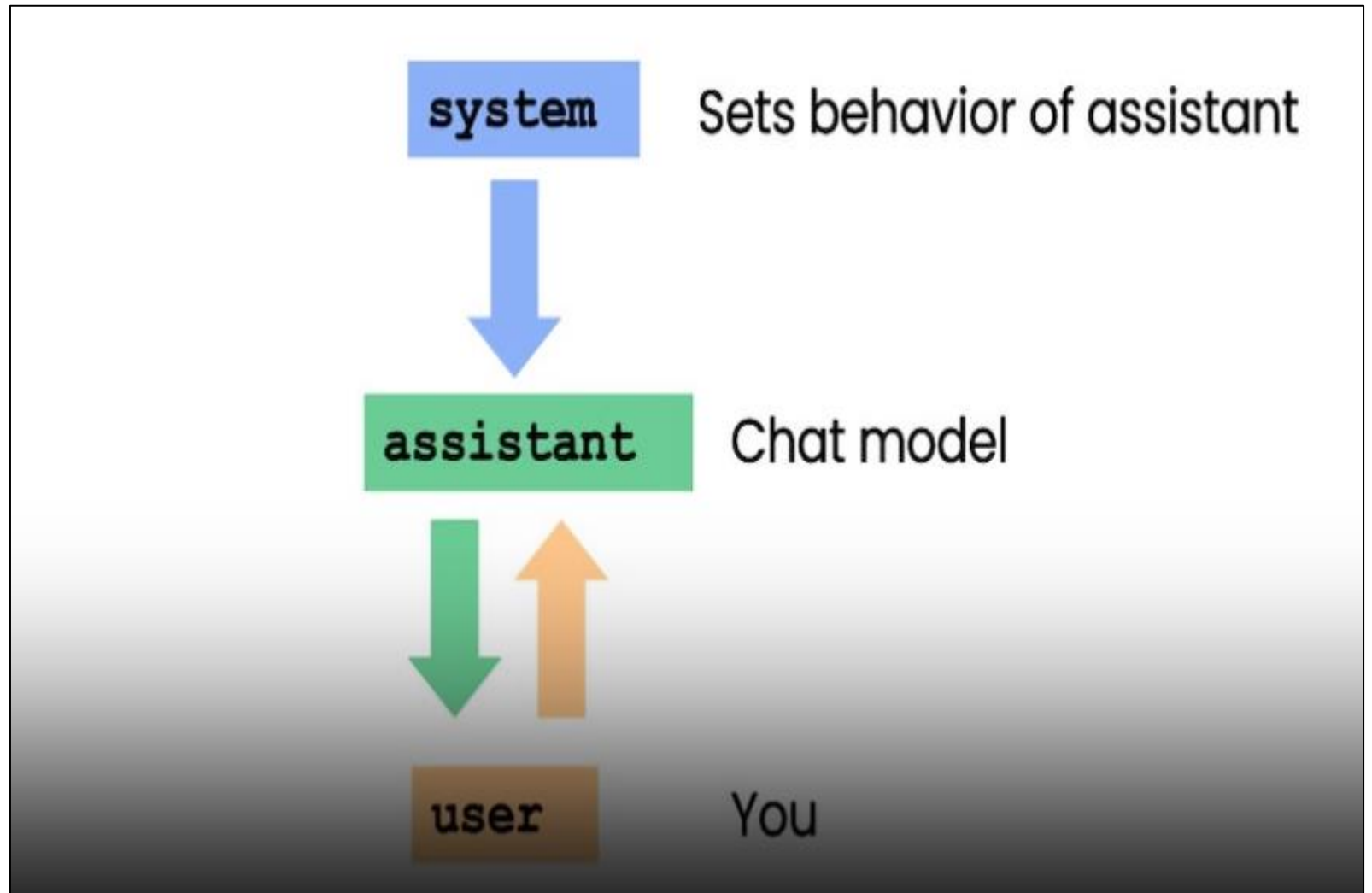


OpenAI ChatCompletion API



- System
 - ✓ 전반적인 AI 역할 지시
- user
 - ✓ 사용자의 요청
- assistant
 - ✓ AI 답변



Role

```
messages =  
[  
  {"role": "system",  
    "content": "You are an assistant... "},  
  {"role": "user",  
    "content": "tell me a joke "},  
  {"role": "assistant",  
    "content": "Why did the chicken... "},  
  ...  
]
```

OpenAI API call

```
def get_completion(prompt,
                    model="gpt-3.5-turbo") :
    messages = [{"role": "user",
                  "content": prompt}]
    response = openai.ChatCompletion.create(
        model=model,
        messages=messages,
        temperature=0)
```



RAG와 벡터 DB

2025년 01월

- 검색 기반 생성 (RAG : Retrieval Augmented Generation)
- 요약

검색 기반 생성

(RAG : Retrieval Augmented Generation)

- 자연어 처리(NLP) = 자연어 이해(NLU) + 자연어 생성(NLG)
 - ✓ NLU : 주어진 문맥을 읽고 이해한다.
 - ✓ NLG : 주어진 질문에 답변한다.
- 임베딩(Embeddings) : 의미를 담은 숫자
 - ✓ 문장의 의미를 비교할 수 있다.
- 토큰(Token) : LLM의 기본 처리 단위
- LLM의 한계
 - ✓ Knowledge Cut-off
 - ✓ Hallucination

