1015 RAG 내용발표

관련 논문 : Retrieval-Augmented-Generation for Knowledge-Intensive NLP Tasks(2020)

https://arxiv.org/abs/2005.11401?ref=pangyoalto.com

RAG (Retrieval검색 - Augmented증강 - Generation생성)

: 질의에 대해 검색하고 이를 바탕으로 응답을 생성하는 모델

sequence - to - sequence 구조 채택 / 검색기와 생성기는 동시에 학습됨 /

R(Retrieval) : 요청된 사항을 어디선가 가져옴

A(Augmented) : 원래의 것에 뭔가 덧붙이거나 보태어 증강됨

G(Generation) : 사용자의 질문에 대한 응답을 텍스트로 생성

[LLM의 한계]

1. 정보의 정확성 문제: LLM은 훈련된 데이터에만 의존하기 때문에 최신 정보나 특정 도메인의 깊이 있는 정보에 대한 답변을 제공하는 데 한계가 있습니다.
2. 모델의 크기와 효율성: 대형 언어 모델은 매우 크고 무겁기 때문에 실시간 응답을 제공하는 데 있어 비효율적일 수 있습니다.
3. 맥락 유지의 어려움: 긴 대화나 복잡한 질문의 경우, 맥락을 유지하면서 정확한 답변을 제공하는 데 어려움을 겪을 수 있습니다.
4. 데이터 편향 문제: LLM은 훈련 데이터의 편향을 그대로 반영할 수 있으며, 이는 부정확하거나 편향된 답변을 초래할 수 있습니다.

[RAG가 LLM의 한계를 극복하는 방법]

1. 실시간 정보 검색

   질문에 따라 외부 데이터를 검색해 답변을 생성하기 때문에 신뢰성 높은 답변 가능

2. 검색을 통한 정보 보완

   기존에 저장되어 있지 않은 정보도 결합해 보완할 수 있다.

3. 정보의 정확성 향상

   LLM 단독으로 작업할 때보다 검색을 통해 세부정보 및 정확한 데이터를 보완해 답변을 제공할 수 있다.

4. 문맥 강화

   검색 결과를 단어가 아닌 문맥을 통해 답변을 생성하기 때문에 복잡한 질문에 대해 더 좋은 답변을 제공할 수 있다.

[RAG의 작동 순서]

- Retriever

1. 사용자의 검색 정보 이해 및 키워드 추출
2. 키워드 바탕의 문서 검색 및 대규모 DB에서 관련성 높은 문서 탐색(딥러닝 기반 문서 임베딩 기술 활용)

- Generator

1. 검색된 문서를 기반으로 정보 추출 및 요약
2. 자연어 생성 모델을 사용하여 자연스러운 응답 제작

[RAG의 두 가지 모델]

1. RAG-Sequence Model
   - 관련 문서를 검색해서 통합하여 최종 답변을 생성
   - 다양한 문서에서 정보를 통합해 더 풍부한 답변 생성 및 연관성 고려 가능
   - 완성된 sequence를 생성하기 위해 같은 문서만을 사용하는 모델

2. RAG-Token Model
   - 관련 문서를 검색해서 필요한 토큰만을 선택해 최종 답변을 생성

- 불필요한 정보는 제외하고 관련성 높은 특정된 정보만 사용가능
- 각 target token마다 다른 문서 사용 가능

[RAG의 장단점]

- 장점

  정보에 기반해서 응답을 생성하기 때문에 질문-응답 시스템에 유리함

  다양한 관점에서의 응답 생성

  특정 주제에 대해 사전 학습된 모델을 사용하기 때문에 매번 학습 불필요 → 시간과 자원 절약

- 단점

  검색을 통해 응답을 생성하기 때문에 문서에 따라 최신 정보가 아니거나 부정확한 정보일 수 있음

  대규모 DB에서 정보를 검색하기 때문에 의존성이 크고 시간도 오래걸림