QA Pair Passage RAG 기반 LLM 한국어 챗봇 서비스

신중민[1,○], 이재욱[2], 김경민[1], 이태민[1], 안성민[3], 박정배[1,*], 임희석[1,*]
[1]Human-inspired AI 연구소, [2]고려대학교 컴퓨터학과, [3](주)오투오
{tlswndals13, jaewook133, totoro4007, taeminlee, insmile, limhseok}@korea.ac.kr[1,2], smahn@o2o.kr[3]

QA Pair Passage RAG-based LLM Korean chatbot service

Joongmin Shin[1,○], Jaewwook Lee[2], Kyungmin Kim[1], Taemin Lee[1], Sungmin Ahn[3], JeongBae Park[1,*], Heuiseok Lim[1,*]
[1]Human-inspired AI Research, [2]Department of Computer Science and Engineering, Korea University, [3]O2O Inc.

요 약

자연어 처리 분야는 최근에 큰 발전을 보였으며, 특히 초대규모 언어 모델의 등장은 이 분야에 큰 영향을 미쳤다. GPT와 같은 모델은 다양한 NLP 작업에서 높은 성능을 보이고 있으며, 특히 챗봇 분야에서 중요하게 다루어지고 있다. 하지만, 이러한 모델에도 여러 한계와 문제점이 있으며, 그 중 하나는 모델이 기대하지 않은 결과를 생성하는 것이다. 이를 해결하기 위한 다양한 방법 중, Retrieval-Augmented Generation(RAG) 방법이 주목받았다. 이 논문에서는 지식베이스와의 통합을 통한 도메인 특화형 질의응답 시스템의 효율성 개선 방안과 벡터 데이터 베이스의 수정을 통한 챗봇 답변 수정 및 업데이트 방안을 제안한다. 본 논문의 주요 기여는 다음과 같다: 1) QA Pair Passage RAG을 활용한 새로운 RAG 시스템 제안 및 성능 향상 분석 2) 기존의 LLM 및 RAG 시스템의 성능 측정 및 한계점 제시 3) RDBMS 기반의 벡터 검색 및 업데이트를 활용한 챗봇 제어 방법론 제안

주제어: NLP, LLM, GPT, Chatbot, Vector Store, RAG, DPR

1. 서론

자연어 처리(NLP) 분야는 지난 수년 간 큰 변화와 발전을 경험하였다. 특히, 초대규모 언어 모델(Large Language Models, LLM)의 등장은 이 분야의 연구 및 응용에 큰 영향을 미쳤다. GPT[1]와 같은 모델들은 다양한 NLP 작업에서 뛰어난 성능을 보이며, 특히 대화형 시스템 및 챗봇 분야에서 주요한 구성 요소로 자리 잡아가고 있다. 상담 분야에서의 인공지능 기반 상담 시스템(Artificial Intelligent Contact Center, AICC)의 요구 증가는 이러한 LLM의 활용 가능성을 더욱 강조하고 있다. 전통적인 챗봇 시스템이 제한된 규칙 기반의 답변을 제공하는 데 반해, LLM은 훨씬 다양하고 맥락에 부합하는 답변을 제공할 수 있다는 큰 장점이 있다. 그러나 이러한 모델의 활용에는 여러 가지 도전 과제와 한계가 존재하는데, 환각(Hallucination)[2-4]과 같이 기대하지 않은 결과를 생성하는 문제는 특히 중요한 연구 주제로 부상하였다. 이러한 문제점을 해결하기 위해 연구자들은 다양한 접근법을 모색하였고, 그 중 하나로 Retrieval-Augmented Generation(RAG) 방법[5]이 큰 관심을 얻었다. RAG는 지식베이스의 검색을 통해 외부지식을 활용하여 언어 모델의 답변을 보완하고 정확도를 향상시키는 방법론이다. 그러나 이러한 방법론도 한계를 가지는데, 지식 기반 대화 모델이 부적절한 지식을 주장하거나 사실과 불일치하는 응답을 생성하는 문제점이 발생하는 것이다.[8-11] 이러한 배경 하에 본 논문은 기존의 딥러닝 시스템을 실제 서비스를 위해 전통적인 규칙화 시스템의 관점으로 응용하는 방법을 제안하고, 실제 어플리케이션을 구축하였다. 이를 위한 실제 방법으로 질의응답 쌍(QA Pair Passage)이 담긴 지식베이스와의 통합을 통해 도메인 특화형 질의응답 시스템의 효율성과 정확도를 개선하는 방안을 제시한다. 또한 Vector Search와 RDBMS(관계형 데이터베이스)를 통합하여 벡터 데이터 베이스를 수정해 챗봇의 답변을 수정 및 업데이트 하는 방안을 제시한다. 이 시스템의 성능은 KorQuad 1.0 데이터셋[6]을 활용하여 평가했으며, 이를 제안하는 방법론의 성능과 한계를 말하고자 한다. 본 논문의 기여는 다음과 같다.

