2023 전기 착수 보고서

LLM(Large Language Model)을 사용한 AI 챗봇 연구

부산대학교
PUSAN NATIONAL UNIVERSITY

팀명	쩝쩝 학사	**담당교수**	김호원 교수님
	201824523		안혜준
학번	201824473	**이름**	박성민
	201824483		박진영

- 목차

- 과제 배경

최근 거대 언어 모델(LLM)의 발전으로 인해 chatGPT와 같은 대화형 챗봇이 많이 개발되고 있다. 이러한 대화형 챗봇은 사용자와의 대화를 통해 사용자의 요구에 맞는 맞춤형 대답을 생성할 수 있다. 이러한 특징을 활용하여 음식 추천 챗봇을 개발하면, 사용자가 대화를 통해 쉽고 편리하게 원하는 음식을 추천 받을 수 있다.

현재 대다수의 음식 추천 사이트들은 사용자의 평점을 기반으로 작동하여 상위권에 위치한 음식을 제한적으로 추천하여 서비스를 제공한다. 또한, 단순히 랜덤으로 음식을 뽑아서 추천을 제공하는 경우도 많다. 이러한 방식으로 음식을 추천 받는 경우, 추천 목록이 제한되고 항상 비슷한 목록을 보게 된다. 사용자 맞춤형으로 데이터를 수집하더라도 사용자는 선호도에 따라 늘 비슷한 추천을 받게 되고, 색다른 추천을 받기가 어렵다.

LLM(Large Language Model)은 딥러닝으로 만들어진 자연어 생성 AI이다. 상황 별 음식 추천 정보를 미리 학습된 LLM에 추가 학습(fine-tuning)시켜 음식 추천 챗봇을 만들 수 있다. AI 챗봇을 사용하여 얻을 수 있는 기대 효과는 다음과 같다.

1) 대화를 통해 복잡한 요구사항에 대한 적절한 대답을 얻을 수 있다.
2) 쉬운 접근성으로 누구나 빠르게 맞춤형 정보를 얻을 수 있다.
3) 다양하고 매번 달라지는 추천을 받을 수 있다.

이러한 방식으로 음식 추천을 제공하면 어떤 사용자에게도 각자가 원하는 맞춤형 서비스를 제공할 수 있다.

- 세부 과제 내용

1. 과제 내용

대화형 음식 추천 챗봇은 음식 메뉴를 정하지 못하는 사용자를 위한 서비스이다. 사용자는 챗봇과의 대화를 통해 자신의 내/외부적인 요소를 고려한 맞춤형 음식을 추천 받을 수 있다.

사용자는 다음과 같은 과정을 통해 챗봇에게서 음식 추천을 받는다.

1. 사용자가 로그인을 하면 챗봇은 보관된 개인정보를 바탕으로 음식을 추천한다.
    - 만약 로그인을 하지 않으면, 챗봇은 보편적인 음식을 추천한다.

2. 사용자는 챗봇에게 음식 추천을 요구한다.
    - 추가로, 사용자는 챗봇과의 대화를 통해 자신의 내부적인 요소(몸 상태, 기분)을 전달할 수 있다.

