LLM을 사용한 AI 챗봇 연구

Team: 모범택시

분과명: D 지능형융합보안

부산대학교 전기컴퓨터공학부

정보컴퓨터공학전공 School of Electrical and Computer Engineering,

Computer Engineering Major

Pusan National University

조장: 201724416 김대영
조원: 201824408 강주호, 201824579 정영진

2023년 8월 4일

지도교수 김호원 교수님

목차

1. 요구조건 및 제약 사항 분석에 대한 수정사항

1.1 요구조건

LLM (Large Language Model)을 이용해 사용자들에게 보다 정확한 의학적 답변을 주고, 한정된 의료진 자원 등으로 인해 다소 아쉬운 의료 서비스를 받던 사람들에게 좀 더 개인화된 의료 서비스 제공이 가능하게끔 의료용 AI 챗봇을 개발한다.

1.2 제약 사항 분석에 대한 수정사항

1.2.1 학습 모델 분석

기존에 의도했던 방향은 오픈소스로 공개되어 있는 LLaMA 모델을 이용해 alpaca 데이터셋으로 fine tuning 한 후, 수집한 의료용 데이터들로 추가적인 fine tuning을 진행하는 것이었다. 하지만 하드웨어의 한계 및 alpaca 데이터셋으로 학습된 모델도 오픈소스로 같이 공개됨에 따라, alpaca 데이터셋으로 fine tuning된 모델을 의료용 데이터로 fine tuning 하는 방향으로 변경하였다.

또한 사전에 조사했던 바로는 LLaMA 및 LLaMA 기반 모델들의 경우 처음에 학습시킨 한국어 데이터가 많이 부족해 해당 모델들에 한국어 데이터로 fine tuning을 진행해도 큰 효과를 보지 못한다고 알고 있었다. 하지만 과제를 진행하면서 오픈소스로 공개된 모델 중 Polyglot 기반 모델이 한국어 추론 능력이 뛰어남을 알게 되었다.
또한 api로 제공되는 번역기 성능이 좋지 않아 질문이나 답변이 왜곡되는 현상이 가끔 발생함에 따라, Polyglot 기반 모델을 이용한 학습도 추가적으로 진행하여 최종적으로 좀 더 자연스럽게 의학적인 답변을 추론하는 모델을 선택하고자 한다.

그리고 해당 모델들이 용량이 매우 크고 (제일 작은 LLaMA 7B 모델의 경우 약 13GB) 그에 따라 많은 gpu vram과 학습시간을 필요로 하여, 이를 단축하기 위해 8bit 내지는 4bit의 quantization을 진행하여 하드웨어 소모량을 줄이고 학습 속도를 높이고자 한다.

[그림 1] alpaca 데이터셋 예시

Hyperparameter	Value
$n_{parameters}$	12,898,631,680
n_{layers}	40
d_{model}	5120
d_{ff}	20,480
n_{heads}	40
d_{head}	128
n_{ctx}	2,048
n_{vocab}	30,003 / 30,080
Positional Encoding	Rotary Position Embedding (RoPE)
RoPE Dimensions	64

[그림 2] EleutherAI의 Polyglot12.8B

LLM Inference Results for Korean Evaluation Set

Type	Base-model	Model	이해 가능성 (0 - 1)	자연스러움 (1 - 3)	맥락 유지 (1 - 3)	흥미롭기 (1 - 3)	지시어 사용 (0-1)	전반적인 품질 (1-5)
Closed	GPT3.5-turbo	GPT-3.5	0.980	2.806	2.849	2.056	0.917	3.905
Closed	GPT-4	GPT-4	0.984	2.897	2.944	2.143	0.968	4.083
Open	Polyglot-ko-12.8b	KoAlpaca v1.1	0.651	1.909	1.901	1.583	0.385	2.575
Open	LLaMA-7b	koVicuna	0.460	1.583	1.726	1.528	0.409	2.440
Open	Polyglot-ko-12.8b	KULLM v2	**0.742**	**2.083**	**2.107**	**1.794**	**0.548**	**3.036**

