롯데이노베이트

한국어 어문 규범 기반 생성 (RAG)

Real Awesome Gyubeom
김광륜, 정민교

2025.10.02

Change Today,
Create Tomorrow .

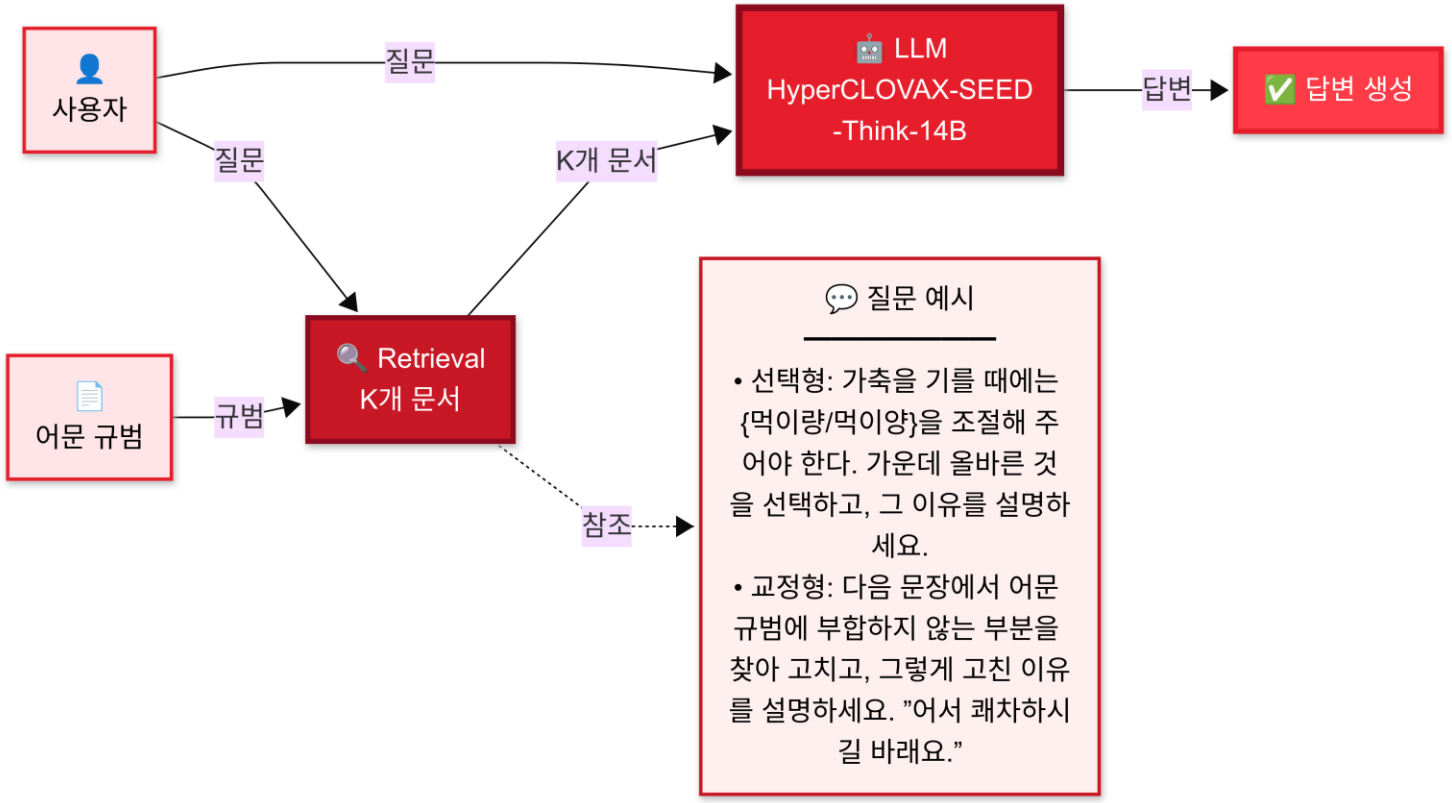
목차

개요

태스크 정의

한국어 어문 규범 관련 질문에 대하여 국어 지식을 참조해 외부의 지식없이 주어진 정보만으로 답변을 생성하는 과제

- **선택형**: 정답과 정답이 아닌 두 선택지에 대해 모델이 올바른 선택지를 고르고, 그 이유를 생성
- **교정형**: 주어진 문장에서 어문 규범에 부합되지 않은 부분을 모델이 수정하고, 수정한 이유를 생성