➔ **RAG (Retrieval Augmented Generation)**

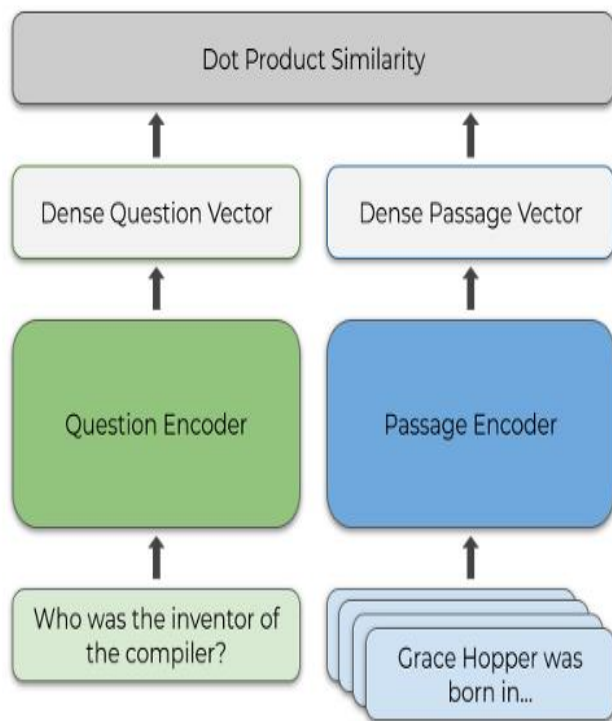
RAG : RAG 개념 → NLU 를 이용



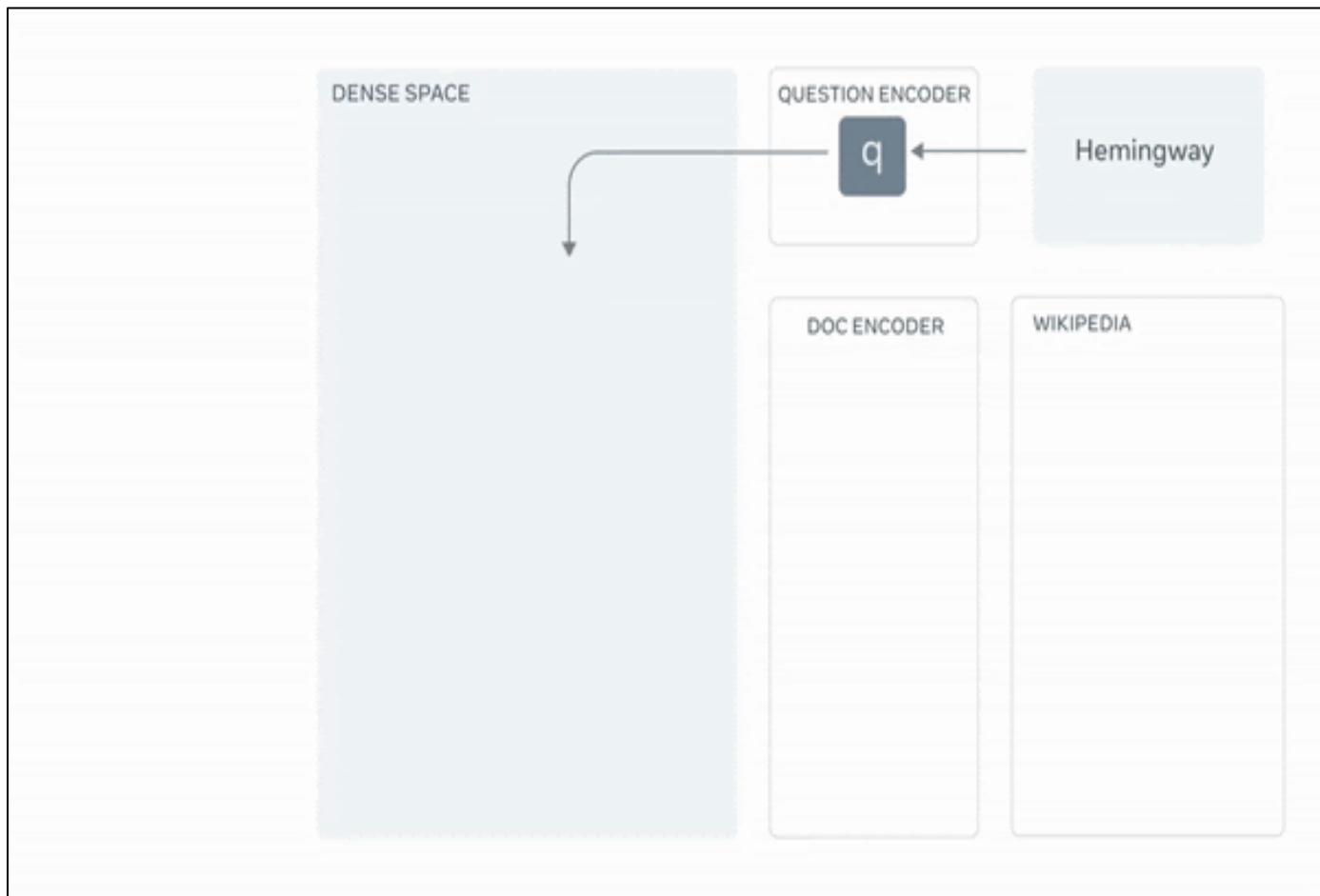
Retrieval Augmented Generation:
Streamlining the creation of intelligent
natural language processing models

