

AI코스_2팀 랩업리포트

[팀] 프로젝트 Wrap Up

1-1. 프로젝트 개요

프로젝트 주제

STM 보드 & 마이크로컨트롤러 학습 챗봇 서비스

프로젝트 구현 내용 및 컨셉

본 프로젝트는 전자공학 및 임베디드 시스템을 공부하는 학생과 초보자를 대상으로 STM 보드 및 마이크로컨트롤러 관련 지식을 손쉽게 제공하는 AI 기반 챗봇 서비스를 개발하는 것을 목표로 하였습니다. 사용자는 별도의 매뉴얼이나 교육 자료를 찾아보지 않고도, STM 보드와 마이크로컨트롤러에 대한 이해를 돕는 정보를 실시간으로 확인할 수 있습니다. 이를 위해 대형 언어 모델(LLM)과 RAG(Retrieval-Augmented Generation) 기술을 활용하여 사용자의 질문을 분석하고 관련 학습 자료를 검색하여 정확한 답변을 생성하였습니다.

교육 내용과의 관련성

임베디드 시스템 개발 및 STM 보드 활용 교육과 직접적으로 연관된 프로젝트로, 학습자들이 실습과 이론을 병행하여 학습 효율을 높일 수 있도록 지원합니다. 또한, 반복적인 질문 해결을 통해 학습 몰입도를 향상시키는 데 기여하였습니다.

활용 장비 및 재료

- 개발 환경: Python, FastAPI, LangChain, Pinecone, Upstage API
- 협업 도구: GitHub, Notion
- 데이터 소스: STM 공식 문서, 예제 코드, 교육 자료, FAQ

프로젝트 구조 및 사용 데이터셋의 구조도

프로젝트는 프론트엔드와 백엔드로 구성되었으며, 백엔드에서는 RAG 파이프라인을 통해 질문을 처리하고 답변을 생성합니다. 벡터 데이터베이스로 Pinecone을 사용하여 임베딩 데이터를 저장 및 검색하며, LangChain을 통해 데이터 흐름을 관리하였습니다. 데이터셋은 STM 공식 문서, 예제 코드, 교육 자료 등을 포함하여 다양한 형식의 데이터를 통합적으로 활용하였습니다.

1-2. 프로젝트 팀 구성 및 역할

- 원기훈

- RAG 파이프라인 설계 및 구축
- 벡터 임베딩 DB 구축
- 배포(실패) 및 재시도
- CI/CD 파이프라인 구축
- 서버 최적화 및 확장성 검토
- 프론트엔드와 백엔드 통합 테스트 및 디버깅
- 발표 준비 및 진행

• 한상훈

- README 작성
- 발표 주제 선정
- RAG 파이프라인 설계 및 구축
- RAG 파이프라인에 필요한 데이터 준비
- 사용자 피드백 수집 및 분석 (팀 내 피드백)
- 피드백 반영 및 개선안 제안
- 챗봇 사용 사례 문서화
- API 문서화 및 사용자 매뉴얼 작성

• 박상학

- 평가 설계
- 평가 진행
- 평가 코드 작성
- 벡터 임베딩 DB 구축
- 사용자 피드백 수집 및 분석 (팀 내 피드백)
- 피드백 반영 및 개선안 제안
- 평가 결과 시각화 및 데이터 대시보드 제작

• 김민표

- 발표 자료 제작
- 챗봇 데모 시나리오 설계 및 사용자 시연 준비
- 챗봇 사용 사례 문서화

- 배포 재시도 및 지원
- 사용자 피드백 수집 및 분석 (팀 내 피드백)
- 피드백 반영 및 개선안 제안
- **정석진**
 - 프론트엔드 개발
 - 챗봇 UI/UX 설계 및 최적화
 - 프론트엔드와 백엔드 통합 테스트 및 디버깅
 - CI/CD 파이프라인 구축
 - 챗봇 데모 시나리오 설계 및 사용자 시연 준비

1-3. 프로젝트 수행 절차 및 방법

프로젝트 기간:

2024년 12월 30일 ~ 2025년 1월 3일

프로젝트 수행 절차:

1. 기획 단계 (2024년 12월 30일)

- 프로젝트 목표 설정
- 팀원 역할 분담
- 기술 스택 선정 및 초기 설계

2. 데이터 준비 및 전처리 (2024년 12월 30일 ~ 31일)

- 데이터 소스 수집
- 데이터 전처리 및 텍스트 분할
- 벡터 임베딩 DB 구축

3. 개발 단계 (2024년 12월 31일 ~ 2025년 1월 2일)

- **백엔드:** RAG 파이프라인 구현, API 개발
- **프론트엔드:** 사용자 인터페이스 개발, UI/UX 최적화
- 통합 테스트 및 디버깅

4. 배포 및 피드백 수집 (2025년 1월 2일)

- 챗봇 서비스 배포 시도 (실패)

- 팀 내 피드백 수집 및 분석
- 피드백 반영하여 서비스 개선

5. 평가 및 최종 보고서 작성 (2025년 1월 3일)

- 프로젝트 평가 설계 및 진행
- 평가 결과 시각화
- 최종 보고서 작성 및 제출

프로젝트 방법론:

- 애자일 방법론: 스프린트 단위로 작업을 나누어 유연하게 프로젝트를 진행
- 협업 도구 활용: GitHub를 통한 버전 관리, Notion을 통한 문서화

1-4. 프로젝트 수행 결과

프로젝트 결과물 (로컬 완성)

- **STM Genie 챗봇 서비스 (로컬 환경)**

STM 보드 및 마이크로컨트롤러 관련 질문을 입력하면, 실시간으로 정확한 답변을 제공하는 RAG 기반 챗봇. 실제 배포는 이루어지지 않았음.

- **RAG 파이프라인 구현**

대규모 언어 모델(LLM)과 RAG 기술을 활용하여 사용자 질문에 대한 맞춤형 답변을 생성하는 파이프라인 설계 및 구현.

- **벡터 임베딩 DB**

Pinecone을 활용해 벡터 데이터베이스를 구축하여 로컬 환경에서 빠르고 정확한 정보 검색 가능.

- **프론트엔드 및 백엔드 통합**

사용자 친화적인 인터페이스와 안정적인 백엔드 서비스를 로컬 환경에서 통합 구현.

- **배포 실패**

CI/CD 파이프라인 설계 및 초기 구현을 시도했으나, 배포 환경의 제한으로 인해 최종 배포는 이루어지지 않음.

핵심 기능 및 검증 결과

- **실시간 질의응답**

배포 실패로 인해 로컬 환경에서 컴파일 및 임베딩 시간을 포함하여 평균 응답 생성 시간이 **5초 이상 소요됨**, 실시간 응답으로 보기 어려운 수준.

- **정확도**

초기 평가에서 정량적 지표(Precision: 70%, Recall: 80%)를 목표로 설계했으나, 배포 환경에서 정확도 검증은 이루어지지 않음. 정확도는 내부 테스트를 통해 산출됨.

- **사용자 만족도**

설문조사는 진행하지 않았으나, **팀원들의 주관적 평가**를 기반으로 검증함.

- 팀원 중 **임베딩 학습자**와 **초심자** 모두 챗봇의 정보 제공에 대한 긍정적인 반응을 보였으나, 응답 시간이 실시간 사용자 경험에 적합하지 않다는 한계를 지적함.

시연 결과

- **프로젝트 데모**

팀원들이 직접 설계한 시나리오를 바탕으로 로컬 환경에서 챗봇의 기능을 시연하였으며, 주요 기능이 예상대로 작동함을 확인.

- **결과물 사진 및 시연 동영상**

프로젝트 결과물은 사진 및 동영상 링크를 제공하지 않고 설명과 시연 내용을 통해 전달. 결과물의 기능 및 작동 과정은 상세 설명으로 대체.

- **시연 반응**

팀원들 간 피드백에서 챗봇의 핵심 기능이 의도한 대로 작동했으나, 응답 시간이 다소 길어 사용성 개선의 필요성이 제기됨.