1) QA Pair Passage RAG을 활용한 규칙화된 RAG기반 챗봇 시스템 제안 및 성능 향상 분석 : 질의응답 쌍을 하나의 Passage로 하여 DPR[14]을 수행하는 방법론이다. 기존 Zero-shot과 비교하여 검색 후보군 Top1-4에서 평균적으로 GPT3.5에서 79.44%, GPT4에서 77.94%의 성능향상이 있었고, 기존 RAG의 근거 문서 검색 방법과 비교하여 GPT3.5에서 23.3%, GPT4에서 25%의 성능향상이 있었다.

2) 기존의 LLM 및 RAG 시스템의 성능 측정 및 한계점 제시 : QA Pair Passage RAG 시스템의 실험을 통해 기존 Zero-shot의 한계와 기존 RAG 시스템 기반 챗봇 시스템이 실제 서비스를 위해 성능에서 한계가 있음을 확인하였다.

3) RDBMS기반 벡터 검색 및 업데이트를 통한 챗봇 제어 방법론 제안 : 이러한 방법론을 통해 챗봇을 제어할 수 있다는 가능성을 확인하였다.

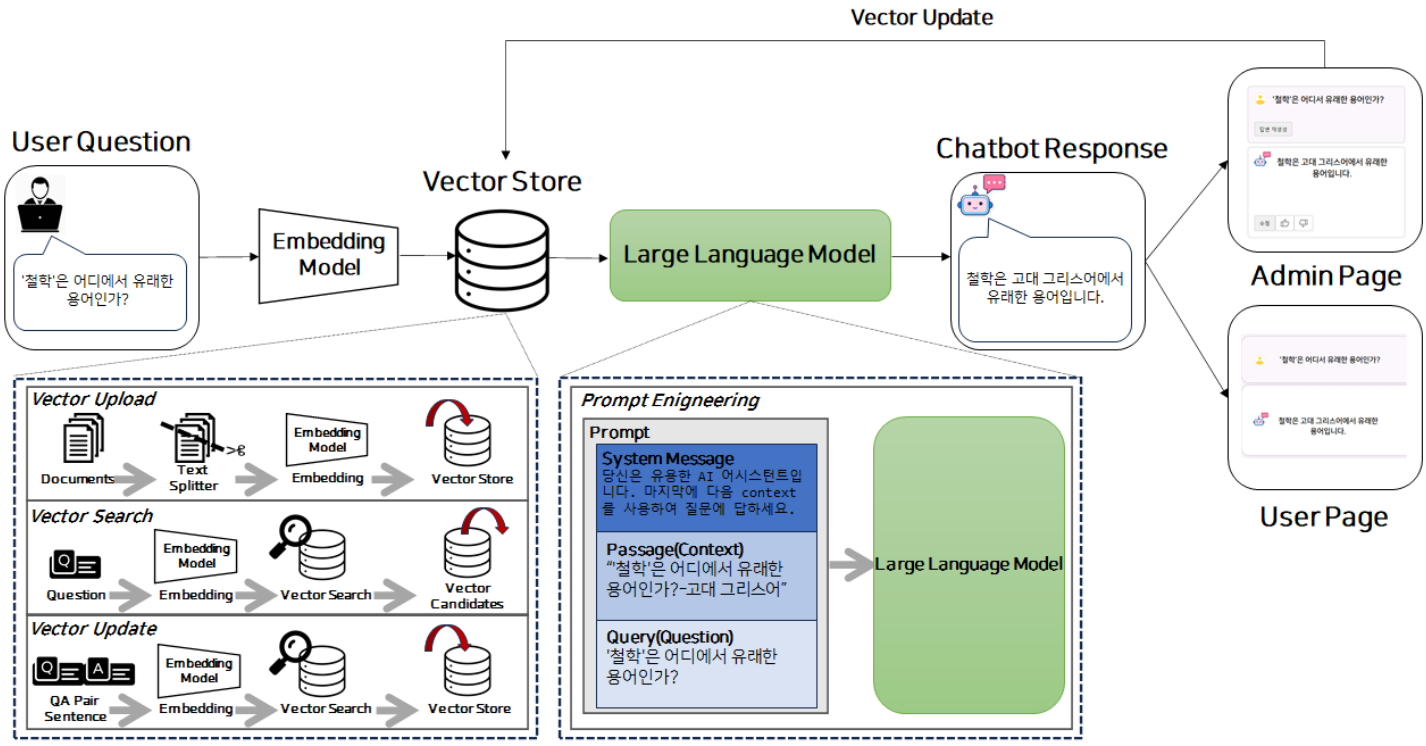


그림 1. 시스템 구조도

2. 관련 연구

최근 몇 년 동안, 자연어 처리(NLP) 분야는 GPT[1]와 같은 초대규모 언어 모델(LLM)의 등장으로 인한 큰 변화를 경험하였다. 이러한 모델들은 대규모의 텍스트 데이터셋을 학습하여 뛰어난 자연어 생성 능력을 보여주며, 요약, 대화 시스템, 기계 번역 등 다양한 응용 분야에서 활용되고 있다. 그 결과로, 챗봇과 같은 대화형 시스템은 사용자의 질문에 더욱 정밀하고 다양한 답변을 제공하는 능력을 갖추게 되었다. 그러나 이러한 발전에도 불구하고, LLM은 여전히 몇 가지 중요한 문제점을 가지고 있다. 특히, 환각(hallucination) 문제는 모델이 실제 데이터에 기반하지 않은, 혹은 입력 정보와 모순되는 정보를 생성하는 현상을 나타낸다. 이 문제는 [3-5] 등의 연구에서 이미 지적되었다. 해당 문제를 해결하기 위한 다양한 연구 방향 중 하나는 지식 기반 대화 시스템의 도입으로, 이 방식은 대화 생성 모델의 성능을 향상시키기 위해 외부 지식 정보를 직접 통합하는 방법론을 사용한다. [6-7] 등의 연구에서는 이러한 접근법의 유효성과 효과를 확인하였다. 그러나, 이러한 지식 기반 방식 또한 완벽하지 않다. [8-11]의 최근 연구에서는 지식 기반 대화 모델이 부적절한 지식을 주장하거나 사실과 불일치하는 응답을 생성하는 문제점을 지적하였다. 