

대규모 언어 모델을 활용한 최적화된 교육용 플랫폼 개발

김규민^o, 길상현, 안승빈, 이민주, 김유섭

한림대학교 정보과학대학 소프트웨어학부

{kyumin6379, rlfkdgus123, asb0729, minju123486}@gmail.com, yskim01@hallym.ac.kr

Development of an Optimized Educational Platform Using Large Language Models

Kyu-Min Kim^o, Sang-Hyeon Gill, Seung-Been Ahn, Min-Ju Lee, Yu-Seop Kim

Division of Software, College of Information Science, Hallym University

요약

본 연구는 생성형 AI, 특히 대규모 언어 모델(LLM, Large Language Model)을 활용하여 자동 문제 생성, 학습 일기 작성, 챗봇을 통한 개인화된 피드백 제공 기능을 통합한 교육 플랫폼을 개발하고, 이를 교육 현장에 적용하는 방법을 탐구하였다. 이 플랫폼은 React.js와 Django 프레임워크를 기반으로 개발되었으며, 문제 생성 서비스는 Retrieval-Augmented Generation(RAG) 기법을 사용하여 실시간으로 높은 품질의 문제를 생성한다. 학습 일기 생성 과정에서는 프롬프트 엔지니어링(Prompt Engineering) 기법을 적용하여 사용자가 제공한 정보를 바탕으로 개인화된 학습 일기를 자동으로 작성하며, 챗봇은 Context-Augmented Retrieval(CAR) 기법을 활용하여 사용자에게 신속하고 정확한 맞춤형 응답을 제공한다. 본 연구는 LLM이 교육 분야에서 어떻게 활용될 수 있는지를 실증적으로 보여주며, 한국어 교육에서의 잠재력을 탐구하였다. 본 논문에서는 이러한 연구 결과를 바탕으로 LLM 기반 교육 도구의 가능성과 향후 발전 방향을 제시한다.

주제어: Large Language Model(LLM), Retrieval-Augmented Generation(RAG), Prompt Engineering, 교육 플랫폼

1. 서론

최근 인공지능(AI, Artificial Intelligence) 기술의 발전은 다양한 분야에서 혁신적인 변화를 일으키고 있으며, 특히 교육 분야에서의 적용 가능성이 주목받고 있다 [1]. 그중에서도 대규모 언어 모델(LLM, Large Language Model)은 자연어 처리와 생성 능력을 통해 교육에 새로운 가능성을 제시하고 있다 [2]. 기존의 교육 시스템은 학생 개인의 학습 필요와 속도에 맞추기 어려운 한계가 있었다. 전통적인 교실 수업은 다수의 학생을 대상으로 하기 때문에, 각 학생의 학습 수준에 따라 맞춤형 피드백을 제공하거나 문제를 생성하는 것이 어렵다. 그래서 LLM을 활용한 문제 생성, 학습 일기 작성, 개인화된 피드백 제공과 같은 기술적 접근이 점점 더 중요해지고 있다 [3]. LLM은 대량의 텍스트 데이터를 학습하여 언어의 문맥과 의미를 이해하고 생성할 수 있는 능력을 갖추고 있으며, 이를 바탕으로 사용자에게 맞춤형 학습 경험을 제공할 수 있다 [4].

본 논문에서 개발한 교육 플랫폼은 이러한 교육적 요구를 충족시키기 위해 개발했다. 특히, 한국의 교육 환경에서는 학생들이 학습에서 겪는 어려움을 실시간으로 파악하고 이에 대한 맞춤형 피드백을 제공하는 시스템의 필요성이 대두되었다 [5]. 기존의 교육 도구들은 대체로 정형화된 콘텐츠와 피드백을 제공하는 데 그쳤으며, 학생들의 개별적인 학습 스타일과 필요에 맞추기에는 한계가 있다 [6]. 이러한 배경에서, 본 연구는 LLM을 활용하여 학습자 개인에게 최적화된 교육 경험을 제공할 수 있는 플랫폼을 개발하게 되었다. 개발된 교육 플랫폼은

