문장 기반 프레임 순서 예측

Qwen3.6-27B QLoRA SFT와 permutation TTA

팀 DeepRed · 2026 SNU AI Challenge 예선 보고서

1. 과제와 접근

문장 하나와 순서가 섞인 이미지 네 장이 주어질 때 각 이미지의 시간 순위를 예측하는 과제다. 정답은 네 원소의 permutation이므로 후보는 24개이고, 평가 지표인 Exact Match는 네 순위가 모두 맞아야 점수를 준다. 학습 데이터는 9,535개, 테스트 데이터는 819개다.

이 과제에는 설계에 활용할 수 있는 두 가지 성질이 있다. 첫째, **Exact Match는 0-1 loss**다. 부분 점수가 없으므로 최적 결정은 posterior의 mode를 고르는 것이고, 위치별 정확도를 조금씩 올리는 것보다 하나의 permutation 전체에 확률을 모으는 편이 유리하다. 둘째, **정답은 이미지 제시 순서에 대해 equivariant**하다. 입력을 permutation σ로 재배열하면 정답도 σ에 따라 deterministic하게 변환되므로, 학습에서는 한 샘플로 최대 24가지 학습 예를 만들 수 있고 추론에서는 같은 문제를 여러 배열로 물어 예측 오차를 상쇄할 수 있다.

최종 모델은 Qwen3.6-27B를 NF4 4비트로 양자화하고 QLoRA로 SFT한 단일 adapter다. 학습에는 이미지 순서 shuffle과 문장 변형 augmentation을 적용했고, augmentation의 구성과 비율은 error analysis와 통제 실험을 근거로 정했다. 추론에서는 permutation TTA로 일곱 개 view를 만들고 Exact Match에 맞춘 position-wise voting으로 집계했다. 모든 채택 판단에는 validation set의 측정 한계에 기반한 동일한 기준을 적용했다.

항목	값
최종 모델	Qwen3.6-27B (Apache-2.0, 공개 2026-04-22), NF4 4비트 + LoRA 단일 adapter
로컬 검증 (477개)	288/477 (60.38%)
Public / Private 리더보드	0.94415 / 0.93902
추론	RTX 3090 24GB 1장, 6시간 36분, peak memory 17.97 GiB
학습	H100 80GB 1장, 합계 약 18 GPU-hour

2. 학습

2.1 백본 선택

모든 실험은 동일한 고정 split(seed 42 → train 9,058 / validation 477)에서 수행했다. Zero-shot 성능은 Qwen3-VL-8B 20.55%, 같은 모델의 Thinking 버전 18.24%, Qwen3.5-9B 17.40%로 모두 20% 내외이며, 예측이 identity permutation으로 강하게 편향된다. SFT 없이 도달 가능한 정확도는 매우 낮다.

Thinking 버전은 Instruct 버전보다 2.31%p 낮으면서 생성 시간이 2시간 26분으로 늘어난다. 병목은 chain-of-thought의 길이가 아니라 프레임 사이의 시각적 시간 단서를 읽는 능력이며, 이 능력은 SFT로만 개선되었다. 이에 따라 chain-of-thought를 사용하지 않고 출력을 permutation 하나로 제한했다.

백본 비교는 SFT 후 성능으로 수행했다. 조건을 맞추기 위해 모두 augmentation과 TTA 없이 epoch 2 체크포인트에서 greedy decoding한 결과다.

모델	학습 방식	Exact Match	eval loss
Qwen3-VL-8B	NF4 QLoRA	52.41%	0.1129
Qwen3-VL-32B	NF4 QLoRA	53.67%	0.1078
Qwen3.5-9B	BF16 LoRA	55.77%	0.1071
Qwen3.6-27B	NF4 QLoRA	**57.23%**	0.1036

규모보다 계열의 차이가 컸다. 32B인 Qwen3-VL-32B가 9B인 Qwen3.5-9B보다 낮았고 Qwen3.5/3.6 계열이 일관되게 유리했다. 이에 따라 NF4 4비트로 24GB 안에 들어가는 최대 규모인 Qwen3.6-27B로 확정했다.

2.2 Augmentation

이미지 순서 shuffle. 매 학습 스텝에서 네 이미지를 24개 permutation 중 하나로 재배열하고 정답도 재배열된 좌표계로 함께 변환한다. 원본 순서를 그대로 쓰면 모델은 "i번째로 제시된 이미지의 정답 순위는 대략 i"라는 shortcut을 학습하는데 shuffle이 이 상관을 끊고, 동시에 정답이 24개 클래스에 고르게 퍼져 특정 출력으로 collapse하는 현상도 완화된다.

