023. She has been the founder and CEO of Data Science Lab, LTD., a venture company specializing in AI and Big Data analytics. Her research interests include retrieval-augmented generation (RAG), Agent-based generative AI systems, Business Intelligence, and AI policy and strategy.

Nammee Moon received B.S., M.S., and ph.D degrees in School of Computer Science and Engineering from Ewha Womans University in 1985, 1987 and 1998, respectively. She served as an assistant professor at Ewha Womans University from 1999 to 2003. From 2003 to 2008, she is a professor of Department Digital Media, Graduate School of Seoul Venture Information. Since 2008, she is currently a professor in the Department of Computer Science and Engineering, Hoseo University. she is current research interests include Social Learning, HCI and User Centric Data, Big-data Processing and Analysis.