[그림 3] 한국어 추론 능력 비교 표

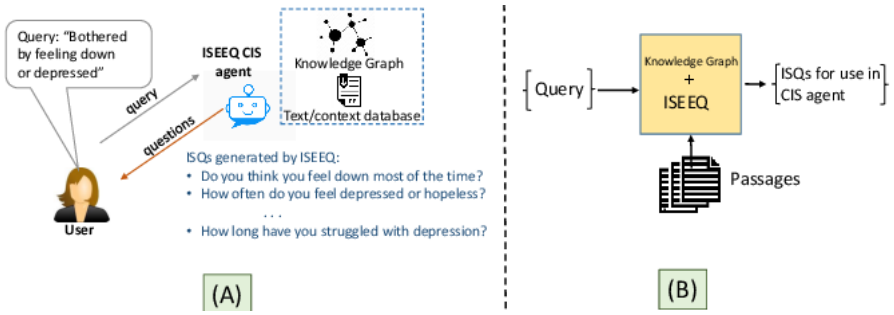
1.2.2 학습 데이터 분석

처음에 학습 데이터를 수집할 때는 의료용 질문 – 답변이 존재하는 사이트에서 단순히 크롤링 한 후 형식만 위 [그림 1]의 alpaca 데이터셋과 같이 맞춰주면 될 것이라고 생각했으나, 몇 가지 문제점들이 있었다.
먼저 질문에 특수문자나 기타 의미를 알기 힘든 말들이 많이 섞여 있어 크롤링 했을 시 제대로 저장이 되지 않는 경우가 있었다. 이는 다소 부정확하지만 크롤링 할 시에 특정 문자나 텍스트가 있는 부분은 제외하고 가져오거나, json 데이터 내에서 직접 제거하는 방식으로 처리하였다.
다음으로 질문자가 질문을 삭제하여 답변만 남아있는 경우였다. 이 부분은 답변을 보고 질문을 대략적으로 유추하여 직접 입력하거나, 해당 데이터는 삭제하고 새롭게 갱신된 다른 데이터로 변경하여 처리하였다.

1.2.3 학습 데이터 추가 확보 방안

앞서 1.2.1. 에서 언급한 바와 같이 Polyglot 기반 모델을 학습하기 위해 한국어 데이터셋도 확보하고자 한다. 현재는 하이닥 사이트와 네이버 지식인의 의료부분 관련 질문 – 답변을 수집하고, 먼저 수집한 데이터를 번역기로 번역한 데이터를 활용할 계획이다. 해당 사이트들의 경우 현직 의료진들만 답변이 가능해 답변의 전문성은 어느정도 보장된다고 볼 수 있다.

다만 질문 내용의 경우 앞선 1.2.2. 에서의 문제가 발생할 수 있을 것으로 생각된다. 따라서 추후에 수집할 데이터셋은 기존 질문 및 답변을 기반으로 질문을 batch decoding을 이용할 계획이다.

