EKLAV: ELICITED KEY COT LEARNING VIA ANSWER ONLY SUPERVISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning distillation trains a student to reproduce both the teacher's chain-of-thought trace and its final answer. We ask whether the trace is better used as privileged training context than as an imitation target. EKLAV keeps the teacher's reasoning in the input sequence but supervises only the final task answer, applying a task-dependent placement that moves part or all of the trace into the prompt. On BRIGHT passage reranking, EKLAV improves average nDCG@10 at all four tested checkpoints across two model families, winning 41 of 48 model-domain comparisons while reducing training compute by up to 31.6% and inference time by up to 32.7%. On four held-out table-reranking benchmarks, it is the best of four conditions on three, exceeding full-trace distillation by 3.0 points and answer-only training without reasoning by 5.5 points on average. Analysis of generated traces reveals that EKLAV students defer their first verbalized verdict later (0.76 vs. 0.44 relative position), reverse fewer decisions (5% vs. 16%), and produce reasoning more predictive of their own rankings than STD-COT's, though less predictive than the base model's. The recipe reuses a fixed teacher corpus, requires no student rollouts or iterative teacher feedback, and applies to both pointwise and listwise formulations. Code and Data are at `https://anonymous.4open.science/r/Eklav-C1AE/README.md`.

1 INTRODUCTION

Reasoning distillation transfers the capabilities of a large teacher model to a smaller student through supervised fine-tuning Guo et al. (2025); Muennighoff et al. (2025); Ye et al. (2025). The standard recipe, which we call STD-COT, trains the student to reproduce both the teacher's reasoning trace and its final answer, implicitly treating the intermediate reasoning as a desirable supervision target even though fluent chain-of-thought need not faithfully describe the computation behind a prediction Wang et al. (2023a); Arcuschin et al. (2026). Separating rationale supervision from answer supervision has been shown to improve out-of-domain generalization Dai et al. (2024), and more broadly, chain-of-thought reasoning has been characterized as a brittle artifact of the training distribution: controlled studies report near-perfect in-distribution performance that collapses under moderate task, length, or format shifts Zhao et al. (2026), and fine-tuning on reasoning traces has been shown to reduce both accuracy and CoT faithfulness outside the fine-tuning task category, more so for smaller models Lobo et al. (2025). We observe the same asymmetry in our own table-reranking results: NAIVE-SFT, an answer-only baseline with no teacher reasoning at all, outperforms full-trace STD-COT on three of four out-of-domain benchmarks we test, even though STD-COT is the stronger of the two on every in-domain benchmark (Section 6). These observations raise a direct question: must a student reproduce the teacher's reasoning to benefit from it?

Reranking offers a controlled setting for studying this question. A reranker must reason about the relevance of a candidate document to a query, while ultimately producing a well-defined and verifiable ranking decision. This structure allows us to separate supervision of intermediate reasoning from supervision of the final task output with fewer confounds from stylistic variation or unconstrained generation. BRIGHT SU et al. (2025) further provides a demanding evaluation across heterogeneous reasoning domains, including scientific literature, mathematics, coding, and logical inference. Performance across these domains therefore tests whether a distillation strategy transfers beyond a narrow reasoning pattern.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We study a simple alternative, EKLAV, that preserves teacher reasoning as training context while removing direct supervision over the reasoning tokens. For pointwise passage reranking, a filtered prefix of the teacher trace is placed in the prompt as a reasoning hint, while the remaining teacher continuation stays in the assistant sequence with its token-level loss masked. For listwise table reranking, the complete teacher trace is provided in the prompt. In both formulations, teacher reasoning conditions the supervised prediction, but the optimization objective applies only to the final task answer. The model therefore learns from teacher reasoning while never being asked to reproduce it. Both formulations operate entirely on a fixed teacher-generated corpus and require neither student rollouts nor iterative teacher feedback. At evaluation time, all teacher-provided reasoning is removed and the student generates its own reasoning before producing the final prediction.

This formulation connects reasoning distillation to learning with privileged information Vapnik & Vashist (2009); Lopez-Paz et al. (2016), where additional information is available during training and withheld at inference. Answer-focused supervision with privileged reasoning context also appears in Masked Distillation Kalwar et al. (2026). EKLAV shares the principle of masking direct supervision over teacher reasoning. It develops this idea for reranking with task-dependent context placement, fully offline training, and both pointwise and listwise formulations, and compares against full-trace distillation and answer-only supervision without reasoning context. We evaluate across four model scales and two model families, complementing task-level results with trace-level analysis of how the supervision objective changes the reasoning the student produces.

Contributions.

- For reranking, we formulate EKLAV as answer-only distillation with teacher reasoning retained as privileged conditioning context, separating the effects of reasoning supervision, context placement, and the removal of teacher context at evaluation.
- Across BRIGHT, EKLAV improves nDCG@10 at all four tested checkpoints spanning two model families and outperforms full-trace distillation in 41 of 48 model-domain comparisons. Across four held-out table benchmarks, it scores highest among four training conditions on three of four, while reducing unparseable outputs from 0–14.5% to 0–2.5%.
- EKLAV reduces training compute by 24–32% and inference time by 15–33%, yielding an accuracy-efficiency Pareto improvement at every tested model scale.
- Trace-level analysis shows that EKLAV students verbalize their verdicts later, reverse fewer intermediate decisions, and produce reasoning whose content is more predictive of their final rankings than STD-COT's, though less predictive than the base model's.

2 RELATED WORK

Trace supervision. Reasoning distillation commonly trains students on teacher-generated traces and answers Guo et al. (2025); Muennighoff et al. (2025); Ye et al. (2025). CasCoD decomposes rationale and answer learning into cascaded stages while retaining explicit rationale supervision, reporting out-of-domain gains Dai et al. (2024). EKLAV removes reasoning supervision entirely and retains the trace only as conditioning context.

