

/* elice */

Enterprise

회고: 퀴즈 + 토론

미니퀴즈 3문항 + 조별 토론

15:30 ~ 16:30

오늘 하루 여정 되돌아보기

데일리 로드맵 : 체험→이해→심화→구축→개선→정리 순서로 진행



오전에 학습한 내용을 오후에 직접 만들었습니다. 이제 확인할 차례입니다.

RAG 5단계 - 직접 말해봅시다

"1단계가 뭐였죠?" → 수강생이 대답 → 정답 확인

1 질문을 _____로 변환한다

벡터

Day1

2 벡터DB에서 유사 _____를 검색한다

문서

Day2

3 검색된 문서와 질문을 _____로 조립한다

프롬프트

오늘

4 LLM이 _____을 생성한다

답변

오늘

5 답변과 _____를 함께 반환한다

출처

오늘

QUIZ

미니퀴즈 TIME

오늘 배운 내용에서 3문항 출제

각 문항 30초 생각 → 답안 확인

틀려도 OK, 이해가 목표

Q1. RAG 파이프라인의 순서

다음 RAG 파이프라인의 5단계를 올바른 순서로 나열하세요:

- | A LLM이 답변을 생성한다
- | B 질문을 벡터로 변환한다
- | C 벡터DB에서 유사 문서를 검색한다
- | D 검색된 문서와 질문을 프롬프트로 조립한다
- | E 답변과 출처를 함께 반환한다



30초 — 올바른 순서를 생각해보세요

Q1. 정답 : B → C → D → A → E

정답

1 질문을 벡터로 변환한다

Day1 복습

2 벡터DB에서 유사 문서를 검색한다

Day2 복습

3 검색된 문서와 질문을 프롬프트로 조립한다

오늘

4 LLM이 답변을 생성한다

오늘

5 답변과 출처를 함께 반환한다

오늘

핵심: Day1~2의 벡터 검색 위에 ③④⑤ 세 단계(프롬프트 조립→LLM 생성→출처 반환)를 추가한 것이 RAG

Q2. "모른다"고 답하게 하려면?

RAG 시스템이 문서에 없는 내용을 질문 받았을 때 "모르겠습니다"라고 답하게 하려면, 어디를 수정해야 할까요?

- | A temperature 값을 0으로 설정한다
- | B ChromaDB의 검색 개수(k)를 늘린다
- | C 프롬프트에 "문서에 없으면 모른다고 답하세요" 규칙을 추가한다
- | D 청크 사이즈를 더 작게 나눈다



30초 — 올바른 순서를 생각해보세요

Q2. 정답: C – 프롬프트 규칙 추가

정답

✓ C가 정답인 이유

프롬프트에 "주어진 문서에 답이 없으면
모른다고 답하세요" 규칙 추가시
LLM이 지시를 따라 할루시네이션을 줄입니다.

이것이 커스텀 프롬프트의 핵심 역할입니다.

나머지 보기는?

- A: temperature=0은 일관성만 높임
- B: k를 늘리면 검색량만 증가
- D: 청크 사이즈는 분할 방법

오후2 실습에서 작성한 프롬프트 규칙:

답변 규칙:

1. 문서에 있는 정보만 사용하세요
2. 문서에 없으면 "해당 정보를 찾을 수 없습니다"라고 답하세요 ← 이 규칙!
3. 수치를 인용할 때는 단위를 반드시 포함하세요



내일 Day4에서 할루시네이션 대응 기법 4가지를 체계적으로 배웁니다

Q3. 출처 표시가 중요한 이유

RAG 시스템이 "RF9000 소비전력은 36W입니다"라고 답했습니다.
출처 표시가 없다면 어떤 문제가 생길까요? 이유를 1가지 이상 말해주세요.

