**[프로젝트 기획안]  
DART 공시자료를 기반으로 삼성전자 취업준비생의 질문에 답변해주는 RAG 기반 챗봇 서비스 구현**

**AI 대전 1 팀 (강성운, 김경환, 김진실, 전제후, 최국진)**

**| 마일스톤 1**

Step 1: 프로젝트 문제 정의   
Step 2: 기획안 작성

Step 3: 데이터 선정

**| 마일스톤 2**

Step 4: RAG 서비스 아키텍처 설계

Step 5: 서비스 기능 요건 정의

Step 6: 서비스 배포

**본 문서의 무단 사용 및 불법 배포 시 법적 조치를 받을 수 있습니다**

**프로젝트 개요**

**프로젝트명 : DART 공시자료를 기반으로 취업준비생의 질문에 답변해주는 RAG 기반 챗봇 서비스 구현**

**프로젝트 개요**:  
 본 프로젝트는 RAG(Retrieval-Augmented Generation) 기술을 활용하여 여러분들만의 챗봇 서비스를 설계하고 구현하는 것을 목표로 합니다. 먼저 설계하고자 하는 서비스의 주제를 기획한 뒤, 해당 서비스에 필요한 데이터를 선정합니다. 이후, 수집된 데이터를 활용해 RAG 파이프라인을 설계하고, 이를 배포하여 챗봇 서비스를 개발하게 됩니다. 이 과정을 통해 RAG의 개념을 경험해보고 실무 적용 능력을 강화합니다.

**프로젝트 목표**

* 주제에 맞는 데이터를 선정, 정제, 활용하는 경험을 쌓습니다.
* 사용자 요구에 부합하는 AI 서비스를 기획하고, 이를 기술적으로 구현하는 과정을 체득합니다.
* RAG 기반 챗봇 설계를 통해 AI 서비스 개발 실무 경험을 배양합니다.

**프로젝트 결과물**:

* 서비스 기획안
* RAG 파이프라인
* RAG 기반 챗봇 서비스 데모

**프로젝트 로드맵**

* **마일스톤 1 (권장 작성 기간 : ~12/24 화)**Step 1: 프로젝트 문제 정의   
  Step 2: 기획안 작성  
  Step 3: 데이터 선정
* **마일스톤 2 (권장 작성 기간 : 12/24 화 ~12/30 월)**  
  Step 4 : RAG 서비스 아키텍처 설계   
  Step 5: 서비스 기능 요건 정의  
  Step 6: 서비스 배포

**Step 1: 프로젝트 문제 정의**

1. **Pain Point(문제점) 찾기**

**1. 정보 과부하와 검색의 비효율성**

취업 준비생은 기업 분석을 위해 대량의 DART 공시자료를 참조해야 하지만, 방대한 양의 텍스트 중에서 원하는 정보를 빠르게 찾기가 어렵다. 이로 인해 챗봇을 사용하더라도 필요한 공시 정보를 효율적으로 추출하지 못해, 시간과 노력을 과도하게 소모하게 된다.

**2. 복잡한 전문 용어와 해석 난이도**

DART 공시자료에는 회계·재무 등 전문 용어가 다수 포함되어 있어, 이에 대한 배경 지식이 부족한 취업 준비생이 정확히 이해하기가 쉽지 않다. 챗봇이 해당 용어들을 단순히 텍스트 형태로 전달만 할 경우, 사용자는 여전히 의미를 해석하지 못하고 혼란스러워한다.

**3. 챗봇 응답의 신뢰도·정확도 부족**

챗봇이 DART 자료를 기반으로 답변을 제시하더라도, 핵심 내용을 요약하거나 맥락을 파악하는 과정에서 오류가 발생할 수 있다. 이러한 부정확한 답변은 취업 준비생의 기업 분석 신뢰도를 낮추고, 실제 면접이나 과제 수행 시 잘못된 정보를 활용하도록 만들어 불이익을 초래할 위험이 있다.

**4. 개인화된 컨텍스트 부족**

취업 준비생마다 관심을 두는 기업이나 직무가 다르지만, 챗봇이 모든 사용자에게 동일한 방식으로만 공시자료를 제공할 경우, 개인화된 분석이나 맞춤형 안내가 어렵다. 그 결과, 사용자는 자신에게 꼭 필요한 정보를 놓치거나 중복된 데이터를 반복 검토해야 하는 비효율을 겪는다.

1. **Expected Solution(예상 해결 방안) 도출 - AI의 역할 정의**

**AI의 역할을 정의하고 문제 해결 방향을 구체화합니다.**해당 문제가 AI로 해결 가능한 영역인지, AI를 활용하면 더 효율적인지 판단합니다.

**2-1/ AI가 잘 할 수 있는 영역**

DART 공시자료를 효율적으로 탐색하고 이해하기 위해서는 대용량 텍스트 처리를 담당하는 자연어 처리(NLP) 역량과, 최신 정보를 검색·반영하는 능력이 결합된 RAG기술을 적용하는 것이 핵심이다. 우선 LLM은 방대한 공시 내용을 자동 요약·구조화해 전문 용어를 친숙한 언어로 변환하고, 패턴 인식을 통해 반복적인 검색 과정을 줄여준다. 그러나 LLM 단독으로는 최신 공시나 세부 재무 정보 등을 놓칠 수 있고, 잘못된 지식을 생성(일명 Hallucination)할 위험이 있으므로, 검색 엔진을 통해 필요한 부분만 선별해 제공하는 RAG 파이프라인이 정확도를 높이는 데 적합하다. 이를 통해 수시로 업데이트되는 DART 공시를 빠르게 반영해 취업 준비생이 최신 기업 정보를 얻을 수 있으며, 일반적인 사전학습 데이터의 한계도 극복할 수 있다. 궁극적으로, AI가 잘할 수 있는 이러한 영역을 적극 활용하면 취업 준비생은 실시간 공시 정보에 기반을 둔 정확한 챗봇 답변을 받게 되어, 면접 준비나 기업 분석 과정에서 큰 도움을 얻을 수 있다.