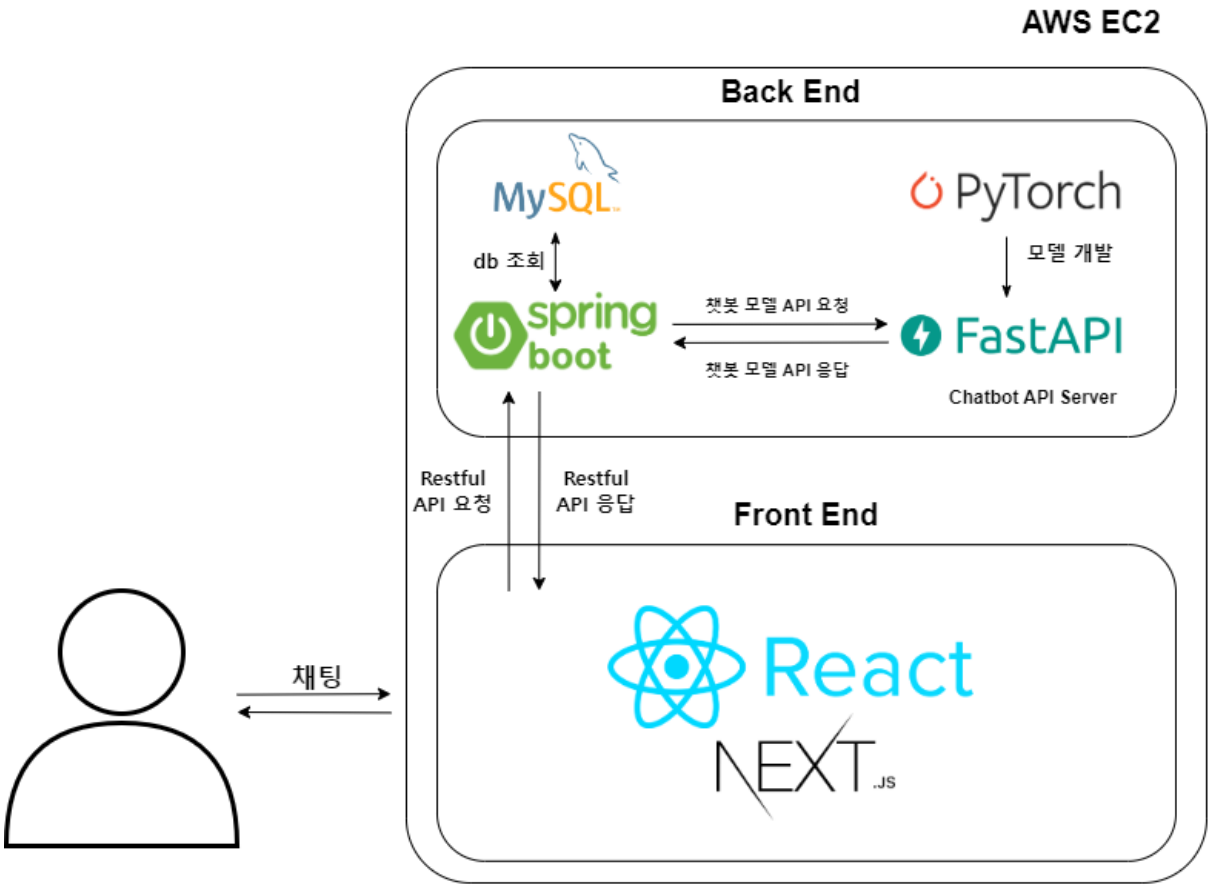
3. 대화를 통해 받은 사용자의 정보를 토대로 챗봇은 계속해서 적절한 음식을 추천한다.

개발은 다음 두 목표를 가진다.

- LLM fine-tuning
  사전학습된 LLM 모델을 선택해 음식 추천을 위한 데이터를 fine-tuning시킨다. 이를 통해 맞춤형 음식 추천 서비스 뿐만 아니라 일반적인 대화도 가능하게 한다.

- 웹 어플리케이션 개발
  대화형 챗봇 서비스를 웹 어플리케이션으로 제공함으로써 접근성을 높이고, 편리한 UI를 통해 사용성을 높인다.

2. 개발환경 / 구상도



분류	이름
Front-end Framework	Next.js
Back-end Framework	spring boot
Chatbot api 서버	FastAPI
데이터베이스	MySQL
클라우드 서비스	AWS EC2(백엔드 서버)