Masked Distillation. Masked Distillation Kalwar et al. (2026) is the most closely related work. In its solution-only setting, teacher feedback conditions on the teacher's reasoning while the student predicts solution tokens. EKLAV shares the core idea of answer-focused supervision with reasoning as privileged information but differs in mechanism: it places the teacher trace directly in the student's training sequence, uses hard-target cross-entropy on a fixed corpus, and introduces task-dependent context placement. We evaluate on passage and table reranking rather than mathematical problem solving.

Student-adaptive supervision. A family of methods makes the supervision depend on the particular student being trained: having the student produce incorrect rationales that the teacher then corrects across multiple rounds Wang et al. (2023b), steering trace generation mid-stream by scoring teacher continuations by perplexity He et al. (2026), or retrieving skill-conditioned teachers that score the same student rollout Huang et al. (2026). These achieve adaptivity at the cost of additional student or teacher evaluations per example, from multiple teacher samples to full student rollouts. EKLAV trades this adaptivity for simplicity: it reuses a precomputed corpus with no student rollouts or iterative feedback, making it applicable to any existing reasoning-distillation dataset.

*Table 1: **Training formulations.** The Masked Distillation row describes its solution-only setting. Context and targets omit chat-template delimiters.*

Method	Student context for answer prediction	Supervision
STD-COT	Task input and teacher trace	Teacher reasoning and answer tokens
NAIVE-SFT	Task input	Answer tokens only
Masked Distillation Kalwar et al. (2026)	Question; teacher reasoning informs feedback	Solution tokens
EKLAV	Task input and teacher trace, with task-dependent placement	Answer tokens only

Shorter reasoning. A separate line of work attacks the verbosity of reasoning traces directly through compression, token skipping, drafting, and budget-aware generation Kang et al. (2025); Xia et al. (2025); Xu et al. (2025); Han et al. (2025), reducing cost but retaining the fundamental relationship that the student still reproduces a reasoning artifact, only a shorter one. EKLAV does not explicitly target length, but its students produce shorter traces as a side effect of removing the imitation objective (Section 7).

Privileged information. Learning Using Privileged Information (LUPI) Vapnik & Vashist (2009) and generalized distillation Lopez-Paz et al. (2016) formalize the setting where training-time information is unavailable at test time. Context distillation trains a context-free model to match the output distribution of a separate context-conditioned model Askell et al. (2021), the reverse direction from ours: it distills across two models, whereas EKLAV trains a single model that sees privileged context at train time and not at test time. EKLAV instantiates LUPI for reasoning traces: the teacher's chain of thought serves as privileged context during training and is withdrawn at evaluation through the reasoning context shift (Section 3).

Table 1 summarizes the central distinction: what the student conditions on versus what its objective supervises.

3 REASONING-CONDITIONED ANSWER SUPERVISION

3.1 PROBLEM FORMULATION

We assume access to a reasoning-distillation corpus produced by a large teacher model. Each example is a triple (x, r, a): a task input, a teacher reasoning trace, and a final answer. Standard distillation (STD-COT) supervises both r and a, training the student to reproduce the teacher's reasoning token by token. EKLAV keeps r in the training sequence as conditioning context but applies the loss only to a:

$$\boxed{\text{reasoning as imitation target} \quad \longrightarrow \quad \text{reasoning as conditioning context}} \tag{1}$$

The student sees the teacher's reasoning during training and learns from it through attention and gradient flow, but is never asked to reproduce it. How the reasoning is partitioned between prompt and response depends on the task.

3.2 TASK-DEPENDENT REASONING TRANSFORMATION

Let $\Phi_\tau(r) = (h_\tau(r), c_\tau(r))$ partition the reasoning into prompt-side context $h_\tau(r)$ and masked assistant-side context $c_\tau(r)$ for task τ.

Pointwise passage reranking. The teacher ends each query–passage judgment with `true` or `false`. Write the trace as sentences $r = (r_1, \ldots, r_m)$ and let $k = \lfloor m/2 \rfloor$. We set

$$\Phi_{\text{point}}(r) = \big(F(r_{1:k}),\ r_{k+1:m}\big), \tag{2}$$

where F is a lexical filter that removes verdict-bearing sentences (Appendix A.1). The filtered prefix $r^H = F(r_{1:k})$ moves into the prompt as a reasoning hint; if filtering empties it, the hint is left empty rather than falling back to unfiltered text, so no verdict-bearing sentence reaches the prompt. The teacher continuation $r^C = r_{k+1:m}$ remains in the assistant response but is excluded from the loss. Only the final verdict token is supervised. The corpus is the public Rank1 training set Weller et al. (2025). The filter targets explicit verdict phrases in the prompt-side hint; the unfiltered continuation r^C still conditions the loss (see the Deployment paragraph below for what this implies about the training signal).

Listwise table reranking. The teacher compares the candidate tables and emits a ranking as a JSON list. We retain the complete trace in the prompt,

$$\Phi_{\text{list}}(r) = (r, \varnothing), \tag{3}$$

and supervise the ranking JSON (see the Deployment paragraph below for how placing the complete trace in the prompt bears on what this training signal can and cannot show). The corpus contains 15,674 DeepSeek-R1 traces. Four evaluation benchmarks contribute no examples to this fine-tuning corpus; four additional benchmarks are evaluated in domain (Section 6 and Appendix B.5).

Context allocation and the prompt ratio. Let $\rho_\tau(r) = |h_\tau(r)|/|r|$ denote the fraction of teacher reasoning placed in the prompt. The pointwise split gives $\mathbb{E}[\rho_{\text{point}}] \approx 0.5$ and the listwise variant gives $\rho_{\text{list}} = 1$. In both cases the full teacher trace is required during training; ρ governs how it is partitioned between prompt and masked response, not whether it is consumed. The pointwise split is by sentence count; after filtering, the token-level split can differ from the midpoint. Both variants are subject to the shared task-specific sequence cutoff, which may truncate the serialized example.

