랩 위드의 AI 기반 솔루션 >

v21까지 자체 sLLM Mi:dm을 21번 고쳤습니다

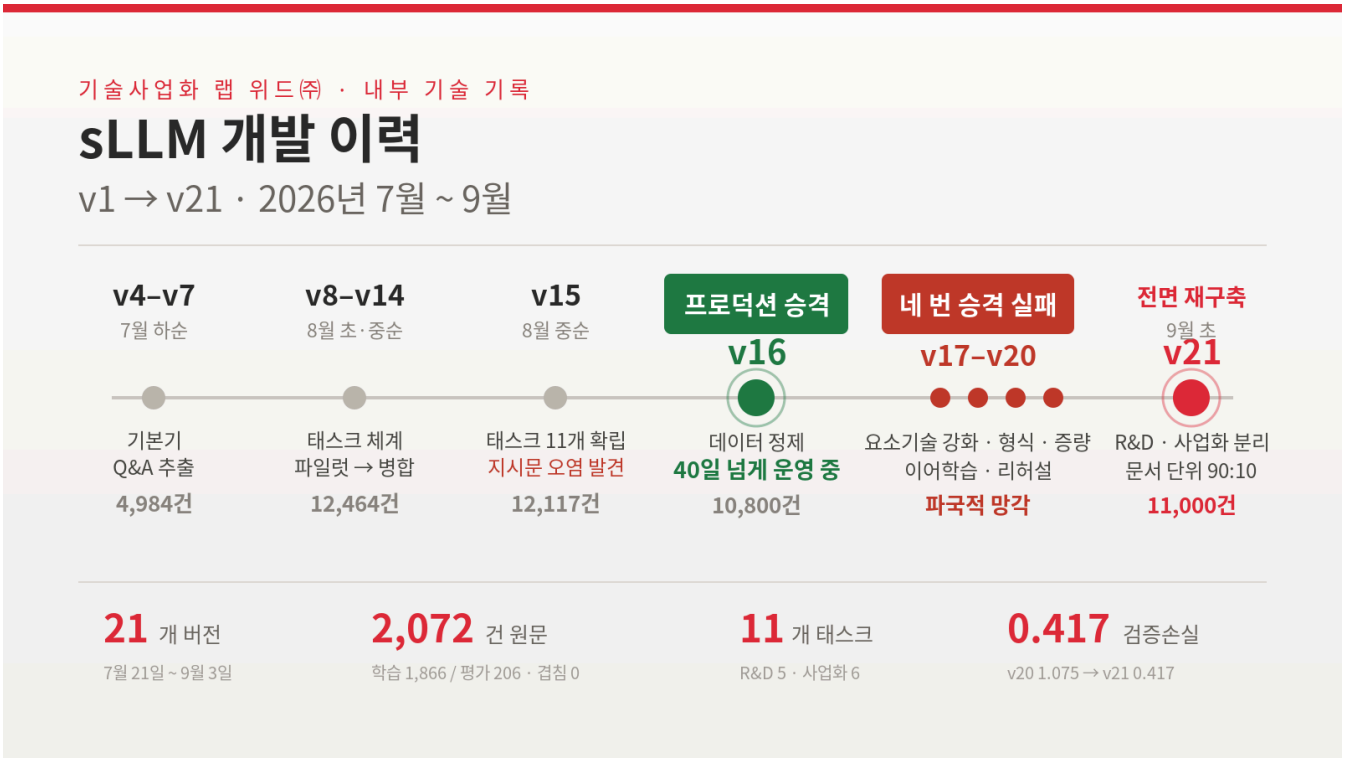
기술사업화 랩 위드
2026.09.03. 17:37 조회 0

댓글 0 URL 복사

구분	내용	소유
기반 모델	KT Mi:dm 2.0 (11.5B, Q4_K_M 양자화)	KT_ 공개 모델
학습 방식	LoRA 어댑터	-
학습 데이터	사업계획서·보고서 2,072건에서 생성한 11,000건	**랩 위드 자산**
결과물	rnd-midm 어댑터(v1~v21), 채점표, 생성 파이프라인	**랩 위드 자산**

저희는 R&D 사전기획과 사업화 전략을 돕는 AI를 직접 만들어 씁니다.

두 달 동안 스물한 개 버전을 만들었고, **지금 쓰는 것은 8월 19일에 올린 v16입니다.**
그 뒤로 만든 네 개가 모두 승격에 실패했기 때문입니다.
무엇을 잘못했는지 적습니다.
기술 자랑이 아니라, 같은 길을 가실 분들이 같은 벽에 부딪히지 않으셨으면 해서입니다.