**2-2/ 비즈니스 임팩트 측정**

**1. Make Money**

취업준비생에게 정확한 기업 분석 정보를 제공함으로써 사용자들이 분석 과정에서 겪는 시간적·오류적 부담을 줄여 구직 효율을 높이면, 서비스 제공 기업은 구독형 수익 모델이나 프리미엄 컨설팅 연계를 통해 부가적인 매출 창출 기회를 확보할 수 있다. 또한 고품질 AI 모델을 바탕으로 차별화된 분석 결과를 제시함으로써 브랜드 인지도를 높이고, 궁극적으로 새로운 고객층을 유입시켜 지속 가능한 비즈니스 모델을 구축할 수 있다.

**2. Save Money 관점**

전문 데이터 엔지니어·서버 인프라·공시 자료를 다룰 수 있는 도메인 지식을 바탕으로 고품질 AI 모델을 운영함으로써, 회계·재무 전문가를 대거 채용할 필요성이 줄어 인건비 부담을 낮출 수 있다. 더불어 데이터 처리 자동화를 통해 핵심 정보를 효율적으로 추출하면 내부 전문 인력의 업무량을 줄이고, 모델 유지·보수에 필요한 비용과 시간을 최소화함으로써 결과적으로 운영 효율을 극대화하는 비용 절감 효과를 얻을 수 있다.

1. **Task(태스크) 유형 정의**

**1. 정보 검색(Information Retrieval)**

공시 문서 중에서 사용자 질문과 연관성이 높은 문서를 빠르게 식별하여, 분석에 필요한 문서만 추려낼 수 있다.

**2. 질의응답(QnA)**

사용자의 질문에 대해 직접 답변을 생성함으로써, 방대한 자료를 일일이 읽을 필요 없이 필요한 정보만 빠르게 제공받을 수 있다.

**3. 텍스트 요약(Summarization)**

수백 페이지에 달할 수 있는 공시 문서를 요약해 핵심 정보를 전달함으로써, 사용자 입장에서 정보 탐색 시간이 획기적으로 줄어든다.

**4. 챗봇 기반 통합 구현 및 사용자 경험 극대화**

태스크들은 챗봇이라는 단일 인터페이스 안에서 통합적으로 구현되어야 하며, 질문 의도 파악부터 문서 검색, 답변 생성, 핵심 요약으로 이어지는 순환 구조가 명확하게 설정되어야 한다.

1. **데이터 요구사항 구체화**

**1. 원천데이터 유형 확인**

필요한 데이터: 취업준비생에게 필요한 기업 정보이므로, 가장 핵심적인 원천데이터는 DART 공시자료가 된다. 구체적으로는 기업의 재무정보, 사업내용, 주주구성 등과 같은 텍스트 정보가 포함되어야 한다.

데이터의 도메인과 특징: DART 공시자료는 금융·회계 분야에 해당하며, 텍스트 중심 데이터가 많다. 분량이 많고 전문 용어(회계·재무 용어 등)가 포함되어 있으므로, 일반 텍스트 처리와 달리 재무·회계 용어 처리가 중요하다.

**2. 원천데이터 형식(format) 확인**

원천데이터의 형식: 일반적으로 DART 공시자료는 PDF나 HTML 형태로 제공된다. 일부 자료는 텍스트 형태로도 다운받을 수 있지만, 가장 흔한 형식은 PDF다.

태스크 유형과의 적합성: 챗봇을 통해 질의응답(QnA), 텍스트 요약(Summarization), 정보 검색(Information Retrieval) 등을 수행하려면, PDF나 HTML을 텍스트로 변환하고 섹션 단위로 분할하거나 구조화하는 과정이 필요하다. 해당 변환 툴이 준비되어 있는지, 변환 후 정확도(오탈자 등 오류 발생 여부)를 점검해야 한다.

**3. 데이터 소스 선정**

데이터셋을 구할 수 있는가?: 국내 기업 공시자료는 DART 시스템에서 무료로 열람이 가능하므로, 공개된 API나 웹 크롤링 등을 통해 수집이 가능하다.

출처가 명확한가?: DART는 금융감독원이 운영하는 공식 사이트이므로, 출처가 매우 명확하며 공공 데이터로 간주할 수 있다. 다만 특정 기업 정보(감사보고서, 사업보고서 등)를 저장·활용하는 과정에서 규정 준수(저작권·허가 등) 여부를 확인해야 한다.

데이터셋의 저작권: DART 자료는 공공 성격이 강하지만, 일부 내용은 기업별 저작권 정책이 혼재될 수 있으므로, 라이선스 및 활용허가 범위를 명확히 파악해야 한다.

**4. 데이터 품질 점검**데이터셋 규모 및 다양성: 국내 상장기업 수가 상당히 많고, 각 기업마다 여러 문서를 주기적으로 업데이트하기 때문에 표본의 다양성은 높다. 다만, 모든 기업을 대상으로 하려면 방대한 데이터를 효율적으로 수집·처리할 수 있는 기술 인프라가 필요하다.

데이터 신뢰도: DART 공시자료는 금융감독원이 운영하는 공식 시스템이므로 신뢰도가 높다. 다만, 문서 내 오탈자나 PDF OCR 과정에서 발생하는 텍스트 인식 오류 등에 대비하여 전처리와 품질 점검이 필수적이다.