데이터셋

한국어 어문 규범 참조 문서

(1) 원천 데이터

개행 문자를 구분자로 사용한 규범(규범 항목명, 규범 내용) pdf 파일

(2) 전처리

모델 학습 및 추론에 사용하기 편리한 json 포멧으로 전처리 수행

- pdf 파서를 이용해 텍스트 파싱
- 규범 제목기준으로 규범을 분리하고, 각 규범을 하나의 문서 단위로 정의

(3) 데이터셋 재구성

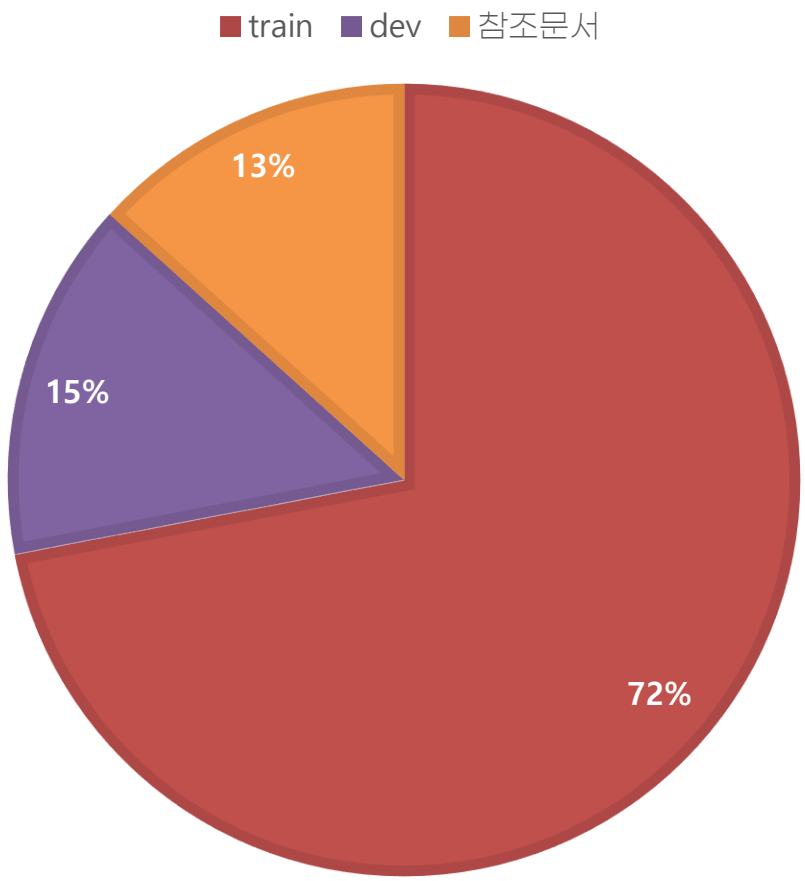
답변 생성에 참조될 문서를 학습과 추론에 적합하게 정의

- gold_context: 답변을 생성하는데 있어 참조하기에 가장 적합한 문서
  - **pseudo labeling 필요성**: 원본 데이터에 gold_context labeling이 되어 있지 않아 임의로 labeling 작업을 수행할 필요성 존재
  - **pseudo labeling 방법**: 질의만을 이용해 gold_context를 labeling 하기에는 정보량이 부족해 정답을 활용
    - 정답의 '옳다' 단어를 기준으로 근거 문장을 추출하고, 이를 query로 하여 가장 유사한 규범을 검색
    - 'BAAI/bge-m3' 임베딩 모델을 이용해 모든 규범을 임베딩하고, faiss DB을 이용해 관리
    - L2 거리검색 알고리즘을 사용하는 faiss의 search 기능을 이용해 pseudo labeling 수행