문장 변형. 이 augmentation의 근거는 error analysis다. 중간 모델의 검증 오답 188개에서 정확도를 지배하는 변수는 문장에 포함된 시간 연결어(then, after, before 등)의 개수였다. 연결어가 0개면 33.3%, 1개 53.0%, 2개 79.5%, 3개 85.1%로 차이가 컸다.

연결어 의존이 인과적인지 확인하기 위해 입력을 직접 조작한 통제 실험을 수행했다. 연결어가 두 개 이상인 문장에서 첫 절만 남기면 80.5%에서 40.3%로 떨어졌고, 문장을 아예 비우면 21.8%로 급락했다. 모델이 문장의 접속 구조에 의존하는 shortcut을 쓰고 있으며 시각 정보만으로 순서를 판단하는 경로가 약하다는 뜻이다. 테스트 데이터의 약 49%가 연결어 0~1개 구간이므로 직접적인 성능 손실이었다.

문장 변형 augmentation은 이 shortcut을 학습 단계에서 차단한다. 문장을 원문 70%, 첫 절만 20%, 내용 제거 10%의 비율로 섞어 제시하고, 문장이 제거된 경우에는 시각 정보만으로 판단하라는 안내문을 대신 입력한다. Prompt에서도 어떤 시각 단서를 보아야 하는지를 반복 등장하는 대상, 물체의 상태 변화, 장면의 진행으로 명시해 같은 목적을 강화했다. 학습과 추론은 완전히 동일한 prompt를 쓴다.

2.3 Hard example에 대한 2단계 SFT

1단계 모델의 오답 중 64%는 일곱 개 view가 모두 틀린 경우였다. 집계로는 개선할 수 없고 학습으로만 좁혀지는 영역이므로, 학습 예산을 어려운 샘플로 재배분하는 짧은 2단계를 두었다.

Train split 9,058개 중 1,132개를 hard example로 선별했다. 난이도 판정에는 1단계와 별개로 팀이 앞서 학습해 둔 체크포인트를 사용해, 각 샘플을 세 개의 view로 예측하고 두 개 이상 틀린 것을 골랐다. 이 체크포인트는 난이도 판정에만 쓰이고 loss와 추론 경로에는 등장하지 않으므로, 재현에 필요한 것은 판정에 쓴 모델이 아니라 판정 결과인 목록이며 확정된 목록을 제출물에 포함했다. 선별에는 train split만 사용했고 validation·test 데이터는 쓰지 않았다.

샘플마다 서로 다른 두 개의 shuffle view를 만들고 두 view 모두 제공된 정답 permutation을 target으로 삼는다. Loss는 두 view의 negative log-likelihood 평균이며 오답을 밀어내는 항은 두지 않았다. Margin ranking loss를 쓴 구성이 검증에서만 이기고 리더보드에서 떨어졌기 때문이다(4.3절). Adapter를 새로 얹지 않고 1단계 adapter를 이어서 학습하므로 최종 산출물은 끝까지 단일 adapter다.

2.4 학습 설정

항목	1단계	2단계
양자화	NF4 4비트, double quantization, BF16 compute	동일
LoRA	r=16, α=32, dropout 0.05, language projection만	동일 (이어서 학습)
학습 파라미터	116.7M (0.4249%), vision encoder freeze	동일
학습량 / Learning rate	3 epoch, batch 8 / 1e-4 cosine	142 step / 1e-6 cosine
소요 시간	H100 1장 약 16시간 20분	H100 1장 약 1시간 38분

Loss는 정답 토큰에만 적용하고 image token과 prompt, padding은 masking한다. 1단계의 epoch별 validation 점수는 242 → 278 → 276으로 3 epoch에서 하락했으므로, cosine schedule을 끝까지 실행한 뒤 epoch 2 체크포인트를 선택했다.

3. 추론

각 테스트 샘플에 대해 고정 seed로 identity 하나와 서로 다른 shuffle 여섯 개, 모두 일곱 개의 view를 만든다. 각 view에서 permutation을 greedy decoding한 뒤 shuffle된 좌표에서 원래 좌표로 역변환하고, 24개 permutation 각각에 대해 위치별 투표 점수를 합산해 최대인 것을 고른다. 동점일 때는 완전한 permutation의 생성 빈도, identity view의 답, 사전순 순서로 해소한다.

마지막 집계가 1절의 mode 추정에 해당한다. View별 답을 다수결하면 어느 한 view가 실제로 생성한 답 중에서만 고를 수 있지만, 위치별 투표를 permutation 공간에서 다시 집계하면 어떤 view도 단독으로 내놓지 않은 permutation이 최종 답이 될 수 있다. 여러 view가 각기 다른 위치를 맞힌 경우를 살려내는 것이 이 규칙의 목적이다. 최종 추론은 이 단일 adapter의 test-time augmentation이며 Kemeny 집계나 likelihood reranking, 별도 verifier는 쓰지 않았다.