- 세부 개발 과정 및 기술 설명

1. LLM & fine-tuning

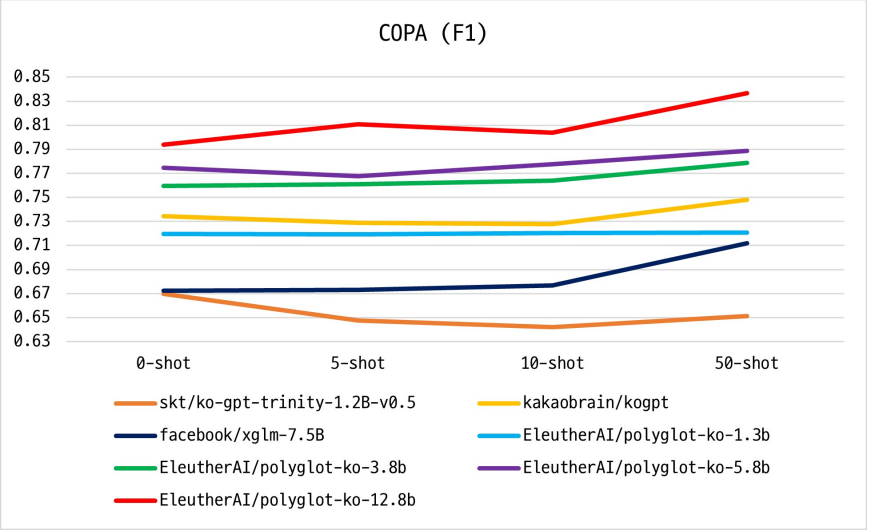
1) LLM(Large Language Model)

LLM(Large Language Model)은 수십억 개 이상의 매개변수로 이루어진 신경망 기반 언어 모델이다. Self-supervised learning과 준지도 학습 기법을 사용하여 레이블이 지정되지 않은 대량의 텍스트 데이터를 학습하며, 특정 작업이 아닌 다양한 분야에서 높은 성능을 보인다.

KoAlpaca

Stanford Alpaca 모델을 학습한 방식과 유사한 방식으로 학습된 한국어 언어 모델이다. LLaMA와 Polyglot-ko 모델을 기반으로 하는 두가지 버전이 존재한다.

LLaMA 모델은 학습 데이터에 한국어가 적게 포함되어 있어 한국어 성능이 낮은 반면, Polyglot-ko는 한국어 데이터로 학습된 다국어 초거대 언어모델 개발 프로젝트로, 한국어 성능이 우수하다는 점에서 차이가 있다. Polyglot-ko는 다른 한국어 LLM 모델과 비교해도 우수한 성능을 보이고 있다.



이런 특징에 근거하여, Polyglot-ko와 polyglot-ko 기반의 KoAlpaca를 참고해 맞춤형 모델을 제작한다.

2) fine-tuning

fine-tuning은 훈련된 모델의 가중치를 새로운 데이터에 대해 업데이트하는 방식을 통한 새로운 훈련이다. 이는 모델의 전체 또는 일부 매개변수에 대해 적용할 수 있다.

fine-tuning은 일반적으로 자연어 처리(NLP), 특히 언어 모델 영역에서 많이 쓰인다. 최근에는 LLM모델에 이 방법을 통해 학습한 언어 모델들이 많이 공개되고 있다.

LoRA(Low-Rank Adaptation)

GPT-3와 같은 거대한 모델을 fine-tuning하려면 많은 계산량과 시간이 소모되었다. LoRA는 원래 parameter는 freeze 시키고 기존 가중치 행렬과 parallel하게 훈련 가능한 rank decomposition 행렬 쌍을 추가함으로써 자원을 절약할 수 있는 fine-tuning 기법이다.

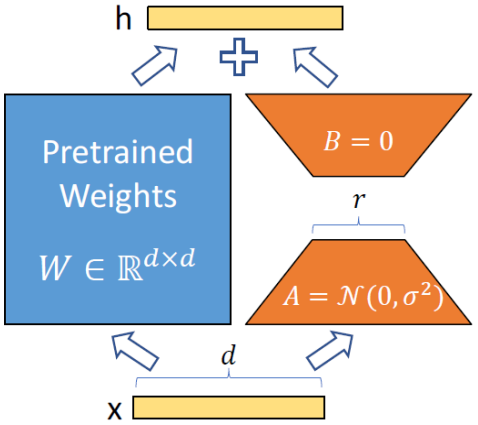


Figure 1: Our reparametrization. We only train A and B.

따라서 사전학습된 모델은 그대로 사용하며 새로 학습시킬 때는 작은 LoRA모듈만 추가시키면 된다. 이 방식을 이용하면 메모리 사용량과 parameter개수를 효과적으로 줄일 수 있다.

3) 모델 제작

1. pre-trained model 선택

KoAlphaca-Polyglot-5.8b, Polyglot 5.8b 모델 중 하나를 선택한다. 12.8b 모델은 파라미터 수가 더 많아 더욱 높은 성능을 가지지만, 학습 및 추론에 더 많은 컴퓨팅 리소스가 필요하다. 따라서 자원 제약으로 인해 고르지 못한다.

