섹션 4: 벡터스토어 검색기 활용 및 성능 평가

- 패키지 설치
    - conda

```
pip install faiss-cpu rank_bm25 kiwipiepy openpyxl
```

    - poetry

```
poetry add faiss-cpu rank_bm25 kiwipiepy openpyxl
```

1. 벡터 저장소 (Vector Store):

1.1 Chroma: 벡터 데이터베이스 솔루션

- 사용자 편의성이 우수한 오픈소스 벡터 저장소
- **langchain-chroma** 패키지 설치
- 주요 사용방법:
    1. 벡터 저장소 초기화 (생성)
    2. 벡터 저장소 관리: 문서 추가, 변경, 삭제
    3. 문서 검색: 유사도 검색
    4. 벡터 저장소 로컬 저장 및 로드

1.2 FAISS: Facebook AI의 유사도 검색 라이브러리

- 효율적인 벡터 유사도 검색 및 클러스터링을 위한 오픈소스 벡터 저장소
- **faiss-cpu** 패키지 설치
- 주요 특징:
    1. 대규모 벡터 세트에서 효율적인 검색이 가능
    2. 대용량 데이터셋 처리 (RAM 효율적 활용)
    3. GPU 가속 지원 (faiss-gpu 설치 필요)
    4. 다양한 인덱싱 알고리즘을 제공하여 속도와 정확도를 조절 가능 [https://github.com/facebookresearch/faiss/wiki/Faiss-indexes]
- 주요 사용방법:
    1. 벡터 저장소 초기화 (생성)
    2. 벡터 저장소 관리: 문서 추가, 삭제
    3. 문서 검색: 유사도 검색
    4. 벡터 저장소 로컬 저장 및 로드

2. RAG 검색기

2.1 Semantic Search: 의미 기반 검색

- 기본 개념:
    - Semantic Search는 쿼리의 문자 그대로의 의미가 아닌, 의도와 맥락을 이해하여 검색을 수행
    - 텍스트를 벡터 공간에 매핑하여 의미적 유사성을 계산

- 작동 원리
    - 문서 임베딩:

        - 모든 문서를 벡터로 변환하여 Vector Store에 저장
        - 주로 사전 훈련된 언어 모델(예: BERT, GPT)을 사용하여 임베딩을 생성
    - 쿼리 임베딩:
        - 사용자의 검색 쿼리도 동일한 방식으로 벡터로 변환
    - 유사도 계산:
        - 쿼리 벡터와 문서 벡터 간의 유사도를 계산
        - 주로 코사인 유사도나 유클리디안 거리를 사용
    - 결과 반환:
        - 가장 유사한 문서들을 검색 결과로 반환

- 장점:
    - 동의어, 관련어 등을 고려한 더 정확한 검색 결과 제공
    - 언어의 뉘앙스와 맥락을 이해하여 검색
    - 키워드 기반 검색에서 놓칠 수 있는 관련 정보도 검색 가능

- 한계:
    - 계산 비용이 높을 수 있음 (특히 대규모 데이터셋에서)
    - 임베딩 모델의 품질에 크게 의존함
    - 매우 특정한 키워드 검색에서는 전통적인 방법보다 성능이 떨어질 수 있음

- 주요 실습:
    1. 벡터 저장소 초기화 (또는 로드)
    2. Top K 검색
    3. 임계값 지정 검색
    4. MMR 검색
    5. metadata 필터링
    6. page_content 본문 필터링

2.2 Keyword Search: 키워드 기반 검색

- 기본 개념: 사용자가 입력한 특정 단어나 구문을 문서 내에서 직접 찾는 전통적인 검색
- BM25: 키워드 검색을 더욱 효과적으로 만드는 알고리즘 중 하나
    - BM25는 TF-IDF의 한계를 보완한 고급 랭킹 함수
    - 주요 특징:
        1. 용어 빈도 (TF): BM25는 용어 빈도가 증가함에 따라 점수 증가율이 감소. 이는 특정 단어가 과도하게 반복되는 경우의 영향을 제한
        2. 문서 길이 정규화: 긴 문서에서 용어가 더 자주 나타날 가능성을 고려하여 조정
- **rank_bm25** 라이브러리 설치

- 장점:
    1. 단순하면서도 효과적인 랭킹 시스템
    2. 계산 비용이 상대적으로 낮음
    3. 특정 키워드나 구문 검색에 매우 효과적

