



Databricks genAI 2024 Hackathon 지원기

염은지



목차

1. Databricks genAI 2024 Hackathon ?
2. 연관 개념
 - a. Databricks, AI Agent, RAG, LangChain, LangGraph
3. 프로젝트 진행 내용
 - a. 참가 동기, 선정 주제, 활용 기술, 구조도, 데모
4. 느낀점
5. Q & A

Databricks genAI 2024 Hackathon

- Databricks라는 회사에서 주최한 기업 해커톤 대회
- Gen AI 기술을 활용해 실생활의 문제를 해결하거나 개선하는 Application 개발
- 팀당 최대 \$500의 플랫폼 크레딧 제공
- 세부 내용
 - AI / Data 직무 종사자 대상 + 만 20세 이상
 - 2~4인의 팀 구성
 - 전 세계를 세 지역으로 나누어 3등까지 수상
- [대회 링크](#)
- [수상 결과](#)



Databricks

- 데이터 엔지니어링, 데이터 분석, ML 및 AI 워크플로를 통합적으로 지원하는 데이터 및 AI 플랫폼
- 주요 기능
 - a. 통합 워크스페이스 제공
 - b. 다양한 클라우드 환경 연동 지원 (AWS, MS Azure, GCP 다 지원)
 - c. 스케일링을 통한 비용 최적화 가능
 - d. Databricks Unity Catalog를 통해 데이터 보안 및 거버넌스 관리 가능
 - e. Delta Lake기반의 데이터의 버전 관리, 거버넌스, ACID 트랜잭션 지원
 - f. ETL 파이프라인을 손쉽게 구축하도록 지원
 - g. Apache Spark와 통합되어 대규모 데이터 처리에 적합
 - h. 사용자에게 친숙한 인터페이스를 가진 Databricks SQL 지원
 - i. MLflow와 통합되어 ML 모델의 실험 추적, 모델 배포 및 관리 가능
 - j. Jupyter Notebook과 유사한 Databricks Notebook 지원 (다른 사용자와 같은 노트북 공동 작업 가능 + 버전 관리 제공)
 - k. AutoML을 통해 코드 없이 머신러닝 모델을 빠르게 생성 가능
 - l. Hugging Face, Mosaic ML 와 통합하여 Gen AI 모델의 빠른 프로토타이핑 가능
- 요약하자면 ... Databricks만 쓰면 AI 관련한 업무 다 해결 가능해요! 입니다.



Databricks

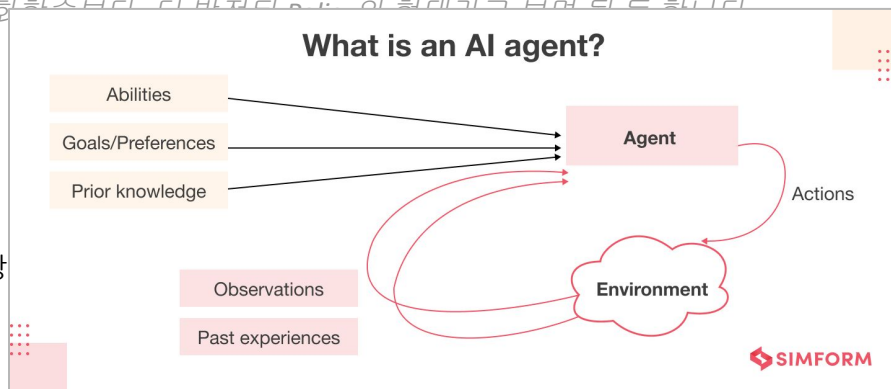
- 기본적으로 Web 기반 사용 (우측 탭을 클릭하여 원하는 페이지로 접속한 후 원하는 작업 진행)
- IDE (VS Code) Extension, Databricks CLI 도 지원하고 있음

The screenshot displays the Databricks web interface. On the left, a sidebar contains navigation options: 'New', 'Workspace', 'Recents', 'Catalog' (highlighted), 'Workflows', 'Compute', 'SQL' (with sub-items: 'SQL Editor', 'Queries', 'Dashboards', 'Genie', 'Alerts', 'Query History', 'SQL Warehouses'), 'Data Engineering' (with sub-items: 'Job Runs', 'Data Ingestion', 'Delta Live Tables'), and 'Delta Live Tables'. The main content area is divided into two sections. The top section, 'Catalog', shows a search bar and a tree view of the catalog structure. The bottom section, 'Quick access', features tabs for 'Recents', 'Favorites', and 'Catalogs', with a search filter. Below this is a table listing recent items:

| Name | Last viewed | Type |
|---|--------------|---------------------|
| fomc_presconf_year_month_day_10years_index workspace.default | 2 months ago | Vector search index |
| meta_llama_v3_1_8b_instruct system.ai | 2 months ago | Model |
| fomc_presconf_10years_index_summary workspace.default | 2 months ago | Vector search index |
| fomc_presconf_10years_index workspace.default | 2 months ago | Vector search index |
| default workspace | 2 months ago | Schema |
| system | 2 months ago | Catalog |
| fomc_presconf_10years workspace.default | 2 months ago | Table |
| workspace | 2 months ago | Catalog |

