

2025년 전남대학교 소프트웨어중심대학사업 소·중·대 산학협력프로젝트(캡스톤디자인) 결과보고서

프로젝트명	깃스타						
Github url 주소	https://github.com/ssotgitstar/gitstar-backend.git						
팀 명	숫			과제수행기간	2025. 9. 24. ~ 12. 19.		
지도교수	학 과	컴퓨터정보통신공학과		성 명	김명진		
프로젝트 수행인원 (※팀장은 첫줄에 기입)	이 름	학과(부·복수전공)	학년	학번	연락처(HP)	E-Mail	
	팀장	박진석	컴퓨터정보통신공학과	3	223400	01064238996	jjinseok3639@gmail.com
	팀원	정의준	컴퓨터정보통신공학과	3	223364	01091778996	apea8888@gmail.com
참여 기업	기업명	멘토명	직위	연락처(HP)	E-Mail		
	(주) 온디	박지현	CEO	01025378579	ondy21@naver.com		
<p>위와 같이 2025년 전남대학교 소프트웨어중심대학사업 산학협력프로젝트 지원 프로그램 결과보고서를 제출합니다.</p> <p style="margin-top: 20px;">2025년 12월 18일</p> <p style="margin-top: 20px;">신청자명(대표학생) : 박진석 </p> <p>지도교수 : 김명진 </p>							
<p>전남대학교 소프트웨어중심대학사업단장 귀하</p>							

산학협력프로젝트(캡스톤디자인) 결과보고서(요약)

프로젝트명	깃스타		
수행기간	2025. 9. 24. ~ 12. 19.	소요예산	434,433원
소요예산 세부내역	-회의비: 295,500원 -(추가지원금)SW활용비: 138,933원		
참여인원	구분	인원수	성명(모두 기재)
	교수	1	김명진
	석박사과정		
	학부생	2	박진석, 정의준
	기업체	1	(멘토) 박지현
	계	4	
추진배경	오픈소스 탐색의 어려움 ○ Github의 공개 레포지토리 수가 4억개 이상으로 오픈소스의 수와 기여자가 늘어난 것과 대비하여, 신생 프로젝트의 홍보 및 대중화가 어려워 지는 문제점이 존재. - Github의 프로젝트에 대한 직접적인 평가 지표의 부재. - 프로젝트 검색 시에 단어 기반으로 검색을 수행하여 일치하는 단어가 들어간 프로젝트만 검색할 수 있다는 단점 존재.		
목표 및 내용	기존 키워드 검색 방식의 한계를 극복하는 AI 기반 GitHub 프로젝트 탐색 및 추천 플랫폼 개발 ○ 동화된 리포지토리 분석 및 벡터화 - 신규 리포지토리의 README.md 파일을 자동으로 분석, 핵심 내용을 추출하여 벡터 임베딩으로 변환하고 데이터베이스에 구축. ○ 시맨틱 벡터 검색 기능 - 사용자가 입력한 자연어 쿼리의 의미를 파악하여, 텍스트가 정확히 일치하지 않더라도 가장 관련성 높은 프로젝트를 찾아주는 시맨틱 검색 기능을 개발. ○ 프로젝트 요약 및 설명 기능 - 사용자가 선택한 프로젝트의 핵심 기능과 사용법을 AI로 요약하여 평균 3줄 이내로 간단하게 보여주는 기능을 개발합니다. 본래 README 상단에 요약 정보를 배치하여 사용자가 프로젝트의 주제 및 기능을 빠르게 파악할 수 있도록 지원.		
기대효과	GitHub 신생 프로젝트가 겪는 구조적 문제점을 해결 ○ 프로젝트의 본질적인 내용(README)을 분석하여 잠재 사용자에게 연결 - 신생 프로젝트의 공정한 노출 기회를 제공하고 기존의 불공정한 홍보 구조를 개선하는 효과 기대. ○ 불필요한 정보 탐색 시간 축소 - 프로젝트의 지속 가능성을 높이고 유저들의 오픈소스 기여도를 증대.		

