셔플된 4프레임의 시간 순서 복원

24순열 우도 채점, 검증이 이끄는 개발, 그리고 24GB 제약의 해소

팀 Lobstar (1인) · SNU AI Challenge 2026 예선 보고서

최종 성과 (Private)	모델	심사환경 추론	학습 비용
0.94308	**27B 단일**	**2.40 h**	**≈ $24**
공동 1위 · Public 0.94415	Qwen3.6 · QLoRA 4bit (4.45% 학습)	RTX 3090 1장 · 819행 · 23.9GB	H100 1장 × 11.4시간, 단 1회

핵심 요약

1. **작게 검증하고, 포화를 확인한 뒤에만 키움.** 경량 모델(r32)로 레버를 걸러내고 포화 신호를 확인한 뒤에야 27B·r256으로 이전했다 — 덕분에 본 학습은 단 한 번이었다(1×H100, vast.ai 시간당 약 $2.06 × 11.4시간 ≈ **$24**).
2. **과제에 맞는 모델을 고르고, 판단은 자체 검증으로 함.** 모델 용량(어댑터 랭크)과 학습 레버(고해상도 등)의 상호작용을 진단해 Qwen3.6-27B(r256, GDN 동결)에 도달했고, 채택·기각은 학습에서 분리해 둔 검증셋 점수와 통계 검정으로 판단했다.
3. **일반화를 위해 눈앞의 점수를 두 번 버림.** 라벨 편향은 증강으로 제거했고, 검증 점수가 오르는 사전확률 보정도 분포가 바뀌면 해가 될 수 있어 버렸다 — 종료 후 Private에서 이 보수성이 옳았음이 확인됐다.
4. **추론 속도도 강점임.** 병목을 계측으로 규명하고(GPU 연산 점유 24%) 추론 파이프라인을 재설계해, RTX 3090 한 장에서 27B로 819행을 **2시간 24분**에 채점한다(9시간 16분에서 단축).

1. 개요 — 문제, 그리고 하나의 관점과 하나의 사이클

과제는 셔플된 비디오 프레임 4장과 한 문장 캡션을 보고 **원래 시간 순서를 복원**하는 것이다. 채점 지표는 예측한 순서가 정답과 완전히 일치한 표본의 비율(EM, Exact Match)이고, 제출물은 RTX 3090 24GB 한 장·24시간 안에서 재현되어야 한다. 그래서 풀어야 할 문제가 둘이었다: 24가지 순서 중 하나를 정확히 고르는 **판별력**, 그리고 27B급 모델을 그 제약 안에서 돌리는 **공학**.

해법은 두 축으로 요약된다. **관점**은 "생성 문제를 닫힌 후보 판별 문제로 바꾼 것", **방법**은 "검증 신호를 보고 다음 개입을 결정하는 사이클을 끝까지 유지한 것"이다.

> 💡 **왜 효과적이었나**
> **기반**은 정렬이다: 모델이 배우는 것(학습 목적)과 심사받는 것(채점 규칙)이 처음부터 끝까지 같았고, 이 정렬을 벗어난 시도는 전부 점수가 떨어졌다(§5.4~5.5). 그 위에서 성과를 만든 것은 실험 전략이다 — 작게 검증하고 포화를 확인한 뒤 이전하는 자원 배분, 모델 용량과 학습 레버의 상호작용을 읽어낸 모델 선택, 그리고 점수가 올라도 버리는 규율.

> ✨ **이 접근의 독창적인 지점**
> 1. **스크리닝→이전 전략(§5.2):** 값싼 r32 라인에서 레버(채점기·증강·TTA)를 확정하고, 포화 신호를 확인한 뒤에야 고급 모델로 이전 — 본 학습이 단일 런(≈$24)으로 끝난 이유다.
> 2. **용량과 레버의 상호작용 진단(§5.2 III):** 모델 확대도 고해상도 학습도 단독으로는 이득이 0이었는데, 어댑터 용량을 r256으로 올리자 같은 레버가 효과를 냈다 — 이 실측이 모델·용량·세대를 한 번에 결정해 줬다.
> 3. **규율 있는 기각(§3, §5.5):** "점수가 올라도 우연이나 분포 특수성으로 설명되면 채택하지 않는다"는 기준을 미리 정했고, 실제로 두 건을 버렸다 — 종료 후 Private에서 보수성이 옳았음이 확인됐다.
> 4. **기술 고안:** 조기 종료와 같은 통계량을 반대 방향으로 쓰는 **마진 기반 적응형 가지치기**(§7).

