

과학 지식 질의 응답 시스템 구축

RAG

IR 2조 발표

IR 2조
김태한
김소현
김준호
최장원

목차

01. 팀원 소개

02. 대회 소개

03. 데이터 소개

04. 모델링

05. 멘토님 피드백 정리

06. 결론

07. 대회 진행 소감 및 회고

01

팀원 소개

팀원 소개

김태한(팀장)	리서치, 데이터생성, 모델링, 후처리, 발표
김소현	리서치, 데이터생성, 모델링, 후처리
김준호	리서치, 데이터생성, 모델링, 후처리
최장원	리서치, 데이터생성, 모델링, 후처리

02

대회 소개

RAG (Retrieval Augmented Generation)의 도입 배경

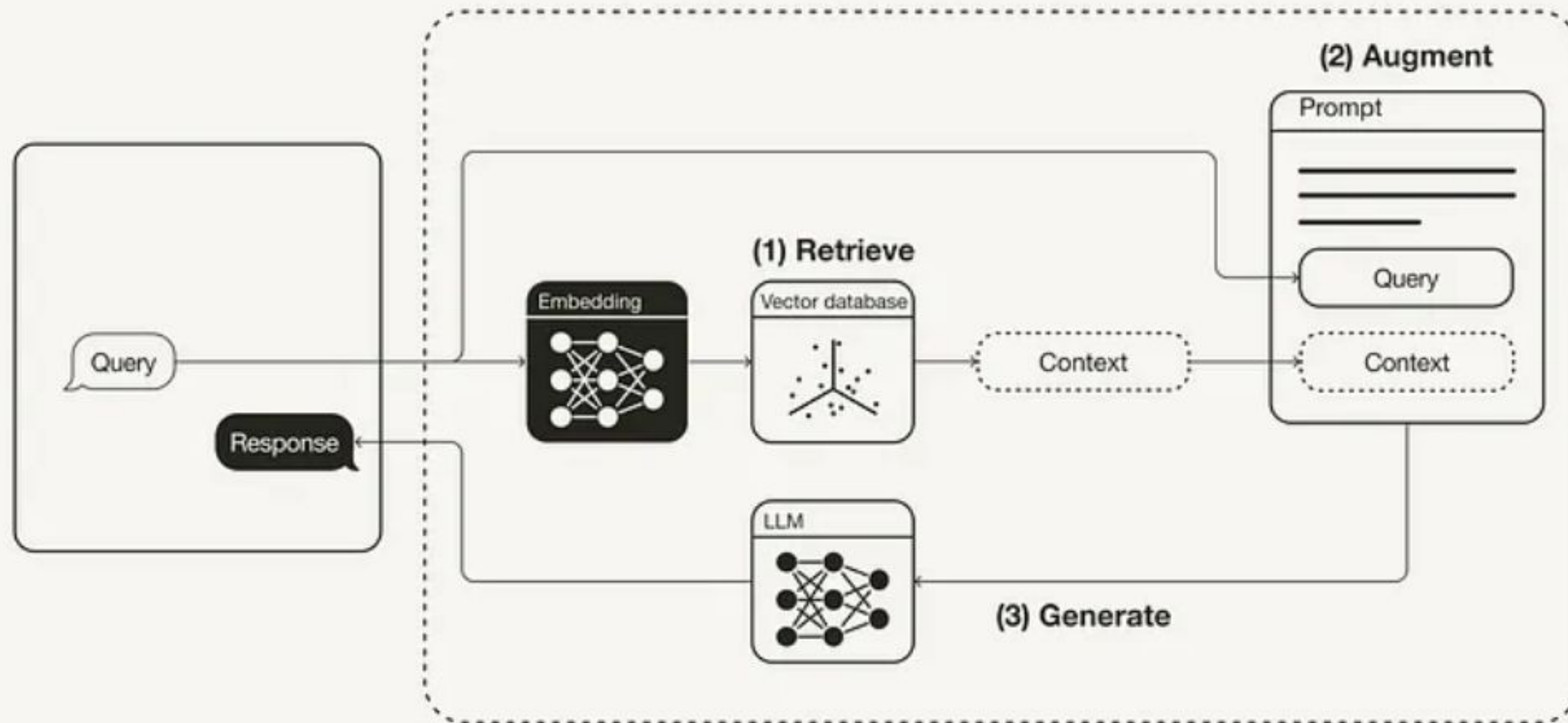
LLM의 문제

- Hallucination
- 내용이 업데이트 되지 않음

해결방법

- 검색엔진의 검색 결과를 활용하여 이 지식에 기반해서 LLM을 추론만을 수행
- 검색엔진의 최적화가 중요하다.
- 검색엔진의 구성과 최적화 등은 Information Retrieval에 속하며 따라서 이름이 Retrieval에 의해 향상된 Generation이 된다.

과학질의응답시스템구축



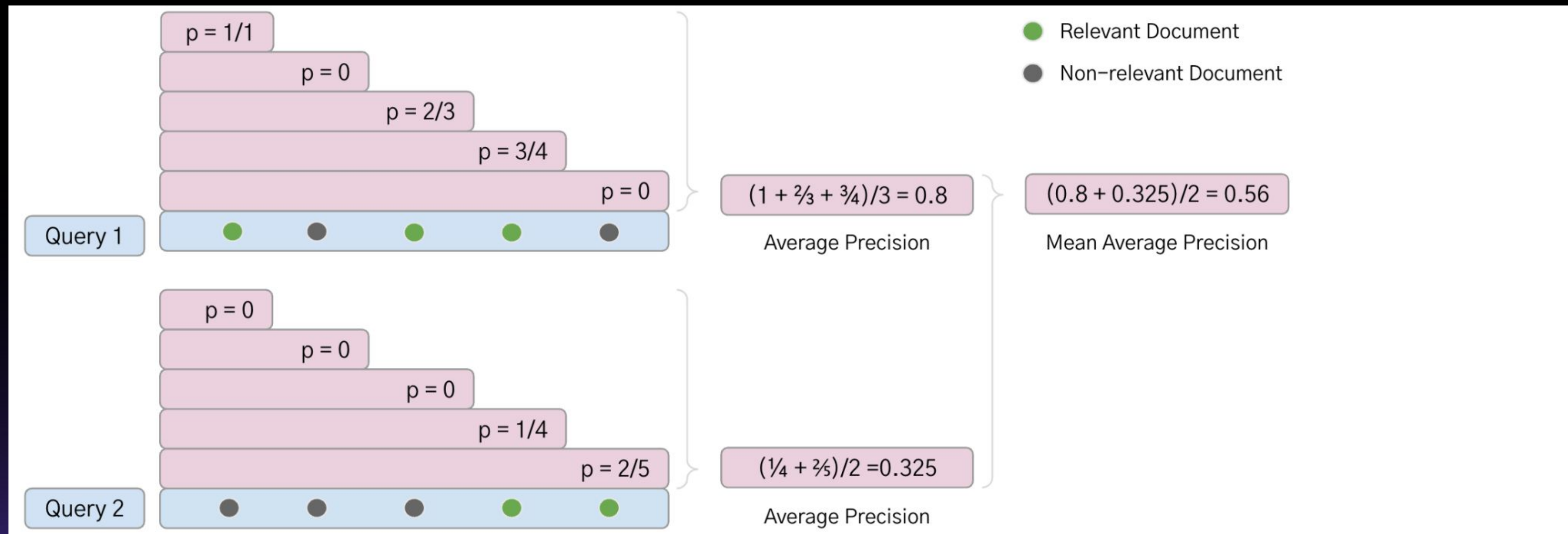
RAG

Information Retrieval을 통해서 LLM의 생성 능력을 향상시키는 태스크다.