3.3 Training Sequences and Loss

We write a chat sequence as [prompt; assistant response], suppressing template delimiters, and use $\oplus$ for concatenation. The three training conditions are

$$y^{\text{STD}} = [x;\ r \oplus a], \tag{4}$$

$$y^{\text{EK}} = [x \oplus h_\tau(r);\ c_\tau(r) \oplus a], \tag{5}$$

$$y^{\text{Naive}} = [x;\ a]. \tag{6}$$

EKLAV changes both the context layout and the loss mask relative to STD-CoT; NAIVE-SFT removes teacher reasoning entirely. All three use token-level cross-entropy under teacher forcing:

$$\mathcal{L}_m(\theta) = -\frac{1}{|\mathcal{M}_m|} \sum_{t \in \mathcal{M}_m} \log p_\theta\left(y_t^{(m)} \mid y_{<t}^{(m)}\right), \tag{7}$$

where $\mathcal{M}_{\text{STD}} = \text{pos}(r) \cup \text{pos}(a)$ and $\mathcal{M}_{\text{EK}} = \mathcal{M}_{\text{Naive}} = \text{pos}(a)$. Under EKLAV, prompt tokens and assistant-side reasoning tokens receive no direct loss; any automatically inserted empty reasoning block is also masked (Appendix A.3). The experimental comparison evaluates the complete recipe — context placement, loss masking, and filtering together — against full-trace distillation.

Isolating the hint: MASK-ONLY. For pointwise passage reranking, we additionally train $y^{\text{Mask}} = [x;\ r \oplus a]$ with $\mathcal{M}_{\text{Mask}} = \text{pos}(a)$: the same masked-loss objective as EKLAV, but with no prompt-side hint, so the complete teacher trace r sits in the masked assistant-side context instead of being split into $h_\tau(r)$ and $c_\tau(r)$. This ablation isolates whether EKLAV's gains come from the hint itself or simply from masking the reasoning loss (Section 5).

Gradient flow under teacher forcing. Masking a token removes its direct loss but not its influence: later answer tokens still attend to its hidden state, so the answer loss can produce a nonzero gradient at masked reasoning positions, in both EKLAV and STD-CoT (Appendix A.4).

Deployment. During training, the answer is conditioned on teacher-derived context $C_T = (h_\tau(r), c_\tau(r))$; for listwise table reranking this is the complete teacher trace, which typically ends by stating the ranking, and for pointwise reranking the unfiltered continuation $c_\tau(r)$ can similarly carry the verdict. Conditioning on C_T can therefore make the supervised answer recoverable from context alone, so training-time performance cannot by itself show that the student is reasoning toward the answer rather than locating it. What the answer-only loss does is shape how that reasoning is produced: gradients push the model toward whatever reasoning, conditioned on C_T, leads to the correct answer, not toward reproducing the teacher's tokens. At evaluation, C_T is removed entirely: the student receives x alone and generates its own reasoning $\hat{R}_S$ before answering (the **reasoning context shift**, $C_T \to \hat{R}_S$). Since C_T is never available at test time, the reported gains reflect only what the student learned to do with its own reasoning (Appendix A.5).

4 EXPERIMENTAL SETUP

Models. We evaluate on four checkpoints across two architecture families: Qwen3-4B, Qwen3-8B, and Qwen3-14B Yang et al. (2025) from one family, and GLM-Z1-9B GLM et al. (2024) from another. All four are trained on passage reranking, comparing STD-COT with EKLAV at each scale; the MASK-ONLY ablation, which keeps EKLAV's masked-loss training but removes the prompt-side CoT hint, is additionally reported for Qwen3-8B (Table 3). Table reranking is trained at a single scale (Qwen3-8B) with the full four-way comparison: the untrained base model, NAIVE-SFT, STD-COT, and EKLAV. All fine-tuned models use LoRA Hu et al. (2022) (rank 32, alpha 64, all linear layers). The NAIVE-SFT and base-model controls, and the MASK-ONLY ablation, are not run at the other three passage scales.

Training. All runs use learning rate 10^{-4}, cosine schedule with 5% warmup, effective batch size 128, bf16 precision, and two epochs in LLaMA-Factory Zheng et al. (2024). Sequence cutoff is 2,500 tokens for passage reranking and 32,768 for table reranking. Within each task, EKLAV and STD-COT share all optimization hyperparameters and are compared at the same training step.

Benchmarks and metrics. Passage evaluation uses the twelve BRIGHT domains SU et al. (2025), a challenging realistic benchmark where reasoning-based rerankers show the largest gains over lexical baselines. Table evaluation uses four held-out (OOD) benchmarks (AITQA Katsis et al. (2021), FeTaQA, OTT-QA, OpenWikiTables), drawn from the IBM Research Table-Text IR Evaluation collection Doshi et al. (2026), and four in-domain benchmarks (SQA Iyyer et al. (2017), TAT-QA Zhu et al. (2021), TabFact Chen et al. (2020a), NQ-Tables; Appendix B.5), drawn from the same benchmark paper as our multi-table training data even though themselves held out of training (Appendix B.2). This is a distributional notion of domain, not literal appearance in the fine-tuning corpus, and it says nothing about overlap with the base model's pretraining data. One partial exception is OTT-QA, built on top of the training split of HybridQA Chen et al. (2020b), which is part of our training corpus; we evaluate on OTT-QA's own test split (Section 6). Table candidate pools are built by lexical and dense retrieval fused via RRF (Appendix B.2), following Rank1's approach Weller et al. (2025), so that pool construction does not favor either training or held-out sources and reranking is compared fairly across conditions. We report nDCG@10 throughout, scaling passage scores by 100 for readability.

Passage scoring. The student generates a reasoning trace $\hat{r}$ for query q and passage d, then scores the pair as $s(q, d) = p_\theta(\texttt{true} \mid q, d, \hat{r})$, a pointwise formulation inspired by Rank1 Weller et al. (2025). We rerank Rank1's released BM25 Robertson et al. (1995) top-100 BRIGHT candidate pools directly rather than rebuilding them, so that all compared methods rerank the same candidates for a fair comparison (Appendix B.4). No teacher context is supplied at evaluation.

Table decoding. Greedy decoding (temperature 0, max 8,192 output tokens). The output parser extracts a JSON ranking; unparseable outputs receive nDCG@10 of zero. The parser and decoding budget are identical across all four conditions, so the reported nDCG is an end-to-end measure combining output validity and ranking quality. Parsing failure rates are reported separately in Section 6. Section 6 also reports an ablation that doubles this budget to 16,384 tokens for STD-COT only.