2. 문제 정식화 — 왜 생성이 아니라 판별인가

이 과제의 정답은 4! = 24가지뿐이다. 무한한 문장을 만들어내는 생성 문제가 아니라, **닫힌 집합에서 하나를 고르는 판별 문제**다. 그런데 자유 생성으로 답을 쓰게 하면 모델은 탐색할 필요가 없는 공간까지 탐색한다 — 실제로 CoT를 포함한 자유 생성 채점은 같은 조건에서 **−5%p**로 실측됐다.

그래서 답 문자열 24개 각각의 시퀀스 로그확률을 teacher forcing으로 전부 계산한 뒤 가장 높은 후보를 고르는 **생성적 채점기**를 택했다. 이 선택에는 이론적 근거가 있다: 답 토큰의 CE 손실을 전개하면 "남은 후보들에 대한 정규화 선택의 연쇄"가 되는데, (i) 각 순위 표지가 단일 토큰이고 (ii) 유효 후보 밖 어휘 확률이 무시 가능할 때, 이는 Plackett–Luce 모형의 listwise MLE와 일치한다. 즉 **학습이 최적화하는 목적과 추론이 사용하는 결정 규칙이 같은 대상**이 된다.

위치를 짚어 두면, 프레임 순서 예측은 자기지도 표현 학습의 고전적 과제이고 [8, 9], 닫힌 후보를 우도 비교로 채점하는 방식도 언어 모델의 다지선다 평가에서 쓰여 온 관행이다 [10]. 우리의 기여는 이 채점 방식을 **학습 목적과 명시적으로 정렬**시키고(Plackett–Luce 관점), 그 정렬을 한 번의 선택이 아니라 **이후 모든 개입을 걸러내는 기준**으로 운용한 데 있다 — 정렬을 벗어난 시도(소프트 라벨, 답 형식 변경, 마진 손실)는 예외 없이 열세로 실측됐고, §5.4~5.5의 기각 목록이 그 기록이다.

📋 **방법 명세 — 모델이 실제로 보는 것과 내는 것**

입력은 셔플된 프레임 4장(제시 순서 1~4)과 캡션, 그리고 "각 이미지가 시간상 몇 번째인지 리스트로 답하라"는 지시문이다. 답은 `[2, 1, 4, 3]` 같은 형태 — "첫 번째로 제시된 이미지는 시간상 2번째"라는 뜻(역순열 형식) — 이고, 각 순위 숫자는 토크나이저에서 단일 토큰이라 위 (i) 조건이 성립한다. 채점기는 이 답 문자열 24가지 각각의 로그확률을 teacher forcing으로 계산해 가장 높은 후보를 제출한다. **셔플 TTA**란 프레임 제시 순서를 바꿔 같은 표본을 최대 K번 다시 채점하고 후보별 로그확률을 누적 합산하는 것으로, 특정 제시 순서에서 오는 편향을 상쇄한다. 추론 상수는 전부 홀드아웃에서 사전 고정했다: 셔플 K=6, 조기 종료 임계 τ=6.0, 가지치기 기준 D=4.0, 후보 배치 G=4.

3. 데이터 활용 — 증강, 검증셋 설계, 일반화 우선

균일 순열 증강 — 라벨 편향의 제거. 데이터 명세상 `No_ordering=True`(15.5%)는 셔플되지 않은 정상 비디오라 정답이 [1,2,3,4]다. 이 표본들을 따로 처리하는 대신, **모든 표본에 균일 무작위 순열을 즉석 적용하고 라벨을 재계산**하는 증강을 썼다. 정답 분포가 균일해지므로 모델이 "데이터가 어떻게 만들어졌는가"라는 편중을 외울 수 없게 되고, 증강이 학습 중 즉석 생성이라 전처리 연산도 0이다. 실제로 균일 취급으로 학습하자 **원래 셔플되지 않았던 표본들(`No_ordering=True`)과 셔플된 표본들** 사이의 검증 EM 격차가 사라졌다(0.629 vs 0.630) — 그 전에는 이 두 집단이 서로 다른 문제처럼 풀리고 있었다는 뜻이다.