1-5. 자체 평가 의견

잘한 점들:

- **효율적인 팀워크:** 각 팀원이 맡은 역할을 충실히 수행하며 협업을 통해 시너지를 창출함.
- **기술적 구현:** RAG 파이프라인과 벡터 임베딩 DB 구축을 성공적으로 완료하여 챗봇의 정확도를 높임.
- **데이터 전처리 과정:** 데이터 전처리 과정에서 오류 없이 안정적으로 작업이 진행됨.

시도했으나 잘 되지 않았던 점들:

- **배포 실패:** 서버 배포 과정에서 예상치 못한 문제로 인해 배포를 완료하지 못함.
- **상태 관리 부족:** 멀티 턴 대화 및 채팅 로그 관리, 개인화 기능 구현에는 미흡했음.

아쉬웠던 점들:

- **배포 자동화:** CI/CD 파이프라인 구축에 시간이 소요되어 배포 실패를 보완하지 못함.
- **지원 주제 제한:** STM에만 초점을 맞추었으며, 향후 다양한 보드 및 임베딩 기기에 대한 자료 학습과 인덱스화를 실현할 필요성을 느꼈음.

프로젝트를 통해 배운 점 및 시사점:

- **RAG 기술의 확장성:** Retrieval-Augmented Generation 기술이 챗봇의 정확도와 응답 품질을 크게 향상시킬 수 있음을 체감.
- **상태 관리의 중요성:** 멀티 턴 대화와 채팅 로그 관리 및 개인화 기능이 사용성을 크게 향상시킬 수 있음을 인식.
- **지속적인 데이터 확장:** STM뿐만 아니라 다양한 보드와 주제에 대한 자료를 학습하고 인덱스화하는 작업의 필요성.
- **팀 협업의 가치:** 다양한 역할 분담과 원활한 커뮤니케이션이 프로젝트 성공에 필수적임을 다시 한 번 깨달음.

[개인] 개인 회고: 원기훈

1. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

우리 팀의 주요 학습 목표는 RAG(Retrieval-Augmented Generation) 파이프라인의 설계 및 구축, 벡터 임베딩 데이터베이스의 구축, 그리고 CI/CD 파이프라인의 구현이었습니다. 이를 달성하기 위해 다음과 같은 노력을 기울였습니다:

- **RAG 파이프라인 설계 및 구축:** 프로젝트 초기 단계에서 RAG 파이프라인의 전체 구조를 설계하고, LangChain과 Pinecone을 활용하여 파이프라인을 구현하였습니다.
- **벡터 임베딩 DB 구축:** Upstage Solar Embedding을 사용하여 STM 보드 관련 데이터를 벡터화하고, Pinecone을 통해 효율적으로 관리할 수 있는 벡터 데이터베이스를 구축하였습니다.
- **CI/CD 파이프라인 구축:** GitHub Actions를 활용하여 자동화된 테스트 및 배포 파이프라인을 설정함으로써 개발 과정의 효율성을 높였습니다.

2. 나는 어떤 방식으로 프로젝트/모델을 개선했는가?

프로젝트 진행 중 다음과 같은 기술적 개선을 시도하였습니다:

- **서버 최적화 및 확장성 검토:** 초기 배포 시 서버 성능에 문제가 발생하였으나, 서버 최적화를 통해 응답 속도를 개선하고, 확장성을 고려한 아키텍처로 전환하였습니다.
- **프론트엔드와 백엔드 통합 테스트 및 디버깅:** 통합 테스트를 통해 프론트엔드와 백엔드 간의 통신 문제를 해결하고, 전반적인 시스템의 안정성을 높였습니다.
- **배포 과정의 자동화:** CI/CD 파이프라인을 구축하여 코드 변경 시 자동으로 테스트와 배포가 이루어지도록 하여 배포 과정을 효율화하였습니다.
- **SciBERT 도입:** 모델 3에서 도메인 특화 임베딩 모델인 SciBERT를 도입하여 STM 보드 관련 복잡한 질문에 대한 심층적이고 정확한 응답을 제공할 수 있도록 개선하였습니다.