React.js와 Django 프레임워크(Framework)를 기반으로 설계되었으며, Retrieval-Augmented Generation(RAG) 기법을 사용하여 실시간으로 높은 품질의 문제를 생성한다 [7]. 학습 일기 작성은 학생들이 학습 내용을 체계적으로 정리하고, 스스로 학습 과정을 돌아볼 수 있도록 돕는 중요한 도구이다. 그러나 많은 학생이 일기를 쓰는 데 귀찮음을 느끼거나 많은 시간을 소모하는 경우가 많다. 이러한 문제를 해결하기 위해, LLM과 Prompt Engineering을 활용하여 자동화된 학습 일기 생성 시스템을 개발했다. 또한 Context-Augmented Retrieval(CAR) 기법을 활용한 챗봇(Chatbot)을 통해 정보 검색 및 답변 생성 시간을 크게 단축함과 동시에 우수한 품질의 답변을 생성한다 [8]. 본 논문은 LLM이 교육 분야에서 어떻게 활용될 수 있는지를 실증적으로 보여주며, 한국어 교육을 포함한 다양한 교육 분야에서의 적용 가능성을 탐구한다. 이를 통해 LLM 기반 교육 도구의 가능성과 향후 발전 방향을 제시하고자 한다.

2. 방법론

본 논문에서는 LLM을 활용하여 자동 문제 생성, 개인화된 피드백, 학습 일기 작성, 그리고 챗봇 기능을 통합한 교육 플랫폼을 개발하였다. 방법론에서는 개발 환경에 대해서 먼저 다루고, RAG 기법을 활용해 생성된 문제를 통해 학생들의 학습 성과를 평가하고, 이를 바탕으로 맞춤형 피드백을 제공하는 과정을 상세히 설명한다. 또한 사용자가 입력한 정보에 기반한 개인화된 학습 일기를 생성하고, 마지막으로 CAR 기법을 활용한 챗봇을 설

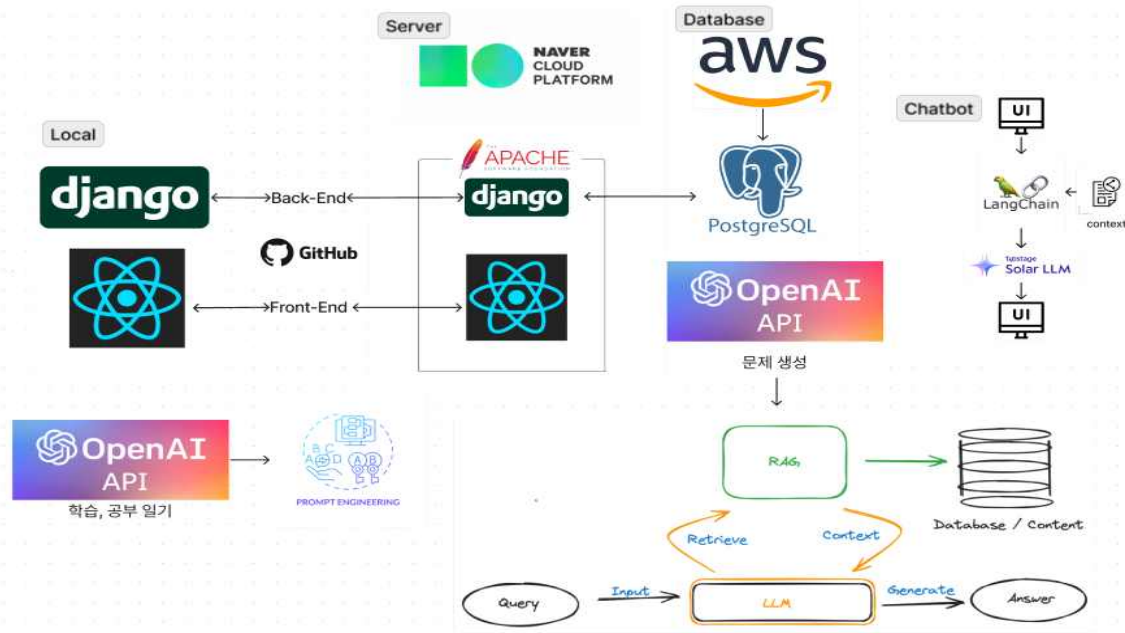


그림 1. 시스템 아키텍처

명한다.

2.1 개발 환경

본 논문에서 개발한 플랫폼은 최적화를 위해 프론트엔드(Frontend)와 백엔드(Backend) 간의 원활한 통신을 기반으로 설계되었다. 그림 1은 본 논문에서 개발한 교육 플랫폼의 시스템 아키텍처다. Frontend는 React.js를 사용하여 사용자 인터페이스(Interface)를 구현하고, Backend는 Django 프레임워크를 사용하여 데이터 처리와 비즈니스 로직(Business logic)을 담당한다.