1. 두 달을 한 장으로

시기	버전	한 일	결과
7월 하순	v4~v7	원문에서 Q&A를 뽑아 기본기를 심음	답변이 원론적
8월 초·중순	v8~v14	태스크 분류 체계를 세움	정의가 뒤섞임
8월 중순	v15	태스크 11개로 정리	지시문이 깨져 있었음
8월 19일	**v16**	데이터 정제	**프로덕션 승격**
8월 하순	v17~v19	약점 보강 시도	**세 번 실패**
8월 말	v20	망각 방어 시도	**실패**
9월 초	**v21**	전면 재구축, 모델 분리	학습 완료·측정 중

기반 모델은 KT의 **Mi:dm 2.0**입니다.
여기에 저희가 쌓아 온 **사업계획서·보고서 2,000여 건**을 학습시켰습니다. 그래픽카드 두 장(RTX 5060 Ti)으로 밤마다 돌립니다.

2. 시작 — 답은 그럴듯한데 쓸 수가 없었다

처음 만든 v4~v7은 원문 PDF에서 텍스트를 뽑아 질문과 답을 자동으로 만드는 방식이었습니다.
건수는 3,797건에서 4,984건으로 늘렸습니다.
그런데 답변이 이랬습니다.
“사업모델을 도출할 때는 먼저 기술의 활용 가능성을 분석하고, 시장에 요구되는 제품 또는 서비스를 정의하는 것이 중요합니다.”
틀린 말은 아닙니다.
다만 **수치도, 기관명도, 절차도 없습니다.**

일반 AI와 구별되지 않습니다. 저희가 원한 것은 “중요합니다”가 아니라 “촉매 활성도 10,000 A/m² 이상”이었습니다.
첫 번째 교훈은 여기서 나왔습니다. 양이 늘어도 깊이는 늘지 않습니다.

3. 이름을 붙이다 _ 그리고 이름이 흔들렸다

v8부터 **태스크 분류 체계**를 만들었습니다.
개발과제명, 개발주제, 요소기술, 연구방법론, 기술적목표 다섯 가지 R&D 항목과, 외부환경·내부역량·시장·전략·BM·정합성 여섯 가지 사업화 항목입니다.
이때 자리 잡은 절차가 하나 있습니다.
새 항목을 추가할 때는 먼저 100건 안팎만 만들어 눈으로 확인하고, 통과하면 본 생성으로 넘어간다.
이 절차는 v21까지 그대로 지켰고, 뒤에서 보시겠지만 저희를 한 번 크게 구해 줍니다.
문제는 다른 데 있었습니다.

버전마다 데이터를 계속 쌓아 올렸습니다.

v9에서 만든 데이터가 v14까지 그대로 실려 왔는데, 그 사이 태스크의 정의가 여러 번 바뀌었습니다.

결과적으로 **같은 이름 아래 서로 다른 기준의 데이터가 섞였습니다.**

v15에서 태스크 11개, 12,117건으로 정리했습니다. 그리고 데이터를 열어 보다가 이런 것을 발견했습니다.

"instruction": "(요청을 실무 질문으로 다듬어)"

"output": "■ 플랫폼 통합 기술: ..."

지시문 자리에 **지시문을 만들라는 메모**가 들어가 있었습니다.

데이터를 만드는 프로그램의 내부 지시가 그대로 학습 데이터가 된 것입니다.

모델은 “요청을 실무 질문으로 다듬어”라는 말을 들으면 요소기술을 답하도록 배우고 있었습니다.

4. 처음 합격 _ v16

v15 데이터에서 깨진 지시문, 중복, 형식 붕괴를 걷어냈습니다.

12,117건에서 10,800건으로 11%를 버렸습니다.

2026년 8월 19일, v16이 프로덕션에 올라갔습니다.

그리고 오늘까지 40일 넘게 그 자리를 지키고 있습니다.

v16의 성적표는 뚜렷하게 기울어 있었습니다.

잘하는 것	점수	못하는 것	점수
사업화 실행전략	1.000	요소기술	0.622
사업화 외부환경	0.933	개발주제	0.589
사업화 정합성	0.867	개발과제명	0.514

사업화는 잘하고 R&D는 약합니다.

특히 요소기술이 문제였습니다.

요소기술은 “이 시스템을 만들려면 어떤 기술이 반드시 필요한가”를 핵심 알고리즘·부품 규격·정량 사양으로 밝히는 항목입니다. R&D 계획서의 심장입니다.