데이터셋

EDA

(1) 데이터셋 크기 및 question type 분포

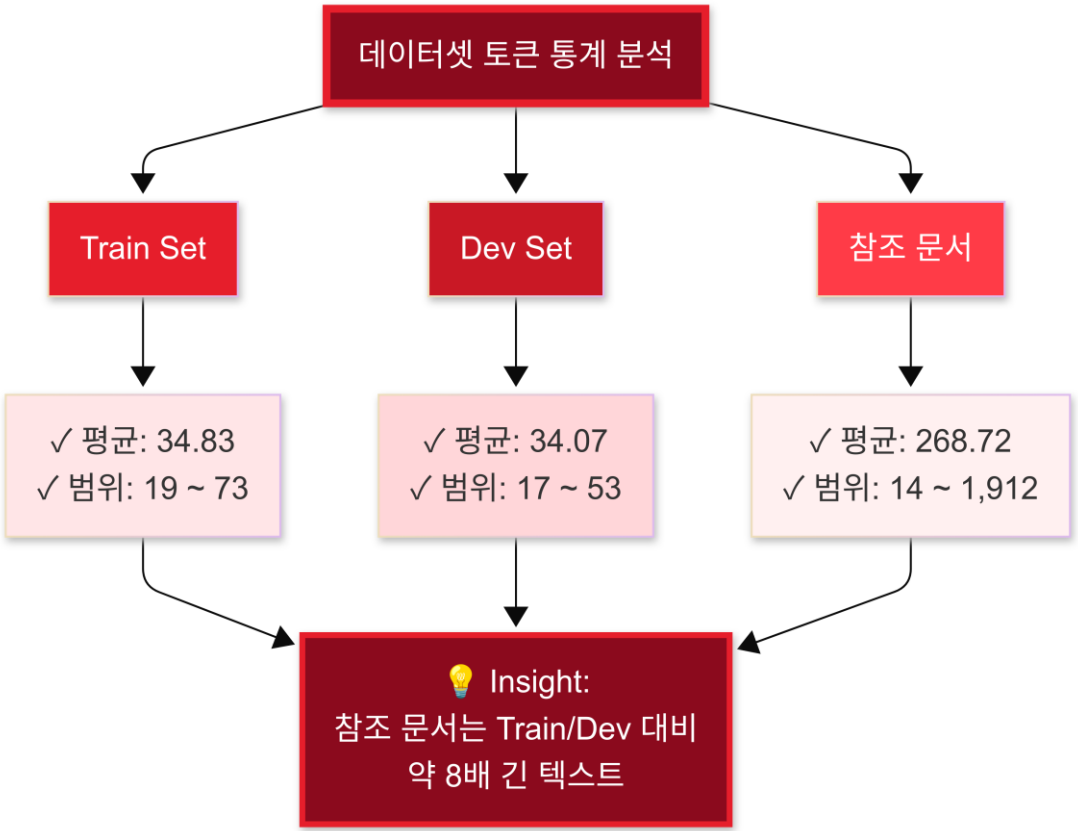


데이터셋

EDA

(2) 토큰 통계

Train: 34.8
Dev: 34.07
참조문서: 268.72



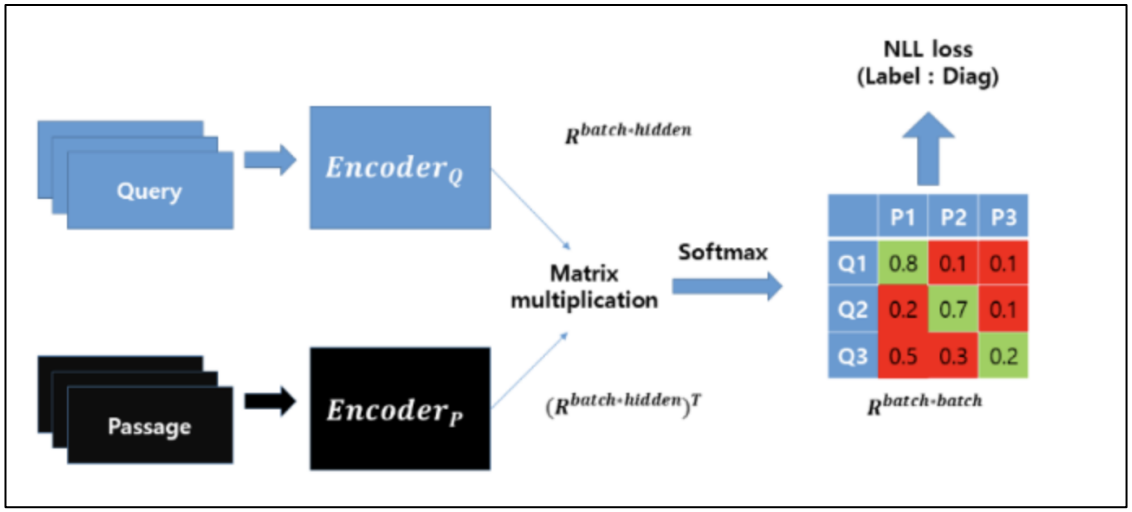
모델

◆ 검색 및 재순위화

(1) 검색

DPR (Dense Passage Retrieval)을 이용해 검색 수행

- dual encoder (query & passage encoder)를 사용해 각 인코더 아웃풋에 대해 코사인 유사도를 계산해 query에 대한 문서 간의 유사도 순위를 계산
- In-batch negative와 hard negative 전략을 사용해 학습