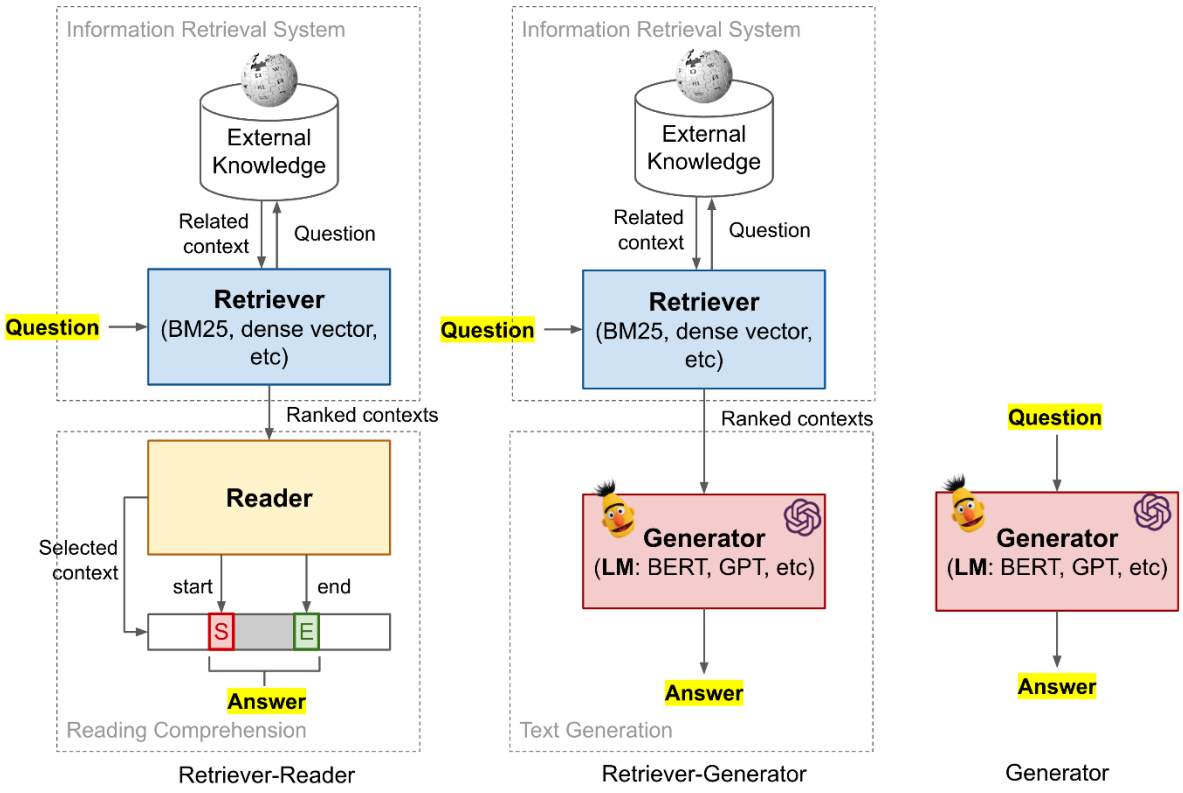


[그림 4] question-generation

2. 설계 상세화 및 변경 내역

2.1 데이터 수집 및 전처리

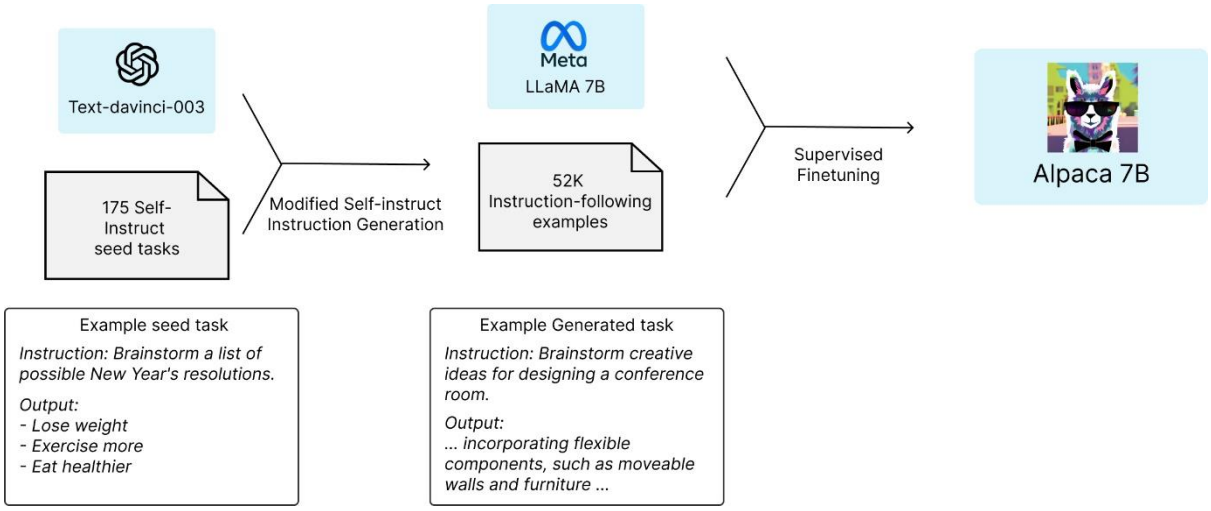
기존에는 영어로 된 데이터만 수집하였으나, 추가로 한글로 된 데이터도 수집할 예정이다. 또한, 앞서 언급한 바와 같이 단순 데이터 크롤링 시 데이터에 일부 문제가 있을 수 있어, 해당 데이터를 그대로 이용하지 않고 ChatGPT에 적절한 prompt를 주어 새로운 질문을 생성하여 활용할 계획이다.



