김광훈

경기도 안양 010-3616-0447 kikiru328@gmail.com 1993.03.28

LLM 기반 개인 맞춤형 학습 플랫폼 1인 서비스 개발

컴퓨터과학과 백엔드 개발을 독학하던 중 기존 온라인 학습 방식의 한계를 직접 체감했습니다. 단순히 책의 목차를 따라가는 방식으로는 전체 학습 흐름을 파악하거나 우선순위, 개념 간 연계성을 이해하는 데 어려움이 있었습니다. 다행히 경험 많은 멘토의 도움을 받아 체계적인 학습 순서와 개념 간 맥락, 그리고 지속적인 피드백을 통해 효과적으로 학습할 수 있었습니다.

이 과정에서 구조적 학습 설계와 피드백 시스템의 중요성을 깨닫고, 관련 연구를 조사한 결과 대부분의 온라인 **학습 플랫폼의 완료율이 5-15%에 불과**하다는 문제를 발견했습니다. 이러한 낮은 완료율의 원인이 체계적인 학습 설계 부재와 개인화된 피드백 시스템의 한계에 있다고 판단했습니다.

이러한 문제를 해결하고자, 멘토가 없는 자기주도 학습 환경에서 발생하는 학습 설계 부재, 피드백 부족, 동기 저하 문제를 LLM 기반 AI를 활용한 개인 맞춤형 학습 코치 시스템으로 개선하고자 했습니다. 단순히 강의 추천만을 제공하는 기존 방식에서 벗어나, 사용자 목표에서 시작해 커리큘럼 자동 생성 → 주차별 요약 제출 → AI 피드백 제공 → 학습 개선으로 이어지는 구조적이고 지속 가능한 학습 사이클을 지원하는 확장 가능한 백엔드 시스템을 구축하였습니다.

프로젝트는 Clean Architecture 원칙에 따라 도메인 중심 설계를 적용하여 유지보수성과 확장성을 확보했습니다. Dependency Injection 패턴으로 각 계층 간의 결합도를 최소화했고, Value Object 패턴을 통해 비즈니스 규칙을 코드 레벨에서 명확히 정의했습니다. 특히 **향후 MSA 전환을 염두에 두고 7개 도메인별로 분리된 모듈과 101개의 RESTful API 엔드포인트**를 통해 Router와 Service 구조를 명확히 분리했습니다.

복잡한 비즈니스 로직을 안정적으로 지원하기 위해 Repository 패턴으로 데이터 접근 계층을 추상화하고, **SQLAlchemy 비동기 ORM 을 활용해 I/O바운드 작업**을 최적화했습니다. 그리고 아키텍처 기반 위에 실제 학습 멘토링 기능을 구현했습니다. OpenAI API를 연동하여 사용자의 학습 목표, 기간, 난이도를 바탕으로 주차 별 맞춤 커리큘럼을 자동 생성하는 시스템을 개발했습니다. 더 나아가 사용자가 제출한 주차 별 학습 요약을 AI가 분석하여 점수와 구체적인 개선 사항을 제공하는 피드백 시스템도 구축했습니다. 이 과정에서 **LLM API 성공률 98% 이상, 평균 응답 시간 6.07초의 안정적인 성능을 달성**하였으며, Value Object를 활용해 제목 길이 제한, 주차 범위, 레슨 개수 제한 등 엄격한 도메인 규칙을 프롬프트 설계 단계부터 적용하였습니다.

시스템의 안정성과 비즈니스 성과를 지속적으로 추적하기 위해 Prometheus와 Grafana를 활용한 실시간 모니터링 대시보드 또한 구축했습니다. LLM 요청 성공률, 데이터베이스 쿼리 성능과 같은 기술적 지표는 물론, 커리큘럼 생성률, 사용자 등록 수 등 비즈니스 KPI 도 실시간으로 추적할 수 있도록 했습니다. 이를 통해 학습 완료율 개선이라는 핵심 목표의 달성 여부를 정량적으로 측정할 수 있는 기반을 마련했습니다.

이러한 체계적인 개발 프로세스를 통해 **101개의 RESTful API 엔드포인트와 13개 데이터베이스 테이블, 400개 테스트 케이스를 완성하여 플랫폼의 핵심 기능을 구현**하였으며, 코드 품질 95% A 등급, 테스트 성공률 93.6% 을 달성했습니다. Docker 컨테이너를 통해 효과적으로 버전 관리 및 배포 환경을 구축했으며, 부하 테스트와 보안 검증을 완료하여 실제 사용자 문제를 해결하는 확장 가능한 시스템 설계 역량을 완성하였습니다.