Controlled comparison. The EKLAV and STD-COT checkpoints compared for each model come from the same optimization configuration and the same training step. The methods differ in their serialized context layout and supervised token set (Section 3.3). Decoding is greedy (temperature 0) and deterministic. All training FLOPs are $6ND$ estimates, which do not account for LoRA updating only a subset of parameters and so overestimate the compute LoRA training actually uses.

5 PASSAGE RERANKING

Table 2 reports BRIGHT nDCG@10 for four models spanning two architecture families.

EKLAV leads at every scale, with absolute gains of 2.2–3.1 points, and holds across the two architecture families we test. Across all four checkpoints and twelve BRIGHT domains, EKLAV improves in 41 of 48 comparisons. Figure 1 shows the per-domain profile at each tier: the EKLAV polygon sits at or outside the STD-COT polygon on nearly every axis. Each tier has one to three losing domains, and the sets only partially overlap across tiers.

Per-domain breakdown. Table 3 gives the full twelve-domain breakdown for Qwen3-8B, including an ablation, MASK-ONLY, that isolates the prompt-side CoT hint: it applies the same masked-

Table 2: ***BRIGHT average nDCG@10*** *(*$\times 100$*)****, across model families and scales.*** EKLAV *improves over* STD-COT *at every scale and across both families.*

Student	STD-COT	EKLAV	Gain	Wins
Qwen3-4B	29.3	**31.5**	+2.2	9/12
Qwen3-8B	31.5	**34.2**	+2.7	11/12
GLM-Z1-9B	30.1	**33.2**	+3.1	11/12
Qwen3-14B	32.5	**34.7**	+2.2	10/12

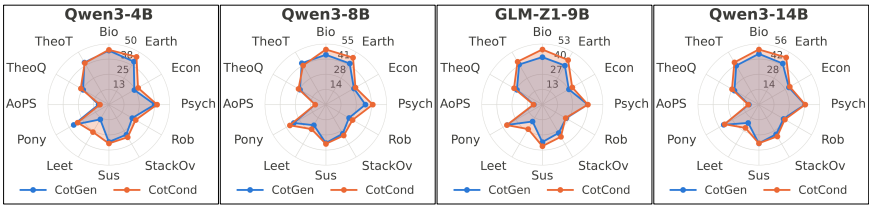


Figure 1: ***BRIGHT per-domain nDCG@10, all four model tiers.*** EKLAV *(orange) vs.* STD-COT *(blue). The losing domains shift from tier to tier, indicating no systematic weakness.*

loss training as EKLAV but without ever placing the hint in the prompt, so the full teacher trace sits in the masked assistant-side context instead. MASK-ONLY trails STD-COT on average (30.33 vs. 31.51) and EKLAV on eleven of twelve domains, so masking the reasoning loss alone does not reproduce EKLAV's gains; the hint is doing the work. EKLAV improves on eleven of twelve domains, with its largest gains on domains that require reasoning across diverse evidence types (Earth Science, Psychology) or algorithmic structure (LeetCode, Pony) rather than lexical overlap; the single regression, TheoremQA-Theorems (-2.4), is a mathematical-proof domain. Regressions are few and shift across model tiers rather than concentrating on a fixed weakness (Appendix B.3).

6 TABLE RERANKING

Because the motivation for this work is generalization, the Out-of-Domain (OOD) evaluation is the most informative experiment. Table 4 evaluates four benchmarks absent from the fine-tuning corpus. All four conditions share the Qwen3-8B backbone and evaluation protocol, differing only in training.

The four OOD benchmarks each shift away from training in a different way: OpenWikiTable keeps single-table questions like NQ-Tables but requires aggregation and comparison across rows rather than single-cell lookup; FeTaQA keeps single tables but requires a free-form generated answer rather than an extracted one; AITQA keeps a similar QA format but moves to airline-industry tables with a complex, hierarchical structure. OTT-QA is built on top of the training split of HybridQA, and we evaluate on its own test split; the reasoning-operation/answer-format-shift story below is more safely established from FeTaQA and OpenWikiTables alone, with OTT-QA as corroborating evidence. EKLAV leads STD-COT on the three benchmarks that change the reasoning operation or answer format (FeTaQA, OTT-QA, OpenWikiTables) and trails only on AITQA, where the shift is in table structure rather than task type.

This pattern extends beyond EKLAV itself: STD-COT generalizes poorly enough that NAIVE-SFT, trained without any teacher reasoning, beats it on the same three benchmarks, and on FeTaQA STD-COT even falls below the untrained base model. Imitating the teacher's trace, then, does not just fail to help once the reasoning operation or answer format changes, it can actively hurt. EKLAV still beats NAIVE-SFT on every OOD benchmark, so dropping the trace is not the fix either; retaining teacher reasoning as context without training the student to reproduce it is what generalizes best here, consistent with the broader pattern reported for CoT fine-tuning Lobo et al. (2025).

Output validity. Under identical decoding budgets, STD-COT fails to produce a parseable ranking on 0–14.5% of queries, while EKLAV fails on 0–2.5% (Table 4). The gap is widest on OTT-QA (14.5% vs. 0.1%) and FeTaQA (8.3% vs. 0.0%). Because unparseable outputs receive a score of zero, the end-to-end nDCG gains in Table 4 reflect both ranking quality and output reliability. We attribute this to the supervision signal: under STD-COT, the student spends training capacity reproducing the teacher's verbose reasoning patterns, which can interfere with structured output format-

Table 3: ***BRIGHT, all twelve domains, Qwen3-8B*** *(*$100 \times$ nDCG@10*). See Appendix B.1 for the* MASK-ONLY *ablation.*