**Step 2. 서비스 기획안 구체화**

|  |  |
| --- | --- |
| **프로젝트 기획 (안)** | |
| **서비스명 및 개요** | **서비스명**: (서비스 이름)  AI DART 컨설턴트  **서비스 개요**:  **- 서비스가 해결하려는 문제점(Problem Statement)**  취업 준비생들은 기업 분석을 위해 방대한 양의 DART 공시자료를 확인해야 하지만, 방대한 텍스트에서 핵심 정보를 찾기가 쉽지 않아 어려움을 겪는다. 게다가 전문 용어와 복잡한 회계·재무 지식이 부족해 자료를 정확히 이해하기도 어렵다. 또한 LLM 단독 챗봇만으로 분석을 진행할 경우, 최신 공시나 세부 정보가 누락되어 잘못된 정보를 제시할 위험성이 존재한다.  **- 제공하고자 하는 주요 기능 및 가치**  RAG 기반 QnA: 실시간 최신 공시자료를 검색하고, LLM이 질의응답을 제공  자동 요약(Summarization): 길고 복잡한 공시를 핵심만 발췌·정리하여 사용자 시간 절감  전문 용어 해설: 회계·재무 용어를 친숙한 언어로 변환해 혼란 최소화  개인화 기능: 관심 기업·직무·조건을 입력하면 해당 영역 위주로 맞춤형 정보를 제공  **- 타겟 사용자와 예상 사용 시나리오**  타겟 사용자: 취업 준비생, 신입·경력 구직자, 기업 분석이 필요한 일반 사용자  예상 시나리오:  1. 사용자가 관심 기업(예: A회사)을 선택해 사업보고서 핵심 요약을 요청  2. 특정 전문 용어(예: 영업이익률) 의미 설명 및 회사 현황 질의  3. 개인 프로필(관심 직무, 지원 포지션 등)을 바탕으로 맞춤형 공시 섹션 안내 |
| **목표 및 기대효과** | **서비스 목표**:  정확하고 신뢰도 높은 기업 분석 정보를 제공함으로써 불필요한 검색 과정을 최소화하고, 최신 공시 자료를 즉각 반영할 수 있다. 또한 전문 지식이 부족한 사용자도 핵심 내용을 쉽게 이해할 수 있도록 정보 접근성을 높인 인터페이스를 제공한다. 더 나아가 사용자의 관심 기업과 직무에 최적화된 맞춤형 정보를 통해 취업 효율을 높일 수 있는 개인화된 분석 지원을 구현한다.  **기대효과**:  - 사용자에게 제공되는 직접적인 이점  공시자료를 빠르고 정확하게 파악함으로써 시간과 노력을 절감할 수 있다. 또한 전문 용어를 자동으로 해설해 주어 정보 이해도가 향상되며, 이를 통해 지원 기업에 대한 면접 준비와 자료 조사가 더욱 간소화된다. 이러한 점들은 취업 준비 과정 전반을 보다 효율적으로 만들어 주는 직접적인 이점이다.  - 조직/팀/사회 등 발전에 기여할 부분 정보 격차를 해소함으로써 취업 과정에서 발생할 수 있는 불균형이 줄어들며, 기업 분석 서비스를 제공하는 회사 입장에서는 구독형 수익 등 새로운 비즈니스 모델을 창출할 수 있는 기회가 열린다. 또한 재무·회계 전문가에 대한 수요가 줄어들고 운영 효율이 극대화되는 등 전반적인 사업 환경에도 혁신적인 변화를 가져올 수 있다. |
| **타겟 사용자** | **타겟 사용자**: 취업 준비생과 구직자  취업 준비생과 구직자는 정확성과 편의성, 그리고 맞춤형 분석을 중요히 여긴다. 최신 공시 자료를 포함하여 잘못된 정보(Hallucination)를 최소화한 정확한 정보를 제공받는 것이 핵심이다. 또한 복잡한 용어나 섹션을 자동으로 요약하고, 대화형 질의응답을 통해 정보를 쉽고 빠르게 획득할 수 있는 편리한 사용 환경이 필요하다. |
| **데이터 구성 및 처리 방법** | **원천데이터 소스**:  금융감독원 DART 시스템, 기업 사업보고서, 감사보고서, 분기·반기 보고서 등 모든 공시자료, 공개 API를 통해 대량 수집 가능  **원천데이터 형식 :** JSON, XML  **원천데이터 처리 방법**:  1. 데이터 수집: DART API 또는 크롤러를 활용하여 주기적으로 신규 공시 데이터를 다운로드한다.  2. 데이터 처리  - 전처리  OCR(광학 문자 인식) 기술과 텍스트 추출 기술을 활용하여 공시 데이터에서 필요한 정보를 추출한다. 추출된 데이터는 불필요한 문장과 특수문자를 정리하는 과정을 거치며, 이 과정에서 데이터의 구조화가 이루어진다. 이후 데이터를 섹션별로 나누어 관리함으로써 효율적인 검색(Retrieval)이 가능하도록 데이터베이스를 설계하고, 이를 기반으로 검색 엔진을 구축한다.  -LLM 결합(RAG, Retrieval-Augmented Generation)  사용자의 질의를 입력받아 검색 엔진을 통해 관련 문서를 탐색한다. 탐색된 문서를 기반으로 LLM(Large Language Model)을 활용하여 적합한 응답을 생성하며, 생성된 응답은 요약, QnA, 용어 해설 등 다양한 형태로 제공된다.  -결과 모니터링 및 품질관리(Human)  정기적으로 결과 데이터를 직접 검토하여 오탈자 및 데이터 누락 여부를 점검하여 데이터 최신성과 Hallucination을 점검한다. |