1. 프로젝트 개요

프로젝트명	깃스타
주제영역	<input type="checkbox"/> 생활 <input checked="" type="checkbox"/> 업무 <input type="checkbox"/> 공공/교통 <input type="checkbox"/> 금융/핀테크 <input type="checkbox"/> 의료 <input type="checkbox"/> 교육 <input type="checkbox"/> 유통/쇼핑 <input type="checkbox"/> 엔터테인먼트
기술분야	<input type="checkbox"/> IoT <input type="checkbox"/> 모바일 <input type="checkbox"/> 데스크톱 SW <input checked="" type="checkbox"/> 인공지능 <input type="checkbox"/> 보안 <input type="checkbox"/> 가상현실 <input checked="" type="checkbox"/> 빅데이터 <input type="checkbox"/> 자동제어기술 <input type="checkbox"/> 블록체인 <input type="checkbox"/> 영상처리 <input type="checkbox"/> 기타()
성과목표	<input type="checkbox"/> 논문게재 및 포스터발표 <input type="checkbox"/> 앱등록 <input type="checkbox"/> 프로그램등록 <input type="checkbox"/> 특허 <input type="checkbox"/> 기술이전 <input checked="" type="checkbox"/> 실용화 <input type="checkbox"/> 공모전(공모전명) <input type="checkbox"/> 기타()

2. 프로젝트 추진배경

Github의 공개 레포지토리 수가 4억개 이상으로 오픈소스의 수와 기여자가 늘어난 것과 대비하여, 신생 프로젝트의 홍보 및 대중화가 어려워 지는 문제점이 있습니다. 이의 원인으로는 Github의 프로젝트를 평가할 지표가 Star 기능 외에 마땅히 없다는 점과 링크드인, 레딧, X와 같은 커뮤니티 중심의 입소문으로 퍼지는 형태를 취하고 있기 때문입니다. 이로 인해 유저들이 유명한 프로젝트에만 머물며 새로운 아이디어를 접하는데 어려움이 있습니다. 또한, 프로젝트 검색 시에 단어 기반으로 검색을 수행하여 일치하는 단어가 들어간 프로젝트만 검색할 수 있다는 단점도 문제로 작용합니다. 따라서 개발자들이 벡터 검색을 통해 Github 프로젝트를 탐색하고 평가 및 피드백 과정을 소셜 서비스와 유사한 형태로 서비스 하여 신생 프로젝트의 발견과 확산을 지원하는 플랫폼을 개발할 것입니다.

3. 프로젝트(주제) 목표 및 내용

본 과제의 목표는 기존 키워드 검색 방식의 한계를 극복하는 AI 기반 GitHub 프로젝트 탐색 및 추천 플랫폼 개발을 목표로 합니다. 이를 통해 잠재력 있는 신규 리포지토리에 조기 노출 기회를 제공하고, 개발자에게는 관심사에 최적화된 맞춤형 탐색 경험을 제공하여 오픈소스 생태계 활성화에 기여하고자 합니다. 또한, 신생 프로젝트의 수집 및 사용자 노출 콘텐츠를 생성하는 자동화 파이프라인을 구축하여 주기적으로 콘텐츠를 최신화 하는 체계를 구축할 것입니다.

과제 수행을 위해 아래의 3가지의 세부 핵심 목표로 구성하여 진행할 것입니다.

- I. 자동화된 리포지토리 분석 및 벡터화: 신규 리포지토리의 README.md 파일을 자동으로 분석, 핵심 내용을 추출하여 벡터 임베딩으로 변환하고 데이터베이스에 구축합니다.
- II. 시맨틱 벡터 검색 기능: 사용자가 입력한 자연어 쿼리의 의미를 파악하여, 텍스트가 정확히 일치하지 않더라도 가장 관련성 높은 프로젝트를 찾아주는 시맨틱 검색 기능을 개발합니다.
- III. 프로젝트 요약 및 설명 기능: 사용자가 선택한 프로젝트의 핵심 기능과 사용법을 AI로 요약하여 평균 3줄 이내로 간단하게 보여주는 기능을 개발합니다. 본래 README 상단에 요약 정보를 배치하여 사용자가 프로젝트의 주제 및 기능을 빠르게 파악할 수 있도록 지원합니다.
- IV. AI 기반 개인화 추천 시스템: 구축된 벡터 DB와 사용자 활동 데이터를 저장하고, 시맨틱 벡터 검색을 통해 개인의 기술 스택 및 관심사에 맞는 프로젝트를 선제적으로 추천하는 시스템을 구현합니다.

위의 목표 달성 이후 UI/UX 구현 및 웹서비스 테스트를 거쳐 실제로 서비스 가능한 형태로 발전시켜 실제 사용자 피드백 및 아이디어 검증 단계로 이어갈 것입니다.

4. 시스템 구성 및 내용

[개발 방법]

2.1 기획 및 기능 요구서 작성

프로젝트 기획 단계에서 구체적인 기능 요구서 작성 및 시스템 아키텍처를 설계. NAVER 클라우드 플랫폼 또는 구글 Colab을 활용한 데이터 전처리, 벡터맵 생성, 추천 서비스 개발을 위한 환경 탐색. 기능 요구서 기반으로 작업 분할 후 협업 진행.