Domain	STD-COT	MASK-ONLY	EKLAV	Δ
Biology	45.25	44.95	**50.00**	+4.75
Earth Science	43.13	41.46	**49.25**	+6.12
Economics	29.27	26.63	**30.88**	+1.61
Psychology	35.77	35.06	**42.42**	+6.65
Robotics	24.18	25.26	**27.87**	+3.69
StackOverflow	30.53	29.15	**32.41**	+1.88
Sustainable Living	34.92	31.92	**35.59**	+0.67
Pony	33.45	34.83	**37.60**	+4.15
LeetCode	21.59	19.92	**25.70**	+4.11
AoPS	9.66	8.28	**9.70**	+0.04
TheoremQA-Theorems	**43.37**	41.44	40.95	−2.42
TheoremQA-Questions	26.96	25.02	**28.46**	+1.50
Average	31.51	30.33	**34.24**	+2.73

Table 4: ***Table reranking results*** *(nDCG@10, scaled to 0–100) on in-domain and out-of-domain (OOD) benchmarks. See Appendix B.1 for column conventions and Appendix B.2 for dataset details.*

	Benchmark	Base	NAIVE-SFT	STD-COT	EKLAV	Δ	Unp.$_{\text{STD-COT}}$	Unp.$_{\text{EKLAV}}$
In-domain	SQA	69.51	71.41	72.05	**73.81**	+1.76	0.0	0.0
	TAT-QA	50.51	49.59	52.03	**54.59**	+2.56	6.4	0.0
	TabFact	60.03	62.20	66.61	**68.90**	+2.29	3.9	0.3
	NQ-Tables	72.10	68.88	**74.73**	72.20	-2.53	2.8	2.3
OOD	AITQA	77.04	68.23	**88.25**	85.99	-2.26	2.3	2.5
	FeTaQA	89.63	91.42	88.06	**91.95**	+3.89	8.3	0.0
	OTT-QA (small)	81.27	88.25	83.20	**90.27**	+7.07	14.5	0.1
	OpenWikiTables	88.68	91.98	90.29	**93.78**	+3.49	8.6	0.1

ting on unseen benchmarks. Under EKLAV, the supervised tokens are exclusively the well-formatted answer, so the model's capacity concentrates on producing valid output.

Generalization gap. EKLAV wins three of four in-domain benchmarks and three of four OOD benchmarks, but the effect sizes differ sharply: the mean gain over STD-COT is $\bar{\Delta}_{\text{OOD}} = 3.05$ out of domain versus $\bar{\Delta}_{\text{ID}} = 1.02$ in domain, about three times larger, which would be surprising if the advantage came only from better optimization or a more favorable loss landscape. Part of the gap is mechanical: STD-COT's unparseable-output rate is higher OOD than in domain (Table 4) and every such failure scores zero, so some of the larger OOD margin reflects output-format reliability rather than ranking quality alone. We hypothesize trace imitation also binds the student to training-distribution surface patterns that transfer poorly, though we cannot isolate this from the reliability effect here. The single in-domain regression (NQ-Tables, -2.53) does not break the overall pattern.

Is EKLAV's in-domain shortfall a truncation artifact? A natural alternative explanation for EKLAV's two in-domain losses is decoding budget: if STD-COT frequently exhausts its 8,192-token budget mid-trace, its answer would be based on an incomplete rationale, and EKLAV could look better for that reason alone. We test this by re-decoding STD-COT with the output budget doubled to 16,384 tokens, changing nothing else about the checkpoint or decoding procedure (Table 5). This ablation covers the four in-domain benchmarks only, and extends the budget for STD-COT alone.

Doubling the budget helps STD-COT on SQA and NQ-Tables, where it now leads EKLAV by 1.2 and 5.1 points, and leaves EKLAV ahead only on TAT-QA (+1.1) and TabFact (+3.0). Averaged over the four benchmarks, extended-budget STD-COT beats EKLAV by 0.55 points. So at least part of EKLAV's in-domain shortfall on SQA and NQ-Tables is consistent with STD-COT being cut off before it finishes reasoning: given enough room to think, STD-COT matches or exceeds EKLAV in domain.

7 EFFICIENCY

Table 6 reports training FLOPs (via the standard $6ND$ estimator) and inference wall-clock time for every passage-reranking model pair.

Table 5: ***Truncation-budget ablation*** *(in-domain, nDCG@10, scaled to 0–100). See Appendix B.1 for the ablation setup and column conventions.*

Condition	SQA	TAT-QA	TabFact	NQ-Tables
STD-COT (8,192 tok.)	72.05	52.03	66.61	74.73
STD-COT (extended, 16,384 tok.)	**75.00**	53.50	65.90	**77.30**
EKLAV	73.80	**54.60**	**68.90**	72.20
Δ (EKLAV − extended)	-1.20	+1.10	+3.00	-5.10

Table 6: ***Training FLOPs and inference time, passage reranking.*** *Training values are $6ND$ estimates in 10^{18} FLOPs. Inference is wall-clock minutes on a single A100 with vLLM, greedy decoding, batch size 1.* EKLAV *uses less compute and is faster at every scale — an accuracy–efficiency Pareto improvement.*

	Training ($\times 10^{18}$ FLOPs)			Inference (min)		
Student	**STD-COT**	**EKLAV**	**Red.**	**STD-COT**	**EKLAV**	**Red.**
Qwen3-4B	10.93	7.55	30.9%	118.5	79.8	32.7%
Qwen3-8B	21.50	14.70	31.6%	132.0	95.5	27.7%
GLM-Z1-9B	23.70	16.61	29.9%	104.4	88.3	15.4%
Qwen3-14B	35.29	26.89	23.8%	176.1	139.0	21.1%

Training compute. EKLAV reduces estimated training FLOPs by 24–32% on passage reranking. The reduction comes from shorter realized training sequences: the lexical filter (Section 3.2) removes verdict-bearing sentences from the reasoning hint before it enters the prompt, so EKLAV's total sequence, prompt plus response, contains fewer tokens than STD-COT's unfiltered trace of the same passage pair. This is a genuine reduction in processed tokens, not an artifact of where the sequence cutoff falls, since masking labels alone does not eliminate transformer computation. Step counts are identical across methods.

