A Test-Time Read-Out Head for Video Frame Ordering: Decoding the Answer from Hidden States Instead of Text

Team talkinglion

July 2026

Abstract

We restore the chronological order of four shuffled video frames given a caption. Our system lets a multitask-SFT'd Qwen3.5-9B reason about a sample under several of the 24 possible frame-presentation orders, then decodes the answer not from the model's text but from its hidden state. A small linear head reads layer 24 at the token just before the forced answer and scores all 24 orderings, its distributions are averaged across presentations, and the result is blended with the verbalized vote share. The motivation is a measurable **exploitable gap**. On a large fraction of views the correct ordering is decodable from the representation even when the emitted answer is wrong. That such gaps exist is well documented [1–3]. What we deliver is a read-out shaped to the output space, a 4×4 assignment matrix rather than a scalar score or a flat 24-way softmax, and convenient to attach: 65,552 parameters, two CPU-minutes on cached states, no change to the generator. It beats every verbalized read-out we could build from the same traces (plurality, Kemeny, Borda) at every view budget in both folds of a two-fold protocol, by about $+15$ points at one view narrowing to $+1.5$ to $+2.7$ at 24. It also composes with generator-side work rather than competing with it, making it an easy and cheap add-on for different generators utilizing CoT in VLMs.

1 Summary

1. **There is an exploitable gap** between what the reasoning encodes and what the answer line states, and it is large enough to build a decode on (§2).
2. **A read-out probe designed for permutation output space exploits it better** than a scalar score, a flat 24-way head, or any verbalized aggregation of the same traces (§3, §5).
3. **It is a convenient thing to attach at test time.** Two CPU-minutes on cached states, no generator change, and it gets better when the generator does (§4).

Setup. The backbone is **Qwen3.5-9B** (bf16, one 24 GB GPU), LoRA-fine-tuned for 900 steps (0.88 epoch) on a multitask suite: pairwise "is B later than A", listwise rank vector (the deployed task), and positional "which slot does frame k occupy". We also screened Qwen3-VL-8B and an int8-quantized InternVL-14B, and choose the model that performed best at its base state. Each training example gets a predetermined random presentation order, so the 24 rank-vector classes are uniform by construction. At inference we elicit a budgeted chain of thought, force a parseable answer, and cache one hidden vector per view.
Protocol. Everything below is measured under a **two-fold protocol** on the two labelled collections from train set (VAL, EVAL; 477 samples(5% of train set) each): fit on one, report on the other, then swap. Hyperparameters are chosen out-of-fold.

2 An exploitable gap

A short fixed-answer SFT already improves the reasoning itself. The SFT targets are terse rank vectors, so it is not obvious that it should help free-form reasoning at all. However, it does boost the performance suprisingly. The table below compares the two generators on 954 validation samples at a 1024-token thinking budget.

