1. Retriever의 정의

리트리버는 이름 그대로 정보를 찾아와서 가져오는 존재입니다. RAG 시스템에서 사용자의 질문에 답변하기 위해 필요한 관련 문서나 지식 조각들을 데이터베이스에서 찾아내는 인터페이스를 말한다.

- 문서를 저장하는 기능보다는 어떻게 하면 더 정확하고 관련성 높은 정보를 검색해 올 것인가에 초점을 맞춘 도구

2. Retrievber의 핵심 기능

리트리버는 단순히 검색만 하는 것이 아니라, LLM이 더 똑똑하게 답변할 수 있도록 돕는 몇 가지 핵심 기능을 수행한다.

- 문서 검색: 수만 개의 텍스트 조각 중 사용자의 질문과 의미적으로 가장 유사한 상위 n개의 문서를 추출
- 검색 결과 확장: 사용자의 질문이 모호할 때, 질문의 의도를 분석하여 검색 범위를 넓히거나 다각화
- 컨텍스트 제공: 검색된 문서들을 LLM에 전달하여, 모델이 학습하지 않은 최신 정보나 내부 데이터를 바탕으로 답변할 수 있는 근거를 제공

3. 자료 中 주목할 부분

1. MultiQueryRetriever : "사용자의 불완전함 보완"(6)

-> 사용자가 대충 이야기를 해도 잘 알아들을 수 있도록 함

보통 사용자는 질문을 완벽하게 하지 않는다. "파이썬 알려줘"와 "파이썬 기초 문법 설명해줘"는 의미는 같지만, 컴퓨터가 계산하는 수학적 거리는 다를 수 있다.

- **문제 해결**: 단 한 번의 검색에 의존하는 '운 가이드' 방식에서 벗어남
- **핵심 가치**: LLM이 질문을 여러 관점에서 재해석해 5개의 그물을 던짐으로써, 검색의 재현율, 즉 관련 문서를 놓치지 않는 확률을 극대화

2. EnsembleRetriever : "검색의 균형과 신뢰도"(3)

-> 단순히 검색기를 두 개 쓰는 것이 아니라, 상호 보완적인 알고리즘을 결합해 검색 결

과의 신뢰성 확보

어떤 검색 엔진도 완벽하지 않다. 의미 중심의 벡터 검색은 고유 명사에 약하고, 키워드 중심의 'BM25'는 문맥 이해에 약하다.

- **문제 해결**: 특정 알고리즘의 편향성 때문에 발생하는 검색 실패를 방지
- **핵심 가치**: 서로 다른 두 검색 엔진의 장점을 합치는 **하이브리드 검색**을 구현합니다. RRF 알고리즘을 통해 두 검색 결과의 교집합과 상위권을 정교하게 재배치하여 정밀도를 높임

3. ContextualCompressionRetriever : "고효율·고품질의 답변"(2)

-> 검색된 날것의 데이터를 AI가 더 정확하고 빠르게 답변할 수 있도록 최적화하는 최종 필터

검색된 문서가 1,000자인데 정작 필요한 답은 중간의 50자뿐일 때가 많다. 무관한 텍스트를 LLM에 다 밀어 넣는 것은 비효율적이다.

- **문제 해결**: LLM의 입력 제한(Token Limit) 문제와 불필요한 정보로 인한 **할루시네이션** 현상을 해결
- **핵심 가치**: 검색된 결과물에서 불필요한 정보를 걸래내고 질문과 관련된 핵심 내용만 압축하여 LLM에 전달한다. 이는 비용절감과 답변 순도 향상에 도움을 줌