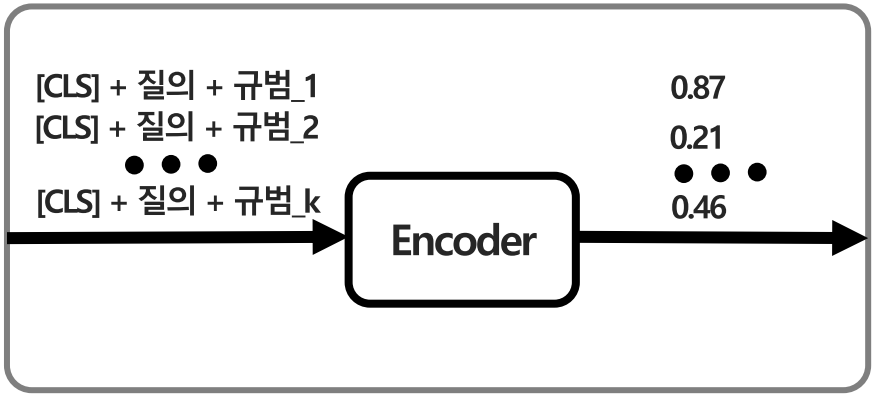
모델

◆ 검색 및 재순위화

(2) 재순위화

검색된 k개의 문서를 대상으로 재순위화를 수행

- query와 각 검색된 문서를 각각 연결하고, 연결된 문장들을 인코더 모델에 입력해 추출된 cls 토큰을 이용해 재순위화 수행
- DPR만 사용해 검색한 성능과 비교해 성능 향상



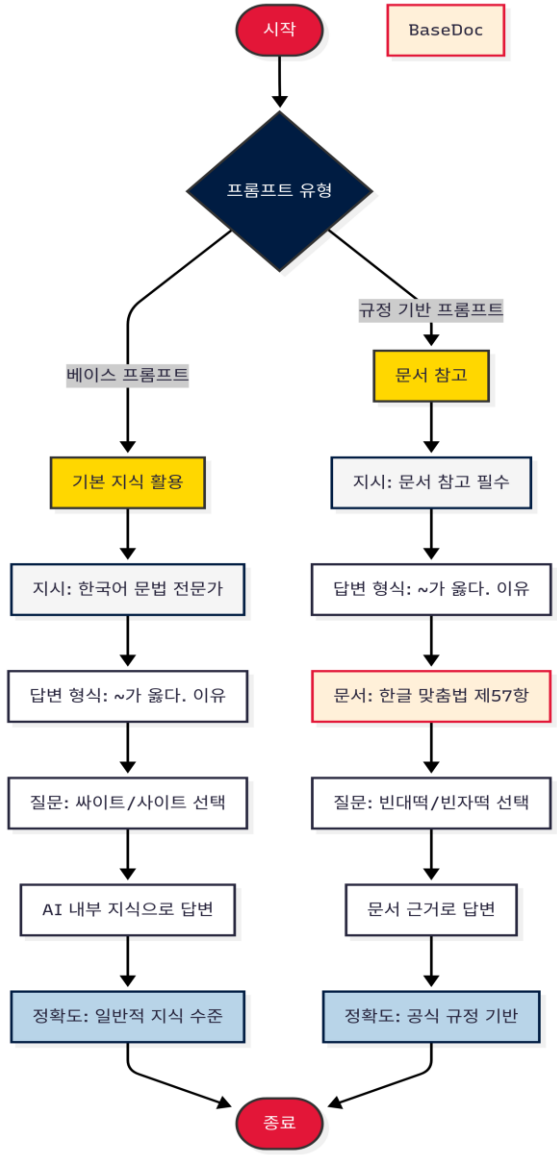
모델

◇ 검색 및 재순위화

(3) 선택 추출 검색

앞서 언급한, 의미 기반 검색인 DPR의 검색 성능이 높지 않고, pseudo labeling을 이용한 평가와 학습이기 때문에 실제 질의 답변에 필요한 규범이 아닐 가능성이 있어 질의에 나타난 선택 추출 검색을 이용해 규범 검색을 수행

- 검색에 성공시 규범을 증강해 답변 생성하는 프롬프트(augmentation prompt)를 사용
- 검색에 실패하면 규범 증강 없이 모델의 LLM의 학습된 knowledge를 사용하는 프롬프트(base prompt)를 사용



모델

◇ 검색 및 재순위화

(3) 슬롯 기반 검색

검색 O

지시: 당신은 한국어 문법에 대해 잘 알고 있는 유능한 AI 어시스턴트 입니다. 문서를 참고하여 한국어 문법에 맞는 것을 선택해 '~가 옳다. 이유' 형식으로 답변하세요. 주어진 문장은 반드시 수정해야합니다. 그대로 출력하지 마세요.
질문타입: 선택형
질문: "막내가 {빈대떡/빈자떡}을 둥글넓적하게 만들었다." 가운데 올바른 것을 선택하고, 그 이유를 설명하세요.
문서: <한글 맞춤법, 표준어 규정 - 한글 맞춤법 제57항>
…
답변:

검색 X

지시: 당신은 한국어 문법에 대해 잘 알고 있는 유능한 AI 어시스턴트 입니다. 질문의 틀린 문장을 올바른 한국어 문법에 맞게 '~가 옳다. 이유' 형식으로 답변하세요. 주어진 문장은 반드시 수정해야합니다. 그대로 출력하지 마세요.
질문타입: 선택형
질문: "이 제품은 공식 {싸이트/사이트}에서만 구매할 수 있다." 가운데 올바른 것을 선택하고, 그 이유를 설명하세요.
답변:

모델

답변 생성

(1) 프롬프트 엔지니어링

검색이 성공했을 때 사용하는 augmentation prompt와 실패 했을 때 사용하는 base prompt를 설정하여 두 가지 버전으로 학습과 추론 수행

```
base_prompt = f"""## 지시: 당신은 한국어 문법 도우미 입니다. 질문에 맞게 '~가 옳다. 이유' 형식으로 답변하세요.
## 질문 타입: {question_data['question_type']}
## 질문: {question_data['question']}
## 답변:"""
```

<base prompt>

```
augmentation_prompt = f"""## 지시: 당신은 한국어 문법 도우미 입니다. 문서를 참고하여 질문에 맞게 '~가 옳다. 이유' 형식으로 답변하세요.
## 질문 타입: {question_data['question_type']}
## 질문: {question_data['question']}
## 문서: {doc_text}
## 답변:"""
```

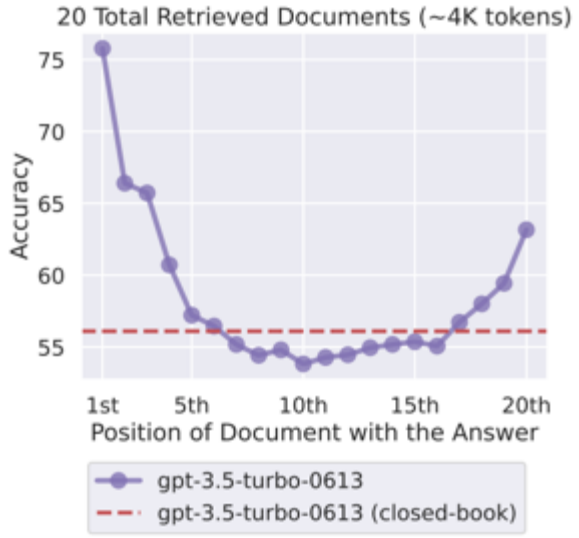
<augmentaion prompt>

모델

답변 생성

(1) 프롬프트 엔지니어링

검색이 성공한 경우 문서를 질의와 함께 요청을 보내는데, 문서의 위치에 따라 모델이 답변을 정확성에 문제 존재
따라서, Lost in Middle에 착안, 질의의 말머리 부분에 명확한 질의, 하단분에 문서를 넣어 모델의 이해력에 집중함.



Lost in Middle

질문에 대한 관련 문서가 컨텍스트 중간에
위치할 경우, LLM 응답 정확도가 낮아진다.

```
if doc_text:
        return f"""## 지시: 당신은 한국어 문법 도우미 입니다.
문서를 참고하여 질문에 맞게 '~가 옳다. 이유' 형식으로 답변하세요.
## 질문 타입: {question_data['question_type']}
## 질문: {question_data['question']}
## 문서: {doc_text}
## 답변:"""
```

모델

양자화(Quantization)

(1) 모델 교정 데이터셋 구축

'HyperCLOVAX-SEED-Think14B' 모델을 기반으로 양자화를 위해 어문 규범 기반 생성에 맞는 교정 데이터셋을 제작함.

```python
def process_dataset(data, docs):
    """데이터셋 처리 - input과 output을 합쳐서 text로 반환"""
    texts = []

    for idx, row in data.iterrows():
        options = extract_options(row['input']['question'])
        if options and len(options) > 1 and len(options[0])+len(options[1]) > 3:
            # 관련 문서 찾기
            relevant_docs = []
            for option in options:
                for doc in docs:
                    if option in doc and doc not in relevant_docs:
                        print(idx, option)
                        relevant_docs.append(doc)

            prompt = create_prompt(row['input'], docs, relevant_docs)
        else:
            prompt = create_prompt(row['input'], docs)

        # input과 output을 합쳐서 하나의 text로 만들기
        combined_text = f"{prompt} {row['output']['answer']}"
        texts.append(combined_text)

    return pd.DataFrame({"text": texts})
```

모델

양자화(Quantization)

(2) gptq

layer-wise 양자화 방식, 모델의 각 레이어를 순차적으로 양자화하면서 양자화 오류를 최소화

naver-hyperclovax/HyperCLOVAX-SEED-Think-14B

Safetensors

Model size	14.7B params	Tensor type	F32	Chat template	Files info

Our !