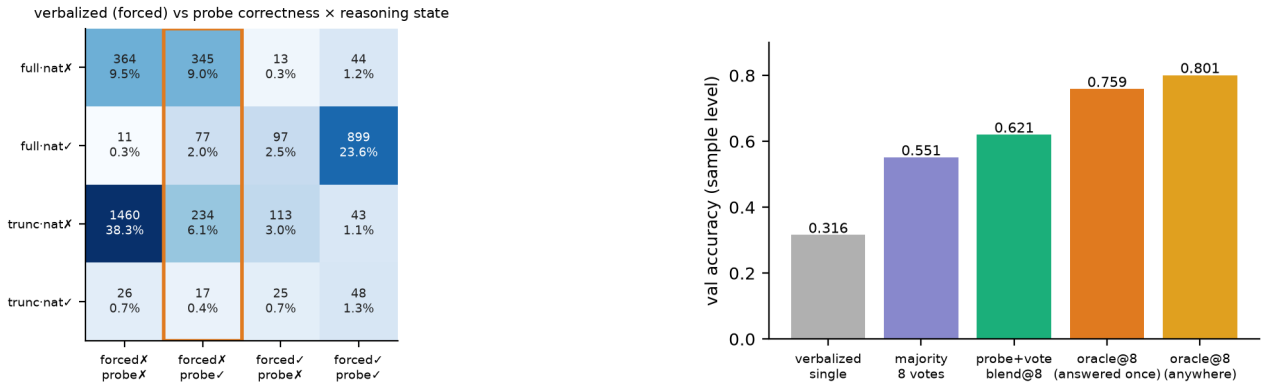

Figure 1: Left: 3,816 validation views crossed by reasoning state (full/truncated × natural answer correct) and decoding route (forced answer × head). The boxed column, forced answer wrong but head right, is 17.6% of all views against 6.5% the other way, a net $+11.1$ points available to a read-out. **Right:** the same gap at sample level: the gold ordering appears among the model's own 8 verbalized answers 75.9% of the time, and majority voting recovers 55.1%.

generator	CoT truncation	single-CoT EM	4-vote EM
base Qwen3.5-9B	96.0%	0.195	0.293
multitask SFT, 900 steps (0.88 **epoch)**	**56.0%**	**0.313**	**0.449**

Its chains close naturally almost half the time where the base model exhausts the budget in 96% of attempts, and it is about 12 EM points stronger at both width 1 and 4.
Even so, chain-of-thought made this task worse to read literally as the model is trained on short answering rather than reasoning. Budget-forced CoT verbalization scores 0.31 exact match against 0.53 for a terse single pass. Reasoning hurt what the model said, which is what sent us looking at what it encoded. Per view, the correct permutation is recoverable from the layer-24 state in 17.6% of cases where the forced answer is wrong, while the reverse costs 6.5%, a net $+11.1$ points for a decoder willing to read the representation (Fig. 1). Per sample, the model produces the gold ordering at least once across 8 presentations 75.9% of the time while majority voting extracts 55.1%. Therefore, we interpret that there is a significant gap between reasoning and final answer which may be exploitable via probing.

Related prior work. That models encode answers they fail to emit is established. Orgad et al. [1] show hidden states holding the correct answer while generation gets it wrong, Gekhman et al. [2] measure a ∼40% relative internal-versus-external gap, and Park et al. [3] show probes beating generations on multiple choice. Reasoning traces have been probed for self-verification, early exit and self-consistency reweighting [5], with CoT unfaithfulness documented separately [4]. Our setting differs mainly in the target. While prior work decodes a binary or few-way scalar, whereas an ordering is a structured object, which is what makes the shape of the head a design question.

3 Read-out head design

For every (sample, presentation) pair we cache the layer-24 residual-stream state $h \in \mathbb{R}^{4096}$ at the token immediately preceding the forced answer. The head is a single linear map $W : h \mapsto U \in \mathbb{R}^{4\times4}$, where $U_{k,t}$ scores frame k occupying chronological position t. An ordering π is scored by the assignment it implies, and the 24 orderings are normalized against each other:

$$u_\pi = \sum_{k=1}^{4} U_{k,\pi(k)}, \qquad P(\pi \mid h) = \mathrm{softmax}(u)_\pi,$$

fit with cross-entropy and L_2 (inverse strength C) by L-BFGS on z-scored features. Adding a constant to a row or a column of U shifts every u_π equally and so changes nothing, which leaves **nine free output directions** out of sixteen. Distributions of this form are standard in permutation learning. At $n{=}4$ the 24 terms enumerate exactly, so we fit by plain maximum

likelihood. The closest task-level precedent is DeepPermNet [7], which orders shuffled visual data with a Sinkhorn permutation head, but trained end-to-end inside a CNN.