2. 프롬프트 튜닝

프롬프트 튜닝은 모델이 특정 작업을 수행하기 위해 fine-tuning되기 전에 수행되는 기술이다. 이는 기존 모델에 fine-tuning 하게 될 데이터의 일부를 제공하여, 모델이 해당 데이터에 대해 추론하게 한다. 모델이 추론해낸 출력을 바탕으로 fine-tuning 시킬 데이터의 형태를 결정한다.

3. 데이터 수집

프롬프트 튜닝을 통해 결정한 데이터 형태로 데이터를 수집한다. 데이터는 인터넷을 통해 정보를 크롤링하여 수집한다. 현재 한국에서 인기있는 메뉴들을 배달어플 통계나 SNS등을 통해 목록을 정리한다. 그 후, 각 메뉴에 대하여 위키문서나 네이버 식당 리뷰 크롤링을 통해서 선호 연령대, 성별, 특성등을 수집한다. 크롤링은 파이썬 라이브러리인 Selenium이나 Beautiful Soup등을 이용해서 진행하며 크롤링한 데이터는 fine-tuning 형식에 알맞게 데이터 전처리 과정을 거쳐서 모델에 학습시킨다.

4. 모델 fine-tuning

5.8b 모델을 튜닝하기 전, 수집한 데이터를 테스트 해보기 위해 먼저 1.3b 모델을 튜닝한다. 1.3b 모델은 5.8b 모델보다 작기 때문에 적은 컴퓨팅 파워로 여러 번의 테스트를 수행해 볼 수 있다. 이후, 테스트를 진행한 데이터를 5.8b 모델에 적용해 fine tuning을 진행한다.

5. 모델 평가

fine tuning한 모델을 평가한다. 이는 모델이 특정 작업에 얼마나 잘 수행되는지 측정하는 단계이다. 또한, 모델이 출시되기 전에 모델이 필요한 수준의 성능을 달성하고 있는지 확인하는 데 도움이 된다.

2. 웹 어플리케이션 및 AI 챗봇 서버 개발

1) AI 챗봇 서버

서버의 부담이 큰 챗봇 AI 모델을 API형태로 분리하여 scale out하면, 로드밸런싱을 도입할 수 있고 서버의 증설과 반환을 간단하게 할 수 있다. 또한, AI 모델은 구동하는 데 필요한 메모리량이 크기 때문에 별도의 고성능 서버를 마련하여 AI 챗봇 API 서버를 구축한다.

모델은 Python의 Tensorflow로 만들어 지기 때문에 제작한 모델을 동작시키기 위해, 같은 Python 환경인 FastAPI를 사용하여 API를 제공한다.

2) 백엔드 서버

사용자의 요청을 직접 처리하는 서버이다. 회원 가입과 로그인을 통해 사용자의 세션을 유지하고, 보관된 사용자 정보(채팅 내역, 선호 정보 등)를 불러와 전달해준다.

사용자의 채팅을 회원가입을 하며 설정한 기본 정보를 바탕으로 가공을 거쳐 챗봇 서버의 API를 요청하게 된다. 이때 전달되는 입력은 채팅 원문에 추가로, 데이터베이스에 저장된 사용자의 선호도를 덧붙여 만들어진다. AI 모델이 생성한 출력 텍스트는텍 서버에서 다시 한번 다듬어져 사용자에게 전달된다. 챗봇이 생성해낸 대답을 분석하여 데이터베이스에 저장된 음식이면, 저장된 기본적인 음식 정보를 사용자에게 추가로 전달해준다.

DBMS로는 MySQL을 사용한다. 데이터베이스는 사용자의 정보를 저장할 users table, 사용자의 채팅 세션들을 저장한 chats table, 채팅 내역을 저장할 messages table, 음식 정보를 저장하는 foods table, 채팅 후 사용자의 선호도를 저장할 preferences table이 존재한다.

채팅 결과를 수집하여 추후 다시 모델을 fine-tuning 시킴으로써 성능 향상을 꾀할 수 있다.