2.1.1 소프트웨어의 구성

Frontend (React.js)

React.js는 사용자와의 상호작용을 처리하며, 데이터를 Backend와 주고받는다. 주요 기능은 다음과 같다.

- 데이터 요청: 사용자가 요청을 수행하면, React는 Fetch API(Application Programming Interface)를 사용해 비동기식으로 Backend에 요청을 전송한다.
- 데이터 표시: Backend로부터 받은 데이터를 UI(User Interface)에 반영한다.
- 상태 관리: React의 상태 관리 시스템을 통해 플랫폼의 상태를 효율적으로 관리한다.

Backend (Django)

Django는 데이터 처리와 API 통신을 담당하며, 주요 기능은 다음과 같다.

- API 엔드포인트(Endpoint): Django REST framework를 통해 API를 제공하여 Frontend와 통신한다.
- 데이터베이스 관리: Django Object-Relational Mapping(ORM)을 사용해 데이터베이스(Database)와 상호작용하며, 데이터를 관리한다.
- 비즈니스 로직 처리: 사용자 요청에 따른 비즈니스

로직을 처리하여 응답을 생성한다.

2.1.2 통신 구조

React와 Django는 Hyper Text Transfer Protocol(HTTP)를 통해 데이터를 주고받으며, 사용자의 요청이 Frontend에서 Backend로 전달되고, Backend는 요청을 처리한 후 응답을 생성하여 Frontend에 전송한다. 이러한 구조는 데이터를 신속하게 처리하여 사용자에게 일관된 경험을 제공한다.

2.2 문제 생성: Retrieval-Augmented Generation

문제 생성 기능은 그림 2에 나와 있는 플로우차트(Flowchart)를 기반으로 실행된다. 교수자가 강의실을 생성하고, 그 안에서 문제 유형과 개수를 선택한 후 특정 주제를 지정하여 문제를 자동으로 생성할 수 있도록 설계했다. 이 과정에서 교수자가 문제를 출제할 주제를 선택하면, 시스템은 해당 주제와 관련된 크롤링이 허가된 공식 문서를 가져와 필요한 정보를 수집한다. 이러한 접근은 문제의 신뢰도와 품질을 높이는 데 기여한다. 이렇게 수집된 정보를 바탕으로 문제를 생성하는 과정을 RAG라고 한다. RAG는 LLM의 프로세스 속도를 높이고, 웹 기반의 실시간 정보를 통합하여 정보에 입각한 응답을 생성하는 데 특히 유용하다. 이 방법은 단순히 기존 데이터에 의존하지 않고, 최신의 외부 리소스를 활용하여 문제를 생성한다. 또한, Prompt를 활용하여 문제의 유형, 난이도, 형식 등을 구체적으로 지정함으로써, LLM이 생성하는 문제의 품질이 기존보다 더욱 높아지는 것을 확인할 수 있다 [9]. 예를 들어, 문제의 구조나 답변 형식을 세밀하게 지정함으로써, 단순한 모델의 기본 응답보다 학습 목적에 부합하는 고품질의 문제를 제공한다. 이렇게 생성된 문제는 교수자가 최종 검토한 후 등록되며, 학생들은 생성된 문제를 실시간으로 받아 풀이하고

제출할 수 있다. 이러한 시스템은 학습의 유연성과 효율성을 높이며, 교수자는 최신 정보를 반영한 고품질 문제를 손쉽게 제공할 수 있게 된다.

