Speculative RAG의 효율적 초안 생성을 위한 다중 클러스터링과 유사도 기반 샘플링 기법

안규원O, 김영재†
서강대학교 AI·SW대학원
kyuwon.ahn@cj.net, youkim@sogang.ac.kr

Multi-Clustering and Similarity-based Sampling for Efficient Draft Generation in Speculative RAG

Kyuwon AhnO, Youngjae Kim†
Graduate School Of AI·SW, Sogang University

요 약

Retrieval-Augmented Generation(RAG)[1]는 외부 문서를 활용해 언어모델의 응답 품질을 향상시키지만, 검색된 다수 문서를 모두 입력에 포함시킬 경우 토큰 수 증가로 인한 추론 지연이 발생하고, Lost-in-the-Middle[2] 현상으로 인해 긴 컨텍스트 내에서 중요 정보를 놓치는 성능 감소가 발생한다. Speculative RAG[3]는 문서를 클러스터링하고 샘플링하여 여러 초안을 생성한 후 최적의 초안을 선택하는 방식으로 이를 해결하였으나, 고정된 군집 수 설정과 무작위 샘플링의 한계가 존재한다. 본 연구는 이를 개선하기 위해 계층적 클러스터링[4]과 그래프 기반 스펙트럴 클러스터링[5], 유사도 기반 샘플링을 제안한다. 실험 결과, 제안 방법은 baseline 대비 더 적은 수의 초안 생성으로 최대 9.0%p의 정확도 향상을 달성하였다.

1. 서 론

최근 대규모 언어모델(LLM)이 빠르게 발전함에 따라, Retrieval-Augmented Generation (RAG) 기법은 외부 지식을 활용해 응답 품질을 향상시키는 핵심 기법으로 자리잡았다[1]. RAG는 벡터 데이터베이스에서 관련 문서를 검색해 프롬프트를 증강하는데, 다수의 문서로 인해 길이가 크게 증가하여 추론 지연과 정확도 감소 문제가 발생한다[2].

이를 완화하기 위해 FILCO[6]는 관련성이 낮은 문서를 필터링하고, RECOMP[7]는 문서를 요약·압축하여 토큰 수를 감소시켰다. 그러나 이러한 접근법들은 필터링과 압축 과정에서 정보 손실로 인해 최종 생성 품질이 저하될 수 있다는 한계를 지닌다.

Speculative RAG[3]는 K-means 클러스터링으로 검색 문서를 군집화하고, 각 군집에서 무작위 샘플링한 문서 서브셋으로 여러 초안을 병렬 생성한 후 검증기가 최적 초안을 선택한다. 이는 정보 손실 없이 입력 토큰 수를 줄이면서도 품질을 확보하여 PubHealth 데이터셋에서 정확도 12.97%p 향상과 지연시간 50.83% 감소를 달성했다.

*본 연구는 정부(과학기술정보통신부)의 재원으로 한국연구재단(RS-2025-00564249)의 지원을 받아 수행되었다.
† Corresponding author

Speculative RAG에는 다음과 같은 개선 가능성이 존재한다. 첫째, K-means는 거리 기반으로 군집 내 분산을 최소화하여 구형 군집에 적합하며, 고정된 군집 수는 불규칙한 형태의 실제 문서 데이터에는 최적의 군집화가 어렵다. 둘째, 무작위 샘플링으로 인해 중요 문서가 누락될 수 있다.

본 연구에서는 이를 극복하기 위해 (1) 계층적 클러스터링[4]과 (2) 그래프 기반 스펙트럴 클러스터링[5]을 적용하며, (3) 질의-문서 임베딩 벡터의 코사인 유사도가 최대인 문서를 선택하는 유사도 기반 샘플링을 도입한다. 또한 실루엣 스코어[8]를 활용하여 클러스터링 품질을 평가하고 동적으로 최적 k 값을 결정한다. 실험 결과, 제안하는 방법은 Speculative RAG 대비 정확도에서 최대 9.0%p, LLM Judge 평가에서 최대 6.4%p의 성능 향상을 달성했다.