검증셋을 진단 도구로. 학습 데이터에서 800행을 떼어 검증셋으로 고정하고(모든 학습에서 미사용), 추론 결과를 **마진**(1위와 2위 후보의 로그확률 격차)·**정답의 순위**(24개 후보 중 몇 등인지)·**오답의 구조 유형**으로 분해해 모델의 약점을 파악했다(결과·처방은 §5.3~5.4). 데이터를 학습 재료로만이 아니라 다음 시도를 알려 주는 **진단의 원천**으로 쓴 것이다.

🚫 **점수가 올라도 버린 것 — 사전확률 보정**

균일 증강의 부수 효과로 모델은 항등 순열을 과소 예측한다(검증 정답 15.5% vs 예측 10.5%). 이를 채점 단계에서 되메우는 label-shift 보정을 설계해 검증셋 +1.25%p, **500회 반복 교차검증**(손해로 뒤집힌 비율 1%), **다른 모델 3종 전이**(전부 양수), 리더보드 방향(+0.35%p)까지 전부 통과시켰다. 그런데도 **기각했다** — 보정이 올리는 것은 "같은 사전확률 분포에서의 점수"일 뿐, 분포가 다르면 손실로 뒤집히기 때문이다. 과소 예측 상태는 분포 이동에 대한 안전 여유로 남겼다. 반대 방향 보정(사전확률을 *빼는* DCPMI)은 −2.63%p(p=0.005)로 실측 실패 — 더하는 쪽과 빼는 쪽을 모두 실험한 끝에 어느 쪽도 넣지 않기로 한 것이다.

정크 표본. 초기에 모델들이 예외 없이 틀리고 마진도 0에 가까운 표본 47건(전체 9,535행의 0.49%)을 플래그하고 학습 분할 내 42건을 제외했다(= 8,693행). 사후 재검토에서 이들은 라벨 오류가 아니라 몽타주처럼 본질적으로 어려운 표본으로 확인됐다 — 제외 근거가 정확하진 않았지만 규모가 0.49%라 영향은 미미하다고 본다.

데이터 누수 없음. 평가 데이터의 정보(라벨·통계·특성)는 학습·전처리·모델 설계 어디에도 유입되지 않았다. 채점 설정(τ, D, K)은 전부 홀드아웃에서 사전 고정 후 무수정 1회 적용했고, 테스트 이미지·캡션을 열어 분석한 일은 없다(§5.1).

4. 모델 설계 — 실제 구조 위에서 무엇을 학습했나

Qwen3.6-27B는 64층 디코더를 가진 dense 비전-언어 모델인데, 64층이 균일하지 않다. **같은 블록이 16번 반복**되고, 한 블록은 「Gated DeltaNet(선형 어텐션) → FFN」 3개와 「Gated Attention(전역 어텐션) → FFN」 1개로 구성된 **3:1 하이브리드**다. 아래 그림에 실제 구조 위에서 학습한 부분(초록)과 동결한 부분(회색)을 표시했다.