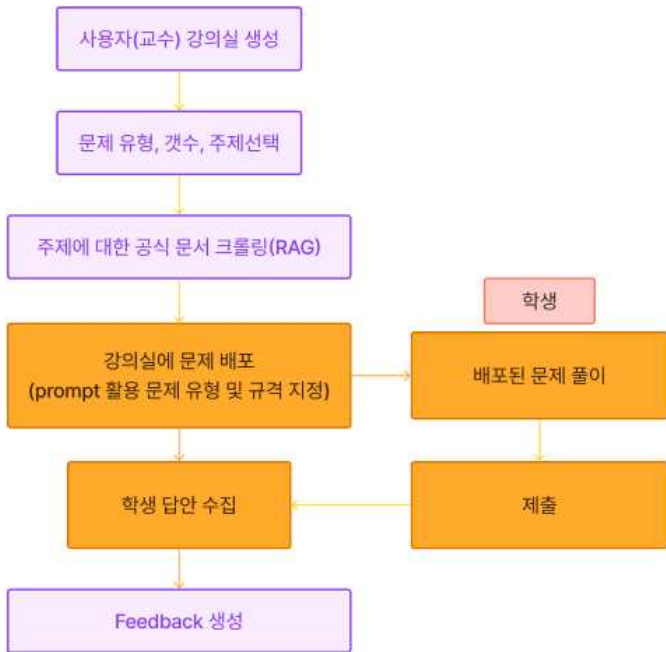


그림 2. 문제 생성 플로우차트

2.3 피드백

피드백 시스템은 문제 생성 후 학생들의 답안을 평가하여, 교수자에게는 학생들의 전반적인 학습 정보를 제공하고, 학생들에게는 개인별 맞춤형 피드백을 제공한다. 이를 통해 학생들의 학습 수준을 실시간으로 파악하고 부족한 부분을 진단하여 개선하는 것에 목적을 둔다. 피드백 제공 과정은 그림 3에 나와있는 플로우차트를 기반으로 실행된다. 학생들이 문제 풀이 후 답안과 실제 정답을 프롬프트와 결합한 뒤 LLM에 입력하여 각 답안의 정답률을 평가한다. 분석된 결과는 UI를 통해 교수자에게 시각적으로 제공된다. 특히, 정답률은 히스토그램으로 표시되어, 교수자가 각 문제의 정답률 분포를 쉽게 확인할 수 있도록 한다. 이를 통해 교수자는 어떤 문제가 학생들에게 어려운지, 어떤 개념이 충분히 이해되지 않았는지를 직관적으로 파악할 수 있다. 분석 결과를 바탕으로 학생들에게는 개별화된 피드백이 제공된다. LLM은 학생들의 답안에서 발견된 오류와 학습의 부족한 부분을 설명하고, 향후 학습 방향에 대한 조언을 제시한다. 이 피드백은 학생 개인의 학습 기록과 성과를 반영하여 맞춤형으로 제공되며, 이를 통해 학생은 자신이 어떤 부분을 보완해야 하는지 명확히 인지할 수 있다. 결과적으로, 문제 생성과 피드백 제공을 통해 학습의 질을 전반적으로 향상하는 데 기여한다.

2.4 학습 일기 생성: Prompt Engineering

학습 일기 생성 기능은 사용자가 제공한 정보와 페르소나(Persona)를 바탕으로 자연스러운 개인화된 학습 일기를 생성한다. 시스템은 그림 4에 나와있는 플로우차트를 기반으로 실행된다. 사용자가 입력한 7가지 정보를

바탕으로 인사말, 본문, 맺음말로 구성된 일기를 자동 생성하며, 각 부분은 사용자의 성향과 학습 스타일을 반영하도록 설계되었다. 인사말은 개인화된 일기를 작성하기 위해, 별도로 개발된 함수와 입력된 프롬프트를 통해, 사람이 직접 작성한 것처럼 자연스럽게 출력된다. 본문은 사용자가 무엇을 공부했는지를 중심으로 섹션별로 정보를 구성하여 작성되며, 맺음말에서는 본문에서 추출된 주요 키워드가 해시태그(Hashtag)로 생성된다. 이렇게 완성된 학습 일기는 마크다운(Markdown) 형식으로 출력되어 시각적으로 확인 가능하며, 상단의 에디터(Editor)를 통해 수정과 저장이 용이하다. 이를 통해 사용자는 단순한 기록을 넘어 자기 주도적 학습을 위한 도구로 활용할 수 있으며, 자유롭게 일기를 작성하고 공유함으로써 학습의 질을 높일 수 있다.



그림 3. 피드백 플로우차트

2.5 챗봇: Context-Augmented Retrieval

챗봇 기능은 교내 연구실 및 인턴십 정보를 포함한 다양한 교내 정보가 기재된 Context 기반의 질의응답을 제공하며, 사용자의 질문 유형에 따라 Context를 활용하거나 기본 모델의 응답을 제공하는 방식으로 그림 5에 나와있는 플로우차트를 기반으로 실행된다. 여기서 사용된 Context는 대화의 맥락을 의미하는 기존의 Context와는 달리, 특정 주제와 관련된 정보 집합체로 정의되며, 이를 통해 챗봇은 단순한 문맥 이해를 넘어, 사용자에게 실질적인 도움을 줄 수 있는 정보를 제공한다.

