SNU AI Challenge 2026 · Final Report

Reordering Shuffled Frames with a Reinforcement-Pretrained VLM

A 7B RL-post-trained checkpoint beat our best 32B model by +0.073 exact match. This report explains why, and shows every experiment that failed on the way there.

YOO YEONGJUN (solo) | Final submission: Public 0.90052 / Private 0.89837 | 2026. 07. 28.

1. Problem framing: the metric dictates the architecture

One caption, four shuffled frames, 24 possible orderings, and a score that pays nothing for being almost right.

Exact Match, not visual difficulty, was the first thing we designed against. The natural engineering instinct is to decompose the problem: train a pairwise comparator for "which frame comes first" and reassemble the ordering from six pairwise votes. We built it and measured it. The comparator reached roughly **81% pairwise accuracy**, a healthy model, yet its Exact Match was **0.290** against **0.420** for a holistic single-shot model trained under identical conditions. The cause is compounding: EM requires all six pairwise relations to be simultaneously correct, and $0.81^6 \approx 0.28$, which matches the measurement almost exactly.

This is the central design argument of our submission. **Under an all-or-nothing metric, decomposition converts local competence into global failure.** Every later decision followed from it: one model, one forward pass, one permutation emitted directly, and all capacity spent on making that single joint decision better rather than on post-hoc assembly or reranking.

The second framing decision was measurement discipline. Three reruns of an identical probe configuration returned 0.3667 / 0.41 / 0.37 (mean 0.382, **σ ≈ 0.02**). A candidate that had looked like a +0.053 winner was, at that spread, a 1.6σ event. We also found a truncation bug that had silently made four supposedly different probe arms train on the same 2,000 samples. From then on we **pre-registered a decision threshold before every launch** and accepted only paired A/B runs with matched sample budgets, seeds and hardware. Several negative results in Section 4 exist only because that rule refused to let us keep a flattering single observation.

2. Method

2.1 Permutation augmentation with synchronised label transport

Each training sample is presented in a fresh random order every epoch, and the target permutation is transported through the same shuffle so that presentation and label stay exactly consistent. One sample therefore yields up to 24 distinct, fully correct training instances. This is our only data amplification: it needs no external data and no generative model, so it is compliant by construction, and the transport map was verified by exhaustive round-trip over all 24 permutations before any training run.

Its value is not simply "more data". The shuffle destroys any positional prior, so the model cannot learn that input 1 tends to come first, and the decision is forced onto caption-to-frame grounding. It also *subsumes* test-time augmentation: permutation-voting TTA gave **+0.04 on a non-augmented model and −0.027 on an augmented one**. The augmentation had already collected the invariance that TTA tries to buy back at inference time.

2.2 The decisive move: swap the base checkpoint, not the recipe

For three weeks we scaled the standard axis: Qwen2.5-VL 7B to 32B, QLoRA 4-bit, 768 tokens per frame, longer continuation runs. That ladder saturated. Continuation gains decayed by an order of magnitude across successive extensions (**+0.034 EM/epoch to +0.0033 EM/epoch**), inference-side tricks were exhausted, and preference optimisation gave nothing.

The breakthrough was a hypothesis about *what kind of checkpoint we were fine-tuning*. Every base we had used was a plain instruction-tuned release. Temporal ordering is a reasoning-shaped task, and the models that are good at reasoning-shaped tasks are the ones that received **reinforcement post-training at the factory**. We could not afford large-scale RL ourselves, and our own attempts are documented as null results below, but we could inherit it. We swapped in **MiMo-VL-7B-RL** (Xiaomi, MORL post-training, released before the eligibility cutoff, same Qwen2.5-VL architecture family so the pipeline transferred with a one-line change) and froze every other hyperparameter.

The result reframed the campaign: **a 7B model matched our best 32B model at 1.84 epochs (val EM 0.5633 versus 0.5633) and beat it by +0.073 at 3.8 epochs (val EM 0.6367)**, at roughly a quarter of the inference cost. The comparison is clean because only the base checkpoint changed. Our conclusion, and the finding we would most want to defend at the finals: **for a compute-constrained team the highest-leverage decision is which pretrained prior you start from, not how hard you fine-tune it.** The reinforcement learning we could not afford to perform was available for free as a choice of checkpoint.