September 28, 2020

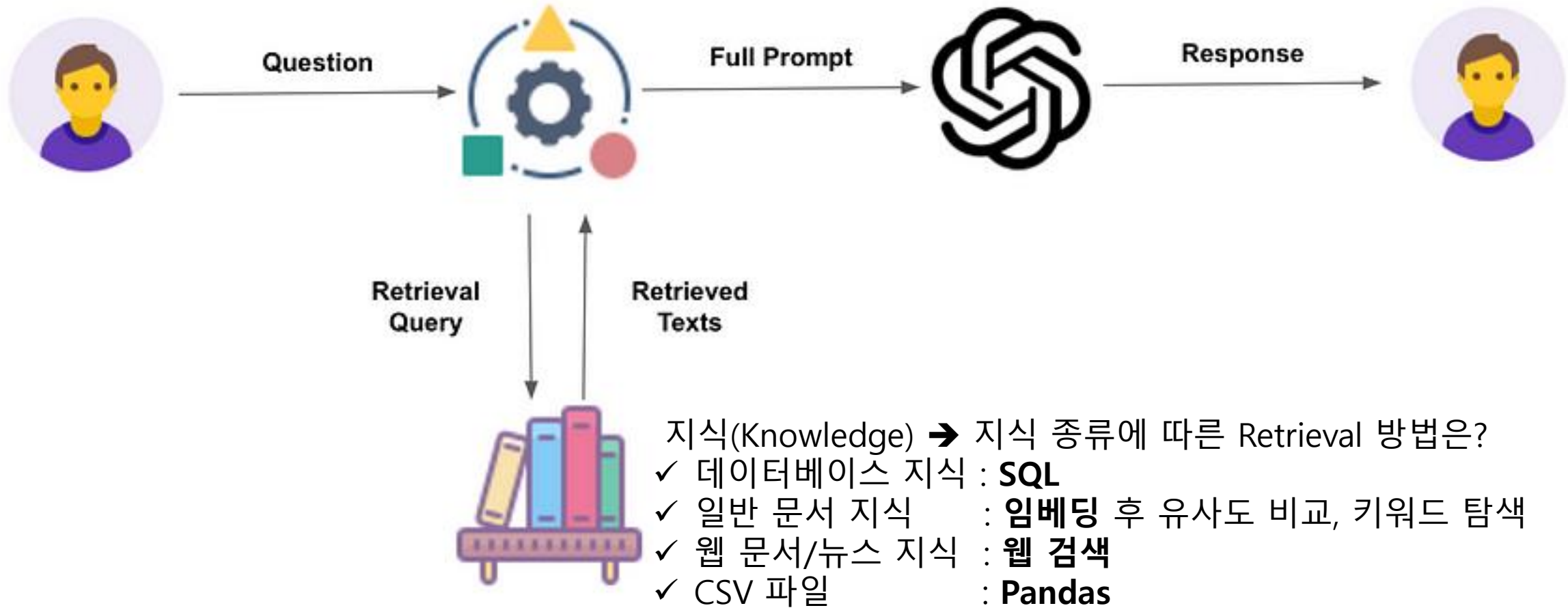
Combining the strengths of open-book and closed-book



Dense Passage Retrieval (DPR).



검색 기반 생성(RAG : Retrieval Augmented Generation)



- 웹 문서를 Context로 제공하여 LLM 답변 요청
 - ✓ 입력된 내용과 연관된 문서 검색
(google, 네이버, ...)
 - ✓ 검색된 내용을 가져오기
 - ✓ 해당 문서 첨부하여 LLM 답변 요청
- 검색 정확도에 따라 답변 정확도가 달라짐