- 단점:
    1. 의미적 유사성을 고려하지 않음
    2. 동의어나 관련어를 자동으로 처리하지 못함
    3. 문맥을 이해하지 못함

2.3 Hybrid Search: 의미 기반과 키워드 기반을 결합한 검색 방식

- 개념:
    - 키워드 기반 검색(예: BM25)과 의미론적 검색(Semantic Search)을 결합한 방식
    - 두 방식의 장점을 활용하여 더 정확하고 다양한 검색 결과를 제공
- 장점:
    1. 정확성 향상: 키워드 매칭의 정확성과 의미적 유사성을 동시에 고려
    2. 다양성 확보: 다양한 관점의 검색 결과를 제공 가능
    3. 강건성: 한 방식의 약점을 다른 방식으로 보완

📊 검색 방법 비교

특성	키워드 검색 (BM25)	의미론적 검색
속도	빠름 ⚡⚡⚡	중간 ⚡⚡
정확도	중간 🎯🎯	높음 🎯🎯🎯
설정 복잡도	낮음 🔧	높음 🔧🔧🔧
의미 이해	제한적 🧠	우수 🧠🧠🧠
적합한 사용 사례	• 정확한 키워드 매칭이 필요한 경우 • 대량의 문서에서 빠른 검색이 필요할 때	• 컨텍스트 이해가 중요한 경우 • 유사 개념 검색이 필요할 때

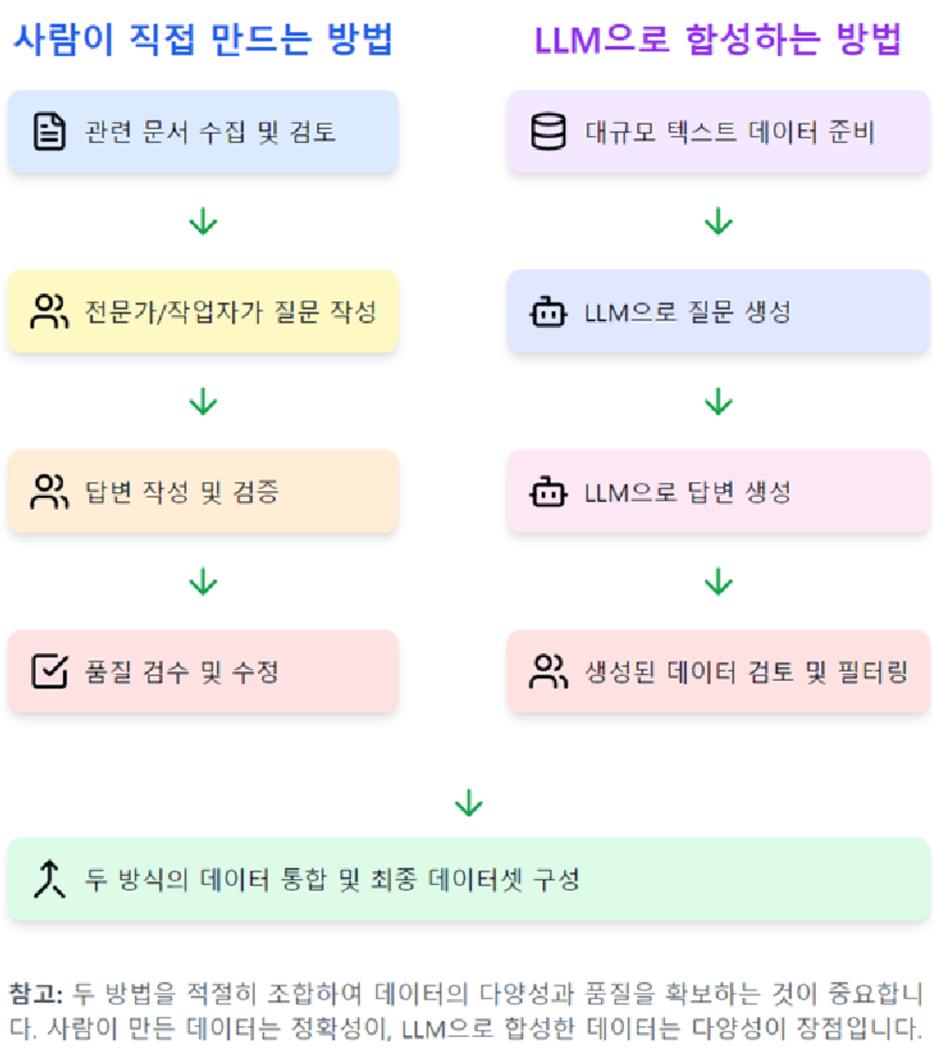
참고: 검색 방법 선택 시 데이터 특성, 성능 요구사항, 구현 복잡도를 고려하세요.