3) 사용자 화면

React 기반의 Next.js 프레임워크를 사용하여 서버 사이드 렌더링으로 클라이언트 페이지를 개발한다. 사용자가 보는 화면의 형태는 OpenAI의 ChatGPT의 UI를 참고한다. 사람들에게 널리 알려진 AI 채팅 서비스와 비슷한 형태를 취함으로써 AI 채팅 서비스로서의 정체성을 확고히 할 수 있다.

카카오맵API를 사용하여 사용자로부터 받은 위치 정보를 기반으로 챗봇이 생성한 추천 음식을 검색한다. 이렇게 함으로써, 사용자는 채팅으로 추천 받음과 동시에 주변의 음식점에 대한 정보까지 얻을 수 있다.

3. 배포 및 운영 계획

AWS EC2 인스턴스를 사용하여 서비스를 배포하고, OpenSSL을 활용하여 HTTPS 보안 연결을 설정하여 안전한 서비스를 제공한다. 서비스 배포 후 1달간의 테스트 기간을 두어 사용자로부터 발생할 수 있는 오류를 예방하고 고칠 수 있도록 한다.

또한, 사용자들로부터 제공받은 다양한 의견과 불편 사항을 수집하기 위해 별도의 문의하기 게시판을 마련하여 피드백을 받는다. 이를 통해 서비스의 개선과 보완을 지속적으로 수행하여 사용자들에게 보다 나은 서비스를 제공하게 된다.

- 기대효과

우리가 고안한 음식 추천 챗봇은 다음과 같은 기대효과가 있다.

- 사용자 맞춤형 추천
  기존의 음식 추천 애플리케이션과는 달리, 챗봇은 대화를 통해 사용자의 취향을 파악하고 맞춤형 추천을 제공할 수 있다.

- 편의성
  챗봇은 사용자들이 쉽게 접근할 수 있으며, 언제든지 대화를 마저 이어나갈 수 있다. 사용자는 단순히 채팅을 하면 되기 때문에 서비스에 대한 진입장벽이 낮다. 또한, 사용자의 위치 정보를 활용하여 가까운 맛집을 추천해주는 등 편리한 기능을 제공할 수 있다.

- 상업적 사용
  음식 추천 챗봇은 음식점에게 매출을 증가시키는 기회를 제공한다. 사용자가 챗봇을 통해 추천된 음식점을 방문하게 되면, 해당 음식점의 매출이 증가할 수 있다. 또한, 챗봇에서는 음식점의 광고를 제공하여, 음식점들의 마케팅에 도움을 줄 수 있다.

- 인기 메뉴 분석
  음식 추천 챗봇은 사용자들의 검색 이력과 만족도를 수집할 수 있다. 데이터를 분석하여 사용자들의 선호도를 분석하고 모델을 개선하여 더 나은 서비스를 제공할 수 있다. 또한, 이는 음식점에서 제공하는 메뉴와 서비스를 개선하는 데 도움되는 지표가 될 수 있다.

- 개발 일정 및 역할 분담

1. 개발 일정

<table>
<tr><th colspan="2">6월</th><th colspan="5">7월</th><th colspan="5">8월</th><th colspan="4">9월</th></tr>
<tr><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td></tr>
<tr><td>DB 설계</td><td colspan="2">UI 디자인</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td colspan="2">API문서 작성</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td colspan="5">데이터 수집</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="4">언어 모델 fine tuning, 테스트</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="2">중간 보고서</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">백엔드 개발</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">프론트 개발</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">배포 서버 환경 구축</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">테스트/배포</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">최종발표</td></tr>
</table>

2. 역할 분담

이름	역할
안혜준	프론트/백엔드 개발, AI 모델 제작, 서버 및 배포 환경 구축
박성민	프론트/백엔드 개발, AI 모델 제작, AI 챗봇 api 제작(FastAPI)
박진영	프론트/백엔드 개발, UI 디자인

- 참고 문헌

- Hu, Edward J., et al. "Lora: Low-rank adaptation of large language models." arXiv preprint arXiv:2106.09685 (2021).

- "Polyglot Ko 5.8B." Hugging Face, EleutherAI, 2021, https://huggingface.co/EleutherAI/polyglot-ko-5.8b.