**Step 3: 데이터 선정 및 전처리 가이드**

1. **기획안 - 데이터셋 내용 정리**

* **원천데이터 소스** :전자공시 DART의 API 시스템을 활용하여 기업 사업보고서, 감사보고서, 분기·반기 보고서 등 모든 공시자료 등 대량의 데이터 수집 / 필요한 경우 원본 형식의 데이터를 크롤링하여 데이터 수집
* **도메인** : 취업 / 금융 및 기업분석
* **원천 데이터 형식** : JSON, XML (DART API 데이터) / PDF, HTML (사업보고서 및 감사보고서 등 원본 형식)
* **출처** : 전자공시 OpenDART 시스템 / 전자공시시스템(DART)
* **저작권** : DART의 공시자료는 공공 목적의 사용이 가능하지만, 상업적 활용 시 별도의 규정 확인. 본 프로젝트에서는 상업적 활용의 목적이 아니기 때문에 자유롭게 사용 가능
  + DART에 공시되고있는 공시보고서 원문 등을 오픈API를 통해 활용할 수 있습니다. 활용을 원하시는 누구든지(개인, 기업, 기관 등) 이용하실 수 있습니다.
  + DART 공시원문 활용 : DART에 공시되는 공시보고서의 원문을 XML형식으로 다운로드받아 원하는 자료를 자유롭게 추출하여 사용할 수 있습니다.
  + 주요공시 및 재무정보 제공 : 사업보고서 주요항목 및 주요재무계정, 지분보고서, 주요사항보고서, 증권신고서 종합정보를 이터 형식으로 바로 활용할 수 있습니다.
  + 대용량 재무정보 제공 : 상장법인(유가증권, 코스닥) 및 주요 비상장법인(사업보고서 제출대상 & IFRS 적용)에서 제출한 정기보고서의 재무제표를 분기별로 다운로드 받을 수 있습니다.

1. **데이터 소스 선정**

본격적으로 데이터를 수집하기 이전에 고려해야할 체크리스트를 다시 한 번 되짚어봅니다.

**2-1. 데이터 소스 선정 방식**

* **사용 가능한 오픈소스 데이터셋 :** 전자공시 OpenDART 시스템에서 JSON, XML 형식으로 공시자료 제공
* **웹 크롤링을 통한 데이터 수집 :** DART 사이트 내 사업보고서 및 감사보고서 등 PDF, HTML 문서

**✅ 체크리스트**

* **오픈소스 데이터 사용 시**:
  + 데이터 출처와 저작권이 명확한가?
    - * DART 공시자료는 금융감독원이 운영하는 공식적인 시스템이기 때문에 출처는 명확하며, 저작권의 경우 공공의 목적으로 사용하면 문제가 없지만 프로젝트 진행과정에서 문제가 발생할 부분이 있는지 추가적인 확인이 필요함.
  + 데이터가 사용 조건에 부합하며, 라이선스 문제가 없는가?
    - * 취업 준비에 필요한 자료인 기업에 대한 공시자료를 공식적으로 얻을 수 있는 데이터이기 때문에 사용 조건에 부합하며, API를 활용해 수집하기 때문에 라이선스 문제는 없음.
* **데이터 수집 시**:
  + 개인정보 보호와 관련된 규정을 준수했는가?
    - * 기업에 대한 공시자료이기 때문에 개인정보 보호와 관련된 규정을 준수한다고 판단.
  + robots.txt 규정 및 크롤링 제한 조건을 확인했는가?
    - * API를 활용한 데이터 수집을 진행하기 때문에 robots.txt 규정 및 크롤링 제한 조건은 없지만, 만약 추가적으로 DART에서 크롤링을 통해 데이터를 수집할 경우에도 공개된 자료이기 때문에 따로 제한 조건은 없음.

1. **데이터 처리**

이제 수집한 원천 데이터를 원하는 방식대로 변환 및 처리하는 방법에 대해서 알아보겠습니다.

**3-1. (데이터 변환 필요 시,) 데이터 형식 변환**

크롤링을 통해 추가적인 데이터를 수집하게 될 경우, PDF 데이터를 OCR을 활용하여 텍스트로 변환하여 활용

**3-2. 데이터 정제**

* 중복 데이터 제거
* 데이터 내 오탈자 수정 및 노이즈 제거
* 불필요한 내용의 문서가 존재하는 경우 삭제

**3-3. 데이터 통합**

* 크롤링을 통해 데이터를 수집하게 될 경우 웹 크롤링 데이터 및 OCR 텍스트 데이터를 API 데이터의 형식(JSON, XML)으로 일괄적으로 통합
* API와 크롤링을 통해 수집된 데이터가 중복되는 경우 중복 데이터 제거

**3-4. 데이터 품질 점검**

* 데이터의 날짜 형식, 숫자 단위, 텍스트 인코딩 방식 등 통일
* 동일한 필드가 종류가 다른 보고서에서도 동일한 형식으로 유지되는지 점검
* 비정상적으로 표시된 데이터나 누락된 항목 검출
* 검증 방법은 샘플링을 통한 데이터셋 검증, 자동화된 데이터 검증 스크립트를 활용하여 데이터 품질을 모니터링 하되 최종적으로 직접 검토하여 데이터 최신성과 Hallucination 등 발생할 수 있는 문제점을 점검

1. **데이터 활용 계획 점검**

**4-1. 데이터와 모델 연계**

* 데이터가 **RAG 파이프라인**과 원활히 연계되도록 설계합니다.   
  데이터가 검색 엔진에 인덱싱되고, 생성 모델이 검색된 데이터를 활용하여 답변을 생성하는 프로세스 입니다.
* **RAG 파이프라인 구조 설계**
  + - 데이터 인덱싱 : 데이터를 검색 엔진에 효율적으로 저장하고 빠르게 검색할 수 있도록 구성
      * 문서를 섹션 단위로 분할 (예: 요약 섹션, 재무제표 섹션)
      * 키워드 및 문서 메타데이터(날짜, 기업명 등) 태그화
      * 벡터 데이터베이스(Pinecone)에 임베딩(Embedding) 저장
    - 질의 처리 : 사용자의 질문을 이해하고, 관련 문서를 정확하게 검색
      * 사용자의 질문을 벡터화 (OpenAIEmbeddings임베딩 모델 활용)
      * 벡터 데이터베이스에서 유사도가 높은 문서 검색
      * 검색된 문서를 LLM이 이해할 수 있는 형태로 전달
    - 답변 생성 : 검색된 문서를 기반으로 사용자의 질문에 최적화된 답변을 생성
      * LangChain을 활용하여 LLM이 검색된 문서 내용을 바탕으로 답변 생성
      * 프롬프트 엔지니어링을 통해 챗봇이 적절한 답변을 생성할 수 있도록 답변 생성의 프레임 작성