이번 대회에서는 과학문서에 대한 질의의 답변을 올바르게 생성하도록 하는게 목적이다.

평가 방법: MAP (Mean Average Precision)

질의 N개에 대한 Average Precision의 평균 값을 구하고, Average Precision은 Precision-recall curve에서 아래쪽 면적을 의미합니다. 계산 과정은 도식화하면 아래 그림과 같습니다.



03

데이터 소개

데이터 소개

- 과학 상식 문서 4272개
- ko_ai2_arc__ARC_Challenge와 ko_mmlu 데이터
- 총 63개의 데이터 소스 (ko_mmlu__human_sexuality__train, ko_mmlu__human_sexuality__test 등을 별개로 카운트, 또한 ko_mmlu__human_sexuality__train과 ko_mmlu__conceptual_physics__train 도 별개로 카운트)
- 파일 포맷은 각 line이 json 데이터인 jsonl 파일
- 트레이닝 데이터 예시

```
{"docid": "7a3e9dc2-2572-4954-82b4-1786e9e48f1f", "src": "ko_ai2_arc__ARC_Challenge__test", "content": "산꼭대기에서는 중력이 아주 약간 변합니다. 이는 무게에 영향을 미칩니다. 산꼭대기에서는 무게가 감소할 가능성이 가장 높습니다. 중력은 지구의 질량에 의해 결정되며, 산꼭대기에서는 지구의 질량과의 거리가 더 멀어지기 때문에 중력이 약간 감소합니다. 따라서, 산꼭대기에서는 무게가 더 가볍게 느껴질 수 있습니다."}
```
- 평가 데이터 예시

```
{"eval_id": 0, "msg": [{"role": "user", "content": "나무의 분류에 대해 조사해 보기 위한 방법은?"}]}  
{"eval_id": 1, "msg": [{"role": "user", "content": "각 나라에서의 공교육 지출 현황에 대해 알려줘."}]}  
{"eval_id": 2, "msg": [{"role": "user", "content": "기억 상실증 걸리면 너무 무섭겠다."}, {"role": "assistant", "content": "네 맞습니다."}, {"role": "user", "content": "어떤 원인 때문에 발생하는지 궁금해."}]}  
{"eval_id": 3, "msg": [{"role": "user", "content": "통학 버스의 가치에 대해 말해줘."}]}
```


데이터 소개

- 과학 세부 분야

['nutrition', 'conceptual_physics', 'ARC_Challenge', 'human_sexuality', 'virology', 'human_aging',
'high_school_biology', 'high_school_physics', 'college_biology', 'computer_security', 'anatomy', 'college_physics',
'medical_genetics', 'electrical_engineering', 'college_medicine', 'college_chemistry', 'astronomy', 'college_computer_science',
'global_facts', 'high_school_chemistry', 'high_school_computer_science']

데이터 소개

질의/질문 생성

- 기존의 데이터에는 질문 - 정답(문서) 페어가 아닌 문서만이 존재한다.
- 따라서 질문을 생성할 필요가 있는데, 4000개가 넘는 문서에 대한 질문은 만들기도 어렵고 과학 분야기 때문에 관련된 지식이 없다면 만들 수 없다.
- 따라서 LLM을 활용하여 질문을 생성한다.
- OpenAI 크레딧이 있지만 크레딧을 아끼기 위해서 무료 API인 구글의 Gemini의 API를 사용해서 만든다.

“””

문서 내용

위 문서에서 질문 5개 만들어줘

“””

위의 간단한 프롬프트를 활용해서 질문을 생성했다.

- 문서마다 3개 혹은 5개의 질문을 생성했으며 학습에는 1개 혹은 3개의 질문을 사용했다.

Positive and Negative Pairs

- Positive pairs는 질의와 답변이 서로 관련이 있는 경우를 의미한다.
- Negative pairs는 질의와 답변이 서로 관련이 없는 경우를 의미한다.