특히 Clean Architecture와 도메인 중심 설계를 프로젝트에 적용하면서, 단순 기능 구현을 넘어 실제 비즈니스 가치를 창출하는 경험을 쌓았습니다. AI 커리큘럼 생성과 5단계 피드백 시스템을 완전히 구현하여 실제 사용자의 학습 완료율 개선이라는 핵심 목표를 기술적으로 해결하는 의미 있는 백엔드 개발을 경험했습니다.

도메인 기반 온프레미스 RAG+LLM AI 서비스 MVP 개발로 사업 확장

클라이언트의 **폐쇄망 환경**에서도 사용이 가능한 AI 서비스를 개발해야 하는 상황이었습니다. A100 80GB GPU 서버가 있었지만 할당받은 **메모리는 20GB**수준으로 제한되어 있었고, 외부 네트워크 속도 제한으로 대용량 모델 다운로드도 어려운 상황이었습니다. 이러한 제약 조건 하에 온프레미스 환경에서 도메인 특화 문서를 학습한 **RAG+LLM 시스템을 구축**하여 MVP를 검증하고 향후 사업 확장 가능성을 입증해야 했습니다.

이를 해결하기 위해 먼저 Bllossom-70B 모델을 활용하려고 시도했지만 **8-bit 양자화**, GPU 메모리 분산 처리, acclerate 라이브러리 등 모든 최적화 시도에도 불구하고 Memory OOM 오류가 지속적으로 발생했습니다. 따라서 근본적 해결을 위해 Bllossom-8B로 모델을 다운사이징하여 메모리 사용량을 80% 절감했습니다. 또한 도메인 기반의 모델을 구축하기 위해서 약 5,000건의 크롤링 데이터를 활용하여 파인튜닝을 하였고, RAG 모듈을 구축하였습니다. 클라이언트가 제공한 PDF 파일은 chunk size 500, overlap 200으로 나누고 한국어 특화 임베딩 모델과 FAISS 벡터 데이터베이스를 조합하여 검색 시스템을 구축했습니다.

하지만 파인튜닝과 RAG 모듈을 추가했음에도 일반 질문에서 도메인 정보가 혼입되는 문제가 발생했습니다. 이 문제를 해결하기 위해서 **3단계 필터링 시스템**을 추가했습니다. 3단계 필터링은 '금지어 키워드 사전 매칭', 'SentenceTransformer 기반 질문 분류', '질문 카테고리 별 차별화 처리'로 구성하여 시스템 효율성과 응답 품질을 동시에 확보했습니다.

그 결과 시스템 구축 완료 후 문서 검색 정확도는 **평균 0.7, 응답 품질 0.67**을 달성했습니다. 30개 도메인 특화 질문-답변 세트를 통한 검증에서 성능을 확보했습니다. 이로서 MVP 검증은 성공적으로 완료되었고, 클라이언트로부터 시스템 고도화 사업 확장의 발판이 되었습니다. 이 경험을 통해 AI 서비스 개발의 전 과정을 직접 설계부터 개발할 수 있었고, 외부 API 의존 없이 완전한 로컬 AI 서비스를 개발할 수 있었습니다. 이 경험을 통해 보안이 중요한 기업 환경에서도 데이터 보안 요구 사항을 충족하는 도메인에 특화된 AI 서비스 개발에 기여하겠습니다.

헬스케어 자동화 큐레이션 서비스로 상담 전환율 8.5배 향상

상담 전환율 2% → 19% (8.5 배 증가), 일일 8-15건 자동 처리

헬스케어 스타트업에서 개인 맞춤형 건강관리 제품 추천 자동화 서비스를 단독 개발했습니다. 기존 상담 프로세스는 "사이트 확인 → 직접 문의" 방식으로 전환율이 2%에 불과했고, CS 담당자가 매번 처음부터 제품을 설명해야 하는 비효율성이 있었습니다.

저는 이 문제를 개인화 추천과 자동화로 해결하기로 했습니다. 3개월간 풀 스택 개발자로서 기획부터 배포, 운영까지 전 과정을 담당했습니다.

첫째, 채널톡 서포트봇 이벤트 분리 시스템을 구축했습니다. 여러 서포트봇이 동시 운영되는 환경에서 큐레이션 전용 봇에만 선택적으로 반응하도록 Webhook 이벤트 필터링을 구현하기 위해 서포트봇 ID 매칭을 통해 정확한 분리 처리를 달성했습니다.

둘째, 5개 영역 평가 기반 개인화 추천 로직을 설계했습니다. 건강 상태, 식습관, 운동량, 음수량, 활동량을 종합 분석하여 목표별 (감량 / 유지 / 증량) 로 고객을 분류했습니다. 복잡한 AI 알고리즘 대신 규칙 기반 매칭으로 ≤ 1초 내 즉시 응답하는 실용적 시스템을 구현했습니다.

