MODULE 12 · 실습

허깅페이스에서 모델 읽는 법 — 도서관에서 원하는 책 고르기

지금까지 우리는 모듈 3에서 양자화를, 모듈 4에서 서빙 엔진을 배웠습니다. 그런데 그 모델 파일들은 애초에 어디서 오는 걸까요? 대부분 **허깅페이스(HuggingFace)**라는 한 곳에서 옵니다. 이번 모듈은 이론이 아니라 실습입니다 — 화면을 실제로 열어가며, 모델 페이지에서 무엇을 어떤 순서로 읽어야 하는지 손에 익힙니다.

01 허깅페이스가 뭔가 — AI 모델의 앱스토어이자 도서관

HuggingFace는 전 세계 연구자·기업이 만든 AI 모델을 올리고 누구나 받아갈 수 있는 사이트입니다. 앱스토어처럼 "검색 → 상세 페이지 → 다운로드"의 흐름이고, 도서관처럼 같은 주제의 책(모델)이 버전별·번역본별로 여러 권 꽂혀 있습니다.

> 💡 **비유**
>
> 계정 없이도 서가를 둘러보고 책을 펼쳐볼 수 있는 도서관과 같습니다 — **huggingface.co 검색창에 모델 이름을 치는 것**부터 시작하면 됩니다. 로그인은 직접 파일을 내려받거나 올릴 때만 필요하고, 페이지를 읽고 정보를 확인하는 데는 필요 없습니다.

- ✔ 검색창에 `gemma`, `qwen`처럼 모델 계열 이름만 쳐도 관련 저장소가 쏟아집니다 — 처음엔 결과가 너무 많아 당황스러운 게 정상입니다.
- ✔ 각 결과 하나하나가 **저장소(repository)**입니다. 저장소 하나 = 책 한 권이라고 생각하면, 이번 모듈은 "표지와 목차 읽는 법"입니다.

02 모델 페이지 첫 화면 해부

저장소 하나를 열면 맨 위에 이름이 있습니다. 이 이름 자체가 이미 많은 정보를 담고 있습니다.

구성 요소	예시	뜻
조직/제작자	`google`	누가 만들었나 — 공식 조직이면 신뢰도가 높습니다
모델명	`gemma`	모델 계열 이름
크기	`-2b` , `-9b`	**파라미터** 개수(모듈 2) — 20억, 90억 개
용도 접미사	`-it`	instruct(지시 따르기)용으로 추가 학습됨
포맷/배포 접미사	`-GGUF`	로컬 실행용 포맷으로 변환된 저장소(모듈 3)

예를 들어 `google/gemma-2-9b-it`는 "구글이 만든 gemma 2세대, 90억 파라미터, 지시를 따르도록 추가 학습된 버전"이라는 뜻입니다. 여기서 **-it가 붙지 않은 base 버전**도 따로 있는데, base는 "다음 글자를 잇는 것만 잘하는 원재료" 상태라 질문에 대답하도록 다듬어지지 않았습니다 — 우리가 챗봇처럼 쓰려면 거의 항상 **it(또는 instruct, chat) 버전**을 받아야 합니다.

> 💡 **비유**
>
> base는 "글은 잘 쓰지만 대화 예절은 안 배운 신입"이고, it는 "같은 신입인데 응대 매뉴얼까지 익힌" 버전입니다. 사내 업무에 쓸 모델은 거의 항상 후자입니다.

이름 아래에는 세 가지 신호가 나란히 뜹니다.

- **다운로드 수 / 좋아요 수** — 많이 쓰인다는 뜻이지 품질 1위라는 뜻은 아닙니다. 참고용 신호일 뿐입니다.
- **라이선스 배지** — 가장 먼저 눈에 담아야 할 정보(섹션 3에서 자세히).
- **마지막 업데이트 날짜** — 오래 방치된 저장소인지, 최근에 관리되는지 감을 줍니다. 너무 오래됐다면 더 최신 저장소가 따로 있는지 검색해보는 게 좋습니다.