**Step 4: RAG 서비스 파이프라인 설계**

1. **데이터 최적화**

**1) 필요 데이터셋 Load하기**  
정확한 공시 정보를 제공하기 위해서는 먼저 DART에서 공개된 PDF, HTML 등 다양한 형식의 공시자료를 안정적으로 확보해야 한다. 취업 준비생 관점에서 기업별 핵심 내용을 얻으려면, 회사개요·재무제표·주요 사업내역 등으로 분류된 파일을 일괄 로드하고, 파일별 메타데이터(기업명·공시일·문서유형 등)도 함께 수집해 빠른 검색 및 식별이 가능하도록 준비한다.

**2) 데이터 전처리: 노이즈 제거**  
수집된 원시 데이터에는 불필요한 각주, 공백, 중복된 표나 이미지 등이 포함될 수 있으므로, 이를 제거해 텍스트 품질을 높이는 작업이 필요하다. 예컨대 문서의 하단에 반복적으로 삽입된 저작권 문구나 광고 문장은 오탐을 유발할 수 있으므로 배제하고, 표나 수치는 취업 준비생이 활용하기 쉽도록 간단한 형태로 변환·정리해야 한다.

**3) 데이터 분할: 청크 크기 및 오버랩 설정**  
DART 문서는 방대하기 때문에, 최대 1000 토큰 단위로 텍스트를 청크화하면 검색·질의응답 단계에서 적절한 맥락을 유지하면서도 처리 속도를 높일 수 있다. 또한 청크 간 100 토큰의 오버랩을 둠으로써 문단이 걸쳐 있는 중요한 맥락이 끊기지 않도록 방지하며, 이 방식으로 사용자 질문에 대한 정확한 답변을 얻을 가능성을 한층 높인다.

**4) 데이터 사이즈에 따른 성능 평가**  
청크와 오버랩 설정이 적절한지 판단하기 위해서는 RAGAS 등의 평가지표를 활용해 답변 정확도와 응답 시간을 비교한다. 특정 기업의 공시자료를 여러 형태로 분할·테스트해보고, 가장 높은 정확도와 합리적인 처리 시간을 보장하는 세팅을 최종 적용한다. 이를 통해 취업 준비생은 방대한 공시자료 속에서도 효율적으로 정보를 얻을 수 있으며, 챗봇 서비스는 지속적인 데이터 최적화로 정확한 답변 품질을 유지할 수 있다.

1. **벡터데이터베이스 선택 및 임베딩 성능 최적화**

**2-1/ 벡터 데이터베이스 선택**

**✅ 최종 프로젝트 벡터데이터베이스 : Pinecone**

DART 공시자료를 기반으로 챗봇을 통해 정확한 기업 정보를 얻으려면, 대규모 임베딩 데이터의 효율적 처리가 필수이며 이에 적합한 벡터 데이터베이스를 선정하는 것이 중요하다. 프로젝트 초기에는 로컬 환경에서 활용 가능한 ChromaDB나 FAISS 등 오픈소스 벡터 DB를 실험하였으나, 궁극적으로 확장성과 성능이 우수한 클라우드 기반 Pinecone이 최종 선택되었다. 로컬 방식은 네트워크 의존도가 낮고 데이터 제어권이 높지만, 데이터가 많아질수록 레이턴시와 관리 부담이 커질 수 있다. 반면 Pinecone은 클라우드 환경에서 손쉽게 확장 가능하고 빠른 검색 속도를 제공하며, 프리 티어 내에서 다양한 테스트를 진행해도 비용 부담을 최소화할 수 있다. 이러한 선택을 통해 챗봇은 방대한 공시자료를 임베딩하고, 시맨틱·키워드 검색 등 다양한 알고리즘으로 취업준비생의 질문에 더욱 정확하고 신속한 답변을 제공할 수 있다.

**2-2/ 임베딩 성능 최적화**

최적화하기 위해 Upstage Solar Embedding 모델을 선택해 실행 효율과 정확도를 동시에 높일 수 있다. 사전에 Hugging Face의 작은 모델로 빠르게 실험해본 뒤, 대규모 데이터에도 적용 가능한 Upstage Solar Embedding으로 확장함으로써 공시자료 전반에서 유사도 검색이나 의미 기반 검색 품질을 극대화한다. 또한 Pinecone 벡터 데이터베이스와 결합해 안정적인 검색 속도를 유지하면서, 방대한 기업 정보를 다루는 취업준비생의 질문에도 정확하게 대응할 수 있다.

1. **검색 기능 구현하기**

검색 방식을 결합한 Ensemble Retriever를 적용해 검색 품질을 높일 수 있다. Sparse Retriever가 제공하는 키워드 기반 검색과 Dense Retriever의 문맥적 의미 파악 능력을 0.3 대 0.7 비율로 가중치 합산하면, 빈도 높은 핵심 용어나 추상적 표현 모두 효과적으로 포착된다. 이를 통해 대규모 공시 문서 속에서도 문서 맥락과 세부 키워드를 놓치지 않고 챗봇 답변의 정확성을 높이며, 취업준비생은 기업 분석 과정에서 원하는 정보를 더욱 빠르고 정교하게 얻을 수 있다.