4. 실험 결과와 분석

4.1 Augmentation 구성

Qwen3.6-27B를 고정하고 augmentation 구성만 바꾸어 비교했다. Direct는 TTA 없이 greedy decoding한 결과이며 각 구성의 best epoch를 사용했다.

구성	Direct	TTA	Public LB (환산)
Augmentation 없음	273/477	281/477	—
Shuffle만	—	289/477	0.93193 (534/573)
Shuffle + 문장 변형	277/477	—	—
Shuffle + 문장 변형 + pair task	274/477	—	—
Shuffle + 문장 변형 + prompt 개선	**278/477**	287/477	**0.94240 (540/573)**

문장 변형과 prompt 개선은 기준선 대비 direct 5문제, TTA 6문제를 개선한다. 반면 보조 task를 추가하는 방향은 효과가 없었다. 인접 쌍의 순서를 묻는 pair task를 15% 섞은 구성은 오히려 3문제 낮다. 이후 verifier 계열 시도가 모두 실패한 것과 같은 경향이다(4.3절).

이 표의 핵심은 **로컬 검증과 리더보드의 방향이 어긋난다**는 점이다. 검증에서는 shuffle만 쓴 구성이 289로 가장 높고 문장 변형을 더한 구성이 287로 두 문제 낮지만, Public 리더보드에서는 여섯 문제 높다. 477개 validation set은 이 augmentation의 일반화 이득을 관측하지 못하며, 채택 근거는 2.2절의 통제 실험이다.

4.2 TTA와 error analysis

최종 adapter를 고정하고 사용하는 view 수를 바꾸어 재집계하면 1개 272, 3개 280, 5개 284, 6~7개 288문제로 개선된다. **view 하나에서 일곱 개로 늘릴 때의 +16문제가 단일 요소로 가장 큰 이득이다.** 6개와 7개가 동점이므로 사전에 정한 tie-break 기준인 평균 위치 정확도에 따라 7개를 택했다.

이 이득이 shuffle augmentation을 전제하지는 않는다. Shuffle 없이 학습한 Qwen3-VL-8B에서도 TTA는 리더보드 0.82897에서 0.86038로 개선되었고, augmentation 없이 학습한 27B에서도 273에서 281로 올랐다. 두 기법은 각각 독립적으로 기여한다.

오답의 구조에서 세 가지가 확인된다. View 간 일치도는 신뢰도 신호로 쓸 수 있다. 일곱 개가 만장일치면 92%가 정답인 반면 서로 다른 답이 네 개 이상 나오면 정답률이 10% 이하였다. 그러나 오답의 64%는 모든 view가 틀린 경우여서 집계 개선의 상한은 약 43문제에 불과하다. 2.3절의 2단계를 집계가 아니라 학습 단계에 둔 근거가 이것이다. 마지막으로 학습 데이터에서 정답이 identity인 샘플과 순서를 정할 수 없음을 나타내는 컬럼이 1,478건 전부 일치했다. 일부 샘플은 애초에 유일한 정답을 갖지 않는다는 뜻이지만 테스트 데이터에는 이 컬럼이 없어 활용할 수 없었다.

4.3 측정 한계와 기각한 접근

Validation set 477개에서 Exact Match 60% 부근의 표준오차는 약 11문제다. 따라서 1~3문제 차이는 노이즈로 간주하는 채택 기준을 사전에 고정했고, 이 기준이 세 차례 검증되었다.

접근	로컬 검증	리더보드	판정
24 permutation likelihood reranking	+7문제	하락	기각
Hard example + margin ranking loss	+3문제	0.93019 (기존 0.93193 미달)	기각
학습된 Yes/No verifier로 24후보 재채점	−4문제	미제출	기각

이 밖에 입력 해상도를 두 배로 올린 실험은 유의한 차이를 만들지 못했고, view 불일치가 클 때 identity로 전환하는 규칙은 모델을 바꾸면 재현되지 않아 폐기했다. 두 번째 항목은 2.3절 구성의 근거다. 오답을 밀어내는 loss는 검증에서 개선되고 리더보드에서 하락했으므로, hard example이라는 부분집합은 유지하고 loss에서 negative 항을 제거했다.

같은 한계가 최종 점수에서도 확인된다. 2단계를 포함한 최종 제출은 Public 0.94415(541/573) · Private 0.93902(231/246)이고 1단계까지만 쓴 제출은 Public 0.94240(540/573) · Private 0.94308(232/246)으로, 두 제출의 차이는 양쪽 모두 한 문제다. Private가 246문제이므로 한 문제가 0.4%p에 해당한다. 이 구간의 점수 차이는 방법의 우열로 해석하지 않는다.