AI Agent

- 특정 작업이나 목적을 위해 설계된 시스템
- 사용자와 상호작용 하거나 주어진 목표를 수행하기 위해 자율적으로 의사결정을 내리는 AI 시스템
- 강화학습과 유사한 면이 있죠? 실제로 비교가 많이 됩니다.
 - a. 기존의 *state + reward*에 기반하여 *Policy*를 가지는 강화학습보다 더 발전된 형태가 그 보면 된 듯 합니다
- LLM 모델 활용
- 추론 능력이 중요
 - a. CoT, Prompt Chaining 기반의 추론 과정 포함 가능
 - b. 부족한 부분은 외부 환경(검색, DB)을 연동해 보강
- 예시
 - a. 커머스 기업 고객센터 Agent (문의 처리)



AI Agent

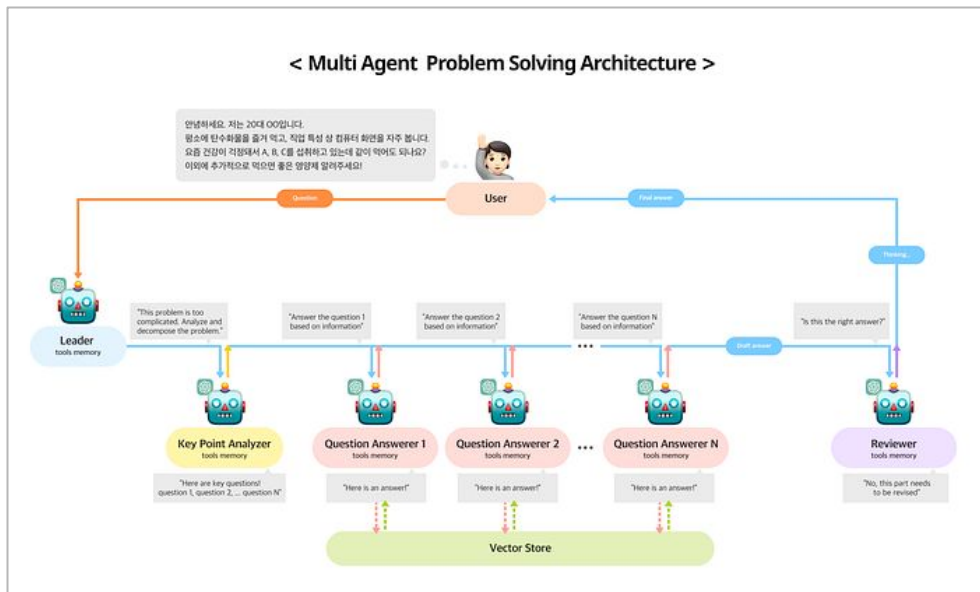
- Fine-tuning과 비교

| 특성 | Fine-tuning | AI Agent |
|----------|-------------------|--------------------------|
| 핵심 목표 | 특정 작업에서 성능 향상 | 자율적으로 문제 해결 및 작업 수행 |
| 학습 데이터 | 특정 도메인 또는 태스크 데이터 | (학습 X) 실시간으로 의사결정과 학습 수행 |
| 작업 범위 | 하나의 작업에 집중 | 복합적이고 다단계 작업 수행 가능 |
| 외부 도구 사용 | 없음 (내부 지식으로만 응답) | API, DB, 클라우드 도구와 통합 가능 |
| 성능 제한 | 학습 데이터와 태스크에 한정 | 다양한 환경에서 동적으로 문제 해결 |

