Jarvis

온프레미스 LLM + RAG 서비스

Github Docs

개발기간: 2024.09 - 2025.03 (6개월) | 역할: AI Agent 개발자 | 기여도: 90% (서비스 MVP 개발)

Python 3.8+ LangChain Latest Streamlit UI FAISS Vector DB

프로젝트 개요

목적	**보안이 중요한 온프레미스 환경에서 도메인 특화 문서 기반 질의응답 MVP 서비스 구축**
환경	A100 80GB GPU 서버, 폐쇄망 환경, 외부 네트워크 제한
핵심 도전	제한된 GPU 메모리(실제 20GB)에서 대용량 LLM 배포 및 질문 범위 제어

프로젝트 성과

BIlossom-70B→8B 모델 전환과 LoRA 파인튜닝으로 80% 메모리 절약과 성능 유지 동시 달성	3단계 필터링 시스템으로 불필요한 LLM 호출 60% 감소
FAISS 벡터 검색으로 문서 검색 정확도 0.70, LLM 응답 품질 0.68 달성	Streamlit 기반 실시간 디버깅 시스템 구축 및 MVP 완성

정량적 성과

지표	달성값	측정 방법
메모리 효율성	약 80% 절약	70B → 8B 모델 전환 (파라미터 추정치)
검색 정확도	0.7034	Cosine Similarity 기반
응답 품질	0.6796	의미 유사도 측정

문제 해결 과정

1) 대용량 모델 배포

상황: 온프레미스 환경에서 BIlossom-70B 모델을 사용하기 위해 A100 80GB 서버에 배포 하려고 함
제약: 1. 서버 메모리 할당 제한으로 실제 사용 가능 메모리가 20GB 수준
2. 외부 네트워크 속도 제한으로 70B 모델 다운로드 어려움
해결 과정:
1. 물리적 모델 전송으로 해결 → USB를 통한 물리적 모델 파일 전송. 외부에서 다운로드 후 A100 서버에 직접 설치
2. 메모리 최적화 시도
1) 8-bit 양자화 (bitsandbytes)
2) GPU 메모리 분산 (device_map="auto"), 모델 샤딩 (accerlerate) 적용
→ 모든 최적화 시도에도 불구하고 CUDA OOM 지속적 발생
3. **전략적 모델 다운사이징 시도**
1) BIlossom 70B 모델에서 8B 모델로 전환 + vLLM 최적화 적용
결과: **메모리 사용량 80% 절약, 안정적 서비스 운영**

2) 질문 범위 제어 실패

상황: LoRA 파인튜닝 후 일반 질문 ("안녕", "자전거가 뭐야?")에도 도메인 특화 정보가 혼입되는 현상 발생
제약: 질문 범위를 제어하고 각 질문에 대해서 안전하고 정확한 응답 시스템이 필요
해결 과정:
1. 파인튜닝 데이터 확장 검토 → 시간 제약으로 불가
2. 프롬프트 수정만으로 해결 → 효과 불확실.
3. 리소스 효율성과 확실성 확보를 위한 **3단계 필터링 시스템 구축**
1) 금지어 키워드 사전에 매칭하여 질문 사전 차단
2) 질문 타입에 대한 키워드 분류 및 SentenceTransformer 기반 질문 분류 (90개 질문 사전) 활용
3) 질문 타입 별 차별화된 프롬프트 적용
결과: **불필요한 RAG 연산을 감소시켜 시스템 효율성과 리소스를 최적화하였고 도메인 외 질문에 대한 안전한 응답 체계 구축**

3) RAG 파이프라인 최적화

상황: 도메인 특화 문서(1개)의 구조와 참조 관계로 인한 청킹 어려움
제약: 문서 구조 파괴 시 문맥 손실, 너무 큰 청크 시 검색 정확도 저하 예상
해결 과정:
1. 다양한 청크 사이즈와 OCR method의 조합으로 테스트 진행
2. 결과를 육안으로 확인 후 Chunk_size=500/overlap=200, pymupdf4llm 최종 설정
3. 한국어 특화 임베딩 모델와 FAISS를 활용하여 문서 vector화
4. 일반 질문에 대한 RAG는 무시, 도메인 일반/특화 질문에 대해서 FAISS-DOCS-context로 파이프라인 구성
결과: 질문에 대한 **검색 정확도 0.7034 도출**