

데이터 전처리 학습된 인공지능 모델

산출물 단계	데이터 전처리
평가 산출물	학습된 인공지능 모델
제출 일자	2025.10.01
깃허브 경로	SKN14-Final-1Team
작성 팀원	이나경, 안윤지, 정민영, 김준기

1. 모델 목적 및 개요

목적 1: 사내 내부 문서 기반 질의응답

- 모델: Qwen3-8B (LoRA 파인튜닝)
- 목적: 회의록, 업무 가이드 등의 문서를 바탕으로 구성원 질의에 적절히 응답할 수 있는 내부 QA 시스템 구축

목적 2: API 문서 기반 질의응답

- 모델: OpenAI GPT 모델 + Dense(벡터) + BM25(키워드) 앙상블 기반 RAG 시스템
- 목적: 공식 API 문서에 기반하여 개발자가 자연어로 질문 시 정확한 문서 기반 응답을 제공하는 외부 API 문서 Q&A 시스템 구축

2. 모델 아키텍처 설계

모델 1: Qwen3-8B (LoRA Fine-Tuning 기반)

- 아키텍처 개요
 - 본 모델은 Qwen3-8B를 기반으로 하여 LoRA Fine-Tuning을 적용한 구조를 가지고 있습니다. 아키텍처는 크게 입력층, 본체, 출력층으로 나눌 수 있습니다.
 - 입력층에서는 Qwen 전용 토큰라이저를 사용하며, 사용자 질의를 ChatML(QWEN 토큰라이저가 인식하는 채팅 형식)으로 변환하고 모델이 인식할 수 있도록 토큰화합니다.
 - 본체는 Qwen3-8B 모델과 LoRA 어댑터로 구성되며, 이를 통해 도메인 지식을 내장하고 적합한 응답을 생성합니다.
 - 마지막 출력층에서는 생성된 결과를 대화형 응답 형태로 가공하여 자연스러운 문장으로 반환합니다.
- 설계 근거
 - Qwen3-8B는 다양한 언어적 맥락을 처리할 수 있으며, LoRA Finetuning을 통해 내부 문서 지식을 효율적으로 내장할 수 있습니다.
 - 입력은 ChatML 대화 포맷(예: <|im_start|>user)에 따라 구성되어, 사용자 질문과 응답 역할 구분이 명확합니다.

모델 2: GPT-4 모델 + RAG (검색 기반 생성)

아키텍처 개요

단계	구성 요소	역할
질의 입력	사용자 질문	자연어 형태로 API 질의
문서 검색	Chroma + BM25	Dense(의미 기반, api_tags 메타 필터링) + BM25(키워드 기반, 태그 인덱스) 앙상블
생성기	GPT-4 모델	문서 컨텍스트 기반 응답 생성

설계 근거

- OpenAI GPT 모델:
 - GPT-4o (주요 질의응답, 분류)
 - GPT-4o-mini (일상/불가능 응답)
 - GPT-4.1 (답변 품질 평가)
 - GPT-4o는 구조화된 API 문서에 대해 예제, 오류 설명, 사용법 등 다양하게 대응이 가능합니다.
 - GPT-4o는 고성능 reasoning 능력을 갖추고 있으며, 외부 문서 검색(RAG)과 결합해 정확한 문서 기반 응답이 가능합니다.
-

3. 모델 학습 요약

Qwen3-8B (LoRA)

항목	값
사용한 GPU 종류	B200 x 1 (RUNPOD)
파인튜닝 방식	LoRA (PEFT)
학습 데이터	멀티턴 학습 데이터(사내 문서 기반) 총 2368개
Epoch 수	3
GPU당 배치 크기	4
그래디언트 누적 스텝 수	2
학습률	1e-4
워밍업 비율	0.03
평가 기준	응답 일관성, 멀티턴 답변 기능 체크