2.2 Colab 개발환경 구축

데이터 수집 자동화 및 AI 개발 환경을 위한 Colab 개발 환경을 구축. 예산 관리를 위한 효율적인 데이터 관리 방법 선택. 데이터 수집 자동화 및 전처리, 추천 서비스를 위한 LLM 처리, 벡터맵 생성에 소요되는 컴퓨팅 자원 및 예산 할당. 구글 Drive를 활용하여 수집한 데이터를 저장하고 Colab과 연동하여 핵심 기능 개발.

2.3 데이터 파이프라인 및 백엔드 개발

Python과 FastAPI를 기반으로 핵심 백엔드를 구축. Python의 requests 라이브러리와 GitHub API 클라이언트를 사용하여 README.md 텍스트를 수집하고, SBERT 모델을 통해 벡터로 임베딩하여 FAISS에 인덱싱. 구글 Drive에 업로드 후 Colab의 GPU 서버에서 전처리하여 데이터 전처리. 데이터 수집 과정에서 반드시 permissive 라이선스가 명시된 프로젝트만 수집하여 저작권 문제 예방.

2.4 시맨틱 검색 구현

구축된 FAISS 인덱스를 바탕으로 사용자 쿼리를 처리하는 로직 구현. 사용자 입력을 SNBERT 모델을 활용하여 검색 벡터를 생성하고 FAISS에서 가장 유사한 깃허브 프로젝트 탐색.

2.5 프로젝트 요약 및 설명 기능 개발

사용자가 선택한 깃허브 프로젝트를 경량 LLM 모델로 요약하여 제공하는 API 구축. 사용자 요청 시 LLM으로 핵심 내용을 요약하고 결과를 DB에 저장하여 다른 사용자가 열람 시 LLM을 거치지 않고 DB에서 불러오도록 설계.

2.6 데이터 추천 서비스 개발

데이터 파이프라인을 기반으로 FastAPI로 핵심 API 구현. 사용자 행동 데이터를 PostgreSQL로 저장 후 사용자 관심 키워드 추출. 키워드 기반으로 시맨틱 검색을 수행하여 개인화 추천 API 개발. 유사도 기반으로 Top-N 프로젝트 선별 후 사용자에게 콘텐츠 노출. 추천 서비스를 위한 사용자 행동 데이터 수집 시 개인정보 보호 위원회의 정책에 위배되지 않는 범위 설정.

2.7 백엔드-프론트엔드 연동

웹 기반 UI/UX 개발 및 API연동. 테스트 서버 구축 및 작동 테스트 진행. NAVER 클라우드 플랫폼을 활용하여 웹서비스 검증.