따라서 LLM과 지식 기반 대화 시스템의 통합은 분명히 효과적이지만, 여전히 극복해야 할 중요한 과제들이 남아 있다. 본 연구에서는 RAG 시스템이 가지는 장점과 한계에 대해 실험을 통해 분석하고, 질의응답쌍을 하나의 Passage로 사용하여 벡터 저장소에 저장하는 QA Pair Passge 방법론을 제안하며, 추가적으로 이러한 지식 기반 대화 시스템에 지식 베이스 업데이트를 통해 잘못된 챗봇의 응답을 제어하는 방법을 제안하고자 한다.

3. 시스템 구조 : RAG(Retrieval Augmented Generation)

자연어 처리와 인공 지능 분야에서 대규모 언어 모델(Large Language Models, LLM)은 그 능력을 계속해서 입증하고 있다.특히 질문/답변(Question/Answering) 작업에서 이러한 모델은 그 잠재력을 극대화하며, 복잡한 챗봇 구현에 있어 핵심 요소로 부상하였다. 대량의 데이터로 사전학습된 LLM은 학습되지 않은 새로운 질문에도, 그 맥락(context)을 정확히 이해하고 가장 적절한 답변을 제공할 수 있다. 이러한 접근법은 기존의 Rule 기반 방식과 비교하여 높은 유연성과 정확도를 보이지만 단점도 존재하는데, 때로는 매우 그럴듯하게 보이는 잘못된 답변(hallucination)을 생성하는 문제가 발생한다. 이를 해결하기 위해 추가 데이터를 통해 학습을 하는 파인 튜닝(fine tuning)을 고려할 수 있지만, 끊임없이 늘어나는 데이터를 지속적으로 파인 튜닝으로 처리하는 것은 실질적으로 비효율적이다. 이러한 문제를 해결하기 위해 RAG(Retrieval-Augmented Generation) 방법론이 제안되었다.[5] RAG는 기존의 LLM 파라미터를 변경하지 않는 대신, 지식 데이터베이스에서 추출된 외부 지식을 활용하여 답변의 정확도를 개선하는 전략을 채택

한다. 구체적으로, RAG는 prompt engineering 기술 중 하나로서 외부지식과 사용자 질의를 조합하여 프롬프트 입력값을 조합한다. 이러한 방식은 LLM의 파라미터를 재학습할 필요 없이, 외부 지식을 통합하여 더욱 정확하고 신뢰성 있는 답변을 생성하는 데에 큰 기여를 한다. 본 논문에서 제안한 시스템은 그림1과 같이 RAG를 기반으로 구축하였다. 제안 시스템에선 vector store를 지식 데이터베이스로 활용하고, QA Pair Passage를 데이터 베이스(Vector Store)에 저장하여 이를 검색엔진을 통해 근거 문장을 추출하여 사용자 질의와 함께 프롬프트에 입력하는 방법을 사용하였다.