2. 연구배경 및 선행연구

2.1 Retrieval-Augmented Generation

RAG[1]는 그림 1과 같이 크게 세 가지 핵심 구성 요소로 이루어진다. 첫 번째는 검색(Retrieval) 단계로, 사용자의 질의를 임베딩 벡터로 변환하여 벡터 데이터베이스에서 의미적으로 유사한 문서들을 검색한다. 두 번째는 증강(Augmentation) 단계로, 검색된 문서들을 프롬프트

템플릿에 통합하여 컨텍스트를 구성한다. 세 번째는 생성(Generation) 단계로, 증강된 프롬프트를 언어모델에 입력하여 최종 응답을 생성한다.

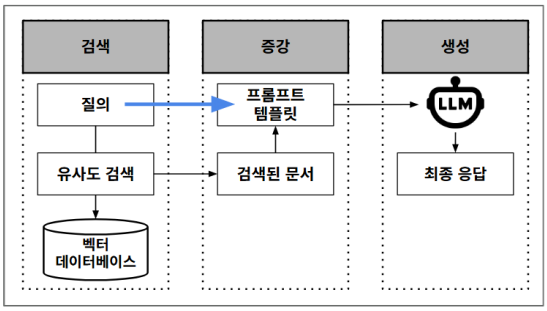

그림 1 RAG 구조

2.2 Lost-in-the-Middle

언어모델은 긴 컨텍스트를 처리할 때 'Lost-in-the-Middle' 현상을 보인다[2]. 다양한 크기와 아키텍처의 언어모델 모두 컨텍스트의 시작과 끝 부분 정보는 효과적으로 활용하지만 중간 위치의 정보는 상대적으로 무시되는 경향이 있다.

이는 RAG 시스템에서 단순히 많은 문서를 검색하는 것이 성능을 보장하지 않으며, 중요한 정보가 중간에 위치할 경우 오히려 성능 저하를 야기할 수 있음을 시사한다.

2.3 Speculative RAG

Speculative RAG[3]의 구체적인 동작 과정은 그림 2와 같다.

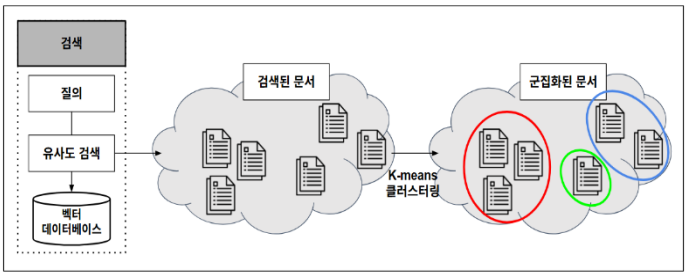

그림 2 Speculative RAG의 클러스터링 과정

검색된 n개 문서를 K-means 클러스터링을 통해 k개의 군집으로 분할한다. 각 군집은 의미적으로 유사한 문서들의 집합으로, 서로 다른 관점의 정보를 담고 있다. 다음으로, 그림 3과 같이 각 군집에서 무작위로 하나의 문서를 샘플링하여 m개의 서브셋을 생성한다.

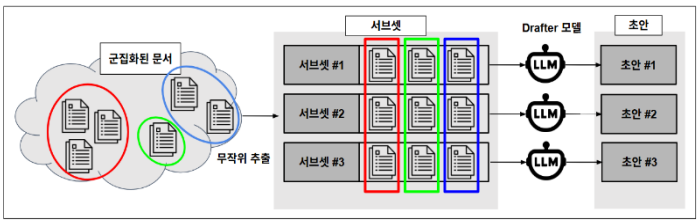

그림 3 서브셋과 초안 생성

Drafter 모델은 각 서브셋을 입력으로 받아 병렬적으로 m개의 초안을 생성한다. 각 초안은 답변과 근거의 쌍으로 구성되며, 서로 다른 문서 조합에 기반하여 다양한 관점의 응답을 제공한다. 이때 병렬 처리를 통해 전체 생성 시간을 단축할 수 있으며, 작은 모델을 사용함으로써 계산 비용을 절감한다. 각 초안에 대해 Verifier 모델은 아래의 세 가지 평가 점수를 계산한다.