5. 세 번의 되돌림

v17 _ 요소기술을 다시 썼습니다.

기존 답변을 교사 모델에게 다시 쓰게 했습니다.

결과입니다.

	요소기술	사업화 정합성
v16	0.622	**0.867**
v17	올라감	**0.633**

요소기술은 올랐는데 **손대지도 않은 정합성이 무너졌습니다.**

정합성은 “외부환경 → 내부역량 → 경쟁환경 → 실행전략 → BM이 서로 이어지는가”를 점검하는 항목으로, 사업화 전략에서 가장 중요한 부분입니다.

v18 _ 형식을 심었습니다.

답변에 [단계별 요약] [정합성 점검] [보완 방향] 머리말을 반드시 쓰도록 지시문에 박아 넣었습니다. **회복되지 않았습니다.**

v19 _ 데이터를 늘렸습니다.

정합성 데이터를 808건에서 1,200건으로 50% 증량했습니다. **0.633에서 0.600으로, 오히려 떨어졌습니다.**

세 번의 측정이 같은 그림을 그렸습니다.

	R&D 능력	사업화 능력
v16	0.643	**0.837**
v18	0.661	0.773
v19	0.658	0.798

R&D를 가르칠수록 사업화를 잊었습니다. 얻은 것보다 잃은 것이 두 배였습니다.

이것을 **파국적 망각(catastrophic forgetting)** 이라고 부릅니다.

사람이 새 언어를 배우면서 모국어를 잊지 않는 것과 달리, 신경망은 나중에 배운 것이 앞서 배운 것을 덮어씁니다.

6. 네 번째 _ 방어도 통하지 않았다

v20에서는 알려진 방어책을 모두 썼습니다.

첫째, 이어학습. 처음부터 다시 배우는 대신 v16이 배운 것 위에 얹었습니다.

둘째, 리허설. v16이 잘하던 항목을 **v16 자신에게 물어** 그 답을 다시 학습시켰습니다.

외부 교사가 아니라 자기 자신이 교사입니다. 잊지 말라고 복습시킨 셈입니다.

셋째, 사업화 체계 정리. 7개로 흩어져 있던 사업화 항목을 5개 블록으로 묶었습니다.

	종합	R&D	사업화
v16	**0.701**	0.643	**0.837**
v19	0.700	0.658	0.798
v20	0.691	0.644	0.800

사업화가 또 내려갔습니다.

복습까지 시켰는데 v19와 같은 수준입니다. R&D는 0.001 올랐습니다. 사실상 제자리입니다.

학습 방법을 바꿔도 결과가 같다. 이 문장이 다음 결정을 만들었습니다.

7. 자를 의심하다

여기서 한 가지를 짚고 넘어가야 합니다.

평가가 잘못됐을 가능성입니다.

저희는 채점표를 만들어 씁니다. 태스크마다 "이 항목들이 답변에 들어 있는가"를 기계적으로 셉니다. 그런데 이 채점표에 결함이 있었습니다.

요소기술 채점에 TRL(기술성숙도)이 들어 있었습니다.

그런데 모델에게 준 지침에는 "TRL은 언급하지 않는다"고 적혀 있었습니다.

금지한 것을 채점에서 요구한 것입니다.

이 하나 때문에 v16의 요소기술 점수가 0.217로 찍힌 적이 있습니다.

모델의 결함이 아니라 자의 결함이었습니다.

또 하나. **필수 항목이 태스크당 2개뿐**이어서 점수가 0점, 50점, 100점 세 단계로만 나왔습니다.

변화를 볼 수 없었습니다.

그래서 채점표를 다시 만들었습니다.

	이전	이후
필수 항목	태스크당 2개	**태스크당 10개**
문항 배분	태스크당 균등	**R&D 42 : 사업화 18**
가중치	없음	**R&D 70% : 사업화 30%**
처음 보는 문제	없음	**20문항**

R&D에 70%를 준 것은 이 모델이 실제로 하는 일이 그렇기 때문입니다.

평가는 실제 업무 비중을 따라야 합니다.

8. 진짜 원인은 둘이었다

자를 고치고 다시 재 보니, 지금까지 안 보이던 것이 보였습니다.

하나 _ 가르친 적이 없었다

세 버전 모두 바닥인 항목이 셋 있었습니다. 개발주제, 개발과제명, 비즈니스모델입니다.

