Structured Permutation Generation and Constrained Scoring for Caption-Guided Four-Frame Ordering

Epoch1
Taehyun Jung, Gyeongmin Kang
SNU AI Challenge 2026

Abstract

We address temporal reconstruction of four shuffled images conditioned on a caption. The label space contains only $4! = 24$ permutations, yet direct 24-way classification hides the compositional structure of the answer, while unconstrained text generation can produce invalid permutations. Our final system combines their advantages. A 4-bit Qwen3.5-9B is adapted with LoRA to generate the rank string $[r_1, r_2, r_3, r_4]$. Training uses label-preserving input permutation augmentation and digit-error-aware reweighting computed from the four rank positions. At inference, we do not sample text: all 24 valid strings are teacher-forced, scored at their informative digit positions, remapped across eight input-order views, and averaged. The final single-adapter system achieves 57.65% exact match (550/954) on a fixed development split, 91.099 public, and 90.243 private. It trains 29.10M LoRA parameters, stores a 111.04 MiB adapter, and uses 16.70GB peak reserved memory in a 24GB-constrained benchmark. The same 20-row benchmark measures 64.3 seconds per row, corresponding to a 14.6-hour linear projection for 819 rows rather than a completed end-to-end timing run. Across the reported studies, preserving full four-frame context and learning the structured output is more reliable than scene graphs, reduced-context judges, caption rewriting, or post-hoc visual reranking.

1 Introduction

Given a caption c and four shuffled frames $X = (x_1, \ldots, x_4)$, the task predicts

$$y = (r_1, r_2, r_3, r_4) \in \mathcal{S}_4, \tag{1}$$

where r_i is the chronological position of *Input_i*. Frame-order prediction has long served as a self-supervised objective [1], but here it is the evaluated task itself under an exact-match criterion, which is unforgiving: one adjacent swap makes the entire answer incorrect. The task is further complicated by near-duplicate frames, captions without explicit transition words, and accidental dependence on the presented input slots.

A natural baseline maps the 24 permutations to arbitrary letters A–X. It is fast, but a letter contains no information about which input rank is wrong. Pure free generation has the opposite problem: its output is structured, but parsing and validity are not guaranteed [2]. Our central design choice is to separate the *training representation* from the *decision constraint*. The model learns the meaningful permutation string, whereas inference performs an exhaustive likelihood comparison over only valid strings.

Our contributions are threefold:

- We formulate four-frame ordering as structured permutation generation and introduce digit-error-aware reweighting while retaining exact-match evaluation.
- We combine exhaustive digit-only candidate scoring with score remapping over eight input permutations, guaranteeing a valid output while reducing sensitivity to the presented input slots.
- We provide a broad, failure-aware ablation analysis. It identifies short and implicit captions as the clearest remaining difficulty and shows why several intuitively attractive auxiliary modules fail to transfer to hidden data.

2 Final Method

2.1 Input, prompt, and efficient adaptation

The final input contains the original caption followed by four images in the order *Input_1* through *Input_4*. A deterministic caption profile is appended: captions containing predefined sequence cues (e.g., "then," "before," or "finally") are marked `EXPLICIT_SEQUENCE`; all others are marked `IMPLICIT_VISUAL_ORDER`. This profile uses only the provided caption and is identical at training and inference. The system instruction requests exactly one permutation in the form `[a, b, c, d]`, using each digit once.

We use Qwen3.5-9B [3] with 4-bit NF4 double quantization and BF16 computation [4]. The visual encoder remains frozen. LoRA [5] is applied only to language-model `q/k/v/o_proj` and `gate/up/down_proj`, with rank 16, scale 32, and dropout 0.05. The final adapter contains 29,097,984 trainable LoRA parameters (111.04 MiB). We report the absolute trainable count rather than interpreting the total-parameter denominator printed by the quantized wrapper, whose accounting depends on the loaded representation.