- LangChain의 EnsembleRetriever
  - 작동 원리:
    - 여러 BaseRetriever 객체(예: BM25Retriever, VectorStoreRetriever 등)의 결과를 집계
    - Reciprocal Rank Fusion (RRF) 알고리즘을 사용하여 결과를 재정렬
  - RRF (Reciprocal Rank Fusion) 알고리즘:
    - 각 검색기의 결과 순위를 고려하여 최종 순위를 결정
    - 공식: score = Σ 1 / (r + k), 여기서 r은 각 검색기에서의 순위, k는 상수(보통 60)

3. 검색 성능 평가

3.1 테스트 데이터: 검색 시스템 평가를 위한 데이터셋

테스트 데이터 준비 방법 비교 흐름도



참고: 두 방법을 적절히 조합하여 데이터의 다양성과 품질을 확보하는 것이 중요합니다. 사람이 만든 데이터는 정확성이, LLM으로 합성한 데이터는 다양성이 장점입니다.

- QA(Question-Answering) 데이터셋을 합성
  1. 문서 준비:
     - 관련 도메인의 문서들을 수집

- 이 문서들은 RAG 시스템의 지식 베이스 역할을 하게 됨

	context	source	doc_id
0	리비안은 디젤 하이브리드 버전, 브라질 원메이크 시리즈를 위한 R1 GT 레이싱 버...	data/리비안_KR.txt	0
1	이 차는 쉽게 교체 가능한 본체 패널을 갖춘 모듈식 캡슐 구조를 특징으로 하며, 2...	data/리비안_KR.txt	1
2	머스크는 최대 주주이자 회장으로서 회사를 현재의 성공으로 이끌었습니다\n회사 이름은...	data/테슬라_KR.txt	2
3	2012년부터 2023년 3분기까지 테슬라의 전 세계 누적 판매량은 4,962,97...	data/테슬라_KR.txt	3
4	리비안은 MIT 박사 출신 RJ 스카린지가 2009년에 설립한 혁신적인 미국 전기차...	data/리비안_KR.txt	4

2. 질문 생성:
    - LLM을 사용하여 각 문서에 대한 다양한 유형의 질문을 생성
    - 다양한 난이도와 유형의 질문을 포함 (예: 사실 기반, 추론 기반, 요약 등).
3. 답변 생성:
    - 생성된 질문에 대한 답변을 LLM을 사용하여 생성
    - 이 답변들은 '정답'으로 간주
4. 메타데이터 추가:
    - 각 QA 쌍에 대해 출처 문서, 질문 유형, 난이도 등의 메타데이터를 추가

	context	source	doc_id	question	answer
0	[리비안은 디젤 하이브리드 버전, 브라질 원메이크 시리즈를 위한 R1 GT 레이싱 ...	[data/리비안_KR.txt]	[0]	리비안은 어떤 버전을 고려했나요?	리비안은 디젤 하이브리드 버전, R1 GT 레이싱 버전, 4도어 세단 및 크로스오버...
1	[리비안은 디젤 하이브리드 버전, 브라질 원메이크 시리즈를 위한 R1 GT 레이싱 ...	[data/리비안_KR.txt]	[0]	리비안의 R1 GT 레이싱 버전은 어떤 시리즈를 위한 것인가요?	리비안의 R1 GT 레이싱 버전은 브라질 원메이크 시리즈를 위한 것입니다.
2	[리비안은 디젤 하이브리드 버전, 브라질 원메이크 시리즈를 위한 R1 GT 레이싱 ...	[data/리비안_KR.txt]	[0]	리비안이 고려한 차량의 종류는 무엇인가요?	리비안은 4도어 세단과 크로스오버 등 다양한 차량 종류를 고려했습니다.
3	[이 차는 쉽게 교체 가능한 본체 패널을 갖춘 모듈식 캡슐 구조를 특징으로 하며, ...	[data/리비안_KR.txt]	[1]	이 차의 구조는 어떤 특징이 있나요?	이 차는 쉽게 교체 가능한 본체 패널을 갖춘 모듈식 캡슐 구조를 특징으로 합니다.
4	[이 차는 쉽게 교체 가능한 본체 패널을 갖춘 모듈식 캡슐 구조를 특징으로 하며, ...	[data/리비안_KR.txt]	[1]	이 차는 언제 생산될 예정인가요?	이 차는 2013년 말에서 2014년 초 사이에 생산이 예상됩니다.