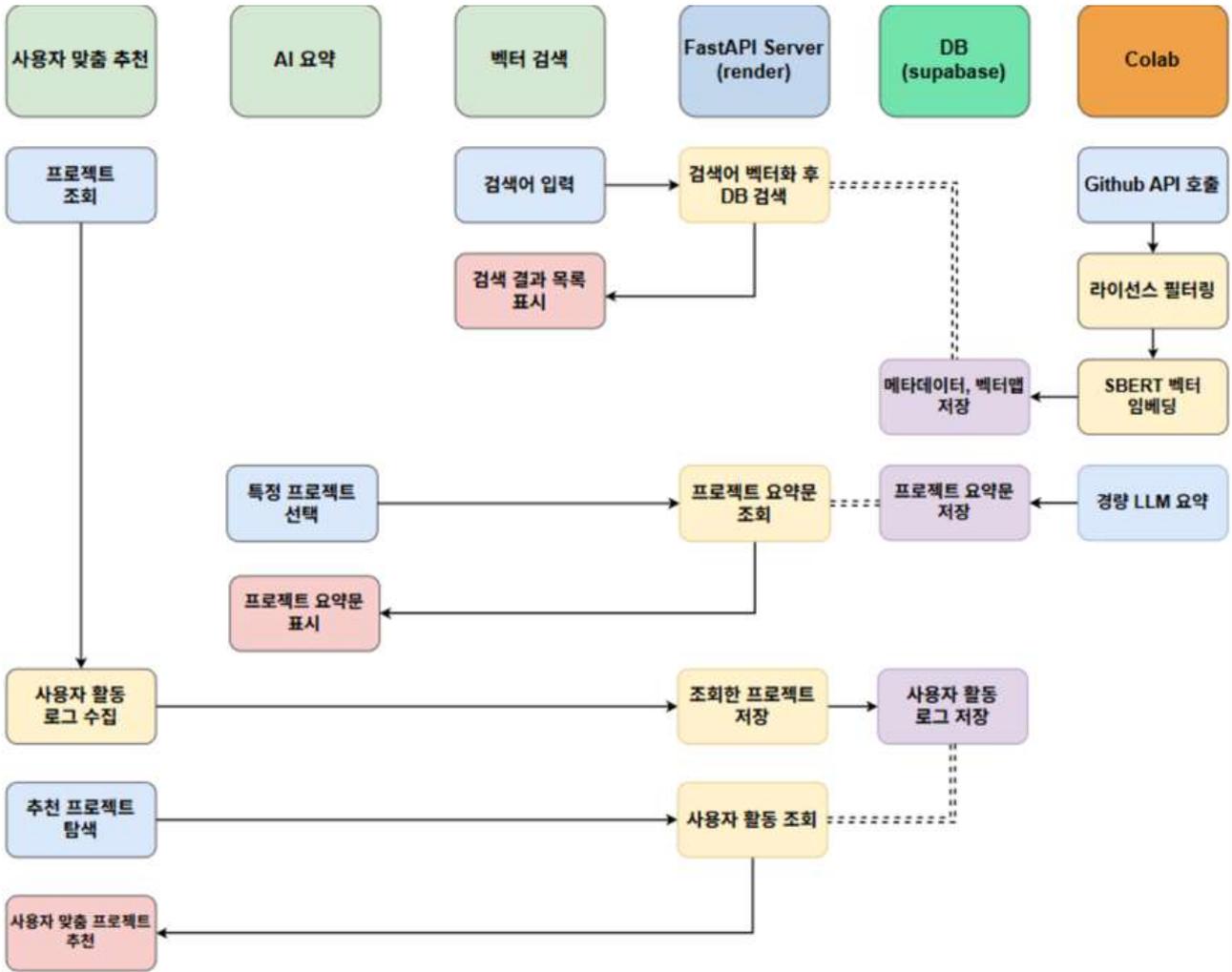
2.8 보안 검토

render 클라우드 플랫폼에서 제공하는 DDoS 등의 외부 위협 차단 방법과 사용자 입력을 받는 코드의 보안성 검토.

2.9 통합 테스트 및 최종 배포

전체 시스템에 대한 통합 테스트와 버그 수정 수행. 사용자 시나리오를 설정하고 핵심 기능의

흐름에 따라 테스트 케이스 작성.



5. 프로젝트 결과물에 대한 기술

3.1 시스템 구축 및 통합

Python과 FastAPI를 기반으로 백엔드 서버를 설계하였으며, Google Colab 환경에서 데이터 수집 및 전처리 파이프라인을 구축하였다. Colab에서 수집한 README 데이터는 Google Drive에 전처리되어 저장되고, DB에는 임베딩 벡터만 업로드하여 DB의 용량을 최적화했다. DB에 코사인 유사도 기반 인덱싱을 생성하여 대용량 데이터의 검색 속도를 향상시켜 기존 평균 25초 대비 15초 이상 감소한 평균 10초 이내로 검색 결과를 제공할 수 있었다.

3.2 LLM 기반 요약 성능 분석

검색된 프로젝트에 대해 경량 LLM 모델인 Mistral-7B-Instruct-v0.2-GGUF 모델을 사용하여 프로젝트의 요약문을 사용자에게 제공한다. 모델의 요약 성능을 평가하기 위하여 gemini-2.5-pro API를 활용하여 평가하는 방법, 요약문과 원문의 벡터 유사도를 계산하는 방법을 선택하였다.

gemini-2.5-pro를 활용하여 평가를 진행하였을 때 0-10점으로 평가하도록 프롬프트를 작성한 결과 평균 7.81, 표준편차 2.09, 최솟값 4, 최댓값 10으로 적절한 성능을 제공하고 있음을 확인했다. (그림 3-3)

요약문과 원문의 벡터 유사도를 비교하는 방법으로 평가를 진행한 결과 평균값이 0.453으로 분포는 정규 분포를 유사하게 따르나 평균값이 낮게 나타났다. (그림 3-4)

두 평가 지표를 고려하였을 때 LLM 기반 요약의 성능은 준수하나, Mistral-7B-Instruct-v0.2-GGUF 모델이 원문의 표현을 그대로 사용하여 요약하지 않고 함축적이거나 추상적인 표현으로 요약하여 고성능 LLM의 평가 점수는 높게 나온 것에 반에 벡터 유사도는 낮게 나왔을 것으로 짐작한다.'