GPT-4 모델 + RAG

항목	값
검색 문서	크롤링한 API 문서 + QA
문서 임베딩	원문 디비 : BGE-m3 기반 1500 token 청크 QA 디비 : QA set 단위로 청크
검색 DB	Chroma + BM25 Hybrid (원문/QA DB 분리) BGE-M3, BM25 가중치 8 : 2
검색 Top-k	1차 실행: 원문 5, QA 20 > 재실행: 원문 15, QA 30
응답 모델	GPT-4o API

4. 저장 및 배포

Qwen3-8B (LoRA)

항목	내용
저장 방식	파인튜닝된 LoRA 가중치를 기본 모델에 <code>merge_and_unload()</code> 로 병합
업로드 위치	허깅페이스 리포지토리 <code>SKN14-Final-1Team/qwen3-8b-informal-formal-merged-09-19</code>

Qwen3-8B + RAG

항목	설명
벡터 DB 위치	./services/chroma_db폴더 내부 cto,frontend, backend, data_ai 벡터 디비
임베딩 모델	BAAI/bge-m3
RAG 설정	Top-k=7
LangChain 구조	일상 질문 -> 일상 답변 사내내부문서 질문 -> Fuction Tool Call로 RAG 조회-> RAG기반 답변







GPT-4 모델 + RAG

항목	설명
벡터 DB 위치	./chroma_db/ (원문), ./qa_chroma_db/ (QA)
임베딩 모델	BAAI/bge-m3
RAG 설정	Hybrid Retrieval (Chroma + BM25 앙상블), [Top-k 원문/QA] 5,20 → 10,30 metadata filtering 적용
LangGraph 구조	<pre> analyze_image ↓ classify — [api] → extract_queries → split_queries → tool → basic → evaluate — [good] → END [bad] → generate_queries → tool → basic → evaluate → END [final] → END [basic] → simple → END [none] → impossible → END </pre> <p>[api 질문 경우] extract_queries (질문과 최근 대화·이미지 분석 내용을 통합) -> split_queries (한/영 쿼리 생성) -> tool (API 태그를 선택하여 검색) -> basic (답변 생성) -> evaluate (답변 평가) -> good → 종료 -> final → 종료 -> bad → generate_queries(대체 질문 생성) → tool (API 태그를 선택하여 검색) -> basic (답변 생성) -> evaluate (답변 평가) → 종료</p>

5. 종합 평가 및 활용 계획

항목	Qwen3-8B (LoRA)	GPT-4 모델 + RAG
활용 분야	내부 QA, 커리큘럼/업무 문서 기반 응답	외부 문서 기반 API 질의응답
강점	빠른 응답, 도메인 적합성 Fuction Tool Call을 도입해서 일상 질문에는 모델의 기본 답변을 하다 사내 내부 문서에 대한 질문이 들어오면 사내 내부 문서를 조회하는 Tool Call을 선택적으로 호출한다.	최신 정보, 문서 기반 정확도
배포 계획	FastAPI	Django + API 연결

부록

- 원문 데이터 저장 경로
 - [Google API 문서 데이터](#)
 - [사내내부 문서 데이터](#)
- QWEN3 파인튜닝된 모델 저장 경로
 - 파인튜닝 모델 : [SKN14-Final-1Team/qwen3-8b-informal-formal-merged-09-19](#)
 - 모델 학습 데이터 : [SKN14-Final-1Team/qwen3-8b-informal-formal-merged-09-19](#)
- 임베딩 모델 설정 및 벡터 저장 경로
 - Chroma Vector DB
 - [구글 API 문서 & QA]
 - 구글 API 문서:  chroma_text_api_final
 - 구글 API QA:  chroma_qa_db
 - [회사 내부 문서]
 - 프론트팀:  com_front_chroma_db
 - 백엔드팀:  com_backend_chroma_db
 - Data&AI팀:  com_data_ai_chroma_db
 - CTO:  com_cto_vector_db