The scalar-ranking head, and why it is not enough. The obvious first thing to try is one number per frame: map h to $(s_1, \ldots, s_4)$ and read the ordering off by sorting. This is the familiar listwise ranking model and it is attractively small, three free directions after its own shift symmetry. But it assumes each frame carries a single position-independent "lateness" score, so it cannot express that a frame is a plausible opener yet an implausible closer, which is exactly the kind of judgement these captions invite. The other extreme, a flat 24-way softmax, is free to express anything but shares no structure between orderings that differ by one swap. The assignment head sits between them:

read-out head (layer 24)	free dirs.	per-view acc	aggregated EM
scalar ranking ($h \to \mathbb{R}^4$, sort by score)	3	0.365	0.577
flat 24-way softmax ($h \to \mathbb{R}^{24}$)	23	0.425	0.631
assignment matrix ($h \to \mathbb{R}^{4\times4}$)	**9**	**0.447**	**0.635**

The ordering of these three is stable across layers 18/24/30 and across training pool sizes, and layer 24 of 32 is the reliable optimum (Fig. 2, right). Structured linear read-outs are not new. The structural probe of [8] is also linear and also carries a shift symmetry, though its structure is metric rather than combinatorial. To the best of our knowledge, a probe designed specifically for a permutation output space is new. We suggest it is cheap and effective precisely because it is built on that structure.

Averaging across presentation orders. The same sample can be shown in any of the $4! = 24$ frame orders. We map each view's distribution back to canonical frame indices and average, the usual way to make a predictor insensitive to presentation order. Using all 24 removes the dependence entirely and smaller budgets approximate it.

Decoding. The averaged distribution $\bar{P}$ is blended with the vote share v of the forced answers, $s = (1-\alpha)v + \alpha\bar{P}$, α=0.85 selected out-of-fold. Averaging in probability rather than logit space caps each view's influence at $1/k$, which matters because a truncated view can otherwise veto a decision. We prefer the blend base on a simple sweep on 2-fold OOF results.

4 Convenient deployment with improvement

(i) Nothing upstream has to change. The head consumes a frozen residual-stream vector and, optionally, the verbalized answer, never the generator's weights, prompt, sampling or token budget. One fitted head decoded views generated at a different thinking budget (4096), under repetition-penalized sampling (1.15), at every width from 8 to 24. However, it collapses with an aggressive penalty (1.3) or upon changing the answer format, both repaired by refitting.

(ii) It is cheap. $W \in \mathbb{R}^{4096\times16}$ plus bias is **65,552 parameters** (7×10^{-6} of the backbone), fit on cached states by a convex solve in **115–126 s on CPU alone**, with one hyperparameter. It is also label-thrifty: 477 labelled samples' worth of views already reach 0.635, within a point of our largest pools.

(iii) Fixed-answer SFT is what makes it work, and better reasoning should make it work better. The head reads one specific token, the last one before a forced, fixed-format answer. That position only carries a decodable summary because the SFT taught the model to emit a rank vector there, and the same SFT also supplies better states to read (§2). So generator-side work is highly orthogonal with our method. We can simply refit the head in two CPU-minutes to match the provided generator. The only hard requirement is white-box activations.

5 Probe beats verbalized read-outs at every budget

A fair worry is that the advantage is an artifact of one view budget. We swept the nested budget k against every pure read-out constructible from the same traces: plurality of the forced