3.3 개인화 추천 기능

사용자가 검색결과를 클릭했을 때, 해당 프로젝트를 DB에 저장하고 추천 검색 버튼을 누르면 랜덤으로 사용자가 조회했던 프로젝트를 선정하고 선정된 프로젝트의 토픽 중 랜덤으로 하나를 선택하여 검색어로 사용하도록 설계했다. 다양한 추천 알고리즘과 DL 기반 추천 방법론이 있으나, 사용 중인 백엔드 서버의 2GB 램 환경을 고려했을 때 가장 효과적인 방법으로 기능을 구현하였다.

구분	기능정의	세부기능 설명
1	자동화된 리포지토리 분석 및 벡터화	자동으로 깃허브 레포지토리 크롤링 및 DB 최신화
2	시맨틱 벡터 검색 기능	사용자의 검색어를 벡터화해서 벡터검색
3	프로젝트 요약 및 설명 기능	LLM으로 프로젝트 요약하여 사용자 제공
4	AI 기반 개인화 추천 시스템	사용자의 활동 내역 기반 추천 프로젝트 제공

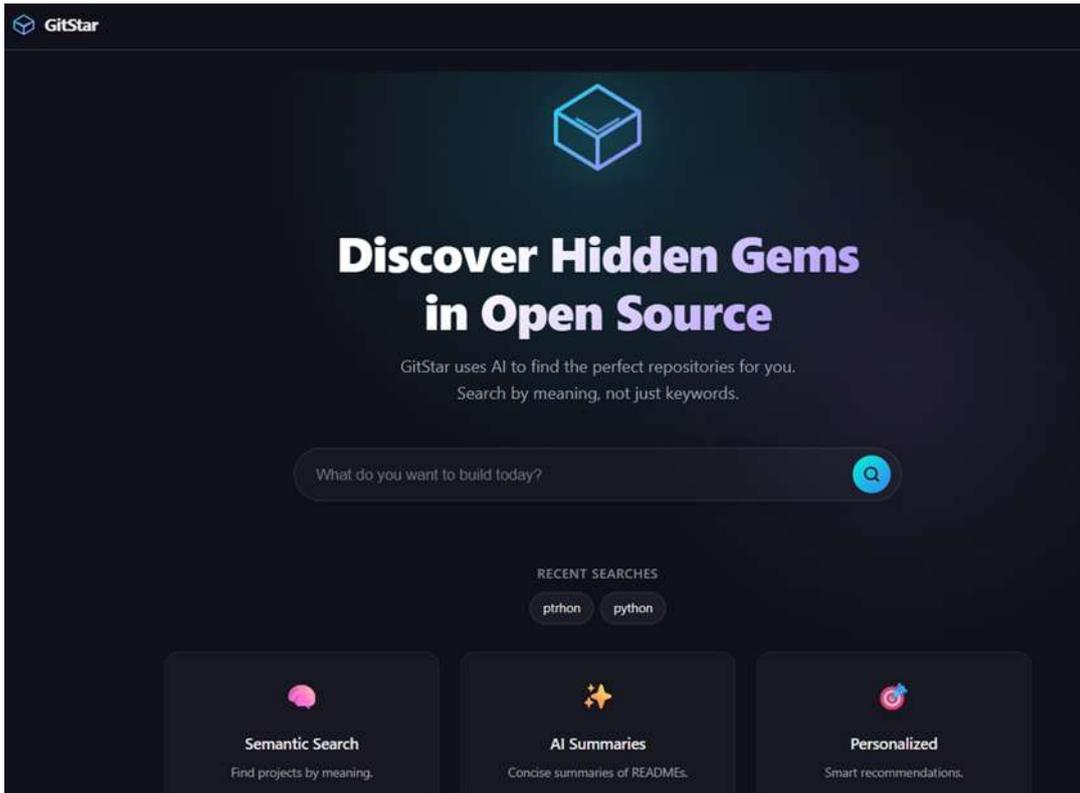


그림 3-1 깃허브 탐색 및 추천 플랫폼(웹페이지)