$$\rho_{Draft,j} = P(\beta_j | Q, d_{j_1}, \ldots, d_{j_k}) + P(\alpha_j | Q, d_{j_1}, \ldots, d_{j_k}, \beta_j)$$

$$\rho_{Self\text{-}contain} = P(\alpha, \beta | Q)$$

$$\rho_{Self\text{-}reflect} = P(\text{"}Yes\text{"} | Q, \alpha, \beta, R)$$

첫 번째 점수($\rho_{Draft,j}$)는 질의(Q)와 검색 문서(d)로부터 답변(α)과 근거(β)가 자연스럽게 도출되는 정도를 평가한다. 두 번째 점수($\rho_{Self\text{-}contain}$)는 질의만으로도 답변이 생성될 수 있는 일관성을 측정한다. 세 번째 점수($\rho_{Self\text{-}reflect}$)는 모든 정보를 종합했을 때 답변의 타당성을 평가한다. 이 세 점수를 종합하여 가장 신뢰할 수 있는 초안을 최종 답변으로 선택한다.

3. 연구방법

3.1 다중 클러스터링과 실루엣 스코어

본 연구에서는 Speculative RAG의 K-means 클러스터링을 확장하여 계층적 클러스터링(Hierarchical)[4]과 스펙트럴 클러스터링(Spectral)[5]을 추가로 적용한다. 계층적 클러스터링은 각 문서를 개별 군집으로 시작한 후 가장 유사한 군집들을 반복적으로 병합하여 계층적 구조를 형성한다. 스펙트럴 클러스터링은 데이터를 그래프로 표현하고, 그래프 라플라시안(Laplacian)의 고유벡터를 활용하여 클러스터링을 수행하는 방법으로, 본 연구에서는 문서 간 공동 등장 개체(entity) 관계를 기반으로 그래프를 구성한다. 군집 수 k는 사전에 지정하지 않고, 군집 내 응집도와 군집 간 분리도를 평가하는 실루엣 스코어(Silhouette Score)[8]를 최대화하는 k를 각 클러스터링 기법마다 적응적으로 결정한다.

3.2 유사도 기반 샘플링

Speculative RAG는 각 군집에서 무작위로 문서를 선택하여 서브셋을 구성한다. 이는 다양한 관점의 초안을 생성할 수 있지만, 질의와 관련성이 높은 중요 문서가 누락될 수 있다. 본 연구에서는 질의-문서 간 유사도를 기반으로 한 결정론적 샘플링 방법을 제안한다.

4. 실험 및 평가

4.1 실험환경

본 연구는 Speculative RAG을 기반으로 다중 클러스터링과 유사도 기반 샘플링의 성능을 검증하였다. 선행연구의

instruction-tuned Mistral Drafter는 리소스 제약으로 Mistral v0.3 기반 프롬프트 엔지니어링으로 대체하였다. 평가는 정확도와 LLM Judge(GPT-4o-mini, Claude-3.5-Haiku)를 활용하였다. 실험의 세부 사항은 표 1과 같다.

표 1 실험 환경

항목	설정
컴퓨팅(런타임)	Google Colaboratory Pro+
컴퓨팅(벡터 DB)	AWS EC2 t2.xlarge(4Core, 16GB RAM)
GPU	NVIDIA A100 GPU VRAM 80GB
데이터셋	TriviaQA(563K)
벡터 DB	Milvus
임베딩(적재/검색)	Contriever-MS MARCO
임베딩(클러스터링)	InBedder-Roberta
LLM(Drafter)	Mistral-7B-Instruct-v0.3
LLM(Verifier)	Mistral-7B-v0.3

4.2 실험 결과 및 분석

클러스터링 수행 시간을 비교한 결과(그림 4), 문서의 수 n=10에서 K-means는 2.73ms, Hierarchical은 6.34ms, Spectral은 504.83ms를 기록하였다. n=50에서는 K-means가 3.47ms, Hierarchical은 48.48ms (약 14배), Spectral은 3541.98ms (약 1,021배)로, Spectral의 높은 수행 시간은 그래프 구축 과정에서 개체명 인식과 관계 추출을 수행하기 때문에 발생한다.