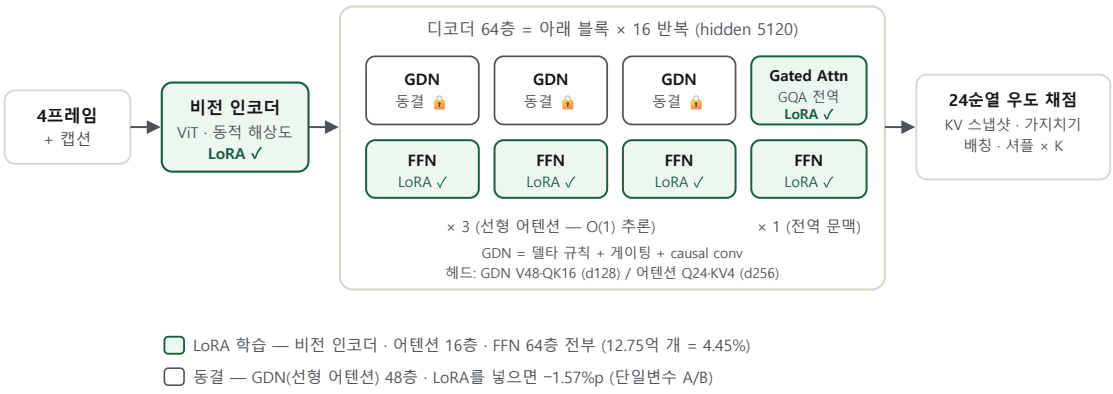

그림 1 — 시스템 구조. Qwen3.6-27B의 실제 3:1 하이브리드 블록 위에 학습(초록)/동결(회색)을 표시. LoRA는 이름 기반으로 q/k/v/o·gate/up/down 투영에만 붙기 때문에 GDN 내부 모듈은 자연스럽게 제외된다. 학습·추론 모두 4bit NF4 양자화 상태로 동일하다.

💡 **왜 GDN만 얼렸나 — 증거**

다른 조건은 전부 같게 두고 **GDN에 LoRA를 넣는지 여부만 바꾼 비교 실험(단일변수 A/B)**을 했다. 검증 EM이 0.6300→0.6188로 떨어졌고, 오답 구조에서도 전역 붕괴형(scramble — 세 개 이상 순위가 뒤엉킨 오답, §5.3 표)이 63→72건, 완전 역순이 9→14건으로 §5.3에서 진단한 취약 유형이 되돌아왔다. 리더보드도 −1.57%p로 같은 방향 — 거시 지표(EM)와 미시 구조(오답 유형)가 일치해 동결을 확정했다.

또 하나 짚을 점: r256 전환은 랭크를 올린 것만이 아니라 **비전 인코더를 적응 범위에 포함**시킨 변경이기도 하다. 고해상도 학습의 이득이 이 구성에서만 나타난 것은(§5.2 III), 순서 판별에 유용한 시각 특징이 함께 조정될 여지가 생겼기 때문이라고 해석한다.

결정	값	근거 (실측)
베이스	Qwen3.6-27B (GDN 하이브리드)	세대 전환 +3.75%p · 24GB 안에서 성능/메모리 균형
적응	QLoRA r256/α512 — 어텐션+FFN+**비전 인코더**	r32 대비 **고해상도 레버 활성화** (§5.2)
GDN층 동결	선형 어텐션 계열 LoRA 제외	**포함하면 −1.57%p (단일변수 A/B)**
양자화	4bit NF4, 학습·추론 동일 상태	배포 시 양자화 불일치가 없음
손실	하드 CE	소프트 타깃은 마진을 압축 (1.73 → 1.13)
해상도	학습 640·28² ~ 1024·28² (업스케일)	원본 프레임(640×360)은 추론에서 원본 그대로 처리
스케줄	3에폭 · 유효배치 32 · 816 step · AdamW · 학습률 1e-4 (cosine, 3% warmup)	총 11.4 H100-시간의 단일 런으로 완료 (§8)

표 — 모델·학습의 주요 결정 요약. "근거" 열은 각 값을 선택하게 만든 실측 결과다: 모든 행이 대안과의 비교 실험 또는 제약 조건에서 나왔다.

5. 실험 전략 — 검증이 이끄는 개발

5.1 평가 프로토콜 — 판단은 자체 검증으로, 리더보드는 방향타로

의사결정 지표는 **둘뿐**이고 둘 다 학습 데이터에서 분리한 홀드아웃이다: **(a) 검증 EM**(800행, 모든 학습에서 미사용), **(b) 층화 하드셋 300**(무작위 150 + 난이도 상위 150) — 여기서 이겼다는 것만으로는 채택하지 않고 McNemar 검정 통과를 요구했다. **테스트 데이터는 어떤 형태로도 분석하지 않았고**, 리더보드는 사전 기준을 정한 뒤 방향 확인에만 썼다. 믿은 것은 절대값이 아니라 **후보 간 순서**다:

모델/변형	검증 EM (800행)	Public LB	결정
초기 생성 채점 (그림 2의 ①)	0.43	0.60383	재정식화의 출발점
경량 스크리닝 기준선 (r32)	0.5925	0.91797	레버 검증용
저해상도 채점 대조	0.6025	0.92670	채점 전처리 정합으로 개선
MoE 변형 (신세대 35B-A3B)	0.5913	0.93019	기각 — 검증·LB 모두 열세
GDN 어댑터 확장	0.6188	0.93542	기각 — 검증·LB 모두 열세
최종 모델 (그림 2의 ⑦·⑧과 동일 가중치)	**0.6300**	**0.95113 / 0.94415**	**채택 — 검증에서도 최고**

표 — 검증 EM과 채택 결정의 일관성. 괄호 번호는 §6 그림 2(개발 로드맵)의 단계 번호다. 검증 채점은 비용 절약을 위해 경량 설정(낮은 셔플 수 등)으로 수행해 절대값이 리더보드 채점보다 낮게 측정된다. 세대가 다른 모델 간에는 검증 채점 설정도 달라 절대값의 직접 비교에는 한계가 있으므로, 채택·기각은 같은 설정으로 잰 쌍의 비교(최종 모델 vs MoE 변형, vs GDN 확장)로 판단했다.

5.2 실험 전략의 핵심 — 작게 검증하고, 포화를 확인한 뒤 이전한다

고급 모델은 학습 한 번이 비싸다. 그래서 우리는 **비싼 실험을 값싼 라인에서 먼저 소진시키는 것**을 전략의 축으로 삼았다.

1. **정식화 스크리닝** — 경량 라인(r32)에서 레버를 저비용으로 검증했다. 생성→판별 전환 하나로 +26%p(그림 2 ②). 여기서 확정한 레버(채점기·증강·TTA)는 이후 모든 세대에서 그대로 유지됐다 — 작은 모델의 결론이 큰 모델로 이월된다는 확인이 이 전략의 전제였다.
2. **데이터·세대** — 균일 순열 증강(§3)과 베이스 세대 전환(검증 +3.75%p가 리더보드로 그대로 이전, 그림 2 ③). 구세대 MoE 베이스(30B-A3B)는 4bit 양자화를 통과하지 못해 24GB 적재 불가로 폐기했다 — §5.1 표의 MoE 변형(신세대 35B-A3B)은 이후의 별개 실험이다.
3. **포화 신호 → 이전** — 경량 라인의 개선이 정체되고(④≈0.911→⑤ 0.918), 모델을 키워도·고해상도로 학습해도 r32에서는 이득이 0으로 나오자, 이를 **"이 라인의 레버가 소진됐다"는 신호**로 읽고 고급 모델·고용량(r256)으로 이전했다. 그러자 같은 레버가 효과를 내기 시작해 0.918→0.949로 도약했다(⑥) — 병목은 용량 단독도 레버 단독도 아닌 **둘의 상호작용**이라는 진단에 이르렀다(이 도약에는 세대 전환도 함께 있어 요인별 분해는 §9의 한계로 남긴다).
4. **진단 정밀화·강건 마감** — 오답 296건 분해(§5.3)로 처방을 표적화하고, 하드 CE 확정(⑦), 재현 보장 산출물 선택(⑧)으로 마감했다.

이 전략의 결과가 §8의 비용 구조다: 실험은 많았지만, 27B 본 학습은 검증된 레시피의 **단일 런(≈$24)**으로 끝났다.

5.3 진단 — 검증셋 오답 296건의 세 축 분해

최종 후보 모델을 검증셋 800행(§5.1의 홀드아웃)에 추론시켜 나온 오답 296건(검증 EM 0.6300)을 전수 분류했다.

축	관찰 ⇒ 함의
마진	정답률이 마진 <0.5 구간 **23%** → ≥5 구간 **99.6%**. ⇒ EM은 사실상 저마진 질량의 함수. 확신 표본의 조기 종료는 안전하고, 셔플 예산은 저마진에 몰아줘야 한다(§7).
순위	오답의 70%는 정답이 top-2 밖(평균 순위 5.83등). ⇒ 채점을 다듬는 것으로는 벽을 못 넘는다 — 모델 표상(모델이 내부적으로 형성한 이해)의 문제. 후처리 계열을 전부 접은 근거.
오답 유형	전역 붕괴(scramble) 오답의 52%가 "시간상 첫 프레임을 못 찍는" 시작점 앵커링 실패. ⇒ 처방의 표적이 정해진다(§5.4).