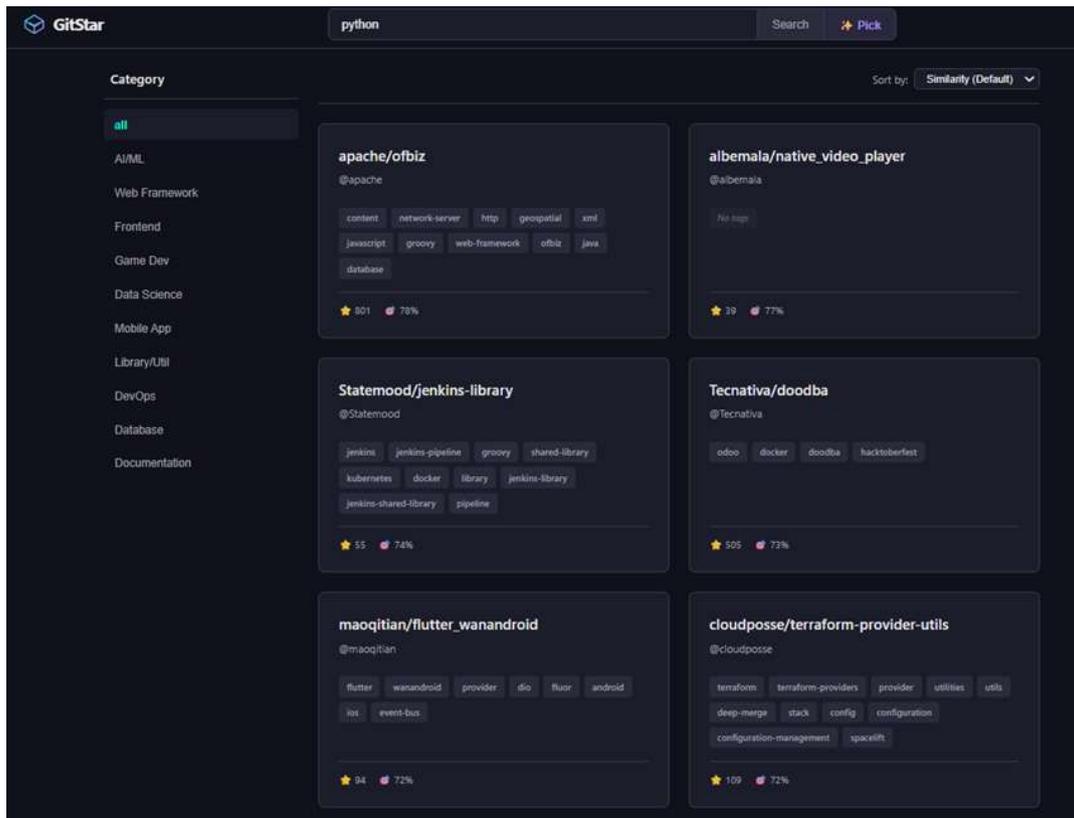


그림 3-2 "python"과 유사한 프로젝트 검색 결과

6. 프로젝트 진행내용

1) 참여인원 및 담당 역할

연번	소속학과	성명	수행역할 분담내용
1	컴퓨터정보통신공학과	박진석	DB 구축, 크롤링 파이프라인 개발
2	컴퓨터정보통신공학과	정의준	프론트엔드 구축, AI 요약 기능 개발
3			
4			

2) 회의 및 SW멘토링 진행

번호	일시/장소	회의/멘토링 내용(상세히 작성)	관련 사진
1	2025. 10. 15. (13:00~15:00) / AI 융합대학 301호	<p>벡터 임베딩 코드 작성: 크롤링된 데이터를 자동으로 벡터임베딩하여 갱신하는 코드 작성.</p> <p>크롤링 코드 블록화: 카테고리 및 스타 수 기반으로 특정 데이터에 대해서도 각각 크롤링 할 수 있도록 Colab에서 직관적이고 실행하기 쉬운 코드 블록으로 변경.</p> <p>매 크롤링 시 중복되는 데이터 관리 및 유지보수 방법 설정: 효율적인 관리를 위해 최신화된 레포지토리와 그렇지 않은 레포지토리를 구분할 방법 탐색. 데이터의 유지보수를 위해 적절한 데이터 최신화 주기 탐색.</p> <p>프론트 UI 레퍼런스 사이트 설정 및 콘텐츠의 썸네일 표시 방법 결정: Github Explore와 Google Open Source 사이트의 UI에서 GitStar의 장점을 부각하기 위한 디자인을 추출하여 웹사이트를 구성하기로 결정. Google Open Source 사이트처럼 크게 3개 항목씩 스크롤하며 확인할 수 있고, Github Explore처럼 ReadMe에서 가장 처음에 나오는 이미지를 썸네일로 설정. 이미지가 없을 경우, 프로젝트 이름으로 그라데이션 배경을 만들고 프로젝트 요약문을 가운데 배치하여 썸네일로 설정.</p>	
2	2025. 10. 17. (13:00~15:00) / 공7 229호	<p>1. 크롤링 범위 설정 및 전처리</p> <ul style="list-style-type: none"> 리드미 크롤링을 위한 적절한 검색 쿼리 설정 마크다운 형식 제거 및 불필요 텍스트 전처리 <p>허브 레포지토리 데이터 수집 현황 공유</p> <p>허브 레포지토리 수집 자동화를 위한 깃허브 검색엔진 분석 및 검색 키워드 설정. 개 항목, 10개의 star 수 범위를 설정하여 프로젝트의 카테고리, 이름, url, 라이선스, 스타, 리드미 수집 현황 공유</p> <p>표, 그림, 영상, 링크 등의 처리 방법 검증</p> <p>및 url, 이미지 등으로 인해 저장된 포맷이 손상되는 경우 확인. 파이썬 라이브러리 활용하여 불필요 서식의 처리 방법 탐구 및 검증</p>	