One implementation detail mattered. Reasoning-post-trained checkpoints emit chain-of-thought before the answer, which breaks a naive first-bracket parser. We switched to a **last-bracket parser** with a 48-token generation budget, so early-training outputs that still carry reasoning traces are read correctly instead of being scored as parse failures.

2.3 Weight-space souping as a legal ensemble substitute

The rules forbid ensembles. Averaging *weights*, however, produces one set of parameters and one forward pass at inference, so it stays inside the rule while recovering part of the variance-reduction benefit. Averaging two independently continued checkpoints of the same lineage gave **val 0.5700 against 0.5533** for the best single member in the same environment, and **+0.0087 public** (0.88481 to 0.89354). We offer this as a general tool for rule-constrained competitions: when ensembling is banned, the weight space is often still open.

3. Pipeline and RTX 3090 constraint engineering

Inference must run offline on a single RTX 3090 (24 GB) within 24 hours. We verified fit by measurement rather than estimate: the 32B lineage ran at **21.8 GB peak VRAM** and **2.3 s/sample** in 4-bit NF4 at 768 tokens per frame, finishing the 819-sample test set in about 31 minutes, a 46× margin against the time limit. The final 7B model is far lighter still, which is the practical reason the base-model swap is a better answer than buying more parameters.

Two operational rules changed outcomes. First, a **submission gate**: a candidate was submitted only if its local validation exceeded the incumbent by a pre-set margin. We used 12 submissions in total, against 20 to 40 for comparable teams, deliberately under-fitting the public leaderboard. Second, **environment regeneration**: library version drift alone moves greedy-decoded validation by ±0.02, so the final model is always re-run in a freshly built environment, worth a measured +0.005.

We also fitted our own validation-to-leaderboard relation over seven submissions, **public ≈ 0.445 + 0.79 × val**, accurate to ±0.003. It let us convert a target rank into a required validation score before spending money on a run, and it is the reason several expensive ideas were never launched.

4. Ablations and error analysis

All comparisons are paired: same seed, same split, same sample budget, same hardware. Rejections are reported at the pre-registered threshold, not chosen after the fact.

Axis tested	Result	What it taught us
Base checkpoint: RL-post-trained vs instruction-tuned	**7B RL 0.6367 vs 32B instruct 0.5633**	**Decisive: +0.073 with 4× fewer parameters**
Permutation augmentation with label transport	adopted, +0.026 public at 7B	Removes positional prior, subsumes TTA
Weight soup of two checkpoints	0.5700 vs 0.5533, +0.0087 public	Legal single-model substitute for ensembling
Pairwise decomposition and reassembly	EM 0.290 vs 0.420	EM compounds errors, $0.81^6 \approx 0.28$

Input resolution 256 to 768 tok/frame	0.3767 to 0.4200	Visual detail is the binding constraint
Learning-rate decay schedule	0.440 vs 0.503	Low LR underfits this task
LoRA r32, 3 epochs (7B)	val tie, public regression	Longer is not better, it overfits
TTA by permutation voting	+0.04 unaugmented, −0.027 augmented	Augmentation already captured the gain
24-permutation log-likelihood rescoring	0.5400 vs greedy 0.5633	Self-likelihood cannot re-rank its own errors
Chain-of-thought self-correction	9 fixed vs 14 broken	Self-verification destroys more than it repairs
Pair-swap targeted augmentation	0.2833 vs 0.4200	Distorting the presentation distribution hurts
Vision-attention LoRA (paired, 1466 samples each)	0.3567 vs 0.3567, **Δ = 0**	No signal on that axis at 7B scale
Model-family swap to InternVL3-38B	0.09	Family swap alone does not transfer at probe scale
Caption dropout 0.50	0.39 vs 0.42	Response curve turns down above 0.30
Video mode with M-RoPE temporal encoding	0.3467 vs 0.382 image-list	Temporal embeddings gave no advantage here
DPO preference optimisation, 3 runs	0.500 / 0.510 / 0.5133 vs 0.510 SFT	Margins learned, EM ranking unchanged
GRPO on-policy RL, 2 runs	0.5133 vs 0.510	Rollouts collapse near SFT init, reward variance ≈ 0
bf16 32B, de-quantised	0.4567	Numerics hypothesis rejected (undertrained confound noted)