AI Agent

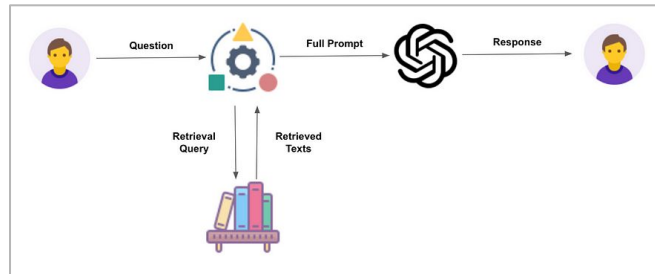
- Multi-Agent

- a. 여러개의 AI Agent를 활용하여 문제를 해결하는 시스템
- b. 각각의 Agent는 자율적으로 동작하며, 상호작용(협력 / 경쟁)을 통해 효율성 상승 및 더 큰 작업 가능
- c. 필요한 경우?
 - 풀어야 될 문제가 복잡한 경우
 - 참고해야 될 데이터 수가 많은 경우
 - 할루시네이션을 낮춰야 될 경우



RAG (Retrieval-Augmented Generation)

- LLM의 응답 성능을 강화하고 할루시네이션을 줄이기 위해 정보 검색(information Retrieval)을 적용한 기술
- 모델 내부의 학습된 데이터를 기반으로 LLM의 답변을 생성
활용해 정확하고 최신의 정보 제공 가능
- 단계
 - a. Retrieval
 - 대상 데이터를 임베딩하여 Vector DB 색인 (Indexing)
 - 질의가 들어왔을 때 Vector DB 검색을 통해 문맥적으로 관련된 텍스트 검색
 - LLM에 추가 입력으로 제공
 - b. Generation
 - 검색된 데이터를 바탕으로 LLM을 사용해 질의에 대한 최종 답변 생성



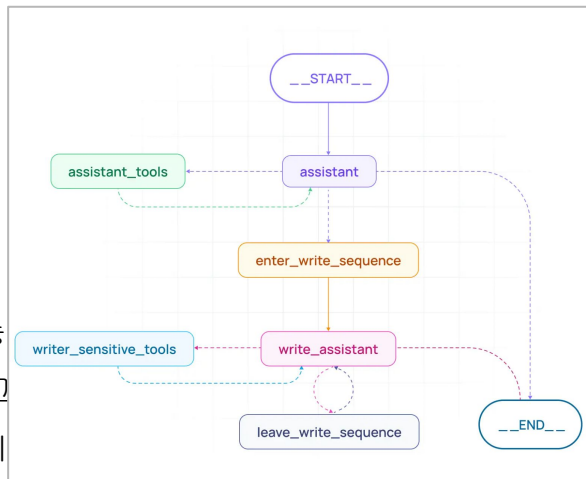
LangChain

- LLM을 활용한 Application 개발을 지원하는 Python 및 JavaScript 기반 오픈소스 Framework
- LLM과 외부 데이터, API, 메모리와 용이하게 결합하고 복잡한 기능을 구현하는데 도움
- 코드 중심의 LLM 워크플로우 설계
- [Wikidocs](#)
- 구성 요소
 - a. Prompt Templates : LLM에 제공할 프롬프트
 - b. Chains : 여러 작업을 연결하여 파이프라인 형태로 실행
 - c. Agents : 외부 도구 (API 등)을 활용해 작업을 수행 가능하도록 설정
 - d. Memory : 대화의 컨텍스트를 유지
 - e. Retrieval : Vector DB에서 유사한 정보를 검색해 LLM 입력으로 제공
 - f. Tools : API 호출, DB Query, 계산 작업 등 다양한 도구와 통합



LangGraph

- LangChain의 기능을 확장하여 복잡한 LLM 기반 워크플로를 그래프 형태로 설계, 시각화 및 관리할 수 있는 도구
- 주로 Multi-Agent 시스템과 작업 흐름을 모델링하고 관리하는 데 사용
- [Wikidocs](#)
- 주요 특징
 - a. 그래프 기반 설계 : 워크플로와 Agent를 시각적으로 구성
 - Node는 Agent의 작업, Edge는 작업 간의 데이터 흐름
 - b. LangChain과 통합 : LangChain의 Chains, Agent 등을 그래프 구조로 변환 가능
 - c. 복잡한 시스템 관리 : Agent 간의 상호작용, 데이터 흐름, 의존성을 관리하고
 - d. 확장성과 유연성 : 새로운 노드를 추가하거나 수정하여 시스템 확장에 용이
 - e. 디버깅 및 모니터링 : 그래프를 통해 작업 흐름을 추적하고 문제를 빠르게 파악 가능