[그림 5] question-answering generator

2.2 모델 구성 및 구현

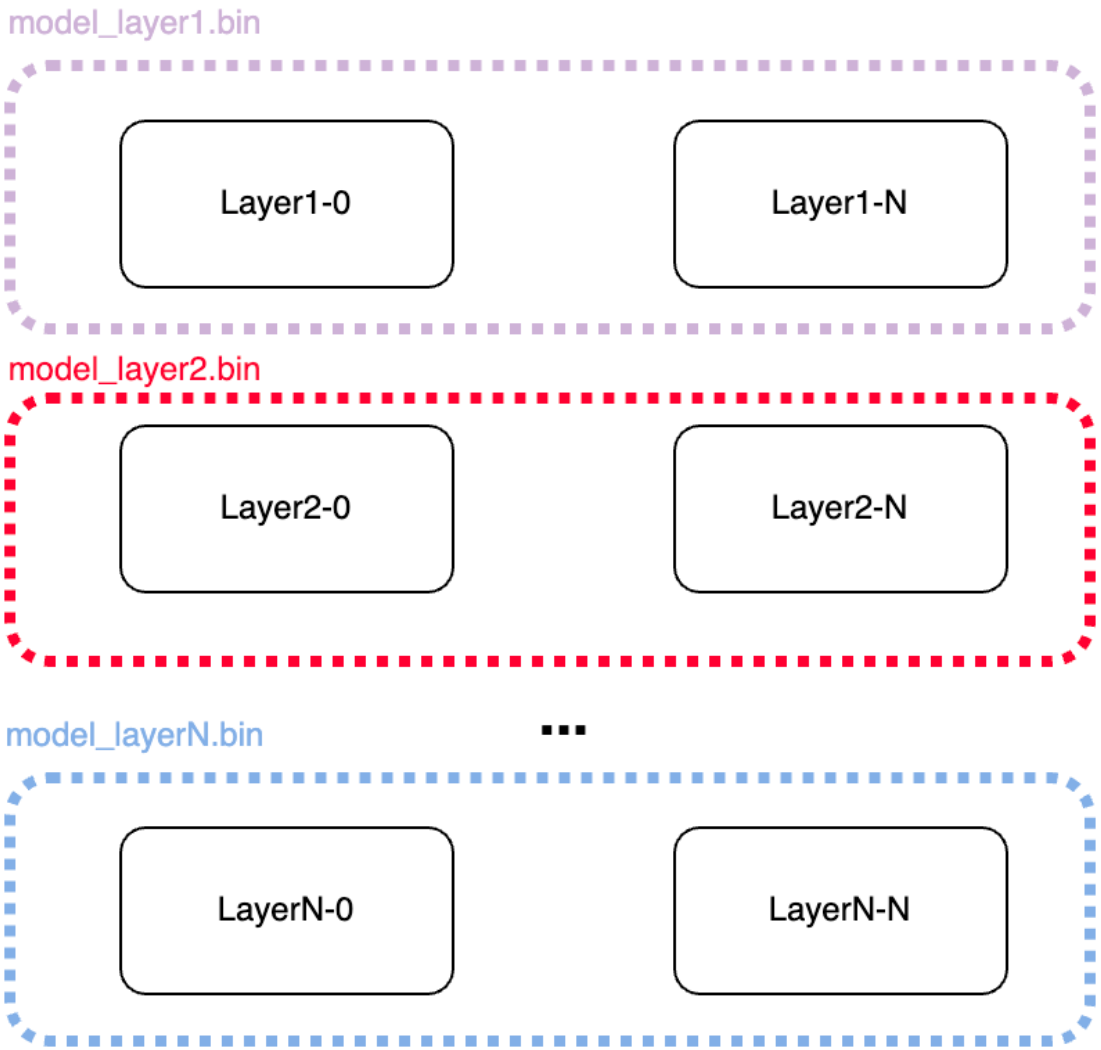
현재 적용중인 alpaca 모델의 구성은 다음과 같다.