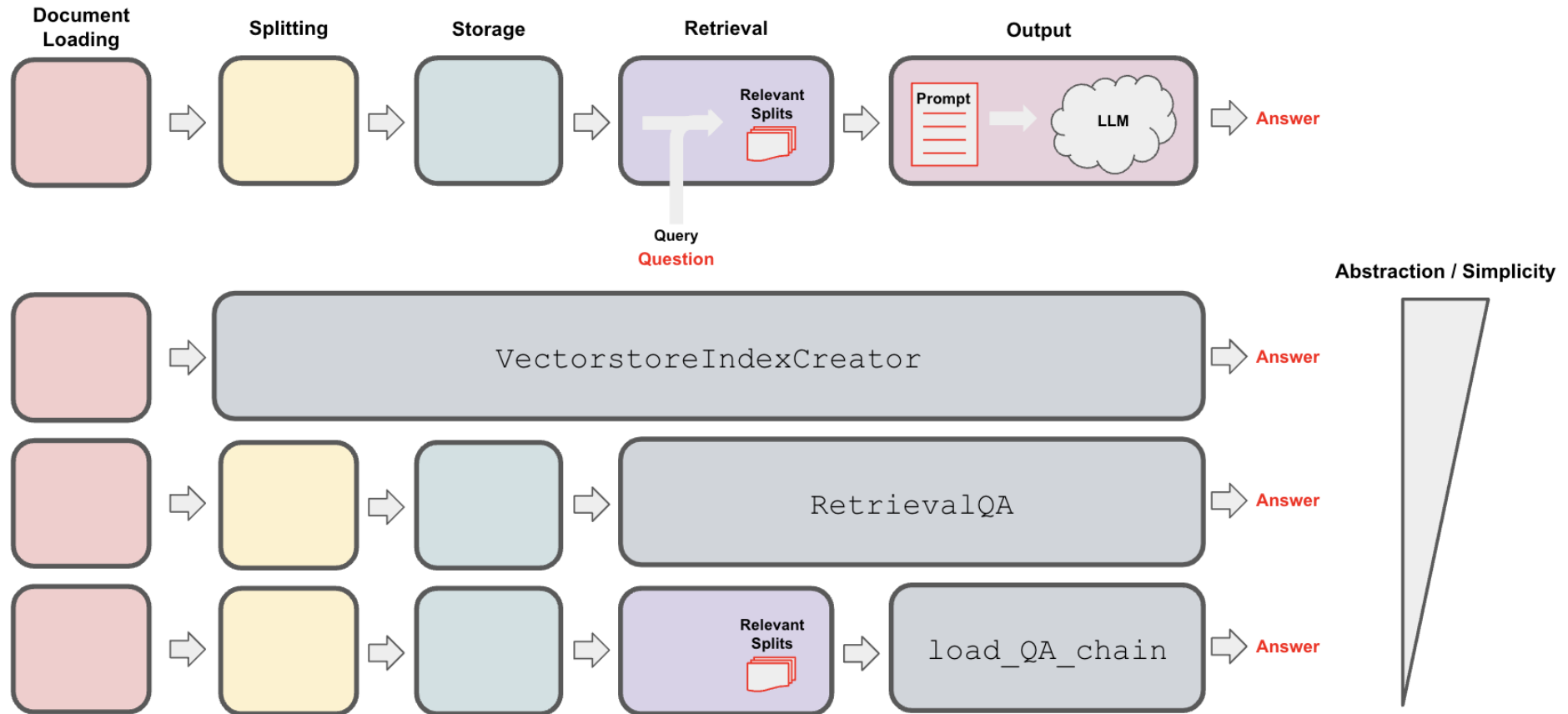
지시 : 프롬프트

컨텍스트 : 웹 문서

웹 조회 기반 생성(RAG) = 프롬프트 + 웹 검색



LangChain Retrieval QA Chain 추상화 단계



RAG : 문서 검색 기반 생성 (RAG : Retrieval Augmented Generation)



초록마을, AI가 상품 찾아준다

| 마이크로소프트 GPT-4 검색엔진 장착

유통 | 입력 : 2023/08/10 08:38

 안희정 기자 |   기자 페이지 구독  기자의 다른기사 보기

[웹비나] ROHM | EEPROM의 기초와 특징, 성능을 최대화시킬 수 있는 테크닉 등을 소개합니다!

D2C 푸드테크 스타트업 정육각이 마이크로소프트의 애저(Azure) 오픈AI GPT-4를 적용한 검색엔진을 자체 개발하고 친환경 유기농 전문 초록마을의 모바일 앱에 전격 도입했다고 10일 밝혔다.

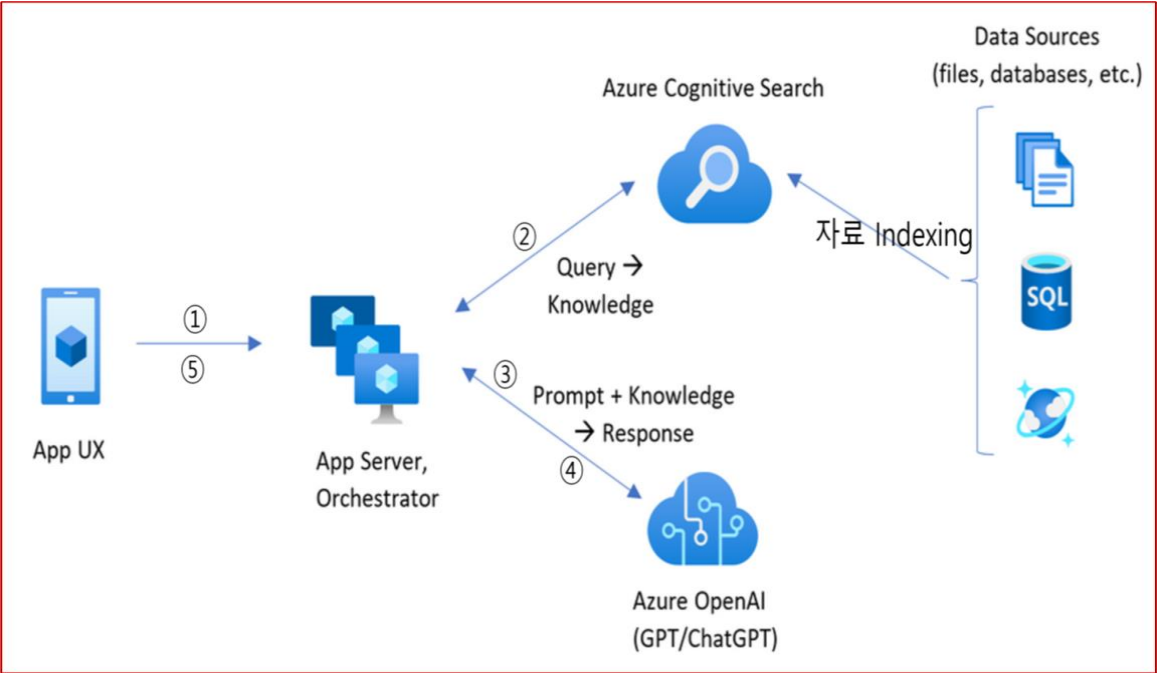
지난달 초 모바일 앱을 네이티브 앱 방식으로 전면 재개발한 데 이어 고객 편의 극대화에 주력한다는 복안이다.

새롭게 적용한 검색엔진은 학습한 검색 패턴을 바탕으로 고객의 의도를 파악하고 관심사가 반영된 개인 맞춤형 결과를 추천해준다. 가령 미역국을 입력하면 초록마을 내 완제품을 포함해 국거리용 한우, 바지락살, 국물용 멸치 등의 부재료 및 간장과 같은 고관여 상품까지 노출하는 식이다.