참가 동기

- 사이드 프로젝트 관련 세미나를 수강한 상태
 - 사이드 프로젝트를 하나 해야겠다고 마음먹음
- 대회 종료 2주 전에, 해커톤이 진행되고 있다는 사실을 우연히 알게 됨
- 수상 가능성은 희박! 공부 목적으로 진행하는 건 좋을 듯 하다는 생각
- 현재 상태
 - Databricks를 이용해 본 적 없음
 - AI Agent 프로젝트 경험이 없음
- 마음맞는 친구를 설득해 2인 1팀으로 참가

프로젝트 진행 과정

- 남은 시간은 2주 + 둘 다 많은 시간을 할애할 수 없는 상황 -> 1주에 1번씩 만나기
- 다음과 같은 순서로 진행
 - a. (각자) 플랫폼, API 이용법 공부
 - b. (오프라인 만남) 주제 선정 + 데이터셋 결정
 - c. (각자) 1명은 데이터 수집하여 Vector DB 색인 / 1명은 코드 개발
 - d. (오프라인 만남) 코드와 데이터 병합 후 데모 영상 녹화
 - e. (리더) 코드 리팩토링, Git repo 정리, 프로젝트 설명서 제작
 - f. 제출

선정 주제

- 경제 전망을 예측 / 설명해주는 GenAI Application 개발
- 선정 이유
 - 즐겁게 프로젝트 하기 위해 관심있는 주제 선택
 - 경기가 안좋아서 AI의 경제 전망이라도 알고자 하는 마음...^^
- 활용 Dataset
 - 2015.03.18 ~ 2024.09.18 기간의 FOMC Press Conference 전문 (63개 파일, 총 11MB)
 - [Federal Reserve System](#) 홈페이지에서 PDF 파일 다운로드 가능
- FOMC (Federal Open Market Committee)?
 - 미국 연방공개시장위원회
 - Fed의 주요 통화정책을 결정하는 핵심 기구
 - 미국 경제의 물가 안정, 최대 고용, 금융 시스템 안정성을 목표로 통화정책 조율



그 유명한 제롬 파월
의장님...

활용 기술

- Model

- a. 임베딩 모델: bge_small_en_v1_5_v3
- b. LLM 모델: llama_v3_8b_instruct

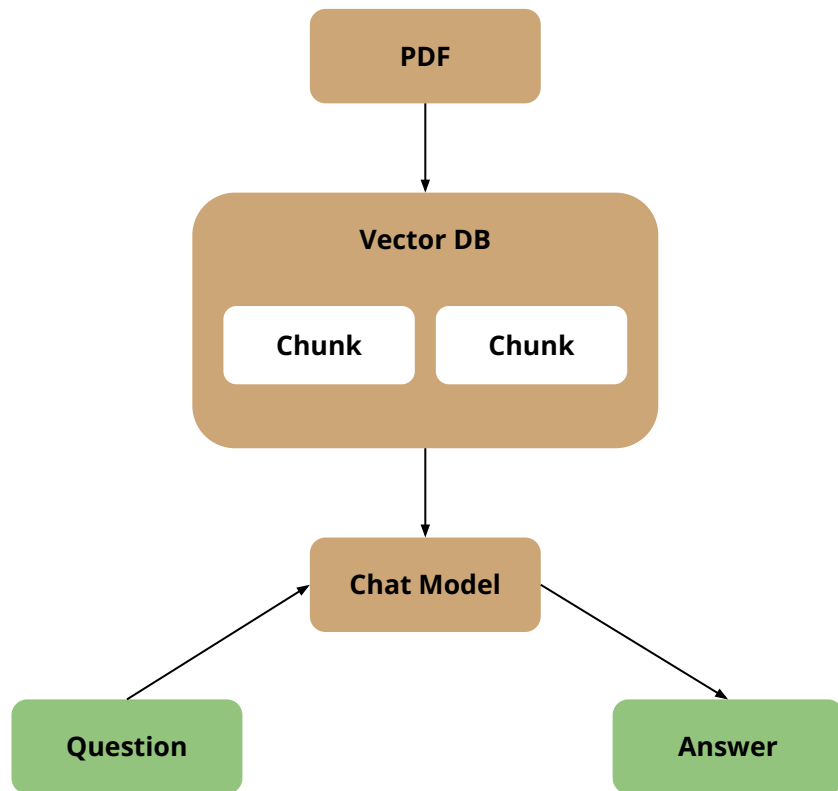
- Databricks Tools

- a. Workspace
- b. Catalog
- c. Compute
- d. Dashboards
- e. Experiments
- f. Registered
- g. Models
- h. Serving Endpoints