2.2 Label-preserving permutation augmentation

Input slots are shuffled deterministically as a function of sample ID, epoch, and seed. If view permutation σ places original frame $\sigma(j)$ in new slot j, the target is remapped as

$$y^{\sigma}_j = y_{\sigma(j)}. \tag{2}$$

Each source example therefore supplies a different, exactly labeled slot arrangement across epochs. This directly teaches

the equivariance required by the challenge without generating images, captions, or pseudo-labels.

2.3 Structured generation and digit-error-aware loss

With teacher forcing, the model learns the complete answer string

$$s(y) = \texttt{[}r_1\texttt{, }r_2\texttt{, }r_3\texttt{, }r_4\texttt{]}. \tag{3}$$

Prompt tokens are masked, and ordinary next-token cross entropy is averaged over answer tokens to obtain $\mathcal{L}_{\text{gen}}^{(i)}$. To emphasize examples with more rank-position errors, we inspect the teacher-forced argmax at the four digit positions. Let $m_i \in [0, 4]$ be the number of correctly predicted rank digits. The discrete count sets only a per-example scalar weight and is detached from gradient computation. The batch loss is

$$\mathcal{L}_{\text{train}} = \frac{1}{B}\sum_{i=1}^{B} \underbrace{[1 + \alpha(4 - m_i)]}_{w_i} \mathcal{L}_{\text{gen}}^{(i)}, \quad \alpha = 0.25. \tag{4}$$

Thus a fully correct example has weight 1, while an example with no correct rank digit has weight 2. This Hamming-style reweighting does not change the exact-match evaluation target; it only concentrates optimization on examples currently wrong at more rank positions.

2.4 Exhaustive valid-candidate scoring

The final decoder never calls open-ended `model.generate`. For every $y \in \mathcal{S}_4$, it teacher-forces $s(y)$ and computes

$$E(y) = -\sum_{t \in \mathcal{D}} \log p_\theta(s_t(y) \mid c, X, s_{<t}(y)), \tag{5}$$

where $\mathcal{D}$ contains exactly the four digit positions. We intentionally restrict scoring to the rank-token positions to reduce sensitivity to syntax-token calibration. This is a scoring choice: shared surface tokens are not assumed to have candidate-independent conditional likelihoods. The normalized candidate score is

$$q(y \mid c, X) = \frac{\exp[-E(y)]}{\sum_{z \in \mathcal{S}_4} \exp[-E(z)]}. \tag{6}$$

This hybrid retains the structured supervision of generation while guaranteeing that every prediction is a valid permutation.

2.5 Eight-view input-order TTA

We use the identity order plus seven fixed deterministic input permutations defined in the released inference configuration. For view v, the 24-way candidate-score vector is mapped back to the original slots with $R_{\sigma_v}^{-1}$ and averaged:

$$\bar{q}(y) = \frac{1}{8}\sum_{v=1}^{8} R_{\sigma_v}^{-1} q_v(y), \qquad \hat{y} = \arg\max_{y \in \mathcal{S}_4} \bar{q}(y). \tag{7}$$

The aggregation is a simple arithmetic mean. All views share one backbone and one adapter; TTA promotes input-order consistency rather than combining separately trained models. Because only eight of the 24 possible slot orders are averaged, Eq. 7 reduces slot sensitivity but does not constitute a proof of full $\mathcal{S}_4$ equivariance.

3 Training and Deployment

Data and selection. The challenge provides 9,535 labeled rows and 819 test rows. We use a fixed, answer-stratified split with 8,581 training IDs and 954 development IDs. A fixed 200-row subset is used for economical per-epoch checkpoint screening with single-view candidate scoring. After freezing epoch 3, we evaluate the final eight-view TTA pipeline on all 954 development rows, obtaining 550/954 = 57.65%. Under the final inference pipeline, the screening subset obtains 114/200 = 57.0%, while the 754-row complement not used for epoch selection obtains 436/754 = 57.82%. The headline development score is therefore selection-inclusive. No development image is used for gradient updates, and the final run uses no external API, generated rationale or STaR-derived data, scene-graph cache, ORB feature, or model-rewritten caption; the only caption-side preprocessing is the deterministic rule-based profile of Section 2.1.