그리고 **세 모델이 똑같은 것을 똑같이 빠뜨렸습니다.**

항목	거의 매번 빠지는 것	빈도
개발주제	기술동향·시장수요·기술개요	27문항 중 24~27
개발과제명	성과지표·적용분야·정량목표	21문항 중 19~21
비즈니스모델	고객군·핵심활동	9문항 중 8

모델 탓이 아니었습니다.

학습 데이터에 그 내용이 애초에 없었던 것입니다. 배운 적 없는 것을 답할 수는 없습니다.

둘 — 시험지가 새고 있었다

AI를 학습시킬 때는 데이터를 둘로 나눕니다.

학습셋으로 가르치고, **검증셋**으로 시험을 봅니다.

검증셋은 모델이 처음 보는 것이어야 합니다.

저희는 만들어 둔 학습 데이터를 무작위로 5% 떼어 검증셋으로 썼습니다. 흔한 방식입니다.

그런데 문서 하나에서 학습 데이터 여러 개가 나옵니다.

A라는 사업계획서에서 요소기술 문제, 개발주제 문제, BM 문제가 각각 만들어집니다.

무작위로 나누면 **A에서 나온 형제들이 학습셋과 검증셋에 흩어집니다.**

모델은 검증셋을 처음 보는 게 아니라 **이미 사촌을 본 상태**입니다.

검증 점수가 좋아도 실제 실력이 아니었습니다.

9. 다시 짓기로 하다 _ v21

v21에서 세 가지를 동시에 바꿨습니다.

하나, 모델을 둘로 나눴다

R&D 모델과 사업화 모델을 따로 만듭니다. **각자 자기 영역만 배우니 다른 영역을 잊을 일이 없습니다.**

파국적 망각을 이기려 하지 않고, 애초에 일어나지 않을 구조로 바꾼 것입니다.

둘, 원문을 먼저 나눴다

학습 데이터를 나누는 게 아니라 **원본 문서를 먼저 나눴습니다.**

사업계획서·보고서 2,072건

├── 학습용 1,866건 (90%) → 여기서만 학습 데이터를 만든다 (11,000건)

└── 평가용 206건 (10%) → 여기서만 검증 데이터를 만든다 (330건)

평가용 206건은 **어떤 버전에서도 학습에 쓰지 않습니다.** 문서 경로의 해시값으로 배정하기 때문에 나중에 문서가 추가돼도 기존 배정이 흔들리지 않습니다. 학습 직전에 겹침이 0건인지 자동으로 확인합니다.

셋, 채점표를 생성기가 그대로 가져다 쓴다

이전에는 **채점은 요구하는데 학습은 안 가르치는** 항목이 있었습니다. 요소기술 채점이 구현방식을 요구했는데 학습 데이터에는 35%만 있었습니다.

이제 데이터를 만드는 프로그램이 채점표를 그대로 불러옵니다. **자와 교재가 같은 출처를 봅니다.**

10. 실패한 버전도 교사가 된다

새 데이터를 만들려면 “정답”을 써 줄 교사가 필요합니다.

저희는 기존 모델들을 교사로 씁니다. 측정해 보니 **항목마다 최고가 달랐습니다.**

항목	교사	이유
요소기술 · 경쟁환경	v20	그 항목에서 가장 높음 (0.789 / 0.833)
연구방법론 · 내부역량 · BM	v19	그 항목에서 가장 높음
기술적목표 · 정합성 · 실행전략 · 외부환경	v16	그 항목에서 가장 높음

개받주제 · 개받과제명	**범용 모델 + 원문**	세 모델이 모두 바닥

승격에 실패한 모델도 잘하는 항목은 있었습니다.

총점으로는 v16을 못 넘었지만, 요소기술 하나만 보면 v20이 0.622 대 0.789로 가장 낮습니다. 그 강점만 떼어 쓴 것입니다.

반대로 **세 모델이 모두 바닥인 항목**은 우리 모델을 교사로 쓰면 결손이 그대로 대물림됩니다.

그래서 범용 모델에게 원문을 주고 **빠진 항목을 명시적으로 요구**했습니다.

11. 만들기 전에 시험한다

앞서 말씀드린 절차, 즉 **먼저 소량만 만들어 눈으로 본다**를 여기서도 지켰습니다.

11개 항목마다 20건씩, 모두 220건입니다.

여기서 문제가 여럿 잡혔습니다.

첫째, 조건이 서로 모순됐습니다.

개발주제에 "필수 항목 9개 이상"과 "1,600자 이내"를 동시에 걸었더니, **146건을 버리고 10건을 건졌습니다.**

항목을 다 채우면 1,700~2,100자가 나오는데 1,600자로 자른 것이 성립할 수 없는 요구였습니다.