03 모델카드(README) 읽는 순서 — 위에서부터 다 읽지 않는다

페이지 본문(모델카드, 사실상 README)은 보통 길고, 처음부터 끝까지 읽으려면 시간이 오래 걸립니다. 실무에서는 **필요한 순서대로 골라 읽습니다.**

- ① **라이선스** — 사내 업무에 써도 되는지부터 확인합니다. Apache-2.0, MIT처럼 상업적 사용을 막지 않는 라이선스인지, 아니면 "NC(NonCommercial, 비상업적 용도만)"나 "연구용(research only)"처럼 사내 업무 활용을 제한하는 라이선스인지 — 이 한 줄이 나머지를 다 읽을지 말지를 결정합니다.
- ② **파라미터 수와 아키텍처** — 몇 B(십억)짜리인지, 그리고 **dense**(전체 파라미터를 매번 다 쓰는 구조)인지 **MoE**(Mixture of Experts — 여러 "전문가" 중 일부만 골라 쓰는 구조로, 표기된 전체 크기보다 실제 연산량이 작을 수 있음)인지 확인합니다.
- ③ **컨텍스트 길이** — 한 번에 몇 토큰까지 읽고 기억할 수 있는지(**컨텍스트 윈도우**). 긴 문서를 다룰 업무라면 이 숫자가 실질적인 상한선입니다.
- ④ **지원 언어** — 한국어가 명시돼 있는지 확인합니다. 언급이 없다고 아예 못 하는 건 아니지만, 공식 지원 언어에 없으면 품질 편차를 각오해야 합니다.
- ⑤ **벤치마크 표는 "참고만"** — 모델카드에 나오는 점수표는 대부분 제작사가 고른 시험지로 자기 채점한 결과입니다. 왜 곧이곧대로 믿기 어려운지: 시험 문제(벤치마크)를 제작사가 고르고, 채점도 제작사 쪽 도구로 하는 경우가 많기 때문입니다. 우리에게 진짜 의미 있는 숫자는 **우리 골든셋으로 직접 실측한 점수**뿐입니다(모듈 6).

> 💡 **비유**
>
> 모델카드는 이력서와 비슷합니다. 자기소개(벤치마크)는 지원자가 스스로 잘 쓴 부분이니 참고만 하고, 자격 요건(라이선스), 경력 연차(파라미터), 업무 범위(컨텍스트·언어)처럼 객관적으로 확인 가능한 항목부터 봅니다. 실제 실력은 결국 우리가 직접 면접(골든셋 실측)을 봐야 압니다.

04 Files 탭 실습 — 내 장비에 들어갈까 어림하기

GGUF 저장소를 열면 **Files and versions** 탭에 여러 파일이 나열됩니다. 파일명에 `Q4_K_M`, `Q6_K`, `Q8_0` 같은 양자화 등급이 붙어 있고(모듈 3), 옆에 파일 크기가 적혀 있습니다. 이 크기를 읽고 "내 장비에 들어갈까"를 어림하는 것이 이번 실습의 핵심입니다.

참조 장비	메모리(대략)
RTX 3090 / 4090	24GB VRAM

RTX 5090	32GB VRAM
맥 미니 (통합 메모리)	16~32GB
맥 스튜디오	64GB 이상

어림 산식은 모듈 3에서 배운 것과 같습니다.

> 💡 **비유**
>
> **장비 메모리 − 버퍼(약 1.5GB) ≥ 파일 크기 + 작업 공간(대화가 길어질수록 커지는 KV 캐시)**
> 버퍼를 먼저 빼고 나머지 자리에 파일 크기와 작업 공간이 들어가는지 확인하는 것 — 딱 이 정도만 계산하면 충분합니다. 예컨대 24GB 카드에서 Q4_K_M 파일이 14GB라면, 버퍼를 빼고도 약 8.5GB가 남아 대화가 길어져도 여유가 있다고 어림할 수 있습니다.