Safetensors

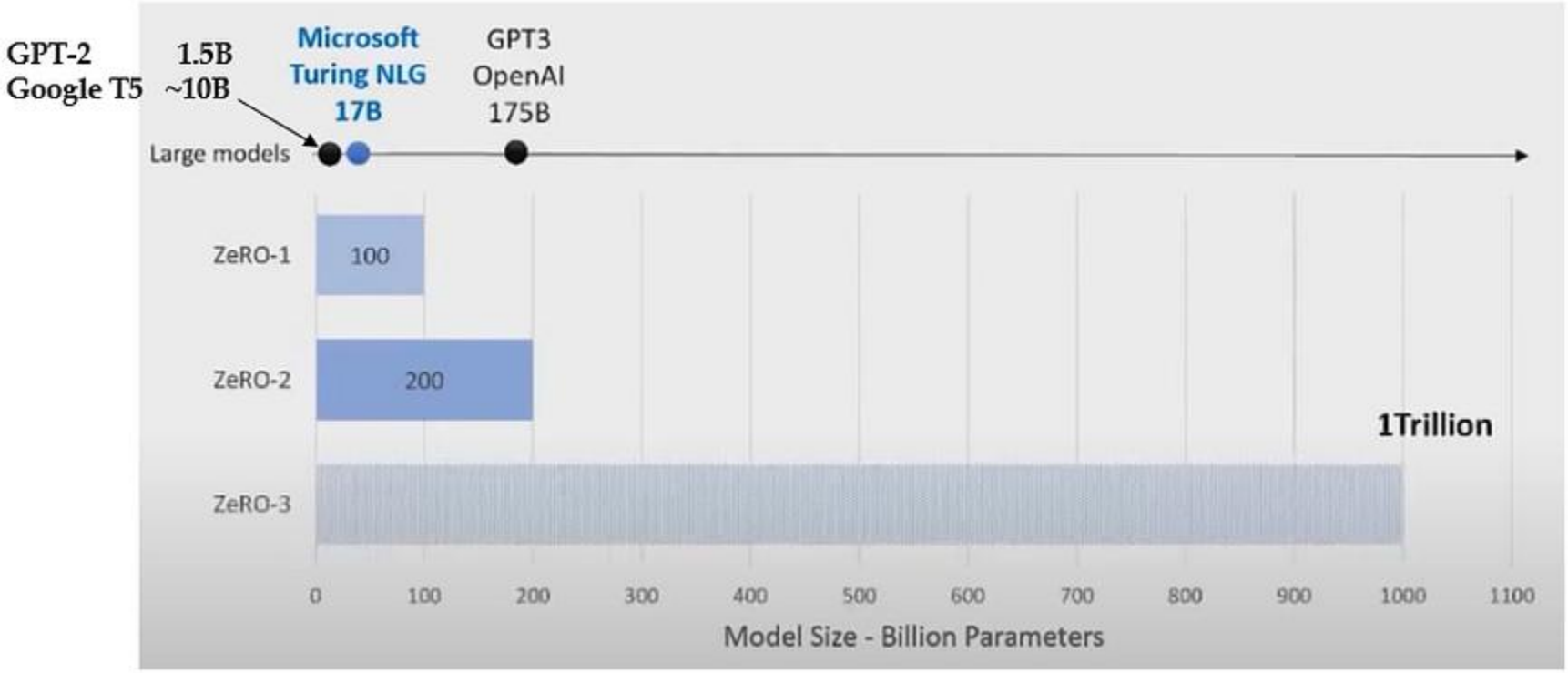
Model size	3.15B params	Tensor type	I32 · BF16	Chat template	Files info

모델

학습

(1) sLM 학습 기법

효율적인 생성형 모델 학습을 위해 deepspeed ZeRO-3 학습을 진행



모델

학습

(2) Iterative Training

앙상블이 금지된 환경에서 seed 앙상블 효과를 누리고, 모델의 강건함을 보존하기 위해 연속적 학습 진행

2 epoch -> 8 epoch -> 1 epoch

2 epoch

```
deepspeed --num_gpus=4 train.py \
    --deepspeed ds3.js
    --model_name_or_pa
    --output_dir naver
    --data_path ./data
    --max_seq_length 1
    --gradient_accumul
    --save_total_limit
    --save_strategy ep
    --num_train_epochs
    --logging_step 1 \
    --learning_rate 3e
    --gradient_checkpo
    --per_device_train
    --overwrite_output
    --bf16 True \
    --use_liger_kernel
    --run_name mm-rag
```

8 epoch

```
deepspeed --num_gpus=4 train.py \
    --deepspeed ds3.jso
    --model_name_or_pat
    --output_dir naver-
    --data_path ./datas
    --max_seq_length 10
    --gradient_accumula
    --save_total_limit
    --save_strategy epo
    --num_train_epochs
    --logging_step 1 \
    --learning_rate 3e-
    --gradient_checkpoi
    --per_device_train_
    --overwrite_output_
    --bf16 True \
    --use_liger_kernel
    --run_name mm-rag
```