- 합성 데이터에 대한 검증 및 수정
    1. 자동화된 검증:
        - 질문과 답변의 길이, 형식 등을 확인
        - 답변이 질문과 관련이 있는지 간단한 관련성 검사를 수행
    2. 사람이 검토:
        - 샘플링된 QA 쌍을 인간 전문가가 검토
        - 질문의 품질, 답변의 정확성, 난이도 등을 평가
    3. 반복적 개선:
        - 검토 결과를 바탕으로 생성 프롬프트를 개선
        - 필요한 경우 특정 QA 쌍을 수동으로 수정
    4. 다양성 확보:
        - 질문 유형, 난이도, 주제 등이 균형있게 분포되어 있는지 확인
    5. 편향성 검사:
        - 생성된 데이터셋에 편향이 없는지 검토

3.2 Information Retrieval 평가 지표:

정밀도 (Precision)
검색된 문서 중 관련 있는 문서의 비율. 검색 결과의 정확성을 측정합니다.

재현율 (Recall)
전체 관련 문서 중 실제로 검색된 관련 문서의 비율. 검색의 포괄성을 나타냅니다.

F1 스코어 (F1 Score)
정밀도와 재현율의 조화평균. 두 지표 간의 균형을 평가합니다.

평균 정밀도 (MAP)
여러 쿼리에 대한 평균 정밀도. 전반적인 검색 성능을 평가합니다.

정규화된 누적 이득 (nDCG)
검색 결과의 순위와 관련성을 고려한 지표. 순위 품질을 평가합니다.

MRR (Mean Reciprocal Rank)
첫 번째 관련 문서의 순위 역수의 평균. 상위 검색 결과의 정확성을 측정합니다.

- Information Retrieval(정보 검색)의 평가 지표를 사용
- K-RAG 패키지를 사용하여 지표 계산
    - **krag** 패키지 설치 (pip install krag, poetry add krag)

1. Hit Rate (적중률):

- 정의: 검색 결과에 관련 문서가 하나라도 포함되어 있는 쿼리의 비율
- 해석: 시스템이 관련 문서를 찾을 수 있는 능력을 나타냄
- 범위: 0 ~ 1 (높을수록 좋음)

2. MRR (Mean Reciprocal Rank):

- 정의: 첫 번째 관련 문서의 역순위의 평균
- 계산: 1 / (첫 번째 관련 문서의 순위)의 평균
- 해석: 시스템이 관련 문서를 상위에 랭크시키는 능력을 나타냄
- 범위: 0 ~ 1 (높을수록 좋음)

3. Recall@k:

- 정의: 상위 k개 검색 결과에서 찾은 관련 문서의 비율
- 계산: (상위 k개 결과 중 관련 문서 수) / (전체 관련 문서 수)
- 해석: 시스템이 모든 관련 문서를 찾을 수 있는 능력을 나타냄
- 범위: 0 ~ 1 (높을수록 좋음)

4. Precision@k:

- 정의: 상위 k개 검색 결과 중 관련 문서의 비율
- 계산: (상위 k개 결과 중 관련 문서 수) / k
- 해석: 검색 결과의 정확도를 나타냄
- 범위: 0 ~ 1 (높을수록 좋음)

5. mAP@k (mean Average Precision at k):

- 정의: 각 관련 문서를 검색할 때마다의 정확도(precision)의 평균
- 계산: 각 관련 문서 검색 시점의 precision의 평균을 모든 쿼리에 대해 평균
- 해석: 검색 시스템의 전반적인 성능을 나타냄
- 범위: 0 ~ 1 (높을수록 좋음)

6. NDCG@k (Normalized Discounted Cumulative Gain at k):

- 정의: 검색 결과의 순위를 고려한 누적 이득(gain)의 정규화된 값
- 계산: 실제 DCG를 이상적인 DCG로 나눔
- 해석: 검색 결과의 순위와 관련성을 모두 고려한 성능 지표
- 범위: 0 ~ 1 (높을수록 좋음)