3.1 User Question & Embedding

사용자가 챗봇에게 특정 도메인에 관련한 질문을 하면, 이를 시스템이 이해하여 처리할 수 있도록 임베딩의 과정을 거치게 된다. 임베딩(Embedding)이란 사람이 쓰는 자연어를 기계가 이해할 수 있는 벡터로 변환하는 것이며, 고차원의 데이터를 저차원의 벡터 공간으로 매핑하는 기법을 나타낸다. 이러한 저차원의 벡터는 원본 데이터의 특성을 보존하면서도 계산상의 효율성을 높이는 데 도움을 준다. BERT와 같이 사전학습을 통해 문맥을 학습한 언어모델을 임베딩 모델로 활용하여 자연어를 의미를 함축한 벡터로 변환할 수 있다. 본 시스템에선 OpenAI에서 제공하는 임베딩 모델(text-embedding-ada-002[12])을 사용하였다.

3.2 벡터 저장소(Vector Store)

시스템의 중요한 구성 요소 중 하나인 벡터 저장소(Vector Store)는 벡터 기반의 데이터를 저장하고 검색할 수 있는 효율적인 저장소 혹은 데이터베이스를 지칭한다. 주로 벡터 공간 모델(Vector Space Model)을 기반으로 하는 정보 검색 시스템에서 활용되며, 이 모델은 문서나 단어를 벡터로 표현하여 유사도 계산 및 검색 작업을 수행하는 핵심 원리를 포함하고 있다. 벡터 저장소는 대용량의 벡터 데이터를 저장하고 필요한 벡터를 검색하거나 유사한 벡터를 찾아오는 기능을 제공한다.

벡터 공간 모델(Vector Space Model) 벡터 저장소의 핵심 원리는 벡터 공간 모델에 근거한다. 벡터 공간 모델은 문서나 단어를 고차원 벡터로 표현하는 모델로, 각 차원은 특정한 의미나 속성을 나타낸다. 이 모델을 활용하면 문서나 단어 간의 유사도를 벡터 간의 거리나 각도로 표현하여 계산할 수 있다. 이는 정보 검색 시스템에서 문서와 쿼리 간의 유사도를 평가하는 데 사용된다.

PostgreSQL RDBMS(관계형 데이터베이스)나 NoSQL(비관계형 데이터베이스)와 같은 기존의 데이터베이스는 다차원의 벡터 임베딩 데이터를 저장하도록 설계되지 않았으며, 필요에 따라 효율적으로 확장하는 데 어려움이 있다. 반면 벡터 데이터베이스는 이러한 임베딩 데이터를 처리하고 저장하기 위해 설계된 특별한 종류의 데이터베이스로 벡터 간의 거리나 유사도를 기반으로 데이터를 조회하므로 더효율적인 벡터 검색을 가능하게 한다. 그러나 벡터 저장소는 전통적인 RDBMS와 다르게 SQL과 같은 체계적인 관리를 위한 구조화된 언어가 존재하지 않아 데이터를 삭제하거나 수정하기가 어렵다. 그래서 본 연구에선 기존 관계형 데이터베이스의 구조를 따르면서 벡터 데이터베이스인 pgvector 확장 기능을 제공하는 PostgreSQL[13]을 사용하였다. 이를 통해 의미적 유사도를 기반으로 검색된 데이터를 SQL을 통해 데이터의 입력, 수정, 삭제가 가능하며, 지식베이스를 지속적으로 업데이트 함으로써 챗봇 시스템 또한 지속적으로 업데이트 및 관리가 가능하다.