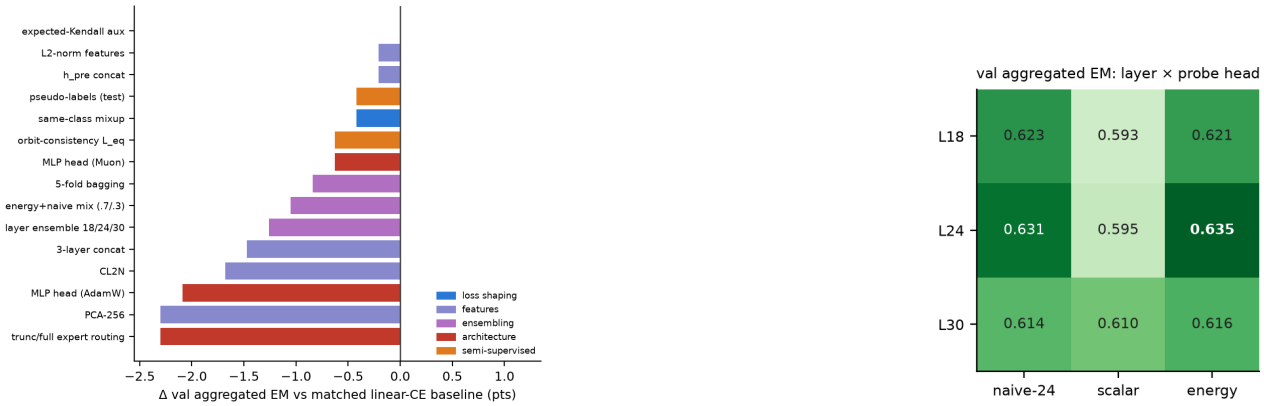

Figure 2: Left: every head-side intervention we evaluated, as the change in validation aggregated EM against its matched plain-CE baseline, under honest selection. Nothing clears the baseline. **Right:** the layer × head grid.

answers (self-consistency [6]), Kemeny/Kendall-median consensus, and Borda positional aggregation. Both folds, 477 samples:

	EVAL→VAL				VAL→EVAL (reverse)			
decode	k=1	8	16	24	k=1	8	16	24
pure verbalized read-outs								
plurality vote	0.375	0.543	0.598	0.608	0.384	0.518	0.566	0.591
Kemeny consensus	0.375	0.497	0.554	0.602	0.384	0.468	0.549	0.574
Borda positional	0.375	0.424	0.484	0.556	0.384	0.411	0.497	0.547
hidden-state head (same traces, same views)								
head only	0.507	0.610	**0.629**	0.631	0.512	**0.604**	0.581	0.600
blend (ours)	**0.528**	**0.621**	0.627	**0.635**	**0.520**	0.600	**0.581**	**0.606**
best head − best read-out	+15.3	+7.8	+3.1	+2.7	+13.6	+8.6	+1.5	+1.5

The head is ahead in all eight cells. At k=1 the three verbalized aggregators necessarily coincide and the margin is widest, $+15.3/+13.6$. Plurality first overtakes the one-view blend at k=8 in the forward fold and between k=8 and k=16 in the reverse, so one view read from the state is worth roughly eight verbalized ones; averaging over all 11,448 single views widens the gap further ($0.322/0.310$ against $0.501/0.495$). The margin narrows with budget, which we read as voting slowly recovering the same signal rather than the head running out of headroom, and which makes the component as much a **compute saving** as an accuracy gain. Accuracy is monotone in k for the blend ($0.6205/0.6247/0.6268/0.635$ at k=$8/12/16/20$) and flat from k=20 to 24, which lets the last block be conditional.We also note that the best terse-answer aggregation we tried reached only 0.614 on VAL. Reasoning with a head on top ends up ahead of the terse pipeline it started well behind.

6 Negative abltaion results

The negative results are worth as much to us as the positive result, because it says the work is being done by the shape of the head rather than by a stack of tricks. Kendall-aware label smoothing, auxiliary losses, whitening, PCA, mixup, bagging, layer and head ensembles, MLP heads, pseudo-labels and orbit-consistency regularizers all fail to beat plain CE with a correct C. We mainly hypothesize this is due to following reasons.

- **Two heads on one vector do not ensemble.** The flat 24-way head disagrees with the assignment head on a third of views, yet their errors correlate at 0.77 and it is exclusively correct on only 5%. Every mixing weight is flat-to-negative in both folds. The one combination that helps is across modalities: hidden state with verbalized vote.
- **Regularization is the only real knob.** The head memorizes its training views perfectly at weak L_2 while held-out accuracy collapses. C is load-bearing and mis-setting it by one grid step costs up to 2 points.