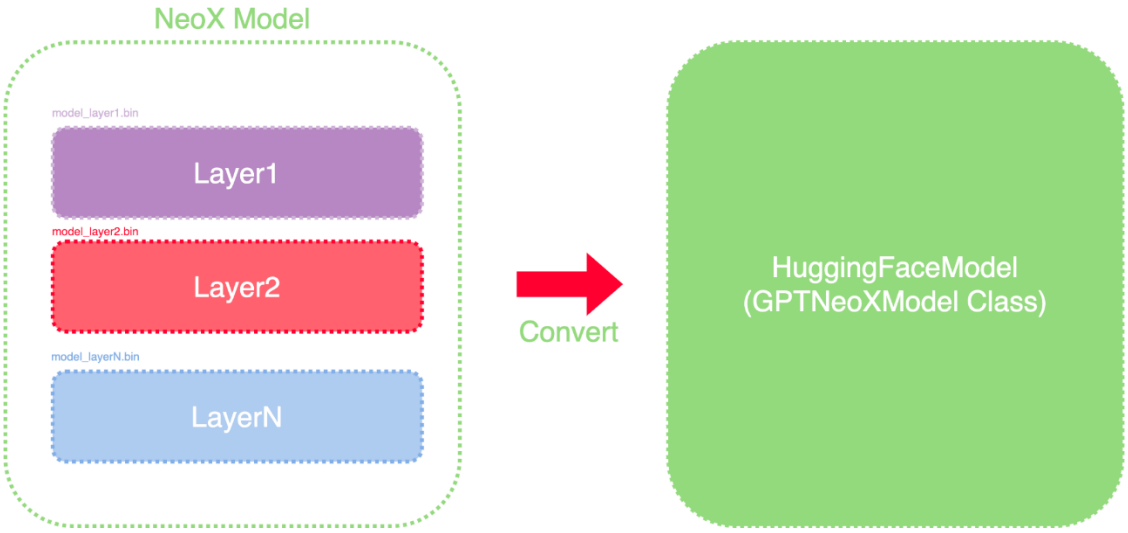
[그림 6] alpaca 모델

추가적으로 학습시킬 Polyglot 모델의 구성은 다음과 같다.

LLM을 사용한 AI 챗봇 연구



[그림 7] checkpoint merging



[그림 8] model upload

현재 모델을 불러오는 함수는 다음과 같다. Transformer 라이브러리의 AutoTokenizer와 AutoModelForCausalLM을 이용해 모델과 토크나이저를 불러온다. 이 때 데이터 학습 외에는 load_in_8bit를 False로 주어 모델이 텍스트를 생성할 때 너무 오랜 시간이 걸리지 않도록 한다. 또한 모델을 fp32로 로드하면 너무 크기가 커 gpu 및 메모리에 업로드가 불가능하므로, low_mem_usage=True 옵션을 주어 모델의 weight를 바로 gpu로 업로드 하게 해 cpu 사용량을 줄인다. 해당 함수의 경우 model_path만 변경하면 모델 종류와 관계없이 모두 활용 가능하다.

```
def load_model(model_path):
    global model, tokenizer, generator

    tokenizer = transformers.AutoTokenizer.from_pretrained(model_path)
    model = transformers.AutoModelForCausalLM.from_pretrained(
        model_path,
        torch_dtype=torch.float16,
        low_cpu_mem_usage=True,
        load_in_8bit=False,
        cache_dir="cache",
        device_map = "auto"
    ).cuda()
    generator = model.generate

load_model(“model_path")
```

3. 갱신된 과제 추진 계획

현재 목표와 이에 따른 계획은 다음과 같다.
1) Polyglot 기반 모델 활용을 위한 추가적인 한국어 데이터 확보 및 전처리
2) 추가적인 fine tuning 진행
3) 모델 성능 비교 후 최종적인 챗봇 완성