둘째, 자가 과했습니다.

채점 항목을 10개로 맞추려다 겹치는 것을 끌어왔습니다. 개발과제명의 성과지표와 정량목표는 같은 것이고, 기술적목표의 비교대상과 국내외수준도 같은 것이었습니다.

7개로 줄였습니다.

셋째, 무의미한 폐기가 많았습니다.

경쟁환경에서 교사가 'A사', 'B사' 같은 빈 기업명을 쓴다고 79%를 버렸습니다.

목표시장·시장규모·전략캔버스·공급망 분석은 멀쩡한데 표의 열 이름 하나 때문에 통째로 버린 셈이었습니다. 그 검사를 뺐습니다.

시험을 건너뛰었다면 수천 건을 버렸을 것입니다.

그리고 그동안 빠지던 항목이 채워지는 것을 확인했습니다.

항목	빠지던 것	채워짐
개발주제	기술동향·시장수요·기술개요	20/20
개발과제명	적용분야·정량목표	20/20
BM	고객군·핵심활동	20/20

12. 지금까지의 결과

학습 데이터 11,000건과 검증 데이터 330건을 새로 만들었습니다.

생성에 사흘, 학습에 20시간이 걸렸습니다.

가장 뚜렷한 변화는 **검증손실**입니다. 모델이 처음 보는 문제를 얼마나 잘 푸는지 나타내는 숫자로, 낮을수록 좋습니다.

	v20	v21
시작	1.0805	0.5811
최저	1.0745	0.4174
하락 폭	0.006	0.164

v20은 거의 평평했습니다.

배울 것이 없었다는 뜻입니다. 같은 데이터를 몇 번을 돌려도 나아지지 않았습니다.

v21은 크게 내려갔습니다.

새로 만든 데이터에서 실제로 배우고 있다는 뜻입니다. 마지막 두 번이 같은 값이므로 과적합 징후도 없습니다.

(최종 측정 결과는 확인되는 대로 이어서 올리겠습니다.)

마무리 — 두 달에서 배운 다섯 가지

하나, 양이 아니라 깊이입니다.

초기에 건수를 3,797건에서 4,984건으로 늘렸지만 답변은 여전히 원론적이었습니다.

수치와 절차가 들어간 데이터 100건이 두루뭉술한 1,000건보다 낫습니다.

둘, 데이터가 모델보다 먼저입니다.

모델이 못하는 것 같아도 대개는 가르친 적이 없는 것입니다.

세 모델이 똑같은 항목을 똑같이 빠뜨린다면 모델 탓이 아닙니다.

셋, 자부터 의심해야 합니다.

점수가 이상하면 모델을 고치기 전에 채점표를 보십시오.

저희는 금지한 항목을 채점에서 요구하고 있었습니다.

넷, 시험지가 새는지 확인해야 합니다.

검증셋을 무작위로 뗐다면 형제 데이터가 섞였을 수 있습니다.

문서 단위로 먼저 나눠야 합니다.

다섯, 안 되는 방법을 붙들지 말아야 합니다.

저희는 같은 벽에 네 번 부딪혔습니다. 세 번째쯤에 구조를 바꿨어야 했습니다.

덧붙이면, **실패한 버전에도 쓸 데가 있습니다.**

승격하지 못한 v19·v20도 특정 항목에서는 가장 나았습니다. 그 강점만 떼어 새 데이터의 교사로 썼습니다.

맺음

기술을 가진 기업이 사업화까지 가는 길에는 이런 시행착오가 늘 있습니다.

AI 개발도 다르지 않았습니다.

저희가 이 과정을 공개하는 이유는, **잘 만들었다고 자랑하려는 것이 아니라 네 번 되돌리면서 배운 것이 누군가의 시간을 아껴 줄 수 있다고 보기 때문**입니다.

R&D 기획이나 사업화 전략에서 막히는 지점이 있으시면 언제든 문의해 주십시오.

이 사업, 신청하면 붙을까요?

R&D 평가위원이 만든 진단으로 우리 기업의 선정 가능성을
항목별 점수로 확인해 보십시오. 무료입니다.

▶ 무료 기업진단 labwith.kr/free-diagnosis.html?utm_source=cafe
▶ R&D 실무교육 아카데미 (연구계획서·PSST·IR DECK)
labwith.kr/training-rnd.html?utm_source=cafe

기술사업화 랩 위드㈜ · 010-6408-8557 · 카카오톡 antipoto

클린봇

기술사업화 랩 위드
댓글을 남겨보세요
등록