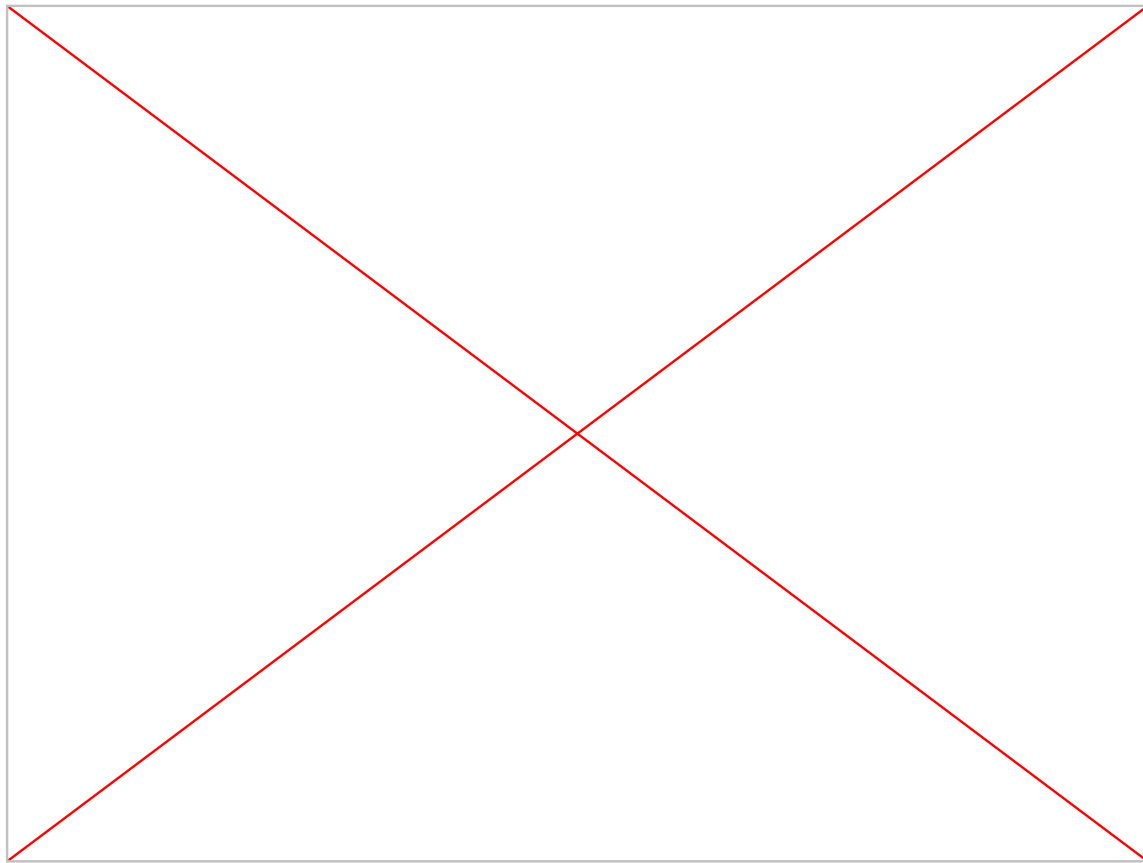
구조도

- Process

- Text Reading
- Chunking
 - 특정 문장 및 단락 수준에서 높은 정확도를 제공하기 위함
 - 긴 벡터를 활용할 경우 연산이 비효율적
- Embedding
- Vector DB Indexing
- Retrieving
- Generate LLM Chat Response
 - Date Extraction
 - 시간이 중요하므로 질의에서 Vector Search 대상 기간 추출
 - Final Answer



데모



느낀점

- 사이드 프로젝트의 일환으로써 해커톤 참여는 좋은 선택이었음
 - 공부가 확실히 되었음 + 이력서도 업데이트 가능
- Databricks 환경 및 사용법 공부 시간이 추가적으로 소요됨
 - 대회 참여 대상자에게 3시간 분량의 교육 영상을 제공해주었음 (모두 다 영어)
 - 개개인마다 공부 시간의 편차가 클 것 같음
 - 퀴즈를 통과하면 수료증을 줌
- 주제 선정의 중요성을 느낌
 - 실제 수급 가능한 데이터인지 / 해당 데이터로 원하는 예측 값이 생성이 가능할지 / 해당 주제에 대해 수요가 있을지
- \$500 credits 이내로 Databricks 플랫폼 자원을 써야 했기에, 맘 편히 실험을 진행하지 못했음
 - 예측 성능도 일부 샘플 케이스에 대해 눈으로 확인하는 정도로 정리

Q & A