6월				7월				8월					9월			
1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주	5주	1주	2주	3주	4주
기본 모델 작성 및 데이터 수집/전처리																
				모델 LoRa fine tuning												

				및 데이터 수집/전처리										
						중간 보고서								
								한국어 데이터 확보 후 추가 fine tuning, 프론트/백 구현 및 API						
												성능 비교 분석 및 서비스 구현		
													최종 보고서 발표 준비	

※ 강조된 부분은 이전 계획에서 변경된 사항임

4. 구성원별 진척도

이름	역할
김대영	학습 데이터 수집 및 전처리 / 프론트-백 API 오류 수정 및 테스팅 모델 fine tuning 및 최적화
강주호	학습 데이터 수집 및 전처리 오류 수정 및 테스팅 UI/UX 디자인 설계 및 학습 결과 시각화
정영진	학습 데이터 수집 및 전처리 오류 수정 및 테스팅 / 백엔드 구현 및 연동

5. 보고 시점까지의 과제 수행 내용 및 중간 결과

아래는 현재까지 수행한 내용이다.

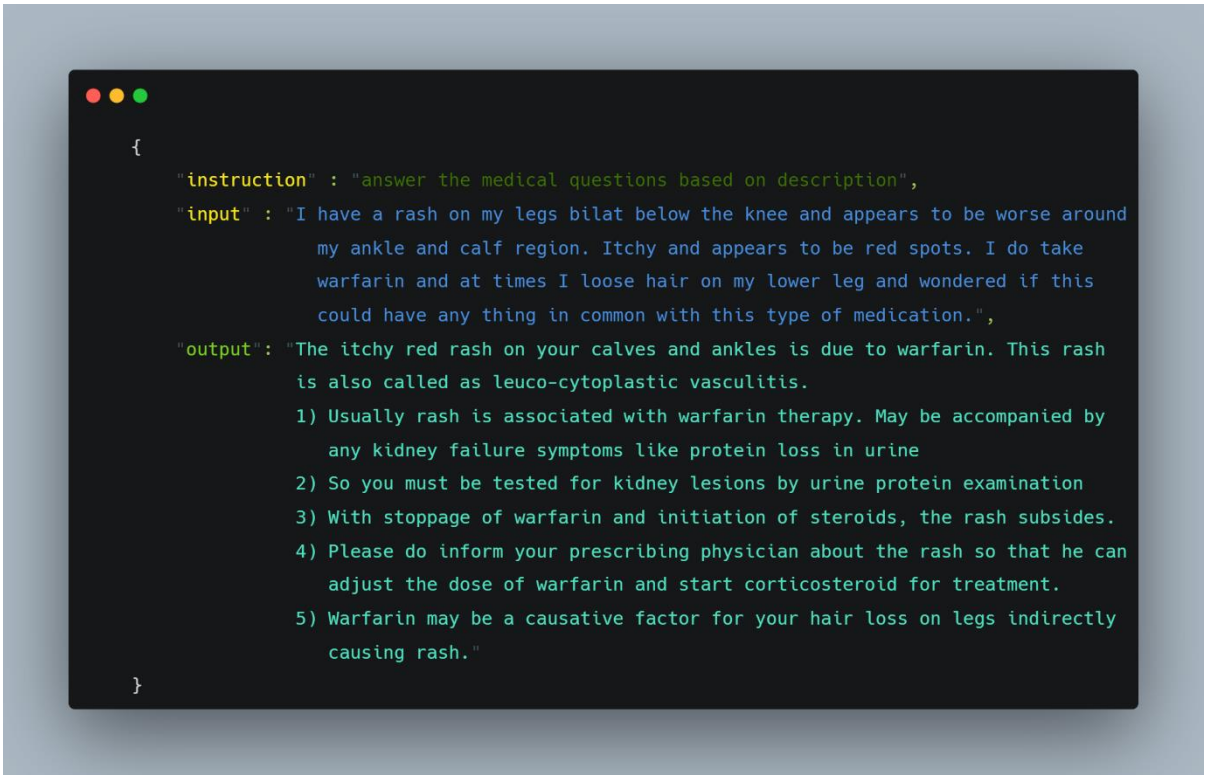
5.1 의료용 데이터 수집 및 전처리

온라인상에서 실제 의사에게 증상에 대해 질문하고 답변을 받을 수 있는 사이트들을 크롤링 한 후 데이터를 json 형식으로 처리하였다.