질의 Q가 "태양의 지름이 얼마야?" 일 때,

관련이 있는 아래의 문서 P는

"태양은 지구 지름의 109배인 139만km, 무게는 지구보다 무려 33만 2,900배나 무겁습니다. 태양계 전체 질량의 99.8% 이상을 차지하고, 태양계의 중심에 위치하여 지구를 포함한 8개 행성과 위성, 혜성 등의 운동을 지배하고 있는 별입니다."

질의 Q와 positive pair를 이루고,

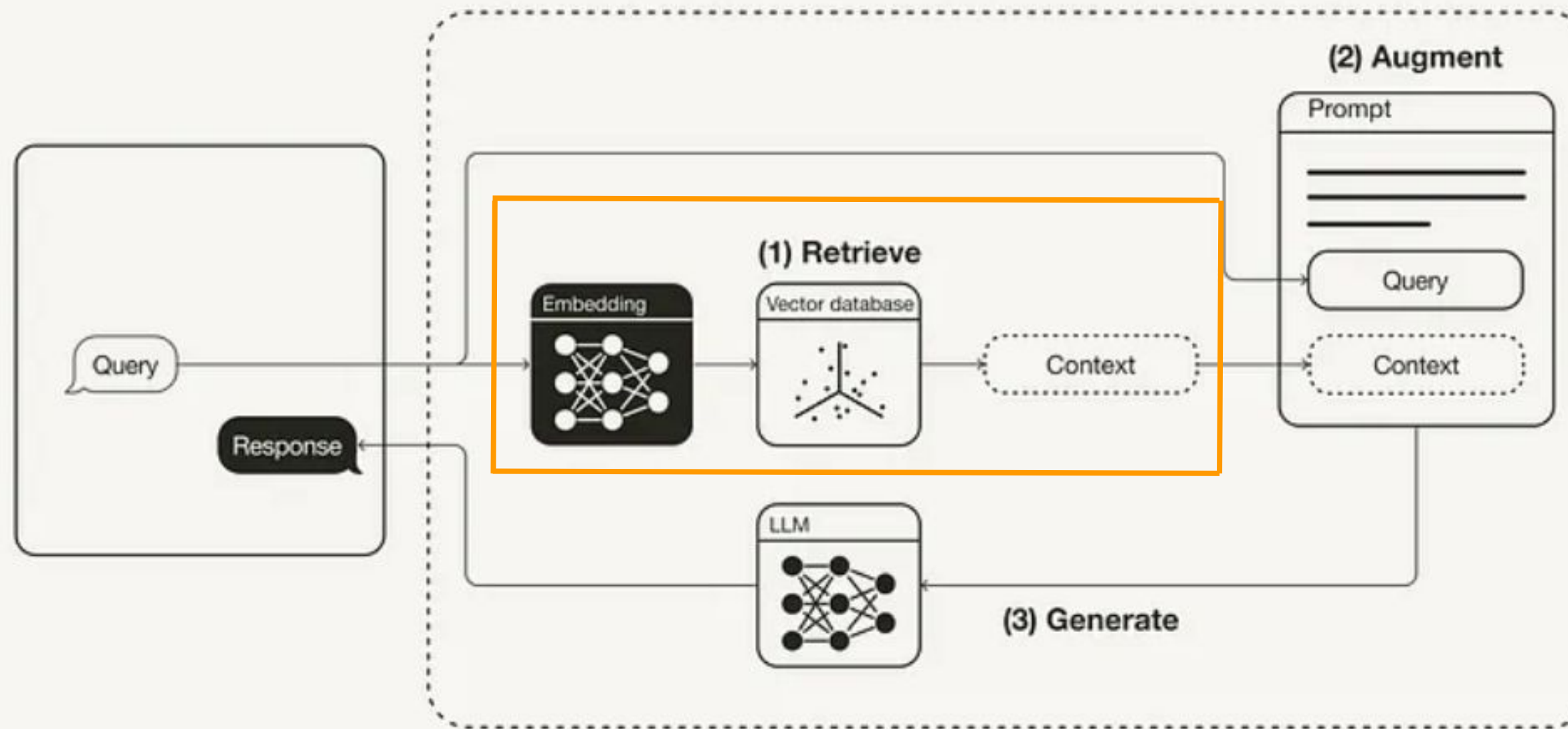
관련이 없는 문서 N는

"인공지능(AI)은 컴퓨터에서 음성 및 작성된 언어를 보고 이해하고 번역하고 데이터를 분석하고 추천하는 기능을 포함하여 다양한 고급 기능을 수행할 수 있는 일련의 기술입니다."