1. **LLM 답변 생성 및 평가**

**4-1/ 프롬프트 구조와 작성 의도**

이번 프롬프트는 크게 세 개의 요소로 구성되며, 각각 System 메시지, Context 삽입 그리고 유저 메시지/모범 응답 예시로 구분된다. 첫째, System 메시지에서는 모델이 수행해야 할 역할과 태스크 범위를 명확히 설정한다. 이를 통해 챗봇이 “삼성전자 다트(DART) 자료를 바탕으로 취업 관련 정보를 제공하는 전문 챗봇”이라는 목표에만 집중하도록 유도한다.둘째, Context 삽입 부분에서는 {context} 플레이스홀더를 사용해 검색된 공시자료 내용을 동적으로 전달한다. 이로써 모델은 문맥을 벗어난 추측 대신, 실제로 검색된 텍스트를 활용해 답변을 생성하게 된다. 셋째, 유저 메시지 예시와 모범 응답 예시를 함께 제시함으로써 모델이 기대되는 응답 스타일과 내용 수준을 보다 구체적으로 학습한다. 예컨대 “자료에서 언급되지 않은 사항은 추측하지 않고 명확히 설명한다”라는 지시문이나 “2023년 3분기 재무 정보”와 같은 실제 기업 분석 사례를 제공하여, 답변 형식과 정확도를 동시에 높이도록 설계했다.

**4-2/ LLM Chain 고도화 전략**

이 프롬프트를 기반으로 LangChain 혹은 유사한 프레임워크를 활용해 LLM Chain을 구성할 때, 체인 내에서 Prompt Template을 단계별로 세분화하고 각각에 의도된 기능을 분명히 한다. 예컨대 첫 번째 체인에서 사용자의 질문을 이해하고, 두 번째 체인에서 DART 공시자료를 검색해 {context} 자리에 삽입한 뒤, 최종 체인에서 Solar Pro Chat Model을 호출해 최적화된 답변을 산출하도록 설계할 수 있다. 이러한 단계적 접근은 모델이 이미 검색된 자료를 바탕으로만 응답하게 만듦으로써, 비문맥적 정보 생성을 억제하고 취업준비생 관점에서 불필요한 정보를 걸러내는 데 도움을 준다.

**4-3/ 정확도 평가와 개선 방안**

챗봇이 생성한 답변이 공시자료 맥락과 일치하는지 확인하기 위해, Upstage Groundness Check API 같은 평가 모듈을 적용하여 답변과 실제 자료 간의 정합도를 측정한다. 만약 답변이 자료의 내용을 제대로 반영하지 못하거나, 허위 정보를 포함하는 경우에는 동일한 Prompt Template과 검색된 Context를 활용해 재생성을 지시한다. 이 과정을 자동화해 반복 수행하면, 매번 답변 품질을 미세 조정할 수 있고, 오답률이 서서히 낮아지면서 전반적인 정확도가 향상된다. 또한 체계적인 로그 분석과 피드백 루프를 운영해, 유저 메시지·답변·Groundness 체크 결과가 축적됨에 따라 모델이 점차 “DART 공시자료 기반 정확한 응답”이라는 서비스 목표에 수렴하게 된다.

1. **RAG 파이프라인 점검 및 최종 프로젝트 보고서 작성**

|  |  |
| --- | --- |
| **RAG 파이프라인 구축 프로세스 보고서** | |
| **RAG  파이프라인 설계** | **데이터 최적화:**   * Chunk Size: 1000 * Overlap: 100   **벡터 데이터베이스 구축 및 임베딩:**   * 벡터 DB : Pinecone * 임베딩 모델 : Upstage Solar Embedding   **Retriever 및 Reranker 구현:**   * Ensemble Retriever : Parse 0.7 Dense 0.3 * 하이퍼파라미터 튜닝 : * 반환할 문서 수(k) = 3   **LLM 프롬프트 설계 및 답변 생성, 평가  :**  **1/ Task 정의:  QA / 취업 챗봇**  **2/ 프롬프트 설계 :**  **3/ 답변 생성 모델** : Upstage Solar Pro  **4/ 답변 사후 평가 :** gpt-4o-mini 으로 Hallucination 평가 |
| **RAG  파이프라인 평가 및 결과** | **평가방법**  **정량 평가 : RAGAS 평가 지표**   * **context\_precision:** 검색한 문서 중에서 진짜로 관련된 문서가 차지하는 비율 * **context\_recall:** 실제로 관련된 문서 중에서 얼마나 많이 검색에 성공했는지 * **faithfulness:** 생성된 답변이 가지고 있는 지식으로 얼마나 뒷받침 되는 지에 대한 비율 * **answer\_relevancy:** 생성된 답변이 주어진 질문과 얼마나 관련성이 있는 지   **정성 평가**   * **샘플링 방식 -** 무작위로 10개의 질문을 선택하여 챗봇의 답변을 평가합니다. - 질문은 제품 매뉴얼의 다양한 섹션에서 추출된 내용을 기반으로 구성합니다. * **평가 항목** - **정확성**: 생성된 답변이 제품 매뉴얼의 내용과 얼마나 일치하는가? - **관련성**: 답변이 검색된 문서/데이터와 관련이 있는가?  - **명확성**: 답변이 쉽게 이해되고 논리적으로 명확한가? * **평가 절차** - 각 질문에 대해 생성된 답변을 매뉴얼의 실제 내용을 기준으로 비교 검토합니다. - 관련성이 낮거나 잘못된 답변은 피드백을 기록하여 개선 방안을 도출합니다.   **평가 결과**   * **정량평가 결과**   **Context\_precision: 0.6**  **Context\_recall: 0.55**  **Faithfulness: 0.75**  **Answer\_relevancy: 0.72**   * **정성평가 결과**   **정확성: 10개의 질문 중 2개는 문맥 이해 부족으로 부정확한 답변 생성.**  **관련성: 10개 중 3개는 답변이 검색된 문서와 관련이 없음.**  **명확성: 모든 답변이 문법적으로 정확하고, 이해하기 쉬운 표현으로 작성됨.** |
| **결론 및  향후 발전 방향** | **결론**  벡터 DB를 구축하고 이를 바탕으로 LLM을 통해 답변을 생성하는 RAG 파이프라인을 설계하여 기업 재무제표, 채용공고, 면접 질문 데이터를 기반으로 취준생의 질문에 대한 신속하고 정확한 답변을 제공하였습니다. 이를 바탕으로 취준생의 기업 분석 시간을 단축하고 면접 준비의 효율성을 증대시키는 서비스를 제공할 수 있습니다.  **향후 발전 방향**  데이터 실시간 업데이트: 기업 공시자료와 채용공고의 실시간 업데이트를 통해 최신 정보 제공  맞춤형 면접 시뮬레이션: 사용자의 지원 직무와 기업에 맞는 면접 질문을 생성하고 답변을 평가하는 기능 추가  데이터 추가 수집: 기업 뉴스, 산업 분석 보고서, 현직자 인터뷰 등의 데이터를 추가하여 더욱 포괄적인 취업 준비 지원 가능 |