데이터 전처리는 다른 alpaca 기반 특정 분야 특화 언어 모델들을 참고하여

instruction은 해당 분야 (의료)의 질문에 대한 적절한 답변 제시 요구로 부여하였고, input과 output에는 질문과 답변을 넣어 수행하였다.

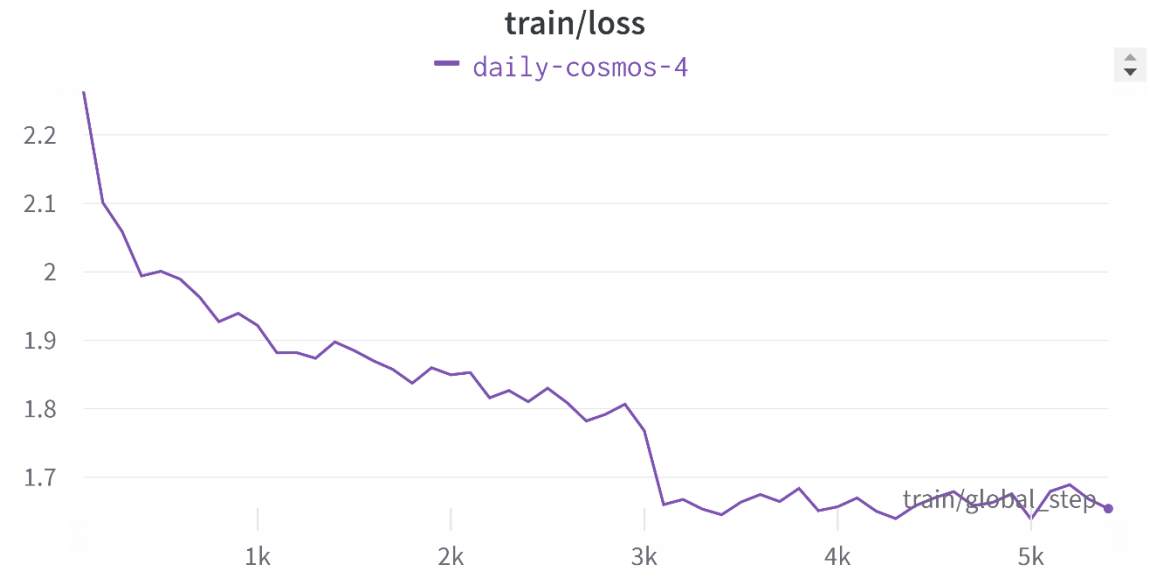
해당 데이터셋의 예시는 다음과 같다.



[그림 9] 데이터셋 예시

5.2 모델 학습 및 평가

기존 alpaca 모델에 fine tuning을 진행하였다. Google Colab에서 학습을 진행하였고, gpu는 A100 40GB 1대를 사용하였다. 초기 Learning rate는 3e-5, adam optimizer, cosine learning rate scheduler를 hyperparameter로 주어 1.5 epoch (5500 step)을 학습하였다.



[그림 10] 현재 모델의 training loss

데이터셋이나, 모델 규모에 따라 다르지만 fine tuning 시에는 일반적으로 1 ~ 3 epoch 정도를 학습시킨다. 현재 모델의 경우 1 epoch 학습 완료 시 loss 값이 상대적으로 크게 떨어졌고, 2 epoch 학습 시에도 비슷하게 떨어질 것으로 예상된다. 다만 학습에 시간이 많이 걸리고, colab에서 할당한 gpu 자원을 거의 다 사용하여 더 이상 학습을 진행하지 못했다. 따라서 현재 가지고 있는 checkpoint에 이어서 학습을 진행하고 데이터를 추가적으로 확보한다면 조금 더 좋은 성능을 보일 수 있을 것으로 예상한다.