Optimization. We train for up to five epochs using AdamW, learning rate 2×10^{-4}, cosine decay, 3% warm-up, weight decay 0.01, maximum gradient norm 1, batch size 4, and four-step gradient accumulation (effective batch 16). Early stopping uses patience 2 and minimum improvement 0.002. Epoch 3 is selected solely from the fixed 200-row single-view screening run: it obtains the highest exact match (113/200=56.5%) and the lowest true-answer NLL (0.0875). Continuing to epochs 4–5 lowers training loss from 0.0921 to 0.0333 but raises validation NLL to 0.1303 and fails to improve exact match, providing a direct overfitting signal. One RTX PRO 6000 Blackwell requires 44,701 seconds (12 h 25 min).

Implementation safeguards. Digit positions are discovered rather than hard-coded. At startup, the tokenizer compares two equal-length reference answers that differ in all four ranks, verifies exactly four differing token locations, and checks that digits 1–4 are single tokens. The run aborts if these assumptions fail. Submission construction independently checks ID order against the provided template, null values, and permutation validity. Together with ID- and epoch-derived augmentation seeds, these checks make the scoring path deterministic and fail-fast under tokenizer or data changes. The UTF-8 source release provides separate training and inference entrypoints, pinned dependencies, relative-path commands, and the Git-LFS adapter at `https://github.com/mynameistaehyun/Snuaichallenge-Epoch1-`.

Inference and resources. The inference path uses one backbone and one adapter. Rows are independent and may be sharded for throughput, but no model ensemble or cross-row state is used; the measured configuration fits within a 24GB memory limit. With candidate batch 1, eight views require $24 \times 8 = 192$ fixed candidate scores per row. A 20-row benchmark on an RTX PRO 6000 Blackwell, with memory usage constrained below 24 GB, peaks at 11.59 GB allocated and 16.70 GB reserved at 64.3 seconds per row. Linear scaling gives an engineering projection of approximately 14.6 hours for 819 rows. Table 1 reports this projection, not a completed single-device end-to-end timing measurement. The submitted implementation also disables the unused KV cache and converts only the four selected digit-logit rows to FP32, rather than upcasting the full sequence-

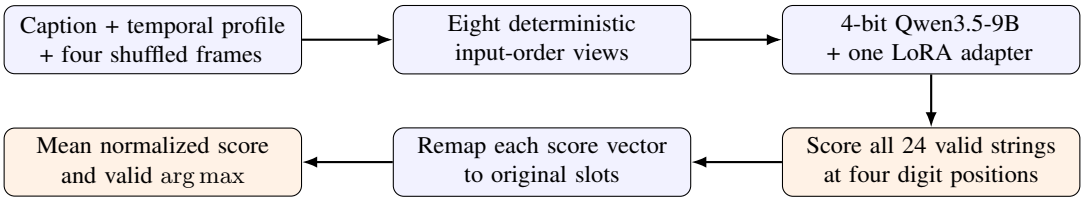

Figure 1: End-to-end pipeline of the final system. Training supervises structured rank strings; inference exhaustively scores valid candidates and reduces input-order sensitivity through remapping and averaging.

Table 1: Final-system efficiency and construction cost.

Item	Final configuration	Cost/result
Preprocessing	slot remap + caption profile	CPU, no API
Adaptation	4-bit LoRA, one GPU	12 h 25 min
Trainable weights	29.10M LoRA	111.04 MiB
Model package	base + adapter + code	approx. 19.5 GB ($<$ 80 GB)
Inference memory	batch 1, TTA 8	16.70 GB reserved
Inference latency	192 scores/row	14.6 h projected on benchmark hardware

by-vocabulary tensor; both are prediction-preserving memory reductions. The 19.3GB public backbone plus the 111.04MiB adapter and repository assets total approximately 19.5GB.

4 Results and Ablation Study

4.1 Final performance and hidden evaluation

Table 2 compares the final generation system with the strongest classifier submissions for which both public and private scores are known. The final model gains 2.032 private points over entropy TTA and 1.219 over ORB reranking. Its public-to-private drop is 0.856 points, less than half the ORB drop and roughly one third of the entropy-TTA drop. This pattern is consistent with improved hidden-set transfer, but public/private differences alone cannot identify which component caused the gain.