새로운 검색엔진은 정육각 개발팀과 마이크로소프트가 협업한 결과물로 구상부터 적용까지 채 한 달이 걸리지 않았다. 약 2만 개가 넘는 초록마을 상품마스터의 전처리 데이터 생성 및 정리에 GPT-4를 적용하는 것을 구상한 후 마이크로소프트의 전문가 지원 프로그램인 패스트트랙을 통해 한국, 호주의 엔지니어들과 접근 방식을 논의했다.

24년 된 초록마을이 빠르게 검색엔진을 갈아끼울 수 있었던 배경에는 클라우드 기반으로 경영 환경을 전면 전환하면서 지난 달 초 도입을 완료한 마이크로소프트 애저가 있다. 애저 CosmosDB로 이전한 기존 상품 정보에 GPT-4로 생성한 원천 데이터를 결합하고 애저 Cognitive Search를 이용하는 등 애저 생태계 내에서 매끄럽고 신속하게 새로운 기능을 구현했다.



기업 데이터와 LLM 연동 : 지시 + 컨텍스트(Context)



기업 데이터를 Context로 제공하자.

- ✓ 토큰 제한 : **데이터가 크면??** → 쪼개자 (split)
 - ✓ 쪼개지면 필요한 **문서를 어떻게 찾을까?** → 질문과 유사한 문서를 찾자 (embedding 비교)
 - 질문 : **무엇을 임베딩**하면 답변에 필요한 **문서를 정확히** 찾을까?
 - ✓ 문서가 많으면 **찾는 속도가 늦지 않을까?** → **인덱싱(indexing)**을 하자.
 - **벡터 데이터베이스(Vector Database)** : 인덱싱된 벡터 관리 시스템
- 지시 : 프롬프트