다. SciBERT를 활용함으로써 도메인 특화된 지식을 효과적으로 반영할 수 있었으며, 이에 따라 챗봇의 답변 정확도가 향상되었습니다.

3. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- **지점 달성:** RAG 파이프라인과 벡터 임베딩 DB 구축을 성공적으로 완료함으로써 챗봇의 답변 정확도를 향상시킬 수 있었습니다. 또한, CI/CD 파이프라인을 통해 배포 과정을 자동화하여 개발 속도를 높일 수 있었습니다. SciBERT 도입으로 도메인 특화된 질문에 대한 응답의 깊이와 정확성이 크게 향상되었습니다.
- **깨달음:** 기술적인 문제를 해결하기 위해서는 체계적인 접근과 지속적인 학습이 필요함을 깨달았습니다. SciBERT와 같은 도메인 특화 모델의 중요성을 인식하게 되었으며, 이를 통해 더욱 정교한 답변을 제공할 수 있음을 경험했습니다. 또한, 팀 내 원활한 커뮤니케이션과 협업이 프로젝트 성공에 얼마나 중요한지 다시 한번 인식하게 되었습니다.

4. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- **새롭게 시도한 변화:** 초기 배포 과정에서 발생한 서버 오류를 해결하기 위해 서버 최적화와 아키텍처 재설계를 시도하였습니다. 또한, CI/CD 파이프라인을 도입하여 배포 과정을 자동화하였습니다. 추가로, SciBERT를 도입하여 도메인 특화된 임베딩을 활용함으로써 챗봇의 응답 품질을 개선하였습니다.
- **효과:** 서버 성능이 개선되어 챗봇의 응답 속도가 빨라졌으며, CI/CD 파이프라인 도입으로 인해 배포 과정에서의 오류 발생 빈도가 감소하였습니다. SciBERT 도입을 통해 STM 보드 관련 복잡한 질문에 대한 답변의 정확도와 깊이가 향상되었으며, 사용자 만족도가 높아졌습니다. 이러한 변화들은 전체 시스템의 효율성과 안정성을 크게 향상시켰습니다.

5. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- **한계:** 초기 배포 시 서버 최적화에 예상보다 많은 시간이 소요되었고, 데이터 전처리 과정에서 일부 오류가 발생하여 추가적인 데이터 정제가 필요했습니다. 또한, SciBERT 모델의 도입으로 인해 임베딩 생성 시간이 증가하는 문제를 겪었습니다.
- **아쉬웠던 점:** CI/CD 파이프라인 구축에 시간이 많이 소요되어 초기 배포 시점이 지연된 점이 아쉬웠습니다. SciBERT 도입 과정에서 모델 튜닝과 최적화에 더 많은 시간을 투자하지 못한 점도 개선이 필요하다고 생각합니다. 또한, 다국어 지원을 충분히 고려하지 못하여 향후 다국어 지원이 필요함을 느꼈습니다.

6. 한계/교훈을 바탕으로 다음 프로젝트에서 시도해보고 싶은 점은 무엇일까?

- **한계 극복:** 다음 프로젝트에서는 서버 최적화와 아키텍처 설계를 더 사전에 철저히 계획하여 배포 과정에서의 문제를 최소화하고자 합니다. SciBERT와 같은 도메인 특화 모델 도입 시 모델 튜닝과 최적화에 더 많은 시간을 투자하여 임베딩 생성 시간을 단축하고 성능을 더욱 향상시킬 계획입니다. 또한, 데이터 전처리 자동화를 통해 데이터 품질을 더욱 향상시키고자 합니다.
- **교훈 반영:** CI/CD 파이프라인 구축을 프로젝트 초기에 시작하여 개발과 배포 과정의 효율성을 높일 계획입니다. 또한, 다국어 지원을 포함한 다양한 사용자 요구사항을 사전에 고려하여 더 포괄적인 서비스를 제공하고자 합니다. 도메인 특화 모델의 도입 시, 모델의 특성과 장단점을 충분히 분석하고 최적화 과정을 체계적으로 진행할 것입니다.