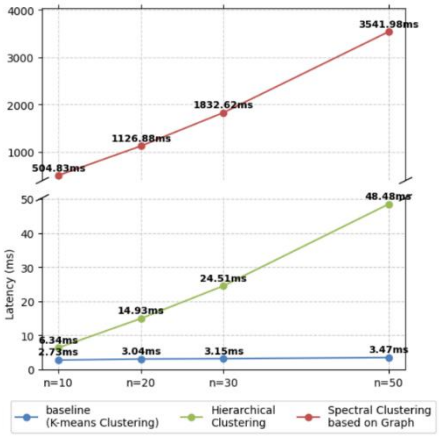

그림 4 클러스터링 알고리즘 별 수행 시간

샘플링 수행 시간 측면에서도(그림 5) 유사도 기반 샘플링은 무작위 샘플링 대비 추가 연산 비용이 발생하였다. n=2에서 무작위 샘플링은 0.26ms, 유사도 기반 샘플링은 1.31ms (약 5배)를 기록하였으며, n=15에서는 각각 1.23ms와 7.00ms (약 5.7배)로 격차가 확대되었다. 이는 각 군집 내에서 쿼리와의 코사인 유사도 계산 및 상위 m개 문서 선택 과정에서 추가 연산이 발생하기 때문이나, 샘플링 수행 시간은 수 ms 수준으로 클러스터링 수행 시간에 비해 상대적으로 미미하다.

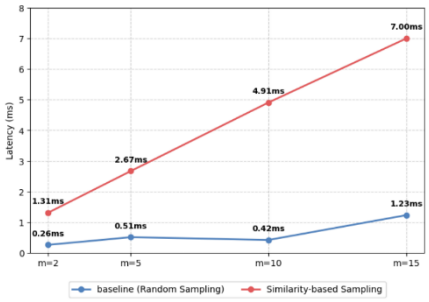

그림 5 샘플링 알고리즘 별 수행 시간

제안 방법과 베이스라인의 통합 성능 비교는 표 2와 같다.

표 2 최종 성능 비교

모델	정확도	LLM Judge (GPT-4o-mini)	LLM Judge (Claude-3.5-Haiku)
baseline(S-RAG)	50.0%	45.4%	58.4%
Hierarchical	56.8%	51.5%	62.3%
Spectral	59.0%	51.8%	64.2%

Spectral은 베이스라인(50.0%) 대비 정확도 9.0%p, LLM Judge 6.4%p, 5.8%p의 향상을 보였으며, Hierarchical도 각각 6.8%p, 6.1%p, 3.9%p 향상되어 두 클러스터링 기법 모두 효과적임을 확인하였다.

5. 결론

본 연구는 다중 클러스터링 기법과 유사도 기반 샘플링을 제안하였으며 베이스라인 대비 일관된 성능 향상을 확인하였다. 성능과 수행 시간 사이의 트레이드오프가 존재하므로 요구사항에 따라 적절한 기법을 선택할 수 있다.

참고문헌

[1] Patrick Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," NeurIPS, 2020.

[2] Nelson F. Liu et al., "Lost in the Middle: How Language Models Use Long Contexts," ACL, 2024.

[3] Zilong Wang et al., "Speculative RAG: Enhancing Retrieval Augmented Generation through Drafting," ICLR, 2025.

[4] S. C. Johnson, "Hierarchical Clustering Schemes," Psychometrika, 1967.

[5] Andrew Y. Ng, "On Spectral Clustering: Analysis and an Algorithm," NeurIPS, 2002.

[6] Zhiruo Wang et al., "Learning to Filter Context for Retrieval-Augmented Generation," arXiv:2311.08377, 2023.

[7] Fangyuan Xu et al., "RECOMP: Improving Retrieval-Augmented LMs with Compression and Selective Augmentation," arXiv preprint arXiv:2310.04408, 2023.

[8] Peter J. Rousseeuw, "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis," Journal of Computational and Applied Mathematics, vol. 20, pp. 53–65, 1987.