컨텍스트 : 벡터 데이터베이스 내의 문서

내부 문서 조회 기반 생성(RAG) = 프롬프트 + 벡터데이터베이스

RAG : 기업 데이터와 LLM 연동

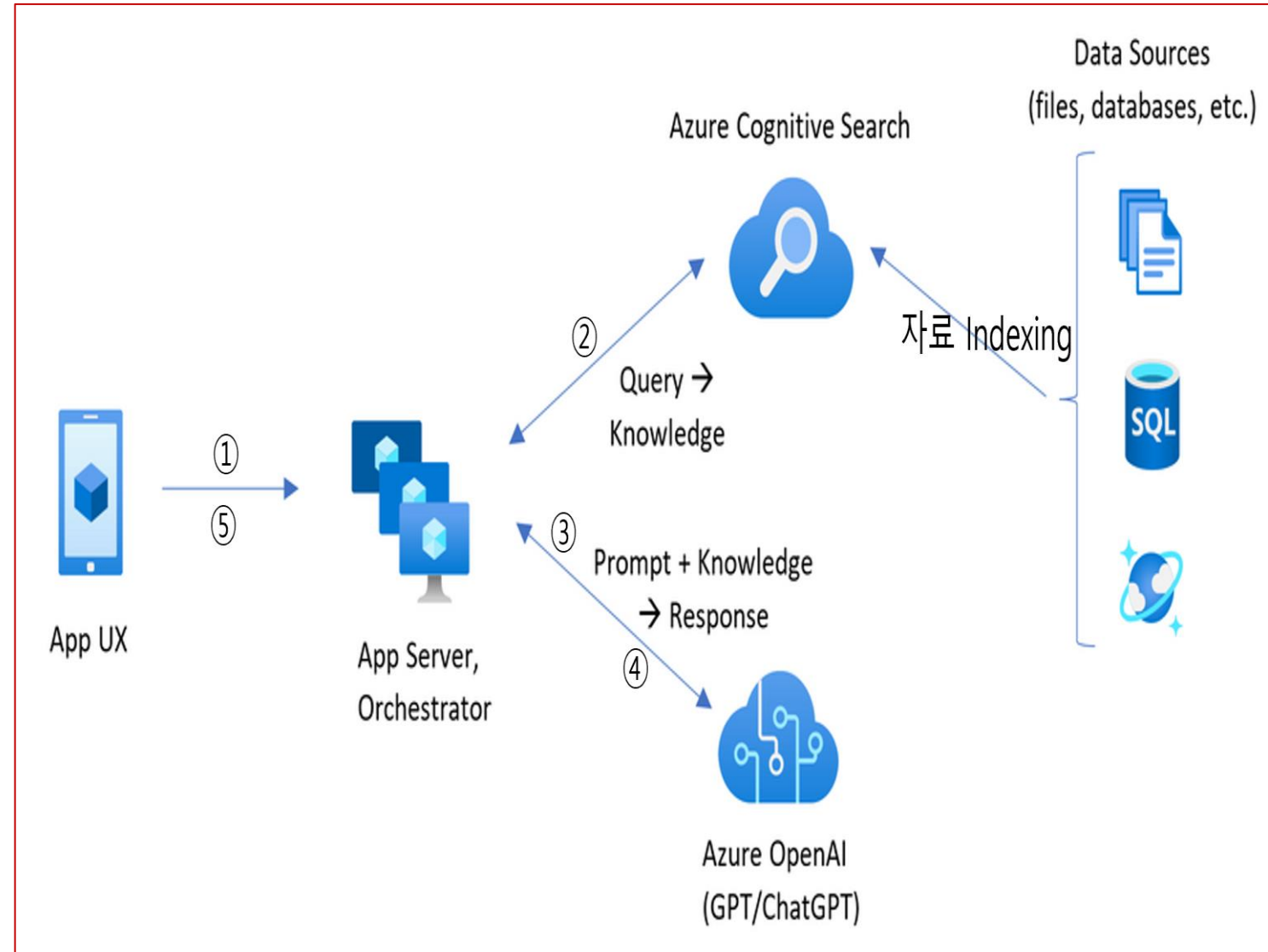


- 준비 사항

- ✓ OpenAI 라이선스 키
- ✓ KMS
 - 기존 자료 탐색을 위한 Index 생성
- ✓ Orchestration Layer

- 처리 절차

- ① 사용자 질의 입력
- ② KMS 에서 연관된 내용 검색
- ③ 검색된 내용과 질의, 응답방법 전달
- ④ 전달된 내용 기반 응답 생성
- ⑤ 사용자 응답 전달



RAG : 기업 데이터와 LLM 연동



- 준비 사항

- ✓ 입력 데이터 : 종합소득세 안내

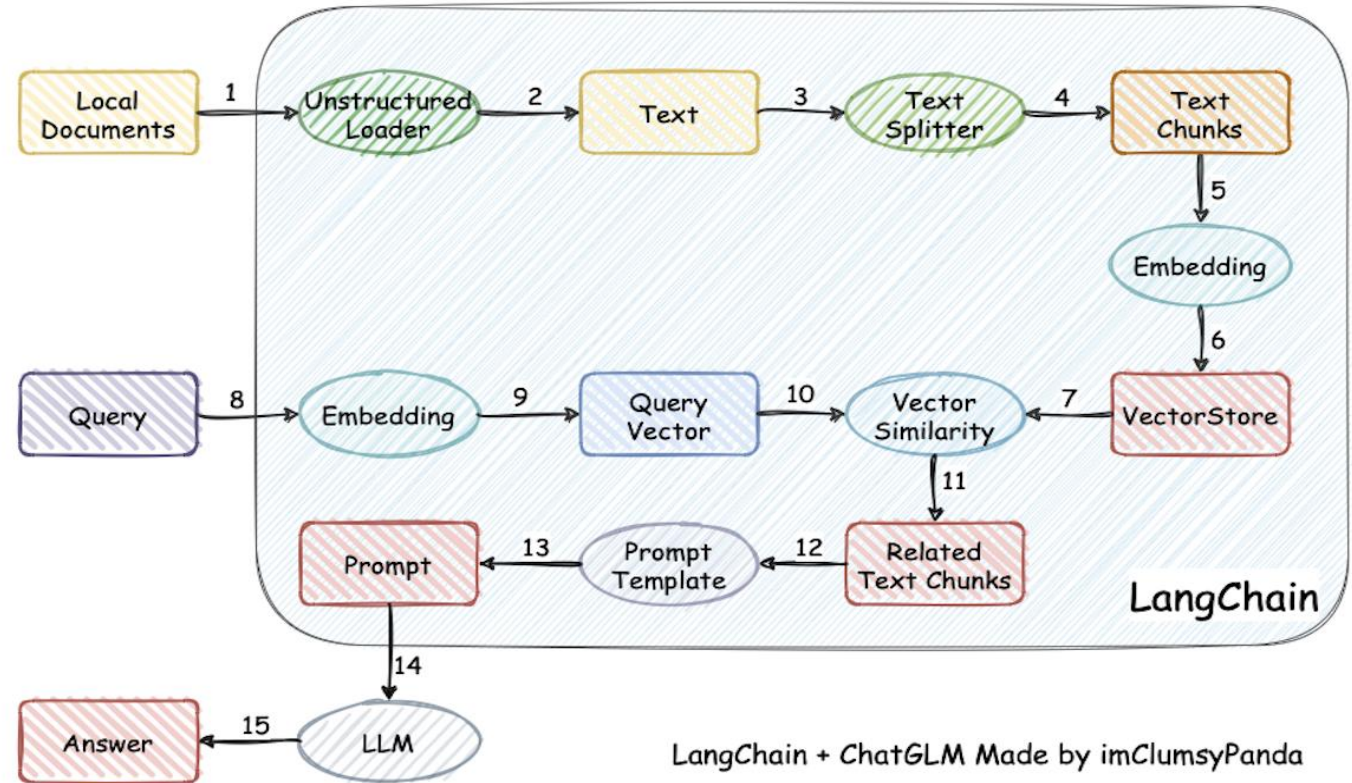
[국세청>국세신고안내>개인신고안내>종합소득세>기본정보>종합소득세 개요 \(nts.go.kr\)](https://nts.go.kr)

- ✓ KMS : FAISS (Facebook AI Similarity Search)

- 기존 자료 탐색을 위한 Index 생성

- ✓ Orchestration Layer : LangChain

- OpenAI, FAISS 연동



KMS와 챗GPT 연동 – 종합소득세 안내

실습 : https://colab.research.google.com/drive/1IMiiNi1m_lvfb1MFjMOYz_swhicFoj8S?usp=sharing

RAG : 테스트 결과



```
2 #####
3 # question = "무신고 가산세 얼마야?"
4 # question = "종합소득세 확정신고 안해도 되는 경우는?" # 오류도 발생
5 # question = "수출 부진 및 산불 피해자 종합소득세 납부 기간은?" #
6 # question = "납부기한 연장 대상자는?" # 표 이해는 어려운 듯
7 # question = "전문사업자는 복식부기 대상인가?"
8 question = "종합소득세 납부 기간은?" # 정보 요약이 상대적으로 좋은 듯
9
10 print(chatbot.run(
11     | prompt.format(query=question)
12 ))
```