Vector Upload(QA Pair Passage) 벡터 업로드는 다차원의 데이터 배열인 벡터를 데이터베이스에 효과적으로 저장하고 검색할 수 있게 하는 기능이다. 특히, 머신러닝 및 딥러닝에서 생성된 임베딩 벡터와 같은 고차원 데이터의 저장에 주로 사용된다. 이를 통해 데이터를 데이터베이스에 저장할 때, 문장을 모델에 입력하기 위해 분절하는 청킹 작업의 단위와 이를 통해 만들어진 Passage의 종류에 따라 검색엔진과 챗봇의 성능에 영향을 미친다. 문서 데이터는 PDF, CSV, Json 등의 다양한 형태로 존재할 수 있으나, 제안 시스템에선 청킹 단위 통일하고 문장이 잘못된 단위로 분절되는 것을 막기 위해 Json line의 형태를 사용한다. Json line으로 입력 문서 데이터를 구성하면 하나의 line에 하나의 Passage를 넣어 벡터 저장소에 Passage 단위로 입력이 가능해진다. 또한 본 연구에선 Json line을 Question과 Answer 쌍의 모음으로 구성하는 QA Pair Passage를 제안한다. 이는 사용자 질문(Query)에 따른 근거 문서 검색(Dense Passage Retrieval)의 성능을 향상시키기 위한 것인데, 실제 챗봇 서비스에선 대부분 공통적으로 묻는 질문들이 존재하고, 이를 규칙화 시켜서 대답하는 경우가 많으며, 오히려 일반적인 MRC와 같이 긴 근거문서를 주는 것이 특정 정보에 대한 편향을 발생시켜 더 비효과적일 수 있다. 그래서 규칙화 시스템과 마찬가지로 자주묻는 질문과 답안 쌍을 하나의 Passage로 구성하여 특정 도메인에서 필요한 질문에 잘 대답하도록 하는 것이 목적이다. QA Pair Passage는 [Question—Answer]의 질의응답 쌍 형태로 Passage를 구성하며, 예는 다음과 같다. **예문 : "'철학'은 어디에서 유래한 용어인가?—고대 그리스어"**

Vector Search(Dense Passage Retrieval, DPR) Passage Retrieval은 정보 검색의 핵심 요소로서, 사용자의 검색 쿼리(Query)와 가장 연관성 있는 텍스트 단위, 즉 Passage를 신속하고 정확하게 찾는 작업을 포함한다. Dense Passage Retrieval (DPR)는 정보 검색 분야에서 전통적인 키워드 기반 방식의 한계를 극복하기 위해 고안된 대규모 텍스트 자료집

합에 밀집 벡터를 적용하는 방법이다.[14] 이 방법은 사용자의 쿼리와 Passage의 의미적 유사도를 계산해 필요한 정보를 추출하는데, Cosine similarity나 K-NN(Kth-Nearest Neighbor), A-NN(Approximate Nearest Neighbor) 알고리즘을 활용한다. 벡터 저장소는 Vector Upload를 통해 임베딩된 Passage Vector들이 저장되어있고, 사용자가 질의를 하게되면 임베딩 모델을 통해 벡터 값으로 변환한 뒤 벡터 저장소에 저장된 Passage Vector를 검색하는 Vector Search가 이루어 진다. 이 과정을 통해 사용자는 필요한 정보를 신속하게 접근할 수 있다. 본 시스템에서는 Vector Search를 활용해 두가지의 기능을 구현하였다.

(1) 사용자 질문(Query)와 가장 연관성 있는 QA Pair Passage 추출 : Prompt에 외부 지식으로 Passage를 입력하기 위한 기능

(2) Vector Search를 통해 벡터 데이터베이스 내의 특정 데이터 위치 값 반 : 특정 위치의 Vector 값을 수정 및 삭제(Vector Update)하기 위한 기능

Vector Update 데이터베이스의 데이터 업데이트는 특정 행(row) 또는 열(column)의 값을 수정하는 작업을 의미하는데, 벡터 업데이트는 기존에 저장된 벡터 베이스에서 데이터를 수정할 수 있도록 하는 기능이다. 이러한 구조화된 데이터베이스에서 값을 수정하는 기능은 관계형 데이터베이스(Relational DataBase Management System, RDMS)의 특징인데, 이 기능은 데이터의 정확성과 일관성을 유지하기 위해 필수적이다. 벡터 업데이트는 데이터베이스에서 벡터 검색(Vector Search)을 통해 특정 벡터의 위치를 찾고, 이 위치 값을 기반으로 SQL을 통해 수정한다. 이를 통해 관리자는 지식 베이스 업데이트를 통해 챗봇의 학습되지 않은 추가 지식을 주입할 수 있고, 챗봇의 잘못된 응답이 존재하거나 답변의 질이 떨어질 경우 챗봇 시스템의 답변을 수정 및 제어할 수 있다.

3.3 Large Language Model(Generator)

기존 RAG 시스템에서 Generator는 검색된 정보를 이해하고 의미 있는 답변을 생성하는 역할을 한다. 이 단계에서 Generator는 벡터 저장소 기반 검색엔진을 통해 추출한 문맥과 질문을 이해하며, 선택된 문서에서 필요한 정보를 추출하여 정확한 답변을 생성한다. 대표적인 자연어처리 모델인 BERT나 GPT와 같은 모델이 독자 역할을 수행하며, 문장의 의미를 이해하고 문맥에 부합하는 답변을 생성한다. 본 연구에서는 LLM(Large Language Model)인 GPT3.5와 GPT4를 Generator로 사용하며, 사용자가 모델에게 자연어 입력으로 주는 Prompt의 예시 형태는 다음과 같다.