Inference speed. Inference is 15–33% faster at every scale, measured on a single A100 GPU with vLLM Kwon et al. (2023). The speedup is consistent with shorter generated reasoning: on 899 matched Qwen3-8B passage pairs, EKLAV generates a mean of 198 words per trace versus 312 for STD-COT, a $1.58\times$ reduction holding across all twelve BRIGHT domains (Appendix B.9 gives the per-domain range and the analogous 1.5–$1.8\times$ reduction on table reranking). The length reduction follows from the changed supervision. Under STD-COT, the student is trained to reproduce the teacher's often-verbose reasoning, and this verbosity persists at inference even when the teacher context is absent. Under EKLAV, the student is never rewarded for generating long reasoning traces; it learns to reason only as much as needed to produce a correct answer. Combined with the accuracy gains, these efficiency improvements yield an accuracy–efficiency Pareto improvement at every tested scale.

8 WHAT CHANGES IN THE STUDENT'S REASONING

We now examine what removing trace supervision changes about the student's generated reasoning. If EKLAV degraded reasoning quality, we would expect its traces to be less structured or less connected to its outputs. We test this by analyzing Qwen3-8B traces on matched inputs across training conditions, using the same lexical extractors for both methods.

Verdict Timing and Stability. A lexical detector identifies relevance verdicts ("relevant," "irrelevant," `true`/`false`) in 899 matched passage-reranking pairs. Table 7 reports when the first verdict appears, how many times it is restated (avg. mentions), and how often the model reverses its stated judgment ($\geq$1 reversal).

EKLAV students state their first verbalized verdict later (median position 0.85 vs. 0.37, Mann–Whitney $p = 1.5 \times 10^{-69}$, holding across all twelve BRIGHT domains), reverse it less often (5.2% vs. 16.0%), and restate it less (1.4 mentions per trace vs. 2.3). The gap widens at the 25th percentile (Q1 = 0.05 for STD-COT vs. 0.66 for EKLAV), and STD-COT detects a verdict in more traces overall (98.6% vs. 89.4%), suggesting some EKLAV traces express judgment only implicitly through the final answer token.

*Table 7: **Verdict timing and reversals** for Qwen3-8B on 899 matched passage-reranking pairs. Position is the fraction of the reasoning trace elapsed at the first detected relevance verdict (0 = start, 1 = end).*

Method	Detection (% traces)	Mean first position	Median first position	Q1 position	≥1 reversal (% traces)	Avg. mentions
STD-COT	98.6	0.44	0.37	0.05	16.0	2.3
EKLAV	89.4	0.76	0.85	0.66	5.2	1.4

*Table 8: **Trace–answer association** for Qwen3-8B table reranking over 132k candidate-level observations. Each cell is $I(\cdot\,;\,\textit{target})/H(\textit{target})$; higher values indicate stronger predictive association.*

	Trace → model's choice $\hat{Y}$			Trace → gold Y
Condition	Mentions	Recency	Both (F)	Both (F)
Base	0.313	0.311	0.373	0.140
NAIVE-SFT	0.278	0.163	0.317	0.097
EKLAV	0.214	0.167	0.264	0.100
STD-COT	0.111	0.101	0.144	0.049

This is consistent with STD-COT imitating the teacher's early-preliminary-judgment-then-elaboration pattern, sometimes contradicting itself when later evidence conflicts with that early commitment, whereas EKLAV has no such pattern to copy and instead gathers evidence before stating a judgment.

Trace-Answer Association. For table reranking, we extract two features from each reasoning trace, *mentions* of "Table k" and the *recency* of its last mention. We combine them into a joint feature F and measure normalized mutual information with the model's top-ranked choice $\hat{Y}$ and the gold label Y over 132,000 candidate-level observations.

EKLAV shows substantially stronger trace–answer association than STD-COT for both its own predictions ($I(F;\hat{Y})/H = 0.264$ vs. 0.144) and the gold label (0.100 vs. 0.049); STD-COT in fact exhibits the weakest association among all four conditions despite directly supervising the teacher's reasoning trace. Removing direct trace supervision therefore does not reduce the measured connection between generated reasoning and final predictions, though these associations remain observational and do not establish causal faithfulness. Further analysis of surface style and voice is provided in Appendix C.2.

9 CONCLUSION

EKLAV studies whether teacher reasoning can serve as training-time context rather than a direct imitation target for reranking. Retaining teacher reasoning while supervising only the final answer improves performance in most evaluated settings, with particularly strong gains on held-out benchmarks (exceptions: AITQA, NQ-Tables, and the Table 5 ablation, where a larger decoding budget lets STD-COT match or exceed EKLAV in domain), while also producing fewer malformed outputs and shorter generations at no cost of student rollouts or iterative teacher feedback. Trace-level analyses further show EKLAV students state verdicts later, reverse them less often, and generate reasoning more predictive of their final decisions than STD-COT's, though still less predictive than the base model's. These gains are task-dependent: exploratory mathematical-reasoning results show the opposite pattern (Appendix C.1), and broader validation across tasks and training formulations remains an important direction.

10 LIMITATIONS

As with any empirical study, our conclusions are drawn from a finite set of models, datasets, and hyperparameter choices, and may not hold under every configuration outside the ones we test. Reported metrics and efficiency estimates rely on standard approximations and measurement conventions common in the literature, which can differ from other studies' methodology. Broader replication across additional tasks, model families, and training regimes would further establish the generality of the trends we observe. Observed associations between measured quantities do not by themselves establish the underlying causal mechanism, and results at the scales and compute budgets studied here may not extrapolate uniformly to substantially larger or smaller settings. Some implementation and design choices were fixed rather than exhaustively tuned, and a different choice could shift absolute numbers without necessarily changing the overall trend.

AI USE STATEMENT

We used generative AI tools for interpreting experimental results and for generating synthetic data: the teacher reasoning traces underlying our fine-tuning corpora were produced by a large language model (DeepSeek-R1 for table reranking; the public Rank1 training corpus for passage reranking), and we used a general-purpose AI assistant to help identify and characterize patterns in our experimental results during analysis. We have not used generative AI tools to design the EKLAV methodology, our experiments, or the training/evaluation code; the training and evaluation pipelines were implemented by the authors without AI assistance. We additionally used generative AI tools for drafting and editing the text and github repository for this paper. All AI-assisted analysis and writing was reviewed by the authors, who verified reported numbers against the underlying experimental logs and take full responsibility for the final content of this work, including all text, claims, and reported results.