종합소득세 납부 기간은 다음해 5월 1일부터 5월 31일입니다. 성실신고확인서 제출자는 6월 30일까지 납부 가능합니다.

```
2 #####
3 # question = "무신고 가산세 얼마야?"
4 # question = "종합소득세 확정신고 안해도 되는 경우는?" # 오류도 발생
5 # question = "수출 부진 및 산불 피해자 종합소득세 납부 기간은?" #
6 # question = "납부기한 연장 대상자는?" # 표 이해는 어려운 듯
7 question = "전문사업자는 복식부기 대상인가?"
8 # question = "종합소득세 납부 기간은?" # 정보 요약이 상대적으로 좋은 듯
9
10 print(chatbot.run(
11     | prompt.format(query=question)
12 ))
```

Yes, 전문사업자 is subject to 복식부기.

RAG : 테스트 결과



```
[13] 1
      2 #####
      3 # question = "무신고 가산세 얼마야?"
      4 question = "종합소득세 확정신고 안해도 되는 경우는?" # 오류도 발생
      5 # question = "수출 부진 및 산불 피해자 종합소득세 납부 기간은?" #
      6 # question = "납부기한 연장 대상자는?" # 표 이해는 어려운 듯
      7 # question = "전문사업자는 복식부기 대상인가?"
      8 # question = "종합소득세 납부 기간은?" # 정보 요약이 상대적으로 좋은 듯
      9
     10 print(chatbot.run(
     11     | prompt.format(query=question)
     12 ))
```

1. 근로소득만 있는 경우
2. 원천징수 의무가 없는 근로소득 또는 퇴직소득이 있는 경우
3. 연말정산을 하지 않은 경우
4. 직전 과세기간의 수입금액이 7,500만원 미만이고, 다른 소득이 없는 보험모집인·방문판매원 계약배달 판매원의 사업소득으로서 소속회사에서 연말정산을 한 경우
5. 퇴직소득과 연말정산대상 사업소득만 있는 경우
6. 비과세 또는 분리과세되는 소득만 있는 경우
7. 연 300만원 이하인 기타소득이 있는 자로서 분리과세를 원하는 경우 등

오류 ~~~ : Hallucination
어떻게 줄일 것인가?

✓ 다음의 경우에 해당되면 종합소득세를 확정신고하지 않아도 됩니다.

- 근로소득만 있는 자로서 연말정산을 한 경우
 - 다만 다음에 해당하는 경우는 확정신고하여야 합니다.
 - 2인 이상으로부터 받는 근로소득·공적연금소득·퇴직소득 또는 연말정산대상 사업소득세를 납부함으로써 확정신고 납부할 세액이 없는 경우 제외)
 - 원천징수 의무가 없는 근로소득 또는 퇴직소득이 있는 경우(납세조합이 연말정산이 규정에 따라 소득세를 납부한 경우 제외)
 - 연말정산을 하지 아니한 경우

```
##%
CHAT_PROMPT_TEMPLATE = ChatPromptTemplate.from_messages(
    [
        SystemMessagePromptTemplate.from_template(
            """
            Answer as thoroughly and truthfully as possible with the following context.
            Find the answer step by step with proofs.
            Answer in Korean with source.
            Use the following format.
            답변 :
                {Answer here}
            출처 :
                {Source fine name here}
            """
        ),
        MessagesPlaceholder(variable_name="history"),
        HumanMessagePromptTemplate.from_template("{input}"),
    ]
)

##%

# S대 챗봇 클래스 정의

class SKK_Talk :

    def __init__(self, dir_base) :

        self.dir_base = dir_base
        os.chdir(dir_base)
        print("Working Directory === ", os.getcwd())
```

```
IPython Console
Console I/A X
In [43]: runfile('C:/MyData/위데이터랩_자료/제품_개발/JJK_1000문(2023Jul)/
test_streaming.py', wdir='C:/MyData/위데이터랩_자료/제품_개발/
JJK_1000문(2023Jul)')

질문 == 등록금 반환 기준 알려줘
Answer =====
답변 :
등록금 반환 기준은 대학의 학칙에 따라 결정됩니다. 일반적으로 등록금은 과오납으로 인한 경우나 관계법령에 특별한 규정이 있는 경우를 제외하고는 반환되지 않습니다. 그러나 학칙에 따라 일부 경우에는 등록금을 반환할 수 있습니다.

학칙에 따르면 등록금 반환의 대상은 다음과 같습니다.
1. 법령에 의해 입학이 불가능하거나 학업을 계속할 수 없는 경우
2. 입학 허가를 받은 자가 입학 포기를 표시한 경우
3. 재학 중인 자가 자퇴를 표시한 경우
4. 휴학 중인 자가 복학하지 않아 제적된 경우
5. 등록한 자가 학기 중에 휴학하는 경우. 단, 병역법상 복무 의무, 임신, 출산, 육아, 창업 등의 사유로 휴학하는 경우에는 등록금을 반환하지 않고 복학 시 등록금을 면제할 수 있습니다.
6. 본인의 질병, 사망, 천재지변 또는 기타 부득이한 사유로 입학을 하지 않게 되거나 학업을 계속하지 않게 된 경우

또한, 계절수업 수강료의 반환 기준도 있습니다. 계절수업 수강료를 납부한 후 본인 사망, 자퇴, 제적, 질병, 천재지변 또는 기타 부득이한 사유로 계절수업에 참가할 수 없는 경우에는 이미 납부한 수강료를 반환할 수 있습니다.

이러한 등록금 반환 기준은 학칙에 따라 상세한 내용이 정해져 있으며, 학교마다 차이가 있을 수 있습니다. 따라서 등록금 반환에 관한 자세한 내용은 해당 대학의 학칙을 참고하시기 바랍니다.

출처 :
- 대학 학칙 (해당 대학의 학칙 파일)
관련 파일 =====
```

www.chatdba.wedatalab.com



안녕하세요 🙋 저는 EZIS ChatDBA입니다.
데이터베이스 관리시스템(DBMS)에 관한 질문을 해주세요.
[예: MySQL과 Oracle Database의 차이를 알려줘.]



inner조인과 outer조인의 차이에 대해 알려줘.