질의 Q와 negative pair를 이룬다.

05

모델링



주황색 박스 부분이 Retrieval이다.
Elasticsearch로 수행

Sparse Retrieval과 Dense Retrieval
두 가지 옵션

Sparse Retrieval은 BM25와 역색인

Dense Retrieval은 KNN 방식이며,
문서의 임베딩 벡터를 사용.
Pre-trained 모델로
sentence_transformers의
"snunlp/KR-SBERT-V40K-klueNLI-au
gSTS" 사용

위 모델을 LoRA로 파인 튜닝도 시도.

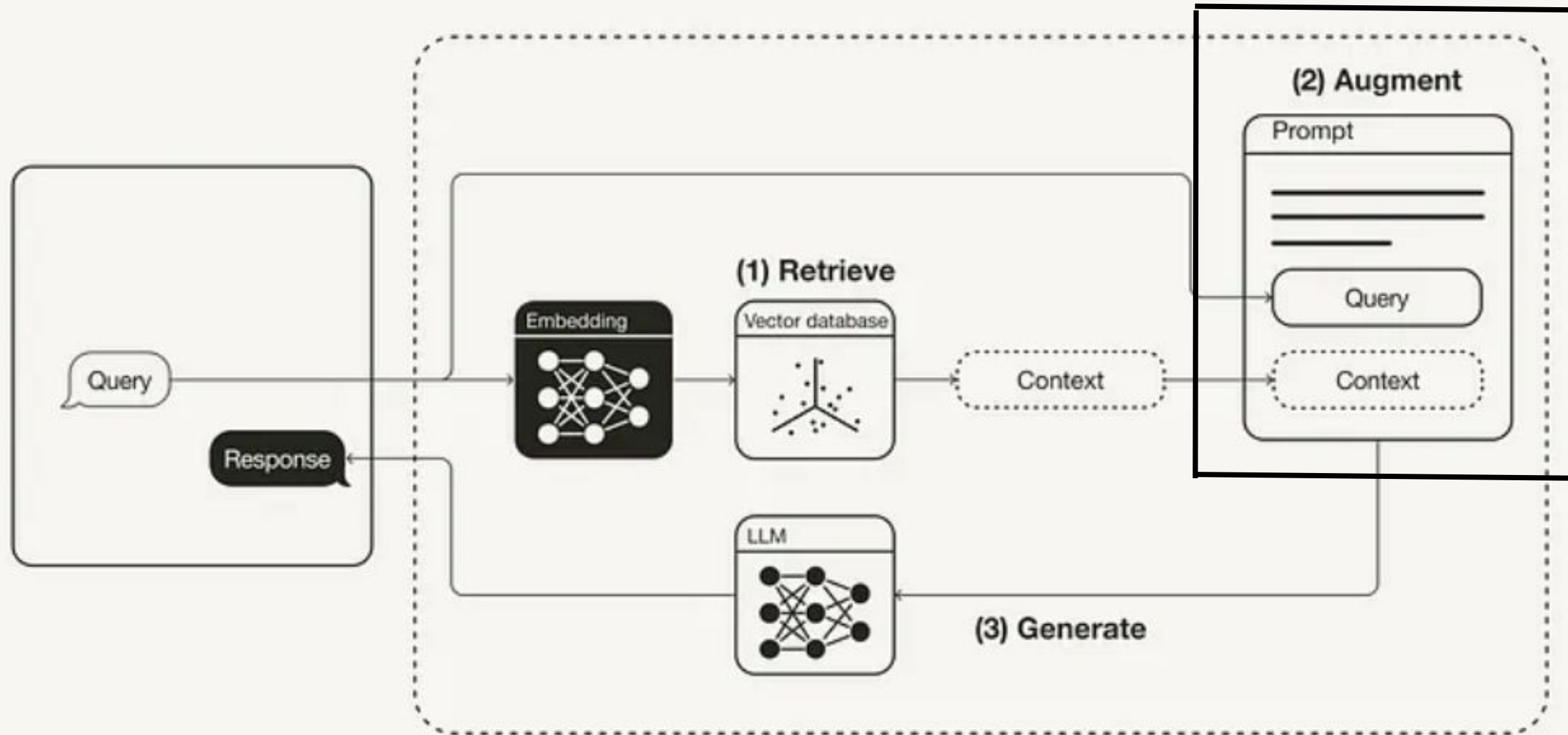
두 가지 retrievals로 topk 레퍼런스
생성.