05 따라 하기 — 10분 실습

> 🔗 **우리 연구회에선**
>
> **직접 해보세요 (10분)**
> 허깅페이스 검색창에서 `google/gemma` 계열의 아무 **GGUF** 저장소를 하나 열어, 다음 네 가지를 적어보세요.
> ① 라이선스는 무엇인가요 (사내 사용 가능한 종류인가요)?
> ② 파라미터 수는 몇 B인가요?
> ③ Files 탭에서 `Q4_K_M` 파일의 크기는 몇 GB인가요?
> ④ 위 산식으로 볼 때, 자기 노트북(또는 위 표의 참조 장비 중 하나)에 들어갈까요?

06 함정 안내

- ✔ **같은 모델의 저장소가 여러 개인 이유** — 원본을 만든 조직의 저장소 하나, 그리고 그걸 양자화해 GGUF로 재배포한 제3자 저장소가 여러 개 있는 게 보통입니다. 원본과 재배포본은 별개의 저장소입니다.

- **배포자 신뢰도** — 공식 조직(google, meta 등)이나 널리 검증된 배포자(꾸준히 GGUF를 올려온 계정)의 저장소를 우선하세요. 무명 계정의 저장소는 변환 과정에서 오류가 섞였을 가능성을 배제할 수 없습니다.
- **파일명의 변형 접두어 주의** — 같은 `Q4_K_M`이라도 `UD` 같은 접두어가 붙은 변형 양자화 파일이 섞여 있을 수 있습니다. 낯선 접두어를 보면 "표준 양자화와 다른 변형"이라는 신호로 받아들이고, 배포자의 설명을 한 번 더 확인하세요.
- **"trending"은 품질 순위가 아닙니다** — 최근 다운로드·좋아요가 몰려 화면 위쪽에 뜨는 것일 뿐, 우리 업무에 맞는지와는 별개입니다. 결국 판단 기준은 라이선스·아키텍처·컨텍스트·언어, 그리고 최종적으로는 우리 골든셋 실측입니다.

🔗 **우리 연구회에선**

출처: HuggingFace 공식 문서. 모델 페이지의 화면 구성(탭 배치, 배지 위치 등)은 사이트 업데이트로 바뀔 수 있습니다 — 이번 모듈의 스크린 요소 설명은 큰 틀을 기준으로 합니다.

↗ **심화 학습**

더 깊이: HuggingFace 공식 문서의 "Model Cards" 가이드와 "Hub" 문서. · 파라미터 산정·양자화 표기는 모듈 3, 벤치마크 신뢰도는 모듈 6을 함께 보면 이해가 빠릅니다.

07 퀴즈 — 가상 모델카드 읽기

CHECK

이해했는지 확인해 봅시다

틀려도 괜찮습니다 — 오답을 고르면 왜 아닌지 설명이 나오고, 정답을 찾을 때까지 다시 고를 수 있습니다. 점수는 첫 시도 기준입니다.

문제 1 / 5

Q1. 다음 모델카드 요약을 보고 답하세요: 『acme/koral-9b-it-GGUF · 라이선스 CC-BY-NC-4.0 · 9B dense · 컨텍스트 32k · Q4_K_M 5.4GB』 — 이 모델을 사내 업무 자동화에 그대로 써도 될까요?

① 그렇다 — 오픈소스라고 적혀 있으니 문제없다

② 아니다 — NC(NonCommercial)는 비상업적 용도만 허용하므로 사내 업무(상업적 활용) 사용은 라이선스 위반 소지가 있다

③ 그렇다 — 다운로드 수가 많으면 상업적 사용도 허용된 것이다

④ 아니다 — 9B 모델은 항상 상업적 사용이 금지된다

← 이전
모듈 11 · 에이전트를 믿을 수 있는 선까지

다음 →
자가진단