We keep the evaluation roles explicit. A fixed 200-row subset selects the checkpoint using single-view candidate scoring. After epoch 3 is frozen, the final TTA8 pipeline obtains 114/200 on those screening rows and 436/754 on the complement not used for epoch selection, totaling 550/954. On the 754-row complement, the entropy-TTA classifier obtains 411/754=54.51%. Generation alone is correct on 80 examples and the classifier alone on 55, giving a net gain of 25 answers; an exact two-sided McNemar test gives $p = .038$. On all 954 rows, the corresponding correct counts are 550 and 520. Controlled comparisons use the fixed development predictions; public and private scores are reported separately as external outcome measurements. Because development and leaderboard accuracy differ substantially, development is used for controlled comparisons rather than calibrated hidden-set estimation.

4.2 Training representation and data ablations

Appendix Table 4 reports the recorded outcomes of the training ablations. On the fixed 8B split, digit-list generation is 0.2 percentage points above A–X classification. For the final 9B run, the preserved 200-row epoch log permits an integer-count audit: epoch 3 is the joint optimum in exact match and true-answer NLL, while epochs 4–5 exhibit overfitting. The final

Table 2: Exact-match development and leaderboard results. "Drop" is private minus public (points).The development score includes the 200-row checkpoint-selection subset; the 754-row complement is reported in Section 4.1.

Submission path	Dev.	Public	Private	Drop
A–X, entropy TTA24	54.51	90.750	88.211	-2.539
A–X, ORB rerank	54.93	90.750	89.024	-1.726
Final constrained generation	**57.65**	**91.099**	**90.243**	$\mathbf{-0.856}$

550/954=57.65% is a separate full-development evaluation of that frozen epoch-3 checkpoint, not the metric used to select it. Historical summaries also list 56.8 for an A–X run and 57.6 for caption-shortening, but their exact denominators and matched prediction files are not preserved; we therefore exclude them from direct quantitative selection rather than presenting them as 200-row exact-match results.

Within the 8B classifier block, permutation augmentation $\times 2$ is 0.74 points below $\times 1$, and label smoothing is 0.21 points below $\times 1$ on the same fixed split. Caption shortening is not part of the final model because it lacks a matched hidden-set evaluation. Hard-error rehearsal retains 98.18% of previously correct examples but does not increase accuracy on the untouched 477-row subset. Contaminated, incomplete, or no-holdout runs remain excluded from performance claims.

4.3 Why auxiliary and post-hoc approaches failed

Appendix Table 5 summarizes every completed, decision-bearing auxiliary family. The failures follow three consistent patterns.

Context reduction loses chronology. Pairwise learning is above chance, but 64.46% is below the predefined 70% gate for safely overriding a 24-way model. Triplet and single-frame anchor methods are near their baselines. Concentrating visual attention cannot compensate for removing the other frames, which often provide the only reference state.

Unsupervised cues lack direction. ORB [6] identifies which pair is visually adjacent in 73.6% of applicable cases, but matching strength cannot determine which frame comes first. Scene graphs similarly describe entities well but over-interpret caption events as visible states. These cues are useful diagnostics, not reliable sequence labels.

Development-tuned corrections can fail to transfer. Epoch 4 is best under 24-view validation (55.24%) but scores only 89.877 public, below epoch 5 despite its lower 54.51% validation score. Entropy TTA and focal blending improve validation

Table 3: Fixed development split by caption length. A–X uses entropy TTA24; generation is the final checkpoint.

Caption words	Rows	A–X top-1	A–X top-8	Final gen.
≤ 10	108	10.19	53.70	**21.30**
11–20	225	24.89	71.56	**28.89**
21–35	519	71.48	93.06	**73.03**
≥ 36	102	80.39	97.06	**81.37**

or public slightly, yet entropy TTA drops to 88.211 private. Center crops, type-hint blending, and same-posterior reranking alter decisions without adding sufficiently independent evidence. The final model therefore uses eight-view *mean* aggregation and no post-hoc correction.