- | A 이 숫자가 사양서에서 온 건지, AI가 지어낸 건지 알 수 없다
- | B R&D 보고서에 이 답변을 인용할 수 없다
- | C 잘못된 답변이 나왔을 때 원인을 역추적할 수 없다
- | D 위의 모두 해당된다



30초 — 올바른 순서를 생각해보세요

Q3. 정답: D – 모두 해당

정답



검증 가능성

답변 근거를 원문 PDF에서 직접 확인 가능 → "사양서 p.2에 있습니다"



신뢰도 확보

R&D 보고서에 AI 답변을 인용할 수 있음 → 출처 없는 수치는 보고서에 쓸 수 없음



오류 추적

잘못된 답변의 원인을 역추적 가능 → "어떤 문서의 몇 페이지에서 온 정보인지"

오후 실습에서 만든 `ask_with_sources()` 함수가 이 3가지를 모두 해결합니다
파일명 + 페이지 번호 → 검증·인용·추적 가능

퀴즈 결과 확인



3문항 전부 정답

완벽합니다! RAG를 정확히 이해하고 계십니다.



2문항 정답

핵심은 잡았습니다. 복습 한 번이면 충분합니다.



1문항 이하

괜찮습니다. 내일 복습에서 다시 한번 정리됩니다.

지금부터 조별 토론 시간입니다. 퀴즈에서 헷갈린 부분도 함께 이야기해보세요.

DISCUSSION

조별 토론

- ① 보드 아이디어 확인 2분 → 아침에 적었던 것을 다시 읽어보세요
- ② 조별 토의 3분 → 만들어본 경험을 반영하여 아이디어 구체화
- ③ 발표 준비 3분 → 발표자 정하고 한 장 정리

토론 가이드: 이렇게 정리해보세요

1

대상 문서

예: 제품 사양서, 설비 매뉴얼, 시험 성적서, 품질 보고서, 안전인증 문서, 고객 VOC 보고서...

2

예상 질문 3개

예: "이 부품의 내열 온도는?", "E7 에러 조치법은?", "이 시험 항목의 합격 기준은?"

3

출처가 왜 필요?

예: "품질 보고서 수치는 근거가 있어야 신뢰 가능", "인증 문서 답변은 원본 확인 필수"

④ 심화 (선택)

"이 시스템에서 할루시네이션이 특히 위험한 상황은?" → 내일 Day4 주제와 연결



4주차 최종 프로젝트에서
조별로 현업 문서를 적용

조별 발표

1조

대상 문서:

예상 질문:

출처 필요성:

2조

대상 문서:

예상 질문:

출처 필요성:

3조

대상 문서:

예상 질문:

출처 필요성:

4조

대상 문서:

예상 질문:

출처 필요성:

오늘의 산출물 확인

커리큘럼 목표: RAG 기반 문서 QA 체인 (출처 포함) | 학습목표 3개 달성 ✓

- ✓ 기본 RAG: RetrievalQA 체인으로 문서에 질문 → 답변 받기

오후1

- ✓ 커스텀 프롬프트: 제조업 전문 AI 역할 + 응답 규칙 설정

오후2

- ✓ 출처 표시: 파일명 + 페이지 번호 함께 출력

오후2

- ✓ "모름" 응답: 문서에 없으면 지어내지 않기

오후2

기본 과제 — 전원 완료 대상

위 4가지 기능이 동작하는 Jupyter Notebook 제출

심화 과제 — A그룹 + 희망자

문서 3종 혼합 + 출처별 필터링 + 신뢰도 점수

Day3 핵심 정리: 한 장으로 보기

1

RAG = 검색 + 생성

벡터DB 검색 결과를 LLM에 전달
근거 있는 답변을 생성하는 구조

2

출처 = 신뢰

return_source_documents
+ metadata로
파일명·페이지를 함께 반환

3

프롬프트 = 품질

커스텀 프롬프트 규칙이
할루시네이션을 방지



Day1 임베딩



Day2 벡터DB



Day3 RAG



Day4 할루시네이션 대응

Day1~3에서 만든 RAG를 내일 더 강하게 만들니다

THANK YOU

수고하셨습니다!

- ① Jupyter Notebook 제출 (산출물 확인용)
- ② 심화 과제(선택) : 문서 3종 혼합 + 출처 필터링
- ③ 내일 예고 : 할루시네이션 대응 4기법