

SK네트웍스 Family AI과정 7기

모델링 및 평가 테스트 계획 및 결과 보고서

□ 개요

- 산출물 단계 : 모델링 및 평가
- 평가 산출물 : 테스트 계획 및 결과 보고서
- 제출 일자 : 2025-04-18
- 깃허브 경로 : <https://github.com/SKNETWORKS-FAMILY-AICAMP/SKN07-FINAL-5Team>
- 작성 팀원 : 김나예

개요

- **AP@K** 평가지표를 이용한 채용공고 추천 시스템 성능 평가 모델
 - 개요
 - 단일 유저의 지원 자료를 기반으로 RAG 검색 엔진과 Hybrid 검색 엔진을 통해 각각 공고를 추천하고, K개의 결과에 대해 Precision Average를 구해 두 검색 엔진의 성능을 비교하는 모델
 - 사용 데이터
 - **Ground Truth**
 - 5가지 평가 기준별 정합성 점수로 도출된 **종합 정합성 점수가 0.83 이상**인 채용공고(최고 점수: 0.85)
 - 임베딩 모델 : OpenAI Embedding Model (“text-embedding-ada-002”)
 - 갯수 : 95건
 - **RAG 추천 공고**
 - ChromaDB에 저장된 데이터를 기반으로 검색한 추천 결과
 - 갯수 : 20건
 - **Hybrid 추천 공고**
 - ChromaDB를 사용하는 RAG 방식에 Elastic Search 엔진을 추가해 검색한 추천 결과
 - 갯수 : 20건

<p>계획</p>	<ul style="list-style-type: none"> ● 기본적인 RAG 와 Hybrid Search 검색 성능 비교 <ul style="list-style-type: none"> a. 지원 자료 : 사용자의 이력서, 포트폴리오, 자기소개서 PDF 파일 b. 수행 과정 <ul style="list-style-type: none"> i. 지원자료 PDF 파일에서 텍스트 파싱 ii. 파싱한 텍스트를 정제하지 않고 검색 시 참고하여 유사도가 높은 공고를 20개 추출하도록 요청 iii. Ground Truth 와 RAG, Hybrid Search 검색 결과를 비교해 AP@K 계산 iv. 두 검색 엔진 성능 비교
<p>수행 과정</p>	<ol style="list-style-type: none"> 1. Ground Truth 생성 (Top-N 공고 리스트) <ul style="list-style-type: none"> a. LLM을 활용해 단일 유저에 대한 지원자료 (이력서/포트폴리오/자소서)의 특성 추출 b. 평가 기준 별 정합성 계산 <ul style="list-style-type: none"> i. 직무 연관성(weight: 0.4) ii. 산업군/도메인 일치도(weight: 0.2) iii. 기술 스킬 매칭도(weight: 0.3) iv. 경력/학력 수준 일치도(weight: 0.05) v. 지역/근무 조건 부합도(weight: 0.05) c. 공고 별 최종 정합성 계산 후 순위 산정 <ul style="list-style-type: none"> i. 평가 기준별 가중치 부여하여 총점 계산 d. 최종 정합성 임계값(0.83) 이상의 공고를 Ground-Truth로 설정 2. 추천 공고 결과 취합 <ul style="list-style-type: none"> a. RAG 추천 결과 b. Hybrid Search 추천 결과 3. AP@K 계산 <ul style="list-style-type: none"> a. 검색 방식별 추천 결과의 Precision Average 계산 후 비교

<p>결과</p>	<ul style="list-style-type: none"> ● 테스트 결과 <ul style="list-style-type: none"> ○ Ground Truth의 채용공고들이 두 검색 엔진의 추천 결과에 전부 포함되지 않음 ○ AP@K는 전부 0.0으로, 두 검색 엔진의 성능 비교가 불가함 ● 결론 <ul style="list-style-type: none"> ○ 지원자료의 raw text를 이용해 검색 할 경우, 두 검색 엔진 모두 성능이 좋지 않음 ○ Ground Truth 생성 방식과 동일하게 LLM을 통해 지원자료의 특성을 먼저 추출한 후 검색을 진행한다면 성능이 향상될 것으로 예상 ● 개선 방향 <ul style="list-style-type: none"> ○ Retriever에 요청하는 query는 적당한 길이로, 유사도를 체크할 대상만 간결하게 포함되도록 작성해 검색하도록 한다. ○ 사용자의 질의에서 문서의 메타데이터에 대한 필터를 추출 하고, 이 필터를 실행하여 관련된 문서를 찾을 수 있는 검색 도구인 Langchain의 'Self Query Retriever'를 이용해 검색 성능을 높인다. <ul style="list-style-type: none"> ■ 여러 개의 카테고리를 메타데이터로 갖는 문서에 대해 조건을 걸어 요청할 수 있으며, 이로 인해 더욱 유사도 높은 검색 결과가 나올 것으로 기대함
-----------	--