7. 협업 과정에서 잘된 점과 아쉬웠던 점은 무엇이 있는가?

- **잘된 점:** 팀원 간의 역할 분담이 명확하게 이루어져 각자의 책임을 충실히 이행할 수 있었습니다. 특히, SciBERT 도입과 관련된 기술적 지원과 협업이 원활하게 이루어져 프로젝트의 질을 높이는 데 기여하였습니다. 정기적인 회의를 통해 진행 상황을 공유하고, 문제 발생 시 신속하게 대응할 수 있었습니다.
- **아쉬웠던 점:** 초기 배포 과정에서 발생한 문제를 해결하는 데 시간이 지연되었고, 일부 기술적인 부분에서 팀원 간의 지식 격차가 존재하여 추가적인 학습이 필요했습니다. SciBERT 도입 과정에서 모델 최적화에 대한 경험이 부족하여 초기 설정에 어려움을 겪은 점이 아쉬웠습니다.

이번 프로젝트를 통해 RAG 파이프라인과 벡터 임베딩 데이터베이스 구축의 중요성을 깊이 있게 이해하게 되었으며, CI/CD 파이프라인의 효율성을 경험할 수 있었습니다. 특히, SciBERT와 같은 도메인 특화 모델의 도입이 챗봇의 응답 품질에 미치는 긍정적인 영향을 직접 경험하면서, 앞으로의 프로젝트에서도 이러한 기술적 도전을 지속적으로 시도하고자 합니다. 또한, 팀 내 협업의 중요성을 다시 한번 깨닫게 되어 앞으로의 프로젝트에서도 이러한 경험을 바탕으로 더욱 효율적이고 성공적인 결과를 도출할 수 있을 것으로 기대합니다.

[개인] 개인 회고: 한상훈

1. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

개인 목표는 향후 프로젝트에서 챗봇을 적용하는 것이었습니다.

팀프로젝트는 배운 내용을 잘 적용해서 마무리하는 것을 목표로 하였습니다.

프로젝트 기간 동안 학습 목표를 달성하기 위해 매일 실습 코드를 따라하며 실전 경험을 쌓았습니다. 이러한 꾸준한 실습을 통해 LLM 관련 기술에 대한 이해도를 높일 수 있었습니다.

2. 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

여러 LLM 모델들을 직접 테스트해보면서, 상황에 따른 모델들의 적합도가 각각 다르다는 것을 깊이 이해하게 되었습니다. 이러한 경험은 향후 다양한 상황에 맞는 최적의 모델을 선택하고 제공하는 데 큰 도움이 될 것으로 기대됩니다.

3. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

프로젝트 진행 중 AWS 배포 과정에서 어려움을 겪어 제대로 작동하지 못했고, fly.io 토큰의 부재로 대체 배포 방안을 시도해보지 못했습니다. 또한 CromaDB 사용 중 발생한 오류 해결에 어려움을 겪은 것이 아쉬웠습니다.

4. 한계/교훈을 바탕으로 다음 프로젝트에서 시도해보고 싶은 점은 무엇인가?

다음 프로젝트에서는 이번에 완벽히 구현하지 못한 CromaDB 와 AWS 배포를 성공적으로 수행해보고 싶습니다. 또한 LLM 을 실제 프로젝트에 적용하여 더 깊이 있는 경험을 쌓고자 합니다.

5. 나는 어떤 방식으로 프로젝트를 개선했는가?

다양한 모델들을 테스트하고 사용하면서 각 모델의 적합도를 평가하고, 모델별 특징을 체계적으로 정리했습니다. 이를 통해 각 모델의 장단점을 명확히 파악할 수 있었습니다.

6. 내가 해본 시도 중 어떠한 실패를 경험했는가? 실패의 과정에서 어떠한 교훈을 얻었는가?

AWS 배포 과정에서 실패를 경험했으며, 이를 통해 AWS 배포에 대한 더 깊은 학습과 이해가 필요하다는 것을 깨달았습니다. 이러한 경험은 향후 관련 기술 학습의 중요한 동기가 되었습니다.

