Retrieval-Augmented Generation (RAG)

작성자: 석현진

1. 개요

Retrieval-Augmented Generation(이하 RAG)은 검색(Retrieval)과 생성(Generation)을 결합한 최신 자연어 처리 기술로, LLM(Large Language Model)이 보다 정확하고 신뢰도 높은 답변을 생성할 수 있도록 지원하는 방식이다. 최근 인공지능의 할루시네이션(hallucination) 문제를 완화하기 위해 다양한 기업과 연구기관에서 활발히 도입하고 있다.

2. RAG 의 동작 원리

기존 LLM 은 사용자 쿼리에 대해 모델이 학습된 범위 내에서 답변을 생성한다. 하지만 이 방식은 학습되지 않은 정보에 대해 허위 답변을 생성할 가능성이 높다. RAG 는 이러한 한계를 극복하고자 다음과 같은 방식으로 작동한다:

1. 사용자의 질문(Query) 입력
2. 유사한 문서(Document)를 검색 시스템을 통해 벡터 기반으로 검색
3. 검색된 문서를 LLM 에 함께 제공하여, 보다 정확하고 맥락 있는 응답을 생성

3. 주요 구성요소 및 파이프라인

전체 파이프라인:

Raw Data → Text Mining → Embedding → Indexing → Retrieval → Generation

상세 단계:

1. 데이터 로드: 다양한 형식의 데이터(HWP, PDF, HTML 등)를 수집
2. 텍스트 분할(Chunking): 긴 문서를 일정 크기의 텍스트로 분할
3. 임베딩(Embedding): 텍스트를 임베딩 모델을 통해 벡터 형태로 변환
4. 인덱싱(Indexing): 벡터를 효율적으로 검색할 수 있도록 인덱스를 구성
5. 검색(Retrieval): 쿼리와 유사한 벡터들을 검색
6. 생성(Generation): 검색된 문서들과 쿼리를 결합하여, LLM 이 최종 응답 생성

4. 특징 및 장점

항목	내용
정확성 향상	외부 지식 참조를 통해 신뢰도 높은 응답 가능
할루시네이션 완화	사실이 아닌 정보 생성을 줄일 수 있음
학습 불필요	실시간으로 최신 데이터를 벡터 DB 에 반영 가능
확장성 우수	회사 내부 데이터, 실시간 검색 결과 등 다양한 소스와 연계 가능

5. 활용 예시

- 사내 데이터 검색 기반 AI 챗봇: 기업의 문서, 정책, 보고서 등을 실시간 검색해주는 어시스턴트
- 구글 검색 자동화 연계: 예: serper.dev API 를 활용해 웹 검색 결과를 실시간으로 불러오고 LLM 의 응답에 반영
- 비즈니스 인사이트 보고서 생성: 외부 뉴스나 실적 정보를 벡터화하여 분석 리포트 자동 작성

6. 한계 및 고려사항

- 할루시네이션 완전 제거는 어려움
- 벡터 생성 및 검색 성능 중요
- 프라이버시 및 보안 이슈 고려 필요

7. 결론

RAG 는 기존 LLM 의 한계를 극복하고, 정확도와 신뢰도를 강화할 수 있는 핵심 기술로 각광받고 있다. 특히 기업에서 자사의 데이터를 AI 에 접목시키기 위한 실용적인 방법으로 활용도가 매우 높다. 향후에도 정보 신뢰성이 중요한 모든 AI 서비스에서 필수적으로 적용될 것으로 기대된다.

📎

부록: 간단한 RAG 구현 예시 코드 (CrewAI + SerperDev)

아래 코드는 CrewAI 프레임워크와 SerperDevTool(구글 검색 자동화 API)을 활용하여 간단한 RAG 예제를 구현한 것입니다.

```
from crewai_tools import SerperDevTool
from langchain.chat_models import ChatOpenAI
from langchain.agents import initialize_agent, AgentType
from langchain.agents.tools import Tool

search_tool = SerperDevTool()
llm = ChatOpenAI(model="gpt-3.5-turbo")

tools = [
    Tool(
        name="Google Search Tool",
        func=search_tool.run,
        description="최신 정보를 검색하고 결과를 요약해주는 도구"
    )
]

agent = initialize_agent(
    tools,
    llm,
    agent=AgentType.ZERO_SHOT_REACT_DESCRIPTION,
    verbose=True
)

query = "2025 년 전기차 시장 전망은?"
response = agent.run(query)
print(response)
```