ETHICS STATEMENT

This work uses only publicly released models (Qwen3, GLM-Z1, DeepSeek-R1) and publicly available academic benchmarks (BRIGHT, SQA, TAT-QA, TabFact, NQ-Tables, HybridQA/MultiTableQA, AITQA, FeTaQA, OTT-QA, OpenWikiTables), each used under the terms of its original release and cited accordingly (Appendix B.2). No new human-subject data was collected, and none of the datasets involve personally identifiable information. We do not anticipate specific harms beyond the general dual-use considerations inherent to information retrieval and reranking research.

REPRODUCIBILITY STATEMENT

We report our full training and evaluation configuration in Section 4 and Appendix A, including hyperparameters, decoding settings, and dataset construction. We will release model checkpoints, training data, and code upon publication to support independent reproduction of our results.

REFERENCES

Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-Thought Reasoning In The Wild Is Not Always Faithful, June 2026.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. A General Language Assistant as a Laboratory for Alignment, December 2021.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. Tabfact : A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, April 2020a.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. HybridQA: A dataset of multi-hop question answering over tabular and textual data. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1026–1036, Online, November 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.91. URL https://aclanthology.org/2020.findings-emnlp.91/.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Wang, and William W. Cohen. Open question answering over tables and text. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Chengwei Dai, Kun Li, Wei Zhou, and Songlin Hu. Improve student's reasoning generalizability through cascading decomposed CoTs distillation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural*

Language Processing, pp. 15623–15643, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.875.

Meet Doshi, Odellia Boni, Vishwajeet Kumar, Jaydeep Sen, and Sachindra Joshi. Tableir eval: Table-text ir evaluation collection. https://huggingface.co/collections/ibm-research/table-text-ir-evaluation, 2026. Hugging Face dataset collection.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-025-09422-z.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24842–24855, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1274.

Chaoqun He, Yingfa Chen, Chaojun Xiao, Xu Han, and Lijie Wen. Student-in-the-Loop Chain-of-Thought Distillation via Generation-Time Selection, April 2026.

Jonathan Herzig, Thomas Müller, Syrine Krichene, and Julian Eisenschlos. Open domain question answering over tables via dense retrieval. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 512–519, Online,

June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.43. URL https://aclanthology.org/2021.naacl-main.43/.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Jiazhen Huang, Xiao Chen, Xiao Luo, Yong Dai, Senkang Hu, and Yuzhi Zhao. Skill-Conditioned Gated Self-Distillation for LLM Reasoning, September 2026.

Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. Search-based neural structured learning for sequential question answering. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1821–1831, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1167. URL https://aclanthology.org/P17-1167/.

Durgesh Kalwar, Vardhan Palod, and Subbarao Kambhampati. Masked Distillation: Internalizing the Chain-of-Thought in Language Models, August 2026.

Yu Kang, Xianghui Sun, Liangyu Chen, and Wei Zou. C3oT: Generating Shorter Chain-of-Thought Without Compromising Effectiveness. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(23):24312–24320, April 2025. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v39i23.34608.

Yannis Katsis, Saneem Chemmengath, Vishwajeet Kumar, Samarth Bharadwaj, Mustafa Canim, Michael Glass, Alfio Gliozzo, Feifei Pan, Jaydeep Sen, Karthik Sankaranarayanan, and Soumen Chakrabarti. Ait-qa: Question answering dataset over complex tables in the airline industry, 2021.

Sunjun Kweon, Yeonsu Kwon, Seonhee Cho, Yohan Jo, and Edward Choi. Open-WikiTable : Dataset for open domain question answering with complex reasoning over table. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8285–8297, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.526. URL https://aclanthology.org/2023.findings-acl.526/.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Elita Lobo, Chirag Agarwal, and Himabindu Lakkaraju. On the impact of fine-tuning on chain-of-thought reasoning. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 11679–11698, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.584. URL https://aclanthology.org/2025.naacl-long.584/.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations*, 2016.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. S1: Simple test-time scaling. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1025.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir Radev. FeTaQA: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49, 2022.

Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pp. 109–126. Gaithersburg, MD: NIST, January 1995. URL https://www.microsoft.com/en-us/research/publication/okapi-at-trec-3/.

Hongjin SU, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Liu Haisu, Quan Shi, Zachary S Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan O Arik, Danqi Chen, and Tao Yu. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. In *The Thirteenth International Conference on Learning Representations*, 2025.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5):544–557, 2009. ISSN 0893-6080. doi: 10.1016/j.neunet.2009.06.042.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. SCOTT: Self-consistent chain-of-thought distillation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5546–5558, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.304.

Zhaoyang Wang, Shaohan Huang, Yuxuan Liu, Jiahai Wang, Minghui Song, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. Democratizing reasoning ability: Tailored learning from large language model. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 1948–1966, Singapore, December 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.120.

Orion Weller, Kathryn Ricci, Eugene Yang, Andrew Yates, Dawn Lawrie, and Benjamin Van Durme. Rank1: Test-time compute for reranking in information retrieval. In *Second Conference on Language Modeling*, 2025.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. TokenSkip: Controllable chain-of-thought compression in LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3351–3363, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.165.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of Draft: Thinking Faster by Writing Less, March 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025.

Chengshuai Zhao, Zhen Tan, Pingchuan Ma, Dawei Li, Bohan Jiang, Yancheng Wang, Yingzhen Yang, and Huan Liu. Is chain-of-thought reasoning of LLMs a mirage? a data distribution lens. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 15231–15261, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.749. URL https://aclanthology.org/2026.findings-acl.749/.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3277–3287, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.254. URL https://aclanthology.org/2021.acl-long.254/.