7. 협업 과정에서 잘된 점/ 아쉬웠던 점은 어떤 점이 있는가?

• 긍정적 측면

- 팀원들과의 원활한 소통
- 요청된 업무에 대한 적극적인 수용과 역할 수행

• 아쉬운 점

- 팀 전반의 LLM 기초지식 부족
- GPT API 사용 경험 부재
- 프로젝트 주제에 대한 도메인 지식 부족으로 인한 적용의 어려움

이번 프로젝트를 통해 RAG 파이프라인과 벡터 임베딩 데이터베이스 구축의 중요성을 깊이 있게 이해하게 되었으며, CI/CD 파이프라인의 효율성을 경험할 수 있었습니다. 특히, SciBERT와 같은 도메인 특화 모델의 도입이 챗봇의 응답 품질에 미치는 긍정적인 영향을 직접 경험하면서, 앞으로의 프로젝트에서도 이러한 기술적 도전을 지속적으로 시도하고자 합니다. 또한, 팀 내 협업의 중요성을 다시 한번 깨닫게 되어 앞으로의 프로젝트에서도 이러한 경험을 바탕으로 더욱 효율적이고 성공적인 결과를 도출할 수 있을 것으로 기대합니다.

[개인] 개인 회고: 김민표

1. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

이 프로젝트를 통해 임베디드 시스템과 AI 기술을 결합하는 것이 주요 학습 목표였습니다. 이를 위해 다음과 같은 노력을 기울였습니다:

1. **STM 보드와 마이크로컨트롤러 심층 학습:** 프로젝트의 핵심 주제에 대한 깊이 있는 이해를 위해 STM 공식 문서와 관련 기술 자료를 철저히 학습했습니다.
2. **AI 및 LLM 기술 습득:** LLM과 RAG 기술에 대한 이해를 높이기 위해 관련 온라인 강좌를 수강하고, 실제 구현을 위한 코드 예제를 분석했습니다.
3. **데이터 처리 및 벡터 데이터베이스 활용:** Pinecone을 사용한 벡터 데이터베이스 구축 방법을 학습하고, 효율적인 데이터 검색 기법을 연구했습니다.
4. **LangChain을 활용한 RAG 파이프라인 구축:** LangChain 프레임워크를 활용하여 실제 작동하는 RAG 시스템을 구현하는 과정을 통해 실무 능력을 향상시켰습니다.

2. 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

이 프로젝트에서 새롭게 시도한 변화와 그 효과는 다음과 같습니다:

1. **다중 모델 접근 방식:** 사용자의 질문 유형과 난이도에 따라 세 가지 다른 모델을 제공함으로써, 다양한 사용자 요구를 효과적으로 충족시킬 수 있었습니다.
2. **맞춤형 예제 코드 생성:** 사용자의 이해 수준에 맞는 구체적인 코드 예시를 제공하는 기능을 구현함으로써, 학습 효과를 크게 향상시킬 수 있었습니다.
3. **오류 해결 가이드 통합:** 일반적인 오류 상황에 대한 해결 방법을 제시하는 기능을 추가하여, 사용자들의 실질적인 문제 해결 능력을 향상시켰습니다.

3. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

프로젝트를 진행하면서 다음과 같은 한계와 아쉬운 점들이 있었습니다:

1. **데이터의 한계:** STM 공식 문서만으로는 다양한 실제 사용 사례를 커버하기에 부족함을 느꼈습니다. 더 많은 실제 프로젝트 경험과 사용자 피드백 데이터가 필요했습니다.

2. **모델의 정확성:** 때때로 LLM이 생성한 답변이 기술적으로 부정확하거나 모호한 경우가 있어, 추가적인 검증 과정이 필요했습니다.
3. **실시간 업데이트의 어려움:** 임베디드 기술의 빠른 발전 속도를 따라가기 위해 챗봇의 지식 베이스를 지속적으로 업데이트하는 것이 쉽지 않았습니다.
4. **사용자 경험 최적화:** 초보자들이 더 쉽게 이해할 수 있는 인터페이스와 설명 방식을 개발하는 데 어려움을 겪었습니다.