4.4 Error analysis: short and implicit captions

All cohort and qualitative analyses in this section use the labeled development split; no test-set prediction analysis is used. Table 3 explains where the final gain originates and what remains unsolved. Constrained generation improves every caption-length bucket, with the largest absolute gain ($+11.11$ points) for captions of ten words or fewer. Nevertheless, this cohort reaches only 21.30%. The classifier's correct answer lies outside its top eight in 46.3% of these cases, showing that a reranker restricted to those eight candidates cannot recover nearly half of them without new candidate mass.

On the same 954 development IDs, generation uniquely corrects 99 examples while regressing on 69, for a net gain of 30. The regression-to-improvement ratio is therefore $69/99 = 0.70$: the development comparison does not show uniform domination. The hidden scores are consistent with positive net transfer, but they do not establish a component-level causal explanation. Explicit-cue captions reach 69.04%, whereas implicit-order captions reach 32.20%.

Qualitatively, the hardest labeled development cases combine two properties: the caption names an action without describing its phase, and consecutive frames differ only in a small pose or object-state change. Camera cuts create the opposite failure mode by making temporally adjacent frames look visually distant. These observations explain why a direction-free similarity feature can detect continuity yet still fail as an ordering rule, but they are descriptive rather than a test-set-derived error taxonomy.

5 Discussion and Conclusion

Why the final approach is effective. The final design aligns the output geometry with the task. Each generated digit refers to one input frame; digit-error-aware weighting emphasizes examples wrong at more rank positions; exhaustive scoring guarantees validity; and permutation augmentation plus remapped TTA promote input-order consistency. Because the strongest classifier and the final generator do not form a fully matched backbone-level ablation, the evidence supports the combined system rather than attributing the gain to any single component.

Advantages. The system is deterministic, always parsable, and reproducible with a small adapter. It uses no external API or costly learned preprocessing, fits the measured 24GB-constrained configuration, and packages to approximately 19.5 GB including the public base checkpoint. The private score is 0.856 points below the public score, a pattern consistent with greater hidden-split stability, although the split comparison is not a causal ablation.

Limitations. Exhaustive inference is slow: 192 candidate scores are required per row, and the reported 14.6-hour figure is a linear projection from a 20-row benchmark rather than a completed single-device end-to-end timing run. Short and implicit captions remain empirically difficult for the current system, especially when adjacent frames are visually similar. The 200-row checkpoint screen is nested within the 954-row development split and is noisy; the epoch-4 reversal demonstrates that sub-point development gains require caution. The 0.70 development regression/improvement ratio further shows that the method redistributes errors rather than improving every cohort uniformly. Because only eight input orders are averaged, the method does not guarantee full permutation equivariance. Finally, matched generation-only ablations are not available for $\alpha = .25$ versus $\alpha = 0$, digit-only versus full-token scoring, or the temporal profile, so we attribute the result to the combined system rather than assigning a separate gain to any one of these components.

Conclusion. CONSTRAINED GENERATIVE PERMUTATION SCORING is our strongest verified configuration, combining a semantically meaningful training target with a strictly constrained decoder. The ablations show that adding plausible but weak evidence is less reliable than improving the learned structured representation. Future work should acquire genuinely directional supervision for short captions—for example through large-scale video pretraining or a single-adapter multi-task objective with pairwise supervision—while preserving full four-frame context and a single deployed adapter.

References

[1] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: Unsupervised learning using temporal order verification," in *European Conference on Computer Vision*, 2016.

[2] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *International Conference on Learning Representations*, 2020.

[3] Qwen Team, "Qwen3.5-9b model card." Hugging Face model card: Qwen/Qwen3.5-9B, released February 2026, 2026.

[4] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," in *Advances in Neural Information Processing Systems*, 2023.

[5] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *International Conference on Learning Representations*, 2022.

[6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *IEEE International Conference on Computer Vision*, 2011.