3	2025. 10. 23. (18:00~19:30) / AI 융합대학 301호	<p>시맨틱 검색 구현 및 테스트: 임베딩한 벡터를 chromadb를 사용하여 인메모리 DB로 사용. 사용자 쿼리에 대해서 유사도가 비슷한 프로젝트를 검색할 수 있는 함수 구현. 메타데이터와 연동하여 조건부 검색과 유사도 기반 검색이 적절하게 이루어지는지 정성적인 평가 시행</p> <p>프론트엔드 UI 검토: 웹사이트 UI 2안의 장단점을 비교하고 적절한 UI 구성과 백엔드 기능 테스트를 위한 최우선 기능 정의</p>	
4	2025. 10. 28. (14:00~15:00) / 공7 229호	<p>백엔드 PaaS 선정: 백엔드의 테스트 및 임시 서비스용 클라우드 서비스로 render 결정 및 깃허브 연동 테스트.</p> <p>프론트엔드 PaaS 선정: 프론트엔드의 테스트 및 임시 서비스용 클라우드 서비스로 vercel 결정 및 깃허브 연동 테스트.</p> <p>벡터 검색 정확도 정성적 테스트: 사용자의 검색 쿼리에 대한 유사도 기반 검색 결과를 정성적으로 평가.</p> <p>일부 항목에서 단편적으로 연관있는 프로젝트가 검색되는 문제 인식. 검색을 위한 태그 도입 고려.</p> <p>임베딩 벡터맵 DB와 프로젝트 INFO를 담은 DB 구축: render를 활용하여 프로젝트 정보를 담고 구글 드라이브에 임베딩 벡터를 저장하여 관리 및 render 연동.</p>	
5	2025. 11. 11. (15:00~16:30) / 공7 229호	<p>프로젝트 요약용 LLM 선정: 크롤링한 리드미 데이터의 내용을 요약할 LLM을 허깅페이스에서 탐색 및 정성적인 테스트 진행.</p> <p>백엔드 비동기 문제 인식: 백엔드의 동작 방식에서 여러 요청이 들어올 시 서버에서 IO wait가 커지는 문제 인식 및 해결방안 탐색.</p> <p>데이터 베이스 테이블 구조 변경: 빠른 검색과 요약 데이터 및 원본 데이터 관리를 위해서 DB 테이블 구조 결정 및 foreign키 활용 방법 모색.</p>	
6	2025. 11. 28. (14:00~15:00) / 공7 229호	<p>검색시간 측정 및 개선: 테이블 구조 개선 및 인덱스 최적화로 검색 시간 7초 이내 평균 5초로 단축.</p> <p>사용자 활동 데이터 수집: 사용자의 관심사를 제시된 프로젝트를 클릭하는 것으로 정의. 사용자가 클릭한 프로젝트의 테이블 ID를 쿠키에 저장하는 API 구현. 쿠키를 저장할 DB 테이블 생성.</p> <p>검색결과 필터링: 10개의 카테고리에 대해서 필터링하는 로직 추가. 스타 수와 유사도를 기반으로 정렬하는 필터 추가.</p>	
7	2025. 12. 2. (15:00~16:30) / 공7 229호	<p>사용자 추천 시스템 추가: 검색 버튼 우측에 추천 버튼 생성. 사용자가 클릭 시 cookie 테이블에 저장된 repository_info를 랜덤으로 가져온 후 해당 프로젝트의 topics 중 하나를 랜덤으로 선택하여 검색어로 사용하도록 추가.</p>	