System Message : "당신은 유용한 AI 어시스턴트입니다. 마지막에 다음 context를 사용하여 질문에 답하세요."

Passage(Context) : "'철학'은 어디에서 유래한 용어인가?-고대 그리스어"

Query(Question) : '철학'은 어디에서 유래한 용어인가?

3.4 User & Admin Page

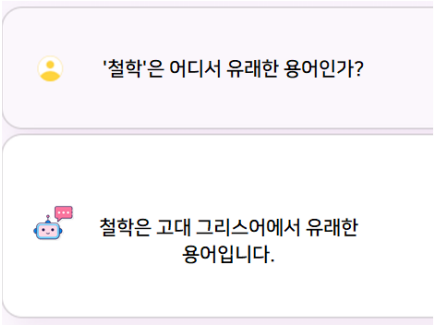


그림 2. User Page

User Page 그림2 은 구축한 시스템의 유저 페이지이다. 유저는 자신이 입력한 쿼리에 대해 RAG 시스템이 생성한 답변을 확인할 수 있다.

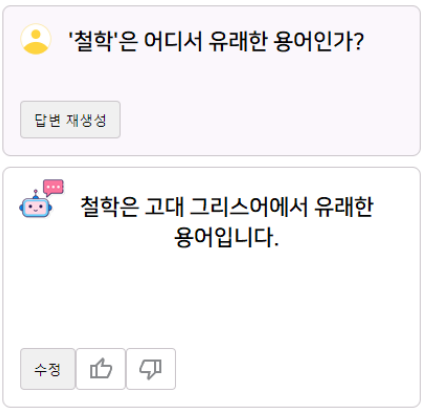


그림 3. Admin Page

Admin Page 그림3는 구축한 시스템의 관리자 페이지이다. 관리자는 유저가 입력한 쿼리와 답변을 확인할 수 있으며, 추가적으로 답변이 잘못되었을 경우 답변을 수정할 수 있다. 관리자가 수정할 텍스트를 입력 후 수정완료를 누르면, 시스템은 지식베이스(Vector Store)에 벡터 검색을 하고, 해당하는 벡터를 찾았을 경우 SQL을 통해 수정된 텍스트 임베딩 벡터를 업데이트 할 수 있다.

4. 실험

4.1 데이터셋과 Metric

본 연구에선 Korquad 1.0[6]의 dev 셋을 사용하였다. Korquad 1.0은 문서를 주고 문서내에서 답변을 찾는 MRC task를 위한 데이터셋이기에 외부에서 검색을 통해 문서를 찾아

문제를 해결하는 Open-domain QA Task[5,14]와는 다르지만, 현재 공개된 텍스트 기반의 한국어 오픈도메인 데이터셋이 존재하지 않고, 이전의 질의응답 시스템에서 빈번히 사용된 데이터셋으로써 모델의 성능에 대해 정성적인 비교가 가능하기에 사용하였다. Korquad 1.0 dev 셋에서 질문 1000개를 랜덤하게 추출하였고, 이를 Zero-shot과 RAG 두가지 방법에서 GPT3.5와 GPT4의 성능을 측정하였다. 또한 MRC는 근거문서에서 정확한 답변인 단어의 위치를 예측하는 task인데, 챗봇은 답변을 문장의 형태로 생성하기에 정확한 성능을 측정하는데 적합한 Metric이 없어 어려움이 존재한다. 그래서 본 연구에선 정답이 챗봇의 생성 답변안에 존재할 경우, 정답을 맞춘 것으로 처리하여 Accuracy Metric으로 성능을 측정하였다.

4.2 실험 결과

4.2.1 검색 성능

RAG 시스템은 검색엔진의 성능에 의존적이므로 좋은 검색 성능을 가지는 것이 중요하다. 본 논문에서는 이러한 검색엔진의 Top 1에서 4까지 성능을 평가하였다. 정답 단어가 검색 결과에 존재할 경우, 정답으로 처리하였다. 또한 의미적 유사도를 구하는 알고리즘은 cosine similarity를 사용하였다.