Language Model Fine Tuning

아래의 Cosine Embedding Loss를 활용해서,
관련있는 Q와 D (document)는 가깝게, 관련없는 Q와 D는 서로 멀도록 학습한다.

$$\text{loss}(x, y) = \begin{cases} 1 - \cos(x_1, x_2), & \text{if } y = 1 \\ \max(0, \cos(x_1, x_2) - \text{margin}), & \text{if } y = -1 \end{cases}$$

- Pre-trained 모델로 sentence_transformers의 "snunlp/KR-SBERT-V40K-klueNLI-augSTS"을 LoRA로 학습,
- Huggingface의 KLUE-RoBERTa를 파라미터 전체에 학습.
- Elasticsearch와의 호환성 문제로 KR-SBERT-V40K-klueNLI-augSTS를 사용



검정 박스에서는 프롬프트를 작성한다.

과학지식인지 아닌지를 LLM이 알아서 판단하고 이에 대한 구체적인 결과를 json 형태로 받는다.

이를 위해

`client.chat.completions.create()`

함수에

```
response_format={"type":  
"json_object"}
```

를 추가한다.

또한 프롬프트 instruction을 변경한다.

RAG 구현에 필요한 Question Answering을 위한 LLM 프롬프트

persona_qa = """

Role: 과학 지식 전문가

Instructions

- 사용자의 이전 메시지 정보 및 주어진 Reference 정보를 활용하여 간결하게 답변을 생성한다.
 - 주어진 검색 결과 정보로 대답할 수 없는 경우는 정보가 부족해서 답을 할 수 없다고 대답한다.
 - 한국어로 답변을 생성한다.
 - 결과는 json 형태로 생성하고, 반드시 'is_science' 필드를 추가하여 질문이 과학 지식에 관련된 내용이면 true를, 과학 지식에 관련된 내용이 아니라면 false를 value로 만든다. 답변 필드의 이름은 'answer'로 통일한다.
- """