내부 조인은 두 테이블 사이의 공통 필드를 기반으로 두 테이블을 결합하는 조인 유형입니다. 쿼리에 지정된 기준과 일치하는 행만 반환합니다. 반면, 외부 조인은 두 테이블 사이의 공통 필드를 기반으로 두 테이블을 결합하지만, 다른 테이블에 일치하는 항목이 없더라도 한 테이블의 모든 행을 반환하는 조인 유형입니다. 외부 조인에는 왼쪽 외부 조인, 오른쪽 외부 조인, 완전 외부 조인의 세 가지 유형이 있습니다. 왼쪽 외부 조인은 오른쪽 테이블에 일치하는 항목이 없더라도 왼쪽 테이블의 모든 행을 반환합니다. 오른쪽 외부 조인은 왼쪽 테이블에 일치하는 항목이 없더라도 오른쪽 테이블의 모든 행을 반환합니다. 완전 외부 조인은 두 테이블 중 하나에 일치하는 항목이 없더라도 두 테이블의 모든 행을 반환합니다.



ORA-00001 에러에 대해 알려줘.



ORA-00001 오류는 고유 제약 조건 위반을 나타내는 Oracle 오류 메시지입니다. 이는 데이터베이스에 삽입되는 데이터가 기본 키나 고유 인덱스와 같은 고유 제약 조건을 위반한다는 의미입니다. 이 오류는 오류 원인에 대한 추가 정보를 제공할 수 있는 Oracle 로그를 확인하여 추가로 디버깅할 수 있습니다. 문제를 해결하려면 삽입되는 데이터의 순서를 조정해야 할 수도 있습니다. 그래도 문제가 해결되지 않으면 Oracle에 문의하여 추가 지원을 받는 것이 좋습니다.



블랙핑크 멤버에 대해 알려줘.



죄송합니다. DBMS 관련 지식에 대해 질문해주세요. 더 나은 답변을 제공해드릴 수 있습니다.

질문을 입력해주세요.

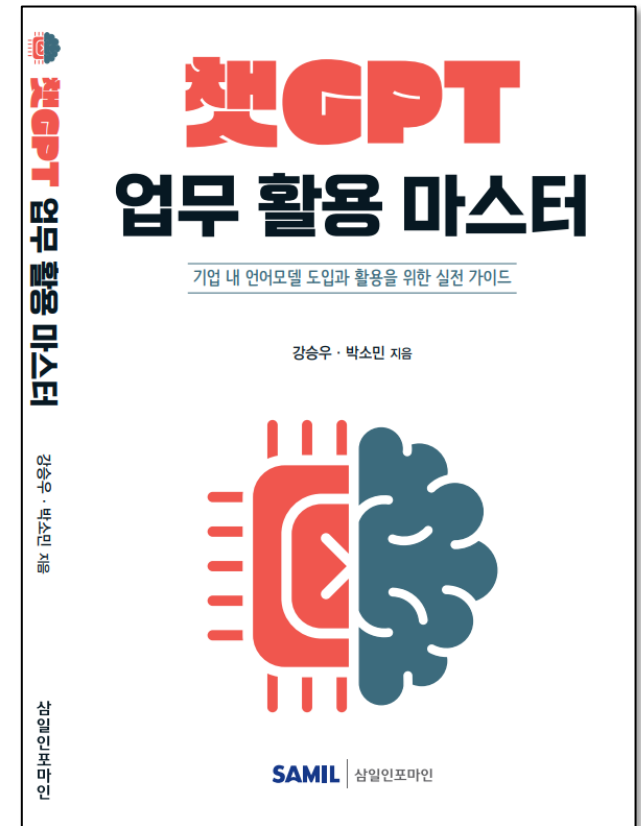


Response in English ☐

EZIS ChatDBA는 Oracle Community ASK TOM과 StackOverflow의 데이터를 기반으로 답변합니다. 해당 데이터가 없을 경우, 구글 검색 결과를 포함합니다. 따라서 사실과 다른 정보를 제공할 수 있습니다.

요약

- RAG = 지시 + 컨텍스트(Context)
 - ✓ 데이터베이스 지식 : SQL
 - ✓ 일반 문서 지식 : 임베딩 후 유사도 비교, 키워드 탐색
 - ✓ 웹 문서/뉴스 지식 : 웹 검색
 - ✓ CSV 파일 : Pandas
- 벡터 데이터베이스 = 임베딩 + 인덱싱
 - ✓ 임베딩 (embedding) : 의미론적 비교
 - 임베딩 방법(모델)에 따른 조회 정확도 차이
 - 임베딩 데이터 처리에 따른 조회 정확도 차이
 - ✓ 인덱싱 (indexing) : (정확하고) 빠른 검색
 - 인덱싱 방법에 따른 조회 정확도와 속도 차이



감사합니다.