챗봇의 주요 기능은 다음과 같다. 사용자가 교내 연구실이나 인턴십 정보에 대해 질문할 경우, 챗봇은 사전에 주어진 Context에서 관련 정보를 검색하고, 이를 기반으로 정확한 답변을 제공한다. 예를 들어, "운영체제 연구실의 지도교수는 누구인가요?"라는 질문에 챗봇은 Context에 기록된 정보를 참조하여 적절한 답변을 생성한다. 반면, Context와 관련이 없는 일반적인 질문에 대해서는 기본 LLM 모델이 응답하게 된다. 챗봇의 중요한 기능 중 하나는 대화 히스토리(history)를 기억하고 참조하여 응답을 제공하는 능력이다. 예를 들어, 이전에 질문했던 내용을 기억하여 사용자가 후속 질문을 할 때

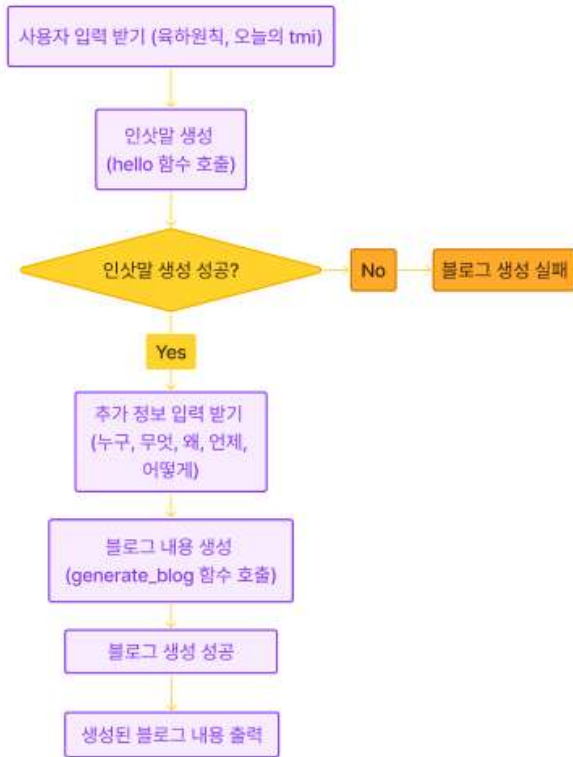


그림 4. 학습 일기 생성 플로우차트

도 일관된 정보를 제공한다. 이러한 기능은 대화의 연속성과 사용자 경험을 향상시키는 데 기여한다. 또한, 챗봇은 탈옥(Jailbreaking)과 같은 유해한 질문에 대한 대응책도 포함하고 있다 [10]. 사전 프롬프트 학습을 통해 이러한 질문을 회피하도록 설계되어 있으며, 안전하고 책임 있는 대화 환경을 유지한다. 챗봇은 Upstage의 Solar-1-mini-chat 모델을 기반으로 구현되었으며, 이 모델은 107억 개의 매개변수를 가진 사전 학습된 작은 규모의 모델이다 [11]. 이 모델의 장점 GPT-4(Generative Pre-trained Transformer-4) [12]와 같은 대규모 모델에 비해 빠른 응답 속도를 제공하며, 응답이 틀린 경우에도 다시 질문하기에 적합한 유연성을 제공한다. 이로 인해, 사용자는 보다 빠르고 적절한 피드백을 받을 수 있다. 챗봇의 CAR 기법은 문제 생성에서 사용된 RAG와 유사한 로직을 사용하여, 정보를 제공하거나 사용자의 질문 유형에 맞는 최적의 응답을 생성하는데 중점을 두었다. 이러한 접근은 Context 기반의 정보를 활용하여 사용자에게 맞춤형 응답을 정확하게 빠르게 제공함으로써 챗봇의 효율성을 높인다.

3. 실험 및 결과

개발된 교육 플랫폼의 주요 기능들의 프롬프트와 실제 출력된 결과를 제시한다. 그림 6은 교수자가 문제 유형과 주제, 개수를 선택해서 제출한 문제를 학생들이 받아보는 실제 UI다.