1 epoch

```
deepspeed --num_gpus=4 train.py \
    --deepspeed ds3.json \
    --model_name_or_path naver-hyperclovax/HyperCLOVAX-SEED-Think-14B \
    --output_dir naver-test \
    --data_path ./dataset/train.json \
    --max_seq_length 10000 \
    --gradient_accumulation_steps 1 \
    --save_total_limit 5 \
    --save_strategy epoch \
    --num_train_epochs 2 \
    --logging_step 1 \
    --learning_rate 3e-6 \
    --gradient_checkpointing \
    --per_device_train_batch_size 1 \
    --overwrite_output_dir True \
    --bf16 True \
    --use_liger_kernel True \
    --run_name mm-rag
```

실험

◆ 검색 및 재순위화 성능 평가

(1) 검색 및 재순위화 성능

DPR을 이용한 검색 및 재순위화 검색 성능

Model	recall@1	recall@5	recall@30	recall@50
DPR	0.02	0.16	0.31	0.68
DPR w/rerank	0.10	0.34	0.48	0.85

- 주어진 질의를 이용해 연관된 규범을 검색하기에는 주어진 질의에 충분한 정보가 없어 검색 성능이 매우 저조함을 확인
- 어문 규범과 같이 특수한 도메인을 일반적인 임베딩 모델을 이용해 pseudo labeling 하였을 때, 정확한 라벨링이 되지 않아 검색 성능 평가가 정확하지 않을 수 있음을 확인
- 의미 기반의 검색 방법이 적절하지 않다고 판단

실험

◆ 검색 및 재순위화 성능 평가

(1) 검색 및 재순위화 성능

모델이 보는 문서수를 늘릴 수 록 정확한 문서를 가져올 확률을 증가되나, 실험결과 문서수와 모델의 성능은 비례하지 않았고, 관련 없는 문서까지 모델이 참조하게되어 오히려 성능이 하락하는 문제로 선택 추출 검색을 적용하였음.

K = 10	답변
참조문서: 문서#0:(1) 서술, 명령, 청유 등을 나타내는 문장의 끝에 쓴다. … 문서#9:(1) 의문문이나 의문을 나타내는 어구의 끝에 쓴다. - 점심 먹었어? … 지시: 당신은 한국어 문법 검사자입니다. 주어진 질문에 대해 질문 타입에 맞게 '~가 옳다. 설명' 형식으로 답변하세요. 질문 타입: 교정형 질문: 다음 문장이 어문 규범에 부합하도록 문장 부호를 추가하고, 그 이유를 설명하세요. — 검( )경찰 답변:	"검(경찰) 경찰"이 옳다. 설명: 문서#6(1)에 따르면, 우리말 표기와 한자어나 외래어를 함께 보일 때 괄호를 사용한다. 또한 "검"이 "경찰"의 약어로 사용된 경우, 괄호로 풀어 쓰는 형태를 지키는 것이 어문 규범이다.

실험

◆ 검색 및 재순위화 성능 평가

(1) 검색 및 재순위화 성능

모델이 보는 문서수를 늘릴 수 록 정확한 문서를 가져올 확률을 증가되나, 실험결과 문서수와 모델의 성능은 비례하지 않았고, 관련 없는 문서까지 모델이 참조하게되어 오히려 성능이 하락하는 문제로 선택 추출 검색을 적용하였음.

선택 추출 검색

"아이가 {목걸이/목거리}를 앓아서 말을 잘 못한다." 가운데 올바른 것을 선택하고, 그 이유를 설명하세요.

검색된 어문 규범

<한글 맞춤법, 표준어 규정 - 한글 맞춤법 제19항>
…,
목거리(목병), 무녀리, 코끼리, 거름(비료), 고름[膿], 노름(도박) [붙임] 어간에 '-이'나 '-음' 이외의 모음으로 시작된 접미사가 붙어서 다른 품사로 바뀐 것은 그 어간의 원형을 밝히어 적지 아니한다.
…

<한글 맞춤법, 표준어 규정 - 한글 맞춤법 제57항>
…
- 목거리: 목거리가 덧났다. - 목걸이: 금목걸이, 은목걸이. - 바치다: 나라를 위해 목숨을 바쳤다. - 받치다: 우산을 받치고 간다., 책받침을 받친다. - 받히다: 쇠뿔에 받혔다. - 밭치다: 술을 체에 밭친다. - 반드시: 약속은 반드시 지켜라. - 반듯이: 고개를 반듯이 들어라. - 부딪치다