		<p>벡터 검색 정확도 평가 방법 탐색: 주요 기능인 벡터 검색의 정확도 평가를 위해서 정답 Label이 필요함을 확인. LLM을 이용한 방법 등을 모색하여 성능 평가를 위한 방법 탐색.</p>	
8	<p>2025. 12. 8. (14:00~15:00) / 공7 누리라운지</p>	<p>캡스톤디자인 결과발표 PPT 제작: 문제정의 및 결과물의 정량적 평가방법 모색. vector와 halfvec 사이의 검색 정확도 측정 및 AI 요약의 정량적 평가 방법 2가지 실험하여 PPT 자료 제작.</p> <p>DB 용량 최적화: 시연 시 검색 시간 단축을 위한 DB 인덱스 추가가 필요하나 용량 부족으로 어려움 확인. 용량을 적게 소비하는 효율적인 인덱싱 방법 탐구 및 적용</p>	

7. 프로젝트 세부일정 및 내용

작업 내용	9월				10월				11월				12월				담당자	
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
프로젝트 기능요구서 작성	■																	박진석, 정의준
기술 스택 선정 및 아키텍처 설계				■														
개발 환경 구축				■														
readme.md 파싱 및 전처리				■	■													박진석
텍스트 벡터 임베딩 구현					■	■	■											
데이터 파이프라인 구현					■	■	■											
벡터검색기능 구현							■	■										정의준
사용자 활동 데이터 모델링								■	■	■								
개인화 추천 서비스 개발									■	■	■							박진석
UI/UX 디자인 및 프론트엔드 개발										■	■	■						
프론트-백엔드 연동											■	■	■					
테스트 및 버그 수정													■	■				박진석, 정의준
서비스 배포 및 검증														■	■	■		
결과보고서 작성 및 제출															■	■	■	

8. 결과물에 대한 향후 활용계획

본 프로젝트를 통해 개발된 '깃스타' 플랫폼은 기존 GitHub 생태계가 가진 '발견'과 '확산'의 구조적 문제점을 해결하는 데 직접적으로 기여할 것입니다.

신생 프로젝트의 공정한 노출 기회 제공: Star 수와 같은 인기 지표에 의존하지 않고, 프로젝트의 본질적인 내용(README)을 분석하여 잠재 사용자에게 연결함으로써, 기존의 불공정한 홍보 구조를 개선하는 효과를 가져올 것입니다. 자연어 기반의 시맨틱 검색과 개인화 추천을 통해 개발자들이 자신의 실제 관심사와 문제 해결에 가장 적합한 프로젝트를 효율적으로 발견하도록 돕습니다. 이는 불필요한 정보 탐색 시간을 극적으로 줄이고, 유명 프로젝트에만 머무르던 개발자들이 새로

운 기술과 아이디어를 접하며 개인의 성장과 기술적 시야를 넓히는 기회를 제공할 것입니다.

오픈소스 생태계 활성화: 잠재력 있는 프로젝트와 적합한 개발자를 효과적으로 연결함으로써, 프로젝트의 지속 가능성을 높이고 건강한 기여 문화를 촉진합니다. 이는 궁극적으로 오픈소스 생태계의 다양성을 증진시키는 데 기여할 것입니다.

[활용방안]

'깃스타'는 실제 운영되는 지속 가능한 서비스로 발전시키는 것을 목표로 합니다.

공식 서비스 론칭 및 사용자 피드백 기반 고도화: 프로젝트 완료 후, 공개 웹 서비스 형태로 공식 론칭하여 초기 사용자 그룹을 확보할 계획입니다. 수집된 실제 사용자 피드백(검색 만족도, 추천 정확도 등)을 바탕으로 AI 모델과 추천 로직을 지속적으로 개선하여 서비스의 완성도를 높여나갈 것입니다.

안정적인 사용자 기반이 확보되면, 고급 분석 기능이나 프로젝트 홍보 부스팅과 같은 프리미엄 기능을 도입하여 SaaS(Software as a Service) 형태의 창업 아이템으로 발전시키는 것을 장기적인 목표로 삼고 있습니다.

9. 참고자료 및 문헌

J. Hellman, E.Jang, C. Treude, C. Huang, and J. L. C. Guo, "Generating GitHub repository descriptions: A comparison of manual and automated approaches," *arXiv preprint arXiv:2110.13283*, 2021.

T. T. H. Doan, P.T. Nguyen, J. Di Rocco, and D. Di Ruscio, "Too long; didn't read: Automatic summarization of GitHub README.MD with Transformers," in *Proc.2023 Int. Conf. Eval. Assess. Softw. Eng. (EASE)*, Oulu, Finland, 2023, pp.267–272.

Y. Liu *etal.*, "G-Eval: NLG Evaluation using GPT-4 with better human alignment," in *Proc. 2023 Conf. Empirical Methods Natural Lang.Process. (EMNLP)*, Singapore, 2023, pp. 2511–2522.