Dataset	유사도 계산	Top-K	Accuracy
KorQuad 1.0	cosine similarity	1	98.4%
		2	99.1%
		3	99.1%
		4	99.2%

표 1. 검색 모델 성능

표1의 결과를 통해 알 수 있듯이, 본 연구에서 사용된 벡터 저장소 기반 검색은 높은 정확도를 보여주었다. 특히, Top-1 설정에서도 98.4%라는 가장 높은 정확도를 기록하여, 단 하나의 결과만을 반환할 경우에도 높은 성능을 나타냈다. 이러한 높은 성능은 검색의 강건함와 일반화 능력을 시사한다. 그러나 Top-2부터 Top-4까지의 정확도 증가 폭은 상대적으로 적었다. 이는 상위 결과 중에서 정답이 포함될 확률이 이미 높기 때문에 추가적인 결과를 반환함으로써 얻을 수 있는 이득이 제한적이라는 것을 암시한다. 이 결과를 통해 벡터 저장소의 벡터 검색은 효과적인 검색 성능을 제공할 수 있음을 확인하였다.

4.2.2 제안 시스템 답변 성능

구현한 시스템의 답변 성능을 평가하기 위해, Zero-shot과 벡터 검색을 통한 QA Pair Passage RAG 방식을 KorQuad 1.0 데이터셋에 기반하여 GPT3.5와 GPT4 모델의 성능을 비교하였다.

표3의 결과를 통해 알 수 있듯이, QA Pair Passage RAG 방법을 적용했을 때 Top-1에서 최대 GPT3.5 92.7%, GPT4 95.7%의 성능을 보였고, Top-3에서 최소 GPT3.5 89.8, GPT4 95.2%의 성능을 보였다. 또한 Zero-shot과 비교하여 평균적으로 GPT3.5에서 79.44%, GPT4에서 77.94%의 성능향상이 있었다. 이러한 결과를 통해 QA Passage Pair가 챗봇 시스템의 성능면에서 우수한 성능을 보임을 확인하였다.

Dataset	Models	Method	Top-K	Accuracy
KorQuad 1.0	GPT3.5	Zero-shot	0	10.88%
		RAG	1	**92.7(+81.82)**%
			2	90.7(+79.82)%
			3	89.8(+77.92)%
			4	90.1(+78.22)%
	GPT4	Zero-shot	0	17.48%
		RAG	1	**95.7(+78.22)**%
			2	95.4(+78.19)%
			3	95.2(+77.99)%
			4	95.4(+78.19)%

표 2. QA Pair Passage RAG 모델 성능

4.2.3 QA Pair Passage 성능 비교

본 논문에서 제안한 방법론인 QA Pair Passage 방법론의 효과을 평가하기 위해, Korquad 1.0 dev 데이터셋의 원본 근거문서를 Passage로 RAG를 한 것과 QA Pair를 Passage로 한 방법론의 성능을 실험을 통해 비교하였다. 둘다 동일하게 RAG 방법론을 사용하였고 Top-1 Passage를 사용하였으며, 오직 벡터 저장소에 업로드된 Passage의 형태를 Original context와 QA Pair로 차이를 두고 실험하였다.

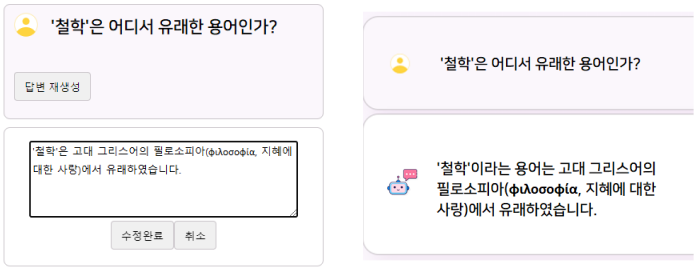
Dataset	Method	Models	Passage	Accuracy
KorQuad 1.0	RAG	GPT3.5	Origianl Context	69.4%
			QA Pair	**92.7(+23.3)**%
		GPT4	Origianl Context	70.7%
			QA Pair	**95.7(+25)**%

표 3. QA Pair Passage 성능 비교

본 실험에서 GPT3.5, GPT4 모두 QA Pair Passage을 사용하면 원본 근거문서 방법과 비교하여 정확도가 각각 23.3%, 25%로 크게 향상되었다. 이는 QA Pair Passage 방법론이 기존의 근거 문서 Context 기반 RAG 모델보다 질문을 이해하고 답하는 데 더 유용함을 알 수 있다.

4.2.4 답변 수정

본 연구에서 구축한 시스템은 챗봇의 답변을 수정하는 기능을 제공한다. 그러나 이를 실험적으로 증명하기가 어려워, 구축 시스템의 관리자 수정화면을 통해 이전의 답변 그림2과 실제 답변이 수정된 그림4을 제시한다. 현재는 이러한 모델의 결과를



(a) Admin Page의 답변 수정 (b) 수정된 User Page

그림 4. 구축 시스템의 답변 수정 결과

통해 제안 방법론의 효과를 확인할 수 있으나, 이를 실험적으로 확인할 수 있는 새로운 데이터셋을 통한 실험이 필요하다.