The DPO and GRPO rows are the most instructive. Our own attempts to add reinforcement learning produced nothing at our scale: five separate runs, all inside the noise band, with DPO loss visibly decreasing while EM stayed flat. The preference margin was learned but did not transfer to exact-match ranking. The same capability, obtained as a *pretrained* RL checkpoint, delivered the single largest gain of the campaign. The contrast is the honest lesson: at small scale, buy RL from the checkpoint instead of trying to reproduce it.

Error analysis. Dumping per-sample log-probabilities over all 24 permutations on a 300-sample validation split showed the gold ordering was the model's top-1 in **54.0%** of cases and inside its **top-3 in 73.0%**; only 22.7% were true misses. Yet rescoring by the model's own likelihood recovered nothing (0.5400 vs 0.5467), which says the model is *confidently* wrong rather than uncertain. The missing ingredient is visual discrimination, not decision policy.

The residual errors are structured. **58% of errors (56 of 97) are a single adjacent pair swapped**, a fine temporal resolution failure rather than global disorientation; complete reversals numbered three, and all three had short captions. The dominant covariate is caption length: **EM 0.276 for captions of 15 words or fewer, against 0.699 for 26 to 40 words**. Short captions carry too little ordering evidence and the model falls back on weak visual cues. We also under-predict the already-ordered class (9% predicted against 12.5% actual), and offline sweeps over margin-based fallbacks to the identity permutation recovered only +0.007, inside noise.

One correction we owe the record. We initially believed the vision encoder was untrained. An adapter autopsy showed that PEFT suffix matching had in fact attached LoRA to the vision MLP blocks (192 visual keys, all 96 lora_B matrices non-zero), so vision MLPs *were* trained throughout; only vision attention and the merger were excluded. The paired experiment on that remaining axis returned an exact tie.

5. Strengths, limitations, and generalisation

Strengths. A single model with a single forward pass, selected under a validation gate and with deliberately few submissions, is a design biased against public-leaderboard overfitting; our public-to-private movement was −0.002, consistent with that intent. The method is cheap to reproduce: the final model is 7B with a 195 MB LoRA adapter. Every claim above is a paired measurement against a pre-registered threshold, including the ones that went against us.

Limitations, stated plainly. Vision attention and the merger were never trained. We never ran large-scale RL ourselves and our small-scale attempts failed. Our 300-sample validation set cannot resolve public-score differences below ±0.01, which twice caused us to gate out a model that later scored higher; we knowingly kept the gate as a catastrophe filter

rather than a fine selector. Total spend was about USD 176 of rented GPU time, and several axes were closed by budget rather than by evidence.

Expected behaviour out of distribution. The error analysis predicts our weak point precisely, and we would rather state it in advance than explain it afterwards. Performance is governed by caption informativeness. On the training distribution, long camera-motion prose with a median of 26 words, the model is strong. On short templated action captions the same model scores 0.276 in our own measurements. An evaluation set built from terse action labels rather than descriptive prose should therefore be expected to sit far below our leaderboard number, not because the ordering ability disappeared but because the text side of the grounding signal is much thinner. The generalisable contribution is the recipe, permutation augmentation with label transport, an RL-post-trained base, and weight-space souping under an ensemble ban, rather than the specific score.

If we continued. The 73% top-3 containment says the answers are nearly in reach and what is missing is finer visual discrimination. We would spend the next unit of compute on vision-side capacity starting from an RL-post-trained base, the one combination our budget never let us test.

Commercial API usage: none. Training, inference and preprocessing were performed entirely on self-rented GPUs at KRW 0 of commercial API spend. Code, environment specification and model weights: https://github.com/yeongjunyoo/snu-ai-2026