Jiaru Zou, Dongqi Fu, Sirui Chen, Xinrui He, Zihao Li, Yada Zhu, Jiawei Han, and Jingrui He. RAG over tables: Hierarchical memory index, multi-stage retrieval, and benchmarking. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 38133–38152, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1902. URL https://aclanthology.org/2026.findings-acl.1902/.

A Method Details

This section records the preprocessing, masking, and deployment details of the training recipe.

A.1 Hint construction

Sentence segmentation and split. The pointwise trace is split into sentences using sentence-boundary detection. If the trace contains m sentences, the split is at $k = \lfloor m/2 \rfloor$. The first part is filtered and placed in the prompt, and the remaining teacher-written sentences are retained as masked assistant-side context. A midpoint by sentence count need not be a midpoint by token count. No split-ratio ablation is reported.

Lexical judgment filter. The recorded filter removes verdict-bearing sentences from the prompt prefix, keeping the prompt-side hint free of the teacher's verdict. Its patterns include relevance vocabulary and direct assertions such as "is true," "is false," and "the answer is." The description also records a heuristic based on following prepositions to distinguish task framing from a judgment. The exact executable patterns are needed to reproduce the transformation.

If filtering removes every sentence from the prefix, the hint is left empty rather than falling back to an unfiltered sentence, so no verdict-bearing text reaches the prompt through this path. The remaining teacher continuation is also available during training, and the full listwise trace is placed in the prompt. Consequently, the method should not be described as eliminating answer information from training context. No teacher-derived hint or continuation is provided at evaluation.

For the exploratory mathematical setting, the recorded filter additionally matches boxed answers and phrases such as "the solution is." The provided summary does not include a separate audit of its accuracy.

A.2 Sample prompts

We show the base prompt structure used for each task, as issued at evaluation (no teacher-derived hint or trace is present at evaluation for either task; Section 3.3). At training time, EKLAV inserts the filtered reasoning hint $h_\tau(r)$ into the pointwise prompt (Section 3.2) or the complete teacher trace r into the listwise prompt (Section 3.2); we omit example hint/trace text here since it is produced by an external data-generation pipeline not included with this submission.

Pointwise passage reranking.

```
System: You are a careful retrieval assistant that judges whether a
passage is relevant to a user's query.
User: Determine if the following passage is relevant to the query.
Answer only with 'true' or 'false'.
```

```
Query: {query}
Passage: {passage}
```

The student's response begins with a <think>...</think> reasoning span followed by `true` or `false`; only the final verdict token is scored, $s(q, d) = p_\theta(\texttt{true} \mid q, d, \hat{r})$ (Section 4).

Listwise table reranking.

```
System: You are a careful retrieval assistant that ranks candidate
tables by their relevance to a user's question.
User: Question: {question}
Candidate tables to rank:
### Table 1
{table_1}
### Table 2
{table_2}
...
Reason through each table's relevance step-by-step.
Then output exactly: {"ranked_tables": [1, 2, 3, ...]}
```

Candidate tables are windowed (at most 20 tables or 20,000 prompt tokens per window, stride 10) and the model's per-window ranking is merged into a single final ranking over the full candidate pool (Appendix B.2).

A.3 Masking auto-inserted reasoning blocks

Some chat templates insert an empty reasoning block into a response that lacks one. Under an answer-only objective, the inserted block must also be masked; otherwise those tokens receive direct next-token supervision. The recorded implementation encountered a boundary-matching issue when tokenizing the closing tag in isolation, because subword tokenization can depend on adjacent characters.

The implementation instead locates the closing boundary in the decoded response and maps the character offset to the smallest token index whose decoded prefix covers the tag. The search is restricted to the assistant response, because the prompt may also mention reasoning delimiters. This behavior still requires verification for each template, including merged boundary tokens and missing or repeated tags. Mask positions must also be aligned with the model's next-token label shift.

A.4 Gradient flow through teacher-forced context

Let $z_t^{(\ell)}$ denote an intermediate representation of a teacher-supplied reasoning token at position t. A masked token has no direct next-token loss. Its representation can nevertheless influence later answer positions through subsequent attention layers, so $\partial \mathcal{L}_A / \partial z_t^{(\ell)}$ can be nonzero along those paths. Loss masking therefore does not delete reasoning context or block all gradients through its processing.

This answer-loss pathway exists in both training conditions. STD-CoT additionally supervises prediction of the teacher reasoning tokens. All tokens supplied as conditioning context during training are teacher-forced, including EKLAV's assistant-side continuation. The student computes their representations; it does not generate those tokens as a training rollout. Gradients update parameters through the processing of fixed context and do not differentiate through sampled discrete reasoning. This observation alone establishes neither causal dependence on generated reasoning nor faithfulness at inference.

A.5 Training and deployment context

Before sequence truncation, the teacher-derived context is

$$C_T = \big(h_\tau(r), c_\tau(r)\big). \tag{8}$$

For pointwise training, this is $(F(r_{1:k}), r_{k+1:m})$; for listwise training, it is $(r, \varnothing)$. The full teacher trace is required before filtering and serialization. The context retained in the model input is subject to the task's sequence cutoff.

At evaluation, the model receives the task input without teacher reasoning. It generates its own reasoning before answering,

$$\hat{R}_S \sim p_\theta(\cdot \mid x), \qquad C_T \longrightarrow \hat{R}_S. \tag{9}$$

This is the reasoning context shift discussed in Section 3. Evaluation also removes the prompt-side framing used to supply teacher context. All reported test results are obtained in this setting. The observed performance therefore measures the trained model after privileged context has been withdrawn, without proving that the teacher's reasoning procedure has been internalized.

B Full Results

This appendix collects the reported benchmark tables, cost records, and exploratory summaries, including regressions.

B.1 Main-text table conventions

Table 3 (BRIGHT, all twelve domains) reports MASK-ONLY, an ablation that keeps EKLAV's masked-loss training but removes the prompt-side CoT hint, to test whether the hint itself, not just the masking, drives EKLAV's gains (Section 5).

Table 4 (table reranking results) bolds the highest score among the four conditions per row and underlines the second highest; Δ is the absolute gain of EKLAV over STD-CoT, and the *Unp.* columns