그림 2에서 나와있는 흐름에 따라 학생들이 답안을 제출하면 개개인 답안에 기반한 피드백을 생성하게 된다. 표 1은 피드백을 생성하는 모델의 프롬프트 일부분을



그림 5. 챗봇 플로우차트



그림 6. 문제 생성 실제 UI

작성한 표다. 개인화된 피드백을 제공하기 위해서 작성된 프롬프트이며, 표 1에서 제시된 프롬프트 이외에도 좋은 출력을 내기 위한 프롬프트도 포함되어 있다 [13].

피드백 프롬프트 예시	
1.	학생들의 수업 내용에 관한 이해를 체크하기 위해 낸 문제들입니다.
2.	답안을 종합해서 문제별 이해도를 반영한 피드백을 작성하세요.

3. 최소 석사 학위 이상의 전문가가 피드백을 한다는 가정하에 자세하게 피드백을 작성하세요.

표 1. 피드백 생성 프롬프트 일부

이렇게 작성된 프롬프트와 학생들의 답안을 종합하여 교수자와 학생들에게는 피드백이 제공된다. 표 2는 실제 교수자에게 제공되는 피드백의 출력을 표로 나타낸 것이다.

피드백 실제 출력(교수자)
스마트 포인터를 이해하는 데는 어려움이 없어 보임. 두 종류의 스마트 포인터인 unique_ptr과 shared_ptr을 구분하는 데는 어느 정도 성공했지만, 그것이 어떻게 작동하는지는 완전히 이해하지 못한 것 같다. 특히, unique_ptr의 소멸자가 무엇을 자동으로 해주는지에 대한 이해가 전혀 없다. 이에 대한 추가적인 설명과 실습이 필요하다.

표 2. 피드백 실제 출력

그림 7은 사용자가 입력한 정보에 따라 생성된 학습 일기의 실제 UI를 보여준다. 학습 일기 생성 과정에서는 인사말, 본문, 맺음말의 세션으로 구성되어 있다. 프롬프트는 사용자 입력을 바탕으로 각 세션의 내용을 작성하는 역할을 하며, 특히 본문 부분은 주제에 대한 7가지 세션을 나누어 세부적으로 작성되도록 설계되었다. 인사말과 맺음말을 통해 개인화된 출력을 생성할 수 있도록 프롬프트를 설계했다. 학습 일기는 Markdown 형식으로 출력되며, 사용자가 직접 수정하거나 저장할 수 있는 에디터 기능을 제공하여, 효과적으로 기록할 수 있는 편리성과 유연성을 동시에 제공한다.

섹션 2: 자연어처리 공부와 역사와 발전 과정

NLP는 1950년대부터 연구되기 시작했으며, 초기에는 규칙 기반 시스템이 많이 사용되었는데, 최근에는 기계 학습 특히 딥러닝 기술의 발전으로 크게 진화했다. 구글의 BERT나 OpenAI의 GPT 시리즈 같은 모델들이 NLP의 혁신적인 발전을 이끌고 있어

섹션 3: 자연어처리 공부에 대한 원리와 구조

NLP는 다양한 알고리즘과 기술로 구성되어 있으며, 예를 들어, 토큰화는 텍스트를 작은 조각으로 나누는 과정이고, 파싱은 문장의 구조를 분석하는 것이다. 이러한 기술들이 복합적으로 작용하여 텍스트에서 의미를 파악하고, 적절한 반응을 생성할 수 있으며, 예를 들어, 챗봇은 이런 원리를 사용해서 우리의 질문에 답변을 해줍니다!

섹션 4: 자연어처리 공부를 활용한 사례

그림 7. 학습 일기 생성 실제 UI

표 3은 챗봇 기능이 사용자와의 대화에서 제공한 실제 응답을 보여주며, 각 질의에 대한 챗봇의 답변을 기재했다. 챗봇은 사용자의 질문에 따라 Context에 포함된 정보를 기반으로 답변을 생성하며, 이전 대화 내용을 기억하고 연속적으로 응답하는 기능을 제공한다. 실제 UI에는 챗봇이 대답한 내용이 개인정보가 포함되어 있어, 프라이버시 보호를 위해 표에 적절히 익명화하여 작성하였다.