5. 효율성과 비용

24GB 제약을 만족시키기 위한 선택은 세 가지다. 27B를 NF4 4비트로 양자화해 peak memory 17.97 GiB에 적재한 것이 이 규모의 백본을 제약 안으로 가져온 핵심 수단이고, LoRA를 language projection에만 적용해 학습 파라미터를 0.42%로 제한했으며, chain-of-thought를 쓰지 않아 출력이 32토큰에 머문다.

항목	실측
테스트 819샘플 × 7 view = 5,733회 생성	6시간 36분
샘플당 / view당 지연 시간	29.62초 / 4.23초
학습 합계	H100 1장, 약 18 GPU-hour
데이터 전처리	없음 (augmentation은 학습 중 수행)
외부 상용 API	0원

추론은 24시간 제한의 27.5%만 사용하며 형식 오류나 fallback은 발생하지 않았다.

지연 시간은 추가로 단축할 수 있다. 4.2절의 일치도 신호를 이용해 상위 두 permutation의 투표 점수 차이가 충분히 벌어지면 생성을 조기 종료하는 방식으로, 동일한 Exact Match를 평균 4.60개 view로 달성했다(계산량 42.5% 절감). 최종 제출에서는 구현의 단순성과 재현성을 위해 고정된 일곱 개 view를 사용했으나, 지연 시간이 중요한 환경에서는 정확도 손실 없이 약 1.6배 빠르게 동작할 수 있다. 학습 측면에서도 분산 학습이 필요하지 않아 H100 한 장으로 하루 미만이면 base 모델에서 최종 adapter까지 재현할 수 있다.

6. 장점과 한계

이 접근의 장점은 각 구성 요소의 근거와 효과를 분리해 확인했다는 점이다. 백본은 동일 조건에서 비교해 선택했고, augmentation은 통제 실험으로 필요성을 확인한 뒤 설계했으며, TTA의 이득이 shuffle augmentation과 독립적으로 성립함도 별도로 검증했다. 최종 구성이 단일 base 모델과 단일 adapter로 단순해 검증과 배포가 쉽고, 학습 18 GPU-hour와 추론 6.6시간이라는 비용도 낮다.

한계는 세 가지다. 첫째, 시각 정보만으로 순서를 판단하는 능력은 여전히 약하다. 문장 변형으로 언어 shortcut 의존을 줄였지만 근본적인 해결은 아니며, 문장을 제거하면 정확도가 21.8%에 머물고 연결어가 없는 구간은 33.3%다. Vision encoder를 freeze한 선택의 대가이기도 하며, 프레임 사이의 상태 변화를 직접 표현하도록 학습하는 것이 다음 과제다.

둘째, 2단계의 기여는 확인되지 않았다. Hard example SFT의 효과는 로컬 검증과 Public에서 각각 한 문제 개선, Private에서 한 문제 하락으로 모두 4.3절에서 정의한 측정 한계 안에 있다. 최종 모델에 포함했으나 이 단계가 개선을 만들었다고 주장하지 않는다. 4.3절의 채택 기준을 다른 접근에 적용한 것과 동일하게 이 단계에도 적용한 결과다.

셋째, TTA에는 상한이 있고 정답이 유일하지 않은 샘플을 다루지 못했다. 오답의 64%는 모든 view가 틀린 경우여서 view를 더 늘려도 회복되지 않으며, 4.2절에서 확인했듯 일부 샘플은 순서 자체가 결정되지 않지만 테스트 데이터에서는 이를 식별할 수 없다.

7. 재현성

상대 경로만 쓰는 독립 실행형 패키지로 제출한다. 개인 경로나 job scheduler에 의존하지 않고 학습과 추론 모두 오프라인에서 실행되며, 두 학습 단계와 검증, 최종 추론이 각각 하나의 스크립트에 대응한다.

제출 전 검사 스크립트가 인코딩, hard example 목록의 해시와 좌표 복원, 고정 split 크기, 목록과 validation·test 데이터의 교집합이 없음, 모델 해시와 총 용량, 집계 함수의 회귀 테스트를 일괄 확인한다. 추론 시작 시 모델과 입력 데이터의 해시를 기록해 두므로 설정이 바뀌면 중단된 작업의 재개를 거부하고, 결과 저장 후 819개 샘플의 순서와 답의 유효성을 다시 검사한다.

실행 환경은 Ubuntu 22.04.5, Python 3.11.15, PyTorch 2.6.0+cu124, Transformers 5.14.0.dev0, PEFT 0.19.1, bitsandbytes 0.49.2이며, 상세한 실행 방법과 가중치 배치는 저장소 README에 정리했다. 제출물 전체 용량은 base 모델과 adapter를 합쳐 약 56.5GB다.