**Step 5. 서비스 기능 요건 정의**

**1. 사용자 시나리오 정의**

|  |  |
| --- | --- |
| **상황** | 사용자가 처음 서비스(취업 챗봇)에 접속하여, 지원하려는 기업의 정보를 신속하게 파악하고 면접 질문에 대비해야 하는 상황 |
| **주어진 것** | - RAG 파이프라인 기반의 취업 챗봇  - 기업 재무제표, 채용공고, 면접 질문, 기업 뉴스 등으로 구성된 벡터 DB  - OpenAI Embeddings를 활용한 임베딩  - gpt-4o-mini 모델로 답변 생성 및 평가 |
| **예상되는 어려움** | - 사용자가 원하는 기업 정보를 구체적으로 제시하지 않으면 챗봇의 검색 정확도가 떨어질 수 있음  - 사용자가 여러 차례 질문을 반복하면서 비슷한 답변을 받게 되어 혼선이 생길 수 있음  - 챗봇이 제공하는 가장 최근 정보가 업데이트되지 않았다면, 실제 채용공고와의 불일치 발생 가능성 |
| **핵심 기능 (Task Flow)** | 1. 사용자 질문 입력: “OO기업 재무 상태가 어떤가요?”  2. RAG 파이프라인 검색: Pinecone 벡터 DB에서 관련 문서(재무제표 데이터 등)를 검색  3. 답변 생성: gpt-4o-mini 모델이 임베딩된 데이터 기반으로 정확한 응답 제공  4. 추가 질문: “면접 질문 예시 좀 알려주세요.” 등 후속 질문 처리  5. 답변 사후 평가: gpt-4o-mini를 통해 답변의 적합성과 정확성 점검 |
| **달성한 결과** | - 사용자가 기업 재무제표 요약과 채용공고 핵심 요건 등을 빠르게 확인  - 면접 질문과 대략적인 답변 예시까지 챗봇을 통해 참고 가능  - 취업 준비 시간이 단축되고, 면접 대비 효율이 높아짐 |
| **참고** | - RAGAS 지표(context\_precision, faithfulness 등)를 활용해 검색 및 답변 품질을 주기적으로 모니터링  - 데이터 업데이트(예: 기업 공시자료 변경) 시, 벡터 DB 재구축 및 모델 재학습 필요 |

**2. 사용자 테스크 정의**

**2-1/ Must Have**

* 1. 기업 정보 검색: 벡터 DB를 통해 지원 기업의 재무제표, 채용공고 등 핵심 문서를 검색.
  2. 질의응답(QA) 시스템: gpt-4o-mini 모델 활용으로 빠르고 정확한 답변 제공.
  3. 기본 UI/UX: 사용자가 질문을 쉽게 입력하고, 답변을 간편하게 확인할 수 있는 화면 구성.

**2-2/ Should Have**

* 1. 면접 예상 질문 자동 생성: 기업 및 직무별로 자주 묻는 면접 질문을 리스트업하고, 사용자에게 제시.
  2. 답변 템플릿 제공: 사용자가 면접 답변을 작성할 때 참고할 수 있는 예시 문구 생성.
  3. 사용자 프로필 연동: 지원 직무나 경력을 토대로 개인화된 질문·정보를 추천.

**2-3/ Could Have**

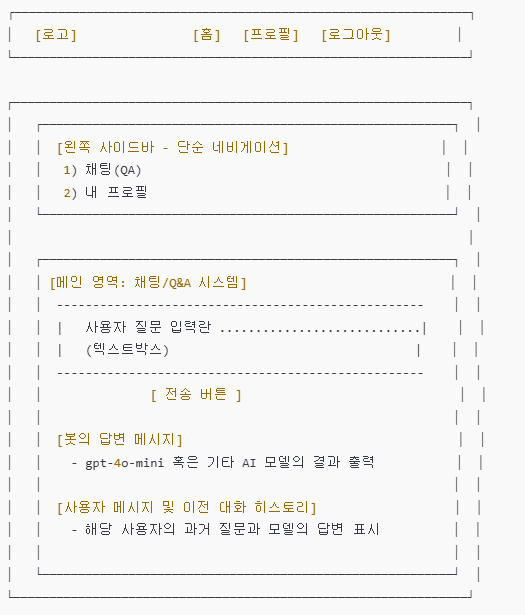
* 1. 실시간 협업 기능: 친구나 멘토와 함께 면접 질문을 공유하고 피드백을 주고받을 수 있는 공동 작업 환경.
  2. 멀티모달 데이터 분석: 텍스트뿐 아니라 영상·음성 데이터까지 분석하여 면접 태도 및 발화 습관 등을 피드백.
  3. 게이미피케이션 요소 도입: 게임 형태의 면접 퀴즈나 랭킹 시스템을 통해 동기부여 강화.

