ETF 상품 분석을 위한 Agentic RAG

21th Junseok Kang, DongYun Nam, Yale Lee
KUBIG(Korea University Data Science & AI Society), Korea University
(junseok0913, ehddus416, yale1234)@korea.ac.kr

고려대학교 KOREA UNIVERSITY

KUBIG DATA SCIENCE & AI 8th CONFERENCE

Introduction

ETF 투자 정보에 대한 지능형 질의응답 시스템

데이터 수집부터 서비스 배포까지 파이프라인을 통해 자동화하고, RAG 구축을 고도화하여 특히 ETF 투자 정보에 최적화된 답변을 생성하는 Agentic RAG 아키텍처를 구축한다.

배경

최근 금융 투자 시장에서 상장지수펀드(ETF)는 개인 투자자들에게 가장 중요한 투자 수단 중 하나로 자리잡았다. 하지만 각 ETF 상품마다 복잡한 투자설명서, 월간 보고서, 신탁계약서 등 방대한 양의 문서가 존재하며, 이들 문서에서 필요한 정보를 신속하고 정확하게 추출하는 것은 투자자들에게 큰 부담이 된다.

기존 AI 활용 금융정보 검색 한계

검증되지 않은 정보 수집:
- 자율적으로 웹사이트를 방문하여 검증되지 않은 정보를 수집할 수 있다.

파편화된 정보:
- 상품 설명, 가격 및 구성 종목 데이터 등 각각 상이한 형태로 분산 관리되고 있다.

멀티모달 정보 처리의 미흡:
- 차트, 표, 텍스트가 혼재된 문서에 대한 통합적인 분석 및 활용 능력이 부족하다.

시계열 특성 누락:
- 시간 정보가 포함된 질의에 대해 정확한 시계열 데이터를 제공하지 못한다.

핵심 아이디어

완전 자동화된 데이터 파이프라인 구축:
- 지속가능한 ETF 정보 수집 및 처리를 위해 자동화된 데이터 파이프라인을 구축하여 매일 업데이트되는 정보를 주기적으로 수집한다.

멀티모달 문서 처리 기술 적용:
- 텍스트, 이미지, 표 정보가 포함된 복합 문서를 통합적으로 분석하고 활용한다.

Agentic RAG 방식 도입:
- 정형 데이터와 비정형 데이터의 효율적인 검색 및 활용을 위하여 Text2sql을 통한 쿼리 작성 및 조회를 포함하는 Tool기반 Agent 검색 방식을 구현한다.

확장 가능한 RAG 파이프라인 구현:
- Self-managed Kubernetes 환경에서 시스템의 확장성과 안정성을 보장하는 RAG 파이프라인을 구축한다.

삼성 KODEX ETF 데이터 수집

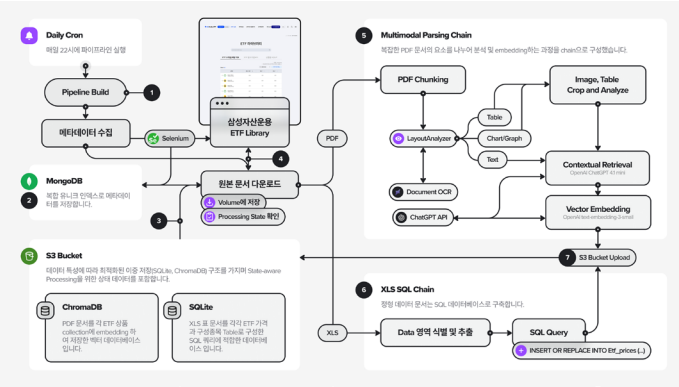
1. 기준가격 XLS 문서
일별 기준가격, 거래량, 순자산가치(NAV) 등의 정량적 시계열 데이터
2. 구성종목 정보 XLS 문서
실제 보유 종목의 주식 수, 시가총액, 포트폴리오 비중 등 상세 정보
3. 투자설명서 PDF
투자목적, 운용전략, 수수료 체계 등 ETF의 기본 투자 정보와 법정 공시 사항
4. 신탁계약서 PDF
법적 계약 조건 및 권리·의무 관계, 펀드 운용 규정 등 계약 세부사항
5. 월간보고서 PDF
해당 월의 운용 성과, 시장 동향, 포트폴리오 변화 등 분석 정보