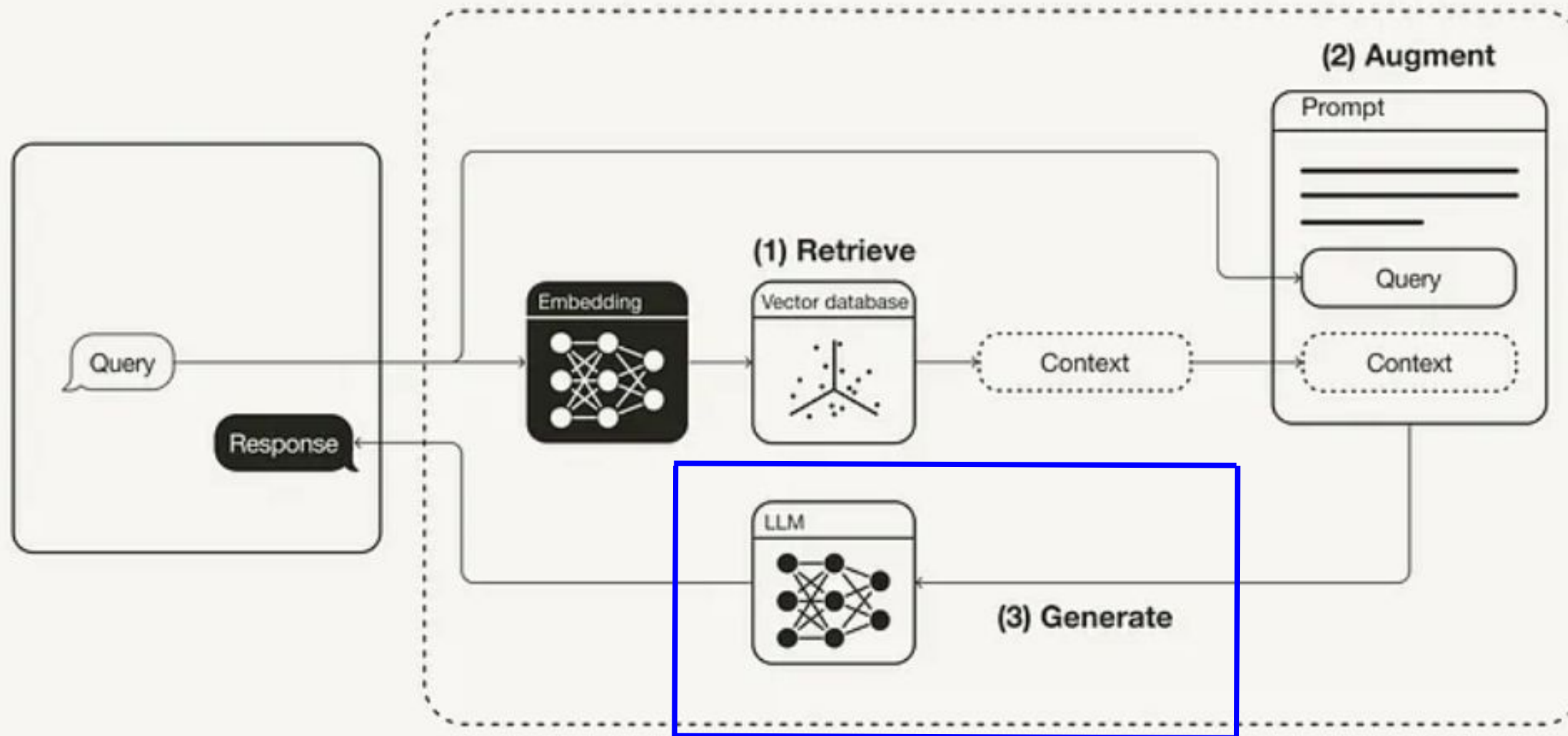
RAG 구현에 필요한 질의 분석 및 검색 이외의 일반 질의 대응을 위한 LLM 프롬프트

persona_function_calling = """

Role: 과학 지식 전문가

Instruction

- 사용자가 대화를 통해 과학 지식에 관한 주제로 질문하면 search api를 호출할 수 있어야 한다.
 - 과학 지식과 관련되지 않은 나머지 대화 메시지에는 적절한 대답을 생성한다.
 - 결과는 json 형태로 생성하고, 반드시 'is_science' 필드를 추가하여 질문이 과학 지식에 관련된 내용이면 true를, 과학 지식에 관련된 내용이 아니라면 false를 value로 만든다. 답변 필드의 이름은 'answer'로 통일한다.
- """



파란 박스 부분에서 LLM을 활용하여 질의에 대한 정답을 생성한다.

OpenAI API를 사용하였으며,
"gpt-3.5-turbo-1106" 모델 사용.