오답 유형의 정의와 분포 — 이 296건을 정답 순열과의 관계로 분류하면:

유형	뜻	건수 (비중)
인접교환	이웃한 두 프레임의 순서만 뒤바뀜	84 (28%)
비인접 교환	떨어져 있는 두 프레임이 서로 뒤바뀜	67 (23%)
3-순환	세 프레임이 자리를 돌아가며 어긋남	67 (23%)
전역 붕괴 (scramble)	세 개 이상 순위가 뒤엉켜 구조를 알아볼 수 없음	63 (21%)

유형	뜻	건수 (비중)
완전 역순	순서 전체가 정확히 뒤집힘	9 (3%)
항등 출력	셔플된 표본인데 [1,2,3,4]로 출력	6 (2%)

이 분포는 **세대 간 이동**도 보여 준다. r32 기준선(오답 326건)에서는 전역 붕괴 82건·3-순환 82건이 최대였지만, 최종 모델에서는 인접교환 84건이 최대다 — 큰 시간 구조는 정복되었고 **이웃 프레임 사이의 판별이 마지막 병목**으로 남았다는 뜻이며, 남은 문제의 성격 자체가 바뀌었음을 의미한다.

분석 자체도 그대로 믿지 않았다 — 초기 가설 두 건은 **스스로 반박을 시도하는 검증**을 거쳐 기각하거나 고쳐 썼다(저마진 구간 해석의 과잉 일반화 1건, "캡션이 빈곤해서 틀린다"는 가설은 문체 차이에 의한 교란으로 재서술). 진단이 자기 확신으로 굳는 것을 막기 위한 절차였다.

5.4 처방과 판정 — 채택 기준을 먼저 정해 두고 실험했다 (사전등록 게이트)

진단	처방	판정
확신 구간은 안전	**적응형 조기 종료**	**채택 — 조기 종료 없이 셔플 6회를 전부 도는 완주(K=6 고정)와 결과가 동일함을 확인**
시각 단서 부족형 오답	부분순서 보조 증강	기각 — 개선이 유의하지 않음(+0.37%p)
시작점 앵커링 실패	첫 프레임 표적 증강	설계 완료 — 시간 예산 소진으로 미판정
표상에 첫프레임 정보 존재	순서좌표 보조헤드 (CORN)	기각 — 사전 기준(134/300) 미달(131)
라벨 노이즈 ~12% 추정	Mallows 소프트 라벨	기각 — −1.12%p (정렬 이탈 비용)

> 🚫 **부정적 결과 — 규칙이 우리에게 불리할 때도 작동했다**
> **직접 순열 실험:** 답 형식을 역순열(각 이미지의 시간상 위치)에서 직접 순열(시간순 이미지 나열)로 바꾸는 가설을 제로샷 페어드 검증(35승 9패, p<0.001)까지 통과시키고 학습했지만, 두 독립 런이 모두 검증에서 붕괴했다(train 92.5% vs val 50% — 암기 과적합). 제로샷 우위 ≠ 학습 후 우위. **점수 오르는 후보의 기각:** 층화 하드셋에서 +7/300이 나온 후보도, 그 정도 차이는 우연으로 설명될 수 있어(McNemar 검정 p≈0.36) 기각했다. 사전확률 보정 기각(§3)과 함께, 둘 다 점수가 오르는 방향인데도 버린 결정이다.