Method

데이터 파이프라인 아키텍처

크롤링부터 Contextual Retrieval를 적용한 이중 저장 구조까지, 완전 자동화된 데이터 처리 시스템

<자동화된 데이터 수집과 RAG 구축 Workflow>

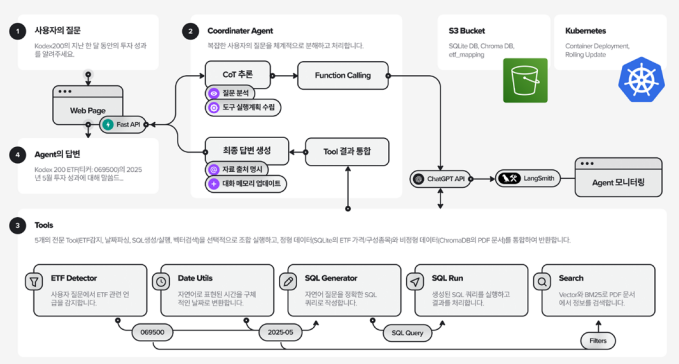


본 프로젝트의 데이터 파이프라인은 RAG기술을 핵심으로 하는 4단계 아키텍처로 구현된다.
2단계 계층 크롤링 구조: 삼성자산운용 웹사이트에서 데이터를 수집한다. Selenium을 활용하여 수집할 상품별 문서의 메타데이터를 우선 수집한 후 MongoDB에 저장하고, 이를 기반으로 ETF 매핑을 생성하여 실제 5종 문서를 다운로드한다. 이처럼 웹 크롤링과 파일 다운로드를 분리함으로써 시스템의 안정성을 확보한다.
멀티모달 파싱 체인: PDF 문서 내의 텍스트, 이미지, 표를 분리하여 분석한다. Upstage OCR과 GPT-4.1mini를 활용한 GraphParser 기반의 멀티모달 처리 방식을 적용하며, 특히 Contextual Retrieval 기법을 통해 각 텍스트 청크에 GPT-4.1mini가 생성한 맥락 요약을 추가함으로써 검색 정확도를 크게 향상시킨다.
이중 저장 전략: 데이터를 정형 및 비정형 특성에 따라 분리하여 저장한다. 정형 데이터는 SQL 쿼리 최적화가 가능한 SQLite 데이터베이스에 저장하고, 비정형 문서는 Hybrid Retrieval(Vector + BM25)이 적용된 ChromaDB에 분리 저장한다.
멱등성 보장 및 자동화된 처리: State-aware Processing을 통해 멱등성을 보장함으로써 대규모 문서 처리 시 중단 및 재시작이 가능하도록 설계한다. 모든 데이터 처리 과정은 Jenkins CI/CD 및 AWS S3 동기화를 통해 완전 자동화된다. 이를 통해 Tool-Augmented RAG 환경에서 정형 및 비정형 데이터의 통합적인 활용이 가능하다.

Agentic RAG 기반 챗봇 아키텍처

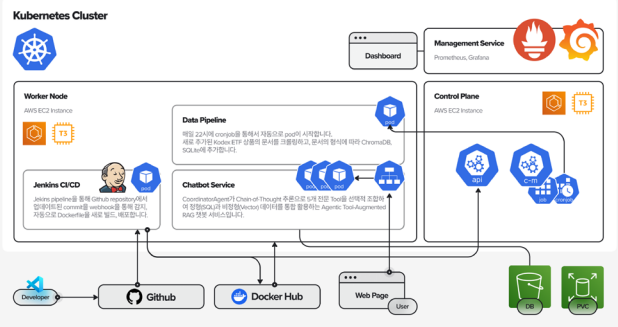
Agent의 추론을 통한 Tool 조합과 Text2sql, Vector+BM25 기반 지능형 ETF 정보 검색 및 질의응답 시스템

<사용자 질의와 챗봇 응답 Workflow>



본 프로젝트의 챗봇 시스템은 Function Calling 기반 Coordinator Agent를 중심으로 구성된다. CoT 추론을 활용하여 사용자 질문을 단계별로 분석하고, 어떤 정보가 필요한지 파악한 다음 최적의 도구 사용 계획을 세운다.
시스템에는 ETF Detector, Date Parser, SQL Generator, SQL Runner, Vector Search 총 5개의 전문 Tool이 준비되어 있으며, Agent가 상황에 맞게 이들을 선택적으로 조합하여 실행한다.
ETF Detector는 질문에서 ETF 티커를 찾아내고, **Date Parser**는 같은 자연어 시간 표현을 구체적인 날짜 범위로 바꿔준다. 가격이나 NAV, 구성 종목 등 수치 데이터가 필요한 질문에서는 **SQL Generator**와 **SQL Runner**가 담당하여 SQL 데이터베이스에서 효율적으로 정보를 가져온다. 반면 투자 전략이나 위험 요소처럼 PDF 문서에 있는 정보가 필요할 때는 **Vector Search Tool**을 사용하고, Hybrid Retrieval를 사용하여 **Vector 검색과 BM25 키워드 검색**을 조합하여 관련 문서를 찾아낸다. 검색 시 날짜나 특정 ETF로 필터링도 가능하며, 검색이 실패할 경우를 대비해 다단계 폴백 메커니즘이 작동한다.
위와 같은 구조를 통해 정형 데이터와 비정형 데이터를 자연스럽게 통합 활용하며, FastAPI/Streamlit 웹 인터페이스와 AWS S3, **Kubernetes 인프라**로 사용성과 안정성을 보장한다.

Result



서비스 배포

AWS EC2 기반 Self-managed Kubernetes 클러스터에서 ETF 전문 챗봇 서비스를 성공적으로 배포하여 안정적으로 운영하고 있다.
이를 통해 기존 부정확한 AI활용 금융상품 검색 방식을 대화형 Agentic RAG 서비스로 발전시켜, 투자자들이 복잡한 금융 문서 없이도 즉시 필요한 투자 정보를 얻을 수 있는 실용적인 서비스를 구현했다.

Conclusion

요약

- Contextual Retrieval과 Vector Search + BM25를 결합하여 기존 AI활용 금융상품 검색 방식보다 검색 정확도를 크게 높였다.
- CoT 기반 Coordinator Agent가 5가지 전문 도구를 조합하여 정형 데이터(가격, 구성종목)와 비정형 데이터(PDF 문서)를 모두 통합적으로 활용할 수 있다.
- Self-managed Kubernetes 환경에 완전 자동화된 RAGOps 파이프라인을 구축하여 데이터 수집부터 서비스 배포까지의 전 과정을 자동화하였다.

한계

- RAG 성능 평가를 위한 정량적 지표의 부재로 시스템의 객관적인 성능 검증에 어려움이 있다.

향후 방향성

- ETF 상품을 넘어 보험, 주식, 채권, 파생상품까지 분석 대상 확장, 실시간 시장 데이터 연동과 뉴스/공시 자동 분석을 통한 종합 투자 의사결정 지원 플랫폼으로 발전할 수 있다.
- 사용자 요청에 맞는 맞춤형 시각화 및 분석 코드 실시간 생성과 같은 동적 분석 도구를 추가하여 확장할 수 있다.