앞에서 retrievals로 만들었던 topk
레퍼런스와 개선된 프롬프트를
토대로 질의에 대한 정답을 생성한다.

06

멘토님 피드백 정리

멘토링 정리




- Elasticsearch 자체를 쓰는것도 좋지만 직접 구현하는 방법도 고려해볼 필요가 있다.
- 쿼리가 과학상식인지 아닌지 구분하는 추가적인 별도의 모델을 둘 수도 있다.
- Elasticsearch에 쓰인 역색인이나 BM25를 건드리는 대신 동의어, 유의어 처리, 토큰나이저 등의 변경 등이 더 중요하다. 데이터의 품질에 집중하는게 좋다.
- Sparse retrieval로 후보군을 추린 다음 dense retrieval로 결정하는 방법도 있다.
- Huggingface나 konlpy 등을 통해서 사용자 사전을 추가하는 방법도 있다.
- 같은 모델이라도 seed를 변경하는 방법도 고려해볼만하다.
- 문서의 주제를 강조하기 위해서 문서의 맨 앞에 주제 단어를 추가한 다음 파인 튜닝하는 방법도 있다.
- Cosine Embedding Loss로 모델을 학습하기 위해서는 positive pairs와 negative pairs를 잘 생성해야 한다. 같은 도메인 내의 다른 주제, 가령 화학 내의 유기화학과 무기화학은 비슷하지만 다른 주제다. 이렇게 큰 틀에서 같지만 세부 내용이 다른 경우를 negative pairs로 만든다면 모델이 더 세밀하게 학습할 수 있다.

07




결론

대회 결과

Public Leaderboard

순위	팀 이름	팀 멤버	MAP ↕	MRR ↕	제출 횟수	최종 제출
10 (1 ▾)	IR 2조	  jw 	0.7076	0.7106	9	1d

Final Leaderboard

순위	팀 이름	팀 멤버	MAP ↕	MRR ↕	제출 횟수	최종 제출
10 (-)	IR 2조	  jw 	0.7152	0.7182	9	3d

08

대회 진행 소감 및 회고

대회 진행 소감 및 회고

- 멘토님이 질의와 응답에 대한 **hard negative pairs**를 생성하라고 조언해주셨다. 이는 똑같은 물리학이라도 전자기학과 양자역학이 서로 다르기 때문에, 커다랄게는 같은 도메인일지라도 세부 내용을 달리하여 보다 섬세하게 학습을 할 수 있도록 한다. 하지만 이는 사람이 수작업으로 매칭을 해야하고 도메인 지식이 많이 요구되기 때문에 수행할 수 없었다.
- **Sentence Transformer**이 아닌 **Huggingface**의 모델을 파인 튜닝 후 적용하지 못해서 아쉬웠다.
- 유저 사전을 만들어서 **konlpy**에는 사전 등록을 해봤으나 **sentence_transformer**에는 적용하지 못해서 아쉽다.
- '통학 버스의 가치에 대해 말해줘.'와 같이, 교육과 관련된 내용인지 교통공학에 관련된 내용인지, 그리고 버스라는 자체로 기계공학이나 전자공학과 관련된 내용인지 애매한 항목에 대한 추가적인 분석이 부족했다.
- 다들 **IR**분야에 생소하기도 하고 **Elasticsearch**를 처음 사용해서 문서 자체에 대한 언어적, 도메인적 분석을 충분히 하지 못한 점이 아쉽다.
- 구체적인 질의 내용이 아닌 내용들을 레퍼런스로 삼아서 **LLM**이 정답을 생성했기 때문에 점수가 낮았다. 예를 들어 "나무 분류"에 대해서 알려달라한 질문에 대해서 "나무 분류"랑 상관없는 나무 내용들이 높은 점수를 얻은 레퍼런스가 되었다. **LLM**의 생성 결과에 대한 분석을 시간 부족으로 인해 하지 못해서 아쉽다.

Q&A

감사합니다.

—