…

실험

◇ 생성 성능 평가

(1) 생성 모델 선정

저작권 문제가 없는 오픈소스 LLM 중 생성 모델 선정

```
# MODEL_PATH = "cognitivecomputations/Qwen3-235B-A22B-AWQ"
# MODEL_PATH = "AMead10/GLM-4-32B-0414-awq"
# MODEL_PAHT = "kakaocorp/kanana-1.5-8b-instruct-2505"
# MODEL_PATH = "MLP-KTLim/llama-3-Korean-Bllossom-8B"
# MODEL_PATH = "LGAI-EXAONE/EXAONE-3.5-32B-Instruct-AWQ"
# MODEL_PATH = "casperhansen/deepseek-r1-distill-qwen-32b-awq"
# MODEL_PATH = "deepseek-ai/DeepSeek-R1-0528-Qwen3-8B"
# MODEL_PATH = "Menlo/Jan-nano-128k"
# MODEL_PATH = "THU-KEG/LongWriter-Zero-32B"
# MODEL_PATH = "POLARIS-Project/Polaris-4B-Preview"
# MODEL_PATH = "letgoofthepizza/gemma-7b-it-finetuned-open-korean-instructions"
# MODEL_PATH = "skt/A.X-4.0-Light"
# MODEL_PATH = "yanolja/EEVE-Korean-Instruct-10.8B-v1.0"
# MODEL_PATH = "Qwen/Qwen3-32B-AWQ"
# MODEL_PATH = "Qwen/Qwen3-8B"
# MODEL_PATH = "gaunernst/gemma-3-27b-it-qat-autoawq"
# MODEL_PATH = "google/gemma-3-27b-it"
# MODEL_PATH = "Qwen/Qwen3-32B"
# MODEL_PATH = "google/gemma-2-9b-it"
# MODEL_PAHT = "kakaocorp/kanana-1.5-8b-instruct-2505"
# MODEL_PATH = "meta-llama/Llama-3.1-8B-Instruct"
# qwen3 8B base
# gemma3 4B
# qwen3 14B
# Qwen/Qwen3-14B
```

+ HyperCLOVAX-SEED-Think-14B

실험

◇ 생성 성능 평가

(2) 'HyperCLOVAX-SEED-Think14B' 양자화 모델 기반 생성 성능 평가

'HyperCLOVAX-SEED-Think14B' 양자화 모델을 기반으로 리더보드에 제시된 'bleurt', "bertscor', 'ROUGE-1'의 평균 성능을 이용해 비교

- base_model: 검색된 규범 없이 질의만으로 답변 생성
- base_model w/rag: 검색 모델(DOR w/reranki)를 이용해 검색된 규범을 사용해 답변 생성
- Base_model w/slot: 슬롯 기반으로 검색한 규범을 사용해 답변 생성

Setting	k	점수
base_model	-	51.62
base_model w/rag	50	54.79
base_model w/rag	40	56.39
base_model w/rag	30	59.23
base_model w/rag	20	61.79
base_model w/rag	slot	69.28

- 검색 모델의 성능이 저조해 사용된 문서가 적을수록 성능이 향상된 것을 확인
- 별도의 검색 모델 없이 슬롯 기반 검색을 이용 했을 때, 성능이 크게 향상된 것을 확인
- 모델의 내제된 지식이 단순한 검색 방식을 이용해도 답변 성능이 높은 것을 확인

결론 및 향후연구

결론

(1) LLM 모델의 사전 학습된 knowledge 사전

국내 8B 이상의 LLM은 사전 학습된 지식이 풍부해 적은 수의 데이터로 fine-tuning을 수행하더라도 충분한 답변 생성 성능을 보임

(2) 선택 추출 검색

어문 규범과 같이 특수한 도메인 문서에 대해 pseudo labeling은 검색의 정확도를 저하시키고, 이를 이용한 답변 생성의 성능이 정확하지 않을 수 있음을 확인

(3) 랜덤성 제어를 위한 반복적 학습 (Iterative Training)

학습 세션을 나누고, 세션별 모델 랜덤성을 이용해 앙상블 효과를 기대할 수 있는 기법을 적용

(4) 양자화

모델 양자화를 통해 대회의 목적에 맞는 자원 효율적인 방법론 적용

향후 연구

(1) Hybrid Retrieval 적용

선택 추출 검색을 실패했을 때, 검색 모델의 검색 결과로 대체하여 답변 생성 성능을 향상할 수 있을 것으로 기대

(2) prompt engineering

검색된 문서의 배치와 구조에 따라 생성 성능이 변화할 수 있어 최적화된 프롬프트를 사용한다면 답변 생성 성능을 향상할 수 있을 것으로 기대

MISSION

사랑과 신뢰를 받는
제품과 서비스를 제공하여
인류의 풍요로운 삶에 기여한다

We enrich people's lives by providing
superior products and services that
our customers love and trust

롯데이노베이트