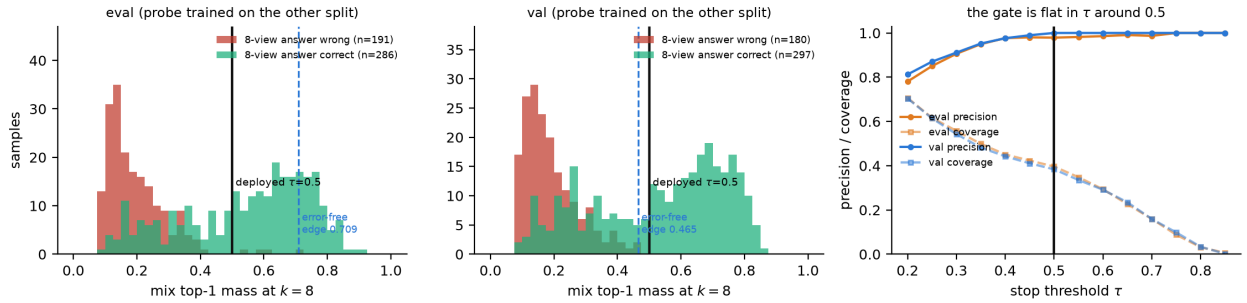

Figure 3: The gate statistic at $k{=}8$. **Left, centre:** its distribution on each labelled split (head fitted on the other), split by whether the 8-view answer is correct. The two populations barely overlap above ≈ 0.45; the black line is the deployed $\tau{=}0.5$, the dashed line the most permissive error-free threshold on that split, which lands at 0.709 and 0.465 and so does not transfer. **Right:** precision is flat near 1 over a wide plateau, which is why an a-priori τ costs nothing measurable.

7 Deployment summary

Because the head is near its asymptote early (§5) while the vote share is not, the blended top-1 mass is already a usable stopping statistic at $k{=}8$: AUROC $0.914/0.917$ across folds. The decode buys views in blocks of eight and escalates only while the mix stays below an a-priori 0.5 (Fig. 3). This is the confidence-gated form of adaptive self-consistency [6], gated on a latent statistic rather than on vote agreement, which is what lets it stop that early. Gate precision is $1.000/0.979$ and the stopped answer equals the full 24-view decode's in 99.5–100% of cases, so early stopping is not a speed/accuracy trade here. The residual hard problem is re-decided by orbit bagging: extra views duplicate presentation orders, so rather than average a composition-biased 32-view pool we enumerate all $2^8{=}256$ complete 24-order combinations and take the plurality. Net: **13.7 views/sample, $0.57\times$ the fixed-orbit cost.**

Reproducibility. The repository ships the adapter, head weights and trainer, the seeded pipeline and the per-view hidden states, which reproduce the submission byte-for-byte on CPU.

8 Limitations

The head needs white-box access to activations, so it does not transfer to text-only API endpoints, and it is tied to the regime it was fitted on. More labelled data does not straightforwardly help either. Pools drawn from a different generation distribution did not consistantly improve both validation folds and still degraded deployment behaviour, so the training distribution has to match rather than merely grow. This can be partially handled if the SFT process has taken probing method into account as we can leave out more data for probe training. The headroom is set by the generator rather than by the head, since on the roughly 24% of samples where the model never proposes the gold ordering no read-out can recover it. Exact enumeration over all orderings is cheap only because $n{=}4$. Larger n would need a relaxation or a restricted candidate set. Finally, the accuracy is bought with inference, at 13.7 views per sample against one for a terse pass, and the adaptive schedule that lowers that average also makes the runtime data-dependent, so it cannot be budgeted in advance. All of it is measured on a single task with a single backbone.

References

[1] H. Orgad et al. LLMs Know More Than They Show. *ICLR*, 2025.
[2] Z. Gekhman et al. Inside-Out: Hidden Factual Knowledge. *COLM*, 2025.
[3] Y. Park et al. Bridging the Knowledge–Prediction Gap in LLMs on MCQs. *ICML*, 2026.
[4] M. Turpin et al. Language Models Don't Always Say What They Think. *NeurIPS*, 2023.
[5] Z. Xie et al. Calibrating Reasoning with Internal Consistency. *NeurIPS*, 2024
[6] X. Wang et al. Self-Consistency Improves CoT Reasoning. *ICLR*, 2023
[7] R. Santa Cruz et al. DeepPermNet: Visual Permutation Learning. *CVPR*, 2017.
[8] J. Hewitt, C. D. Manning. A Structural Probe for Finding Syntax. *NAACL*, 2019