셋째, 기존 상담 프로세스를 변경했습니다. 2분 간단 설문으로 고객 진입 장벽을 낮추고, 설문 완료 즉시 맞춤 제품 추천과 상세 설명을 자동 제공했습니다. 이를 통해 CS 담당자는 기본 설명 대신 배송/가격 안내 등 고부가가치 상담에 집중할 수 있게 되었습니다.

넷째, AWS Lightsail 기반 비용 효율적 운영을 구현했습니다. 월 $3.50 비용으로 EC2 대비 70% 절약하면서도 24시간 무중단 서비스를 제공했습니다. Flask 기반 경량 아키텍처로 외부 서비스 의존성을 최소화하여 99.9% 가동률을 달성했습니다. 그 결과, 상담 전환율을 2%에서 19%로 8.5 배 향상했습니다. 일일 8-15건의 큐레이션 서비스를 안정적으로 처리하며, 설문 시작 고객의 95%가 완료하고 추천받은 고객의 85%가 실제 상담으로 연결되었습니다.

비즈니스적으로는 마케팅 효율성이 개선되었습니다. 고객 진입 장벽이 해소되고 자연스러운 상담 플로우가 구축되어 고객 경험이 크게 향상되었습니다. 또한 체계적인 고객 데이터 수집으로 향후 마케팅 전략 수립에 활용할 수 있는 기반을 마련했습니다. 이 프로젝트를 통해 기술과 비즈니스의 연결점을 명확히 이해하게 되었습니다. 완벽한 기술보다는 실제 사용자 문제를 해결하는 실용적 솔루션의 가치를 깨달았고, 최소 비용으로 최대 효과를 내는 아키텍처 설계의 중요성을 배웠습니다. 또한 정량적 성과 측정을 통해 기술이 비즈니스에 미치는 구체적 임팩트를 입증할 수 있었습니다.

CheXNet 최적화와 데이터 불균형 해결로 F1-Score 6배 달성

소아 흉부 폐질환 진단 및 분류 경진대회에서 저희 팀은 우승을 목표로 참가했습니다. 주최측에서 제공한 4,000장의 소아 흉부 X-ray 데이터로 7가지 질환을 분류하는 과제였지만, 심각한 데이터 불균형 문제(정상 25% vs 과대팽창 3.1%)로 인해 초기 VGG16, InceptionV3 등 6가지 모델을 시도했음에도 F1-score가 0.1~0.4에 머물렀습니다.

저는 팀리더로서 데이터 전처리와 모델 최적화를 주도하며 문제 해결에 집중했습니다. 당시 최신 의료 AI 논문 10편을 분석한 결과, 흉부 X-ray에 특화된 CheXNet 모델이 돌파구가 될 수 있음을 발견했습니다. 그리고 이를 활용하기 위해 팀원들과 역할을 분담하여 체계적으로 접근했습니다.

우선 전처리 파이프라인을 최적화했습니다. CLAHE기법으로 대비를 개선하고 U-Net을 활용한 폐 영역 분할을 시도했습니다. 하지만 일부 질환에서는 오히려 성능이 저하되는 것을 발견하고 데이터를 재 검토하여 흉강을 포함한 데이터를 사용하였습니다. 결과적으로 품질 필터링을 통해 전처리 시간을 20% 단축했습니다.

그리고 CheXNet 모델을 구현하고 최적화했습니다. DenseNet121을 백본으로하는 CheXNet에 GlobalAveragePolling2D 레이어를 추가하여 의료영상에 특화된 아키텍처를 구성했습니다. 학습은 150 epoch의 학습으로 안정적인 성능을 확보했습니다.

또한 데이터 불균형 문제를 해결했습니다. 불균형이 심한 데이터 환경을 해결하기 위해 4배 데이터 증강으로 학습 데이터를 총 16,000장으로 확장하고 class weight를 활용해 소수 클래스에 가중치를 부여했습니다. 이를 통해 소수 클래스에도 분류가 될 수 있도록 하였습니다.

이러한 문제 해결을 통해 F1-Score를 초기 0.1에서 0.63으로 6배 향상시켜 300팀 중 2위를 수상했습니다. 이 경험을 통해 의료 AI 개발의 복잡성과 도메인 지식의 중요성을 깨달았습니다. 특히 단순한 기술적 접근보다는 의료 영상의 특성을 이해하고 임상적 관점에서 문제를 해결하는 것이 핵심이라는 것을 배웠습니다. 또한 팀 내에서 각자의 전문성을 결합하고 체계적으로 접근했을 때 개인과 팀 모두 최고의 성과를 낼 수 있다는 것을 경험했습니다.