A Detailed Ablation Results

The following tables provide the complete training and diagnostic ablations summarized in Section 4.

Table 4: Training and data ablations. The final 9B checkpoint rows use the preserved 200-example epoch log and the fixed 954-example evaluation. Historical results without auditable denominators are omitted from direct comparison.

Block	Variant	Evaluation	Recorded result	Observed comparison
Qwen3-VL-8B objective	A–X restricted classification	fixed val	52.6%	0.2 pp below digit-list generation on the same split.
	digit-list generation	fixed val	52.8%	0.2 pp above A–X on the same split.
Qwen3-VL-8B classifier learning	permutation augmentation $\times 1$	fixed val, raw	52.73%	Reference run for this ablation block.
	permutation augmentation $\times 2$	fixed val	51.99%	-0.74 pp versus $\times 1$.
	label smoothing 0.05	fixed val	52.52%	-0.21 pp versus $\times 1$.
	focal loss $\gamma = 0.75$	TTA24 val	55.14%	Tuned blend: 56.29% validation; 90.0226 public.
	control strip	fixed val	53.25%	$+0.52$ pp versus the $\times 1$ raw reference.
	alternate seed	fixed val	53.04%	$+0.31$ pp versus the $\times 1$ raw reference.
Qwen3.5-9B final selection	epoch 1 / epoch 2	fixed selection 200, single-view	53.0 / 56.0%	Exact match and true-answer NLL improve through epoch 2.
	epoch 3: generation $\alpha = 0.25$ (final)	fixed selection 200, single-view	**113/200=56.5%**	**Highest exact and lowest NLL (0.0875); selected before test evaluation.**
	epoch 4 / epoch 5	fixed selection 200, single-view	53.0 / 55.0%	Train loss falls, but NLL rises to 0.1127 / 0.1303: overfitting.
	frozen epoch 3	fixed full 954, TTA8	**550/954=57.65%**	TTA8: 114/200 on the screening rows and 436/754 on the epoch-selection complement; classifier 411/754.
	hard-error rehearsal	untouched 477	55.1%	98.18% retention; no accuracy increase on the untouched subset.

Table 5: Auxiliary, prompt, and inference ablations. All are excluded from the final model. "Gate/result" reports the metric appropriate to each hypothesis; values across rows are not directly comparable.

Family	Method	Gate/result	Observed outcome
Structured text	cached scene graph (4,187 rows)	44.73% val	Partial coverage and caption leakage; lower validation than direct visual input.
	visual-cue / full graph input	51.68 / 52.83% val	Higher than cached graphs, with no matched leaderboard gain.
	caption expansion	22.50→18.75% (80 rows)	Exact match decreases by 3.75 points on the local subset.
Reduced visual context	frozen / LoRA pairwise judge	59.11 / 64.46% pair acc.	LoRA adds 5.35 points; both remain below the 70% override gate.
	three-frame ordering	17.34% vs. 16.67% random	0.67 points above random; four-frame reconstruction exact is 6.01%.
	single-image E/L/M anchor	42.32% vs. 50% majority	7.68 points below the majority baseline.
	short-anchor generation	45.27%	Below the 50% majority baseline used for the anchor task.
Prompt / reasoning	classifier zero-shot CoT	5.0% exact; 28.3% valid	23.3-point gap between valid-format and exact-match rates.
	classifier type-hint blend	87.958 public	Below the 90.750 public score of the entropy-TTA classifier.
	HOG verticality / pHash transfer	0.6% usable / no threshold	HOG applies to 0.6% of rows; pHash yields no accepted threshold.
Inference	epoch 4 vs. epoch 5, entropy24	55.24 vs. 54.51% val; 89.877 vs. 90.750 public	Epoch 4 is $+0.73$ val but -0.873 public relative to epoch 5.
	entropy-weighted TTA24	90.750 public; 88.211 private	Public-to-private change is -2.539 points.
	ORB posterior rerank	adjacency 73.6%; 90.750/89.024 pub./priv.	Public unchanged; private is $+0.813$ over entropy TTA.