6. 실험 결과 — 개발 로드맵

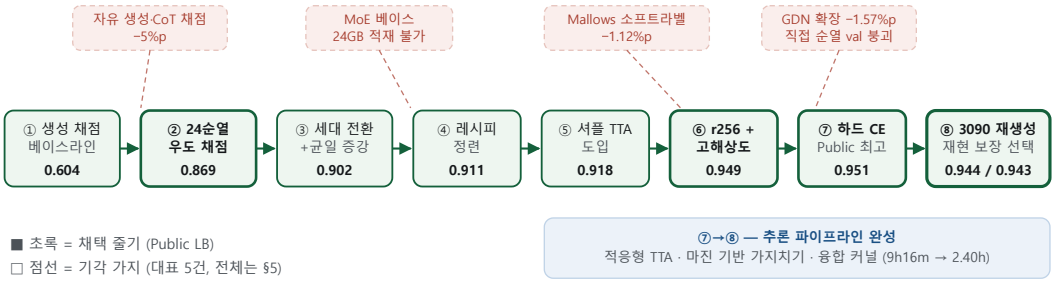

그림 2 — 개발 로드맵. 줄기 ①~⑧은 채택된 변경과 당시 Public LB — 각 단계는 리더보드 제출 이력과 직접 대조할 수 있다. ⑧의 Private는 0.94308(공동 1위).

> 🔪 **해석의 기준자 — 하드웨어 노이즈 밴드 ±0.70%p**
> ⑦→⑧은 동일 가중치·동일 코드에서 GPU만 H100→3090으로 바꾼 재생성인데도 Public이 −0.70%p 움직였다(단일 대조쌍 실측, 보수적 참고선). 그래서 이 밴드 안의 작은 점수 차이는 "기각 확정"이 아니라 "개선 증거 없음"으로 읽었고, 최종 제출도 Public 최고본(⑦) 대신 심사환경 재현이 보장되는 ⑧을 택했다. **사후 확인:** 종료 후 공개된 점수에서 ⑦은 Private −1.21%p 하락, ⑧은 −0.11%p 방어 — 사전 판단과 부합했다.

7. 최적화 — 24GB·24시간 제약의 해소

병목의 규명. 비용의 87~89%가 prefix(24개 후보가 공유하는 이미지 4장+캡션 입력부) 계산인데, 정작 GPU 연산 점유율은 **24%**뿐이었다 — 계산량이 아니라 오버헤드가 병목이다. 원인은 GDN 48층이 융합 커널 없이 실행될 때 **소형 연산마다 CPU→GPU 실행 지시 왕복(커널 런치)**이 시간을 지배하는 구조. 처방의 공통 원리는 하나다 — **왕복 횟수를 줄인다.**

KV 캐시 스냅샷 — 비싼 prefix는 셔플당 1회만 순전파하고 스냅샷, 24후보는 복사본 위에서 짧은 답 토큰만 채점(prefix 재계산 제거). 실행 시작 시마다 순진한 방식과 같은 답인지 자가 검증한다.

적응형 셔플 TTA — 마진이 τ를 넘으면 조기 종료하고 낮은 표본만 K=6 완주(불필요한 셔플 제거) — 아낀 연산을 어려운 표본에 재배분한다.

마진 기반 가지치기 (자체 고안) — 1위 후보와의 점수 격차가 기준 D를 넘는 후보를 다음 셔플부터 제외(가망 없는 후보 제거). 조기 종료와 같은 통계량을 반대 방향으로 쓴다.

후보 배칭 (G=4) — 후보 4개씩 한 forward로 묶어 런치·루프 왕복을 1/4로. 비용식 [24/G]+(K−1)[M/G]에 따라 "배치 경계 안에서 후보를 최대한 남기는" 것이 목표가 된다.

융합 커널 — GDN 48층의 소형 런치를 Triton 융합 커널로 통합(층 내부 왕복 제거)하고, 실제 로드된 커널을 서명으로 검사하는 가드를 붙였다.

✅ 재현성 — 심사환경에서 그대로 재현된다
구성 요소들은 채점 수식을 바꾸지 않는 구현 개선(계산 재사용·배칭)과 연산 재배분이고, 적응형 요소는 완주(K=6 고정)와 결과가 같음을 확인했다. 심사환경(CUDA 12.4)에서 819행을 재실행하면 test 제출 CSV와 **816/819가 일치**하며, 차이 3행은 전부 마진 <0.18의 동점 표본으로 **CUDA 런타임 차이(12.6→12.4)**에서 온다(커널 경로만 달리한 통제 실행에서도 같은 3행·같은 값 — 원인 분리). cuBLAS 워크스페이스 고정 시 같은 환경 재실행 출력은 **바이트 단위로 동일**하다.