챗봇 실제 출력
사용자: 운영체제 연구실 지도교수님 누구야?
챗봇: 운영체제 연구실의 지도교수님은 A교수님입니다.
사용자: 교수님 이메일 좀 줄래?
챗봇: A교수님의 이메일은 A@gmail.com입니다.
사용자: A교수님 연구실에서는 주로 뭘 다루?
챗봇: A교수님의 연구실인 운영체제(OS) 연구실은 운영체제 및 임베디드 소프트웨어를 주로 다룹니다.

표 3. 챗봇 실제 출력

5. 결론 및 향후 연구

본 논문에서는 개인화된 학습 서비스를 제공하기 위해 LLM을 활용하여 문제 생성, 피드백 제공, 학습 일기 생성, 챗봇 기능을 통합한 교육 플랫폼을 개발하였다. 문제 생성에서는 RAG를 사용하여 문제의 신뢰도를 높였다. 또한 문제를 푼 뒤 교수자에게는 문제별 학생들의 이해도에 기반한 피드백, 학생들에게는 개개인 맞춤형 피드백이 제공된다. 프롬프트 엔지니어링을 활용한 개인화된 학습 일기 생성과 챗봇의 기능을 통합함으로써 교육 플랫폼을 구현할 수 있었다. 추후 분야를 확장하고, 초, 중, 고등학교까지 사용 범위 또한 확장할 수 있다. 본 논문에서는 학습 위주로 진행하였지만, ADHD, MBTI 등 다양한 진단검사나 심리검사 등 개인화된 피드백을 활용해 개발할 수 있다.

6. 사사의 글

이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원[RS-2021-II212068, 인공지능 혁신 허브 연구 개발 (서울대학교, AI연구원)]과 2022년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2022R1A5A8019303).

참고문헌

[1] Holmes, W., Bialik, M., & Fadel, C., "Artificial Intelligence in Education: Promises and Implications for Teaching and Learning," Center for Curriculum Redesign, 2019.

[2] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E.,

- Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., "Language Models are Few-Shot Learners," NeurIPS, 2020. arXiv preprint arXiv:2005.14165.
- [3] Sondos Mahmoud Bsharat, Aidar Myrzakhan, Zhiqiang Shen. "Principled Instructions Are All You Need for Questioning LLaMA-1/2, GPT-3.5/4." arXiv 2312.16171v2 [cs.CL] 2024.
- [4] J. Zhang, P. Zhou, M. Li, "Large Language Models for Education: A Survey", arXiv preprint arXiv:2403.18105, 2024.
- [5] Hashim, S., Omar, M. K., Jalil, H. A., Sharef, N. M., "Trends on Technologies and Artificial Intelligence in Education for Personalized Learning: Systematic Literature Review," International Journal of Academic Research in Progressive Education and Development, vol. 11, no. 1, pp. 884-903, 2022. DOI: 10.6007/IJARPED/v11-i1/12230.
- [6] S Musali, M Vermeulen. "AI-Driven Learning: Enhancing Critical Thinking Through Technology-Integrated Pedagogy." ResearchGate, 2024.
- [7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks", arXiv preprint arXiv:2005.11401, 2020.
- [8] S. Ganesh, A. Purwar, G. B, "Context-augmented Retrieval: A Novel Framework for Fast Information Retrieval based Response Generation using Large Language Model", arXiv preprint arXiv:2406.16383, 2024.
- [9] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, D. C. Schmidt, "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," arXiv preprint arXiv:2302.11382, 2023. arXiv.2302.11382.
- [10] P. Chao, A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, E. Wong, "Jailbreaking Black Box Large Language Models in Twenty Queries", arXiv preprint arXiv:2310.08419, 2024.
- [11] D. Kim, C. Park, S. Kim, W. Lee, W. Song, Y. Kim, H. Kim, Y. Kim, H. Lee, J. Kim, C. Ahn, S. Yang, S. Lee, H. Park, G. Gim, M. Cha, H. Lee, S. Kim, "SOLAR 10.7B: Scaling Large Language Models with Simple yet Effective Depth Up-Scaling", NAACL 2024 Industry Track. arXiv preprint arXiv:2312.15166.
- [12] OpenAI, "Gpt-4 technical report," 2023.
- [13] Sondos Mahmoud Bsharat, Aidar Myrzakhan, Zhiqiang Shen. "Principled Instructions Are All