5.3 텍스트 생성

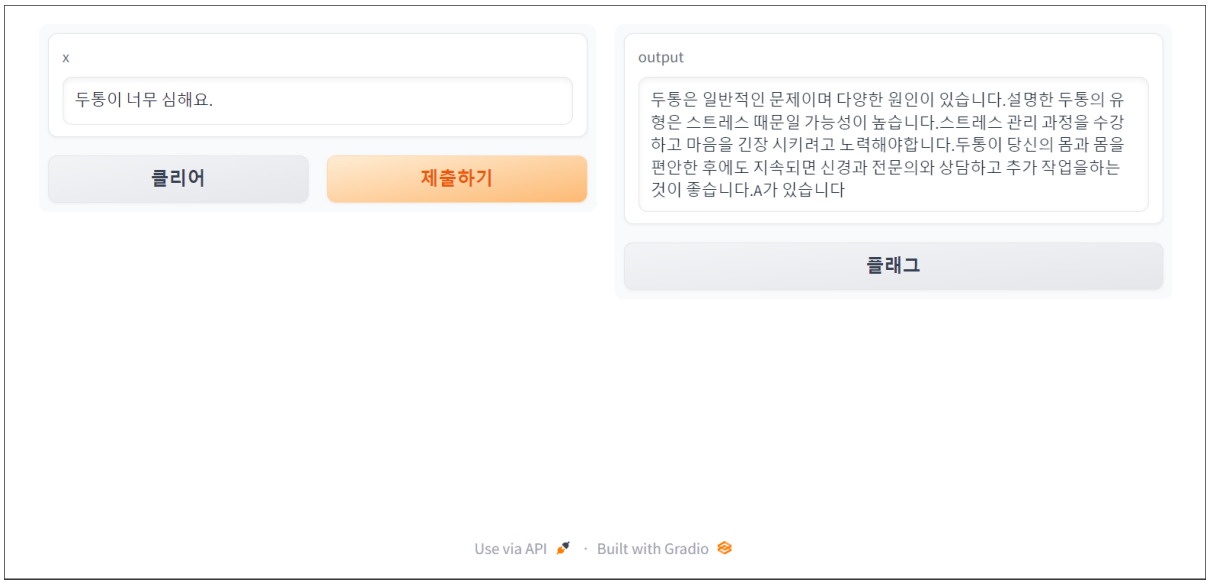
Fine tuning한 모델로 실제 답변을 생성하여 확인하였다. 현재는 영어로 된 데이터셋을 이용하여 학습시켰으므로, input과 output에 영어 번역기를 적용한 후 출력하였다.

다음은 텍스트 생성 코드 및 실행 결과물이다. 실제 텍스트 생성물 출력은 gradio sample code를 이용하였다.

```
pipe = pipeline(
    'text-generation',
    model=model,
    tokenizer=tokenizer,
    device=0
)

def ask(x, is_input_full=False):
    context = "answer the medical questions based on description."
    ans = pipe(
        f"### question: {x}\n\n### context: {context}\n\n### answer:",
        do_sample=True,
        max_new_tokens=200,
        temperature=0.2,
        top_p=0.9,
        return_full_text=False,
        eos_token_id=2,
    )
    return ans[0]['generated_text']
```

Transformers 라이브러리의 pipeline 함수에서 'text-generation'을 이용한다. 이는 주어진 context (데이터셋에서의 instruction) 정보를 추출해 질문에 대한 답변을 제공하는 방식으로 동작한다.



[그림 11] 텍스트 생성 예시

6. 참고 자료

[2, 7, 8] https://wandb.ai/eleutherai/polyglot-ko/reports/Polyglot-Ko-Open-Source-Korean-Autoregressive-Language-Model--VmlldzoyOTYyODcz
[3] https://github.com/nlpai-lab/KULLM#evaluation
[4] https://www.semanticscholar.org/paper/ISEEQ%3A-Information-Seeking-Question-Generation-and-Gaur-Gunaratna/77d2456630d7b22efe84bffcc7d4ad495ce50a6d
[5] https://lilianweng.github.io/posts/2020-10-29-odqa/
[6] https://crfm.stanford.edu/2023/03/13/alpaca.html