**3. 개발 태스크 구체화**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **기능** | **Title** | **설명** | **개발 PIC** | **Status** |
| **검색기능** | 기업 정보 검색 | - 벡터 DB를 통해 지원 기업의 재무제표, 채용공고 등 핵심 문서를 검색  - 자료 수집 지원 | 최국진 | Done |
| **검색기능** | 질의응답(QA) 시스템 | - gpt-4o-mini 모델 활용  - 사용자 질문에 맞춘 실시간 빠르고 정확한 답변 제공 | 강성운 | Done |
| **Endpoint Deployment** | 기본 UI/UX | - 사용자 친화적 인터페이스 제공  - 질문 입력 및 답변 확인을 간편화 | 김경환 | Done |
| **검색기능** | 면접 예상 질문 자동 생성 | - 기업 및 직무별 자주 묻는 면접 질문 자동 리스트업  - 면접 대비 질문 추천 기능 | 김진실 |  |
| **Endpoint Deployment** | 답변 템플릿 제공 | - 면접 답변 시 참고 가능한 예시 문구 및 가이드 생성  - 답변 작성 편의성 증대 | 전제후 |  |
| **Endpoint Deployment** | 사용자 프로필 연동 | - 지원 직무·경력 기반으로 개인화된 질문 및 정보 추천  - 맞춤형 면접 대비 기능 강화 | 김진실 |  |
| **Endpoint Deployment** | 실시간 협업 기능 | - 친구나 멘토와 면접 질문·답변을 공유  - 공동 작업 환경에서 피드백 주고받기 | 김경환 |  |
| **검색기능** | 멀티모달 데이터 분석 | - 텍스트뿐 아니라 영상·음성 데이터까지 분석  - 면접 태도 및 발화 습관 등에 대한 피드백 | 강성운 |  |
| **Endpoint Deployment** | 게이미피케이션 요소 도입 | - 게임형 면접 퀴즈, 랭킹 시스템 등 동기부여 요소  - 재미를 통한 학습 효과 극대화 | 최국진 |  |
| **검색기능** | 기업 정보 검색 | - 벡터 DB를 통해 지원 기업의 재무제표, 채용공고 등 핵심 문서를 검색  - 자료 수집 지원 | 최국진 |  |

**4. UI/UX 설계**

* **UI 요소**:
* **질문 입력창**: 사용자가 텍스트를 입력할 수 있는 창.
* **응답 창**: 생성된 답변을 출력할 수 있는 창.
* **UX 고려사항**:
* **응답 시간**: 검색 및 생성 과정의 속도를 최적화.
* **직관성**: 인터페이스는 사용자가 질문을 쉽게 입력하고, 답변을 명확히 확인할 수 있도록 설계.

**5. 예시 와이어 프레임 초안**

**Step 6. 서비스 배포**

**1. 서비스 준비**

* 데이터 흐름 검증: 질문 → 검색 → 생성 → 응답으로 이어지는 RAG(Retrieve & Generate) 파이프라인이 올바르게 작동하는지 확인합니다.
* 에러 핸들링: 비정상 입력이나 API 호출 실패 시 사용자에게 적절한 오류 메시지가 전달되는지 점검합니다.

**2. 서비스 배포**

* **1/ 개발 단계**:
  + 로컬 환경에서 모든 기능 테스트 완료.
  + 데이터 흐름(질문 → 검색 → 생성 → 응답)이 올바르게 작동하는지 검증.
* **2/ 배포 준비**:
  + **배포 환경 설정**:
    - **Frontend**:
      * Vercel에 Chat UI를 배포합니다.
      * 배포된 UI URL을 확인하고 접근성을 테스트합니다.
    - **Backend**:
      * Fly.io를 사용해 FastAPI 서버를 배포합니다.
      * 벡터 데이터베이스와 연결 상태를 최종 검증합니다.
      * LLM API가 잘 호출되는 지 최종 검증합니다.
    - API 요청/응답 지연 시간을 점검한다.

**3. 서비스 배포 종료**

더 이상 서비스를 유지할 필요가 없거나, 프로젝트를 종료하기로 결정한 경우 아래 단계에 따라 배포를 내린다. 배포를 종료하지 않을 시 비용이 더 청구될 수 있으므로 반드시 최종 프로젝트 마무리 후 배포된 프로젝트를 내릴 수 있도록 한다.

* **Frontend 종료**:
  1. Vercel 배포를 중단하고 해당 프로젝트를 비활성화한다.
  2. 필요시 Vercel 프로젝트를 삭제한다.
* **Backend 종료**:
  1. Fly.io 대시보드에서 FastAPI 서버를 중지하거나 삭제한다.
  2. 관련된 환경 변수 및 API 키를 폐기한다.
* **데이터베이스 종료**:
  1. 벡터 데이터베이스(Pinecone, Chroma 등)를 비활성화하거나 삭제한다.
  2. 저장된 데이터를 백업 후 삭제한다.

**4. 향후 확장 가능성 :  모니터링 및 유지보수**

* **1/모니터링**
  + **API 호출 로그**
    - FastAPI와  Vercel에서 호출된 요청/응답 로그를 수집하여 사용량과 에러 발생 여부를 모니터링한다.
  + **검색 정확도**:
    - 사용자 질문에 대해 반환된 답변의 품질을 주기적으로 확인한다.
    - 필요시 RAG 파이프라인에서 검색 알고리즘을 개선한다.
  + **응답 시간**
    - 각 단계(검색, 생성 등)에서의 처리 속도를 측정하고 병목 구간을 최적화한다.
* **2/ 유지보수**
  + **지속적 배포관리**  
    버그 수정이나 기능 추가 시 CI/CD 파이프라인을 구축하여 자동 배포를 설정한다.
  + **사용자 피드백 수집**  
    설문이나 피드백 폼을 통해 서비스 개선 요소를 파악하고 신속히 조치한다.