5. 결론

자연어 처리 분야에서 초대규모 언어 모델의 중요성은 다양한 NLP 작업과 챗봇 분야에서의 활용 가능성을 통해 명확하게 확인되었다. 본 논문에서는 이러한 모델의 활용과 관련된 도전과 한계를 극복하기 위한 방법을 탐구하였다. 특히, QA Pair Passage(질의응답 쌍)을 활용한 도메인 특화형 질의응답 시스템의 효율성과 정확도 향상 방안을 제시하였으며, 그 결과로 기존 모델에 비해 상당한 성능 향상을 보였다. 또한, RDBMS 기반 벡터 검색과 업데이트를 활용한 챗봇 제어 방법론을 제안하였고, 이를 통해 챗봇의 답변을 보다 효과적으로 제어할 수 있는 가능성을 확인하였다. 그러나, 이 방법론의 정량적인 효과는 추가적인 실험을 통해 더욱 명확히 해야 할 필요가 있다. 본 연구의 결과는 자연어 처리 분야의 연구 및 응용에서 LLM의 활용을 더욱 효과적으로 하기 위한 방향을 제시한다. 향후 연구에서는 QA Pair Passage가 성능을 크게 향상시키는 이유와 이 형식에 맞게 모델을 더욱 최적화하는 방법, 제안 구조에서의 멀티턴의 대화의 성능을 확인하고자 한다. 또한 답변 수정기능을 실험적으로 확인할 수 있는 새로운 데이터셋을 구축하여 실험적으로 증명하고, 벡터 업데이트를 활용해 더욱 챗봇의 성능을 키우는 방안에 대해 연구하고자 한다.

감사의 글

본 연구는 과학기술정보통신부 및 정보통신기술기획평가원의 대학ICT연구센터지원사업의 연구결과로 수행되었음 (IITP-2022-2018-0-01405) 본 논문은 교육부와 한국연구재단의 재원으로 지원을 받아 수행된 3단계 산학연협력 선도대학 육성사업(LINC 3.0)의 연구결과입니다. 이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기술기획평가원의 지원을 받아 수행된 연구임 (No. 2020-0-00368, 뉴럴-심볼릭(neural-symbolic) 모델의 지식 학습 및 추론 기술 개발)

참고문헌

[1] N. R. Tom B. Brown, Benjamin Mann et al., “Lan-guage models are few-shot learners,” ArXiv preprint arXiv:2005.14165, 2020.

[2] Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018.Faithful to the original: Fact aware neural abstractive summarization. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI18 30th, New Orleans, Louisiana, USA, February 2-7, 2018, pages 4784–4791. AAAI Press

[3] Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. “Neural text summarization: A critical evaluation.”, EMNLP-IJCNLP 2019, pages 540–551, Hong Kong, China. Association for Computational Linguistics.

[4] Feng Nie, Jin-Ge Yao, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019a. A simple recipe towards reducing hallucination in neural surface realisation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2673–2679, Florence, Italy. Association for Computational Linguistics

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”, NeurIPS 2020, arXiv:2005.11401

[6] Seungyoung Lim, Myungji Kim, Jooyoul Lee”KorQuAD1.0: Korean QA Dataset for Machine Reading Comprehension

[7] Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing

[8] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In ICLR.

[9] Hannah Rashkin, David Reitter, Gaurav Singh Tomar, and Dipanjan Das. 2021. Increasing faithfulness in knowledge-grounded dialogue with controllable features. In ACL/IJCNLP, pages 704–718. Association for Computational Linguistics.

[10] Yingxiu Zhao, Yinhe Zheng, Zhiliang Tian, Chang Gao, Bowen Yu, Haiyang Yu, Yongbin Li, Jian Sun, and Nevin L. Zhang. 2022b. Prompt conditioned VAE: enhancing generative replay for lifelong learning in task-oriented dialogue. CoRR, abs/2210.07783.

[11] Nouha Dziri, Sivan Milton, Mo Yu, Osmar R. Zaïane, and Siva Reddy. 2022. On the origin of hallucinations in conversational models: Is it the datasets or the models? In NAACL, pages 5271–5285. Association for Computational Linguistics.
[12] OpenAI,“New and improved embedding model”, https://openai.com/blog/new-and-improved-embedding-model
[13] pgvector, https://github.com/pgvector/pgvector
[14] Vladimir Karpukhin, Barlas Oğuz, Sewon Min and others, ”Dense Passage Retrieval for Open-Domain Question Answering”EMNLP 2020, arXiv:2004.04906