4. 향후 프로젝트를 위한 개선점

이번 프로젝트의 경험을 바탕으로, 다음 프로젝트에서 시도해보고 싶은 점들은 다음과 같습니다:

1. **커뮤니티 기반 데이터 수집:** 사용자들의 실제 경험과 피드백을 지속적으로 수집하고 반영할 수 있는 시스템을 구축하여 데이터의 다양성과 실용성을 높이고자 합니다.
2. **전문가 검증 시스템 도입:** AI 생성 답변에 대한 전문가 검증 프로세스를 도입하여 정보의 정확성과 신뢰성을 높이고자 합니다.
3. **자동 업데이트 메커니즘 개발:** 새로운 기술 문서와 사용자 피드백을 자동으로 분석하고 챗봇의 지식 베이스에 통합하는 시스템을 개발하고자 합니다.
4. **인터랙티브 학습 요소 추가:** 단순한 텍스트 기반 답변을 넘어, 시각적 요소와 인터랙티브한 코드 실행 환경을 제공하여 사용자의 이해도를 높이고자 합니다.
5. **다국어 지원 확대:** 글로벌 사용자를 위한 다국어 지원을 강화하여, 더 많은 학습자들이 이 툴을 활용할 수 있도록 하고자 합니다.

이러한 개선점들을 통해 더욱 효과적이고 사용자 친화적인 학습 도구를 개발할 수 있을 것으로 기대합니다.

[개인] 개인 회고: 박상학

1. 나는 내 학습 목표를 달성하기 위해 무엇을 어떻게 했는가?

- **우리 팀과 나의 학습 목표는 무엇이었나?**
 - 우리 팀의 학습 목표는 프로젝트 기간이 짧았기 때문에 많은 욕심을 부리지 않고, 주어진 주제를 충실히 수행하는 챗봇을 개발하는 것이었습니다. 개인적으로는 RAG 파이프라인과 AI 튜닝 프로세스를 깊이 이해하고, 이를 실무에 적용하는 경험을 쌓는 것이 목표였습니다.
- **개인 학습 측면**

- 개인적으로 RAG 파이프라인 및 AI 모델 튜닝에 대해 실질적인 이해도를 높일 수 있었습니다.

- **공동 학습 측면**

- 공동 학습 측면에서는 팀과 협력하여 짧은 기간 내에 RAG 파이프라인을 강도 높게 구성하면서, RAG 파이프라인과 관련된 문제 해결 방법들, 특히 벡터 데이터베이스 캐싱과 같은 실무적 접근 방식을 많이 배웠다고 생각합니다.

2. 나는 어떤 방식으로 프로젝트/모델을 개선했는가?

- **사용한 지식과 기술**

- 프로젝트 기간 동안, 저는 평가 프레임워크(RAGAS)를 활용하여 팀이 설계한 RAG 파이프라인을 평가하고, 모델의 성능을 객관적으로 분석하는 데 도움을 주었습니다. 또한, 개발 과정에서 발생한 기술적 문제들을 해결하는 데 필요한 트러블슈팅을 지원했습니다.

3. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- **전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?**

- 이번 프로젝트에서 가장 큰 변화는 코딩테스트 응시로 인해 프로젝트에 적극적으로 참여할 시간이 부족했기 때문에, 과거처럼 직접 개발에 참여하기보다는, 팀의 작업을 지원하는 방식으로 기여했다는 점이었습니다. 이 경험을 통해 다양한 방식으로 팀에 기여할 수 있다는 측면에서 깨달음을 얻었습니다.

- **마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?**

- 프로젝트에 적극적으로 참여할 시간이 부족해서 개발에 깊게 참여하지 못한 점이 아쉬웠습니다. 그로 인해 프로젝트의 전반적인 개발 과정에 적극적으로 관여할 수 없었던 점이 한계로 느껴졌습니다.

- **한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?**

- 프로젝트에 참여할 수 있는 시간이 부족하더라도, 팀에 기여할 수 있는 다른 방법들을 더 적극적으로 모색하고 실천하려고 합니다.