8. 자원 효율과 구축 비용

구성 (1×3090)	표본 수	표본당	총 시간	24h 대비
test · 최적화 이전	819	40.7 s	9h 16m	38.6%
test · 최종	**819**	**10.5 s**	**2.40 h**	**10.0%**
외부 데이터셋 · 최종	3,884	15.9 s	17.16 h	71.5%

외부 데이터셋의 표본당 지연이 더 긴 것은 **적응형 재배분의 의도된 결과**다 — 어려운 표본이 많을수록 자동으로 더 오래 생각한다. VRAM 피크는 23.9GB(한도의 97%)로, fp32 어댑터의 bf16 상주와 단편화 방지 할당자 둘 다 필수다.

구축 비용: 제출 어댑터는 1×H100 SXM 80GB(vast.ai 대여, 시간당 약 $2.06)에서 11시간 23분 — **약 $24짜리 단일 런**의 산출물이다. 이어 학습이 아니라 숨은 학습 시간이 없고, 증강이 즉석 생성이라 전처리 연산도 0이다. 비용 효율의 핵심은 §5.2의 구조다 — 비싼 실험은 r32에서 걸렀고, 본 학습은 한 번이었다.

9. 한계와 결론

잔차의 성격
오답의 70%는 정답이 top-2 밖 — 상당 부분은 채점이 아니라 표상 수준의 한계이거나 본질적 모호성이다.

진단과 처방의 간극
모델 내부 표현을 간단한 분류기로 읽어 보면 첫 프레임 정보를 56% 복원할 수 있었지만(무작위 추측은 25%), 이 여지를 실제 성능으로 바꾸는 처방(보조헤드·표적 증강)은 성공시키지 못했다.

원인 미분리
직접 순열 붕괴가 형식 단독 효과인지, III단계 도약에서 세대 전환의 기여분이 얼마인지는 통제 실험을 하지 못했다.

설계의 적용 범위
24순열 전수 채점은 4프레임(4! = 24)이라서 가능한 설계다. 프레임 수가 늘면 순열 수가 계승적으로 폭발하므로(5! = 120), 그대로 확장할 수 없고 후보 축소(쌍별 비교 등)와 결합해야 한다.

향후 지표 제안 — 저마진 질량
EM은 저마진 질량의 함수이므로(마진 <1 질량 1%p ≈ EM 0.61%p) 라벨 없이 계산되는 저마진 질량 축소를 대리 지표로 쓸 수 있다고 본다. 다만 확신-오답 모드가 지표를 속일 수 있어(직접 순열 실험), 이번에는 표본 내 용도(조기 종료·가지치기)에 한정하고, 채택 판정은 정답 라벨 기반 기준으로 유지했다.

요체는 세 원칙의 일관된 적용이다: **정렬**(학습 목적 = 채점 규칙), **분리**(모델의 판단 ≠ 분포 사전지식), **규율**(검증 신호 없이 방향 없고, 게이트 없이 채택 없다). 제출은 검증이 이끄는 사이클의 기록이며, 기각된 처방들도 시스템의 일부다.

참고문헌

[1] Plackett (1975) · [2] Luce (1959) — 순열 선택 모형
[3] Xia et al. (ICML 2008) — listwise MLE (ListMLE)
[4] Hu et al. (ICLR 2022) LoRA · [5] Dettmers et al. (NeurIPS 2023) QLoRA
[6] Yang et al. (ICLR 2025) Gated Delta Networks · [7] Yang & Zhang (2024) FLA 라이브러리
[8] Misra, Zitnick, Hebert. Shuffle and Learn: Unsupervised Learning using Temporal Order Verification. *ECCV* 2016. · [9] Lee, Huang, Singh, Yang. Unsupervised Representation Learning by Sorting Sequences. *ICCV* 2017. — 프레임 순서 과제의 선행 계보
[10] Brown et al. Language Models are Few-Shot Learners. *NeurIPS* 2020. — 닫힌 후보의 우도 비교 평가 관행
[11] McNemar (1947) · [12] Shi et al. (2021) CORN · [13] Saerens et al. (2002) label-shift

설치·재현 절차는 제출 저장소 README에, 제출 CSV 산출 기록과 학습 1차 증거는 `docs/PROVENANCE.md`·`docs/TRAINING.md`에 기재. 모델 구조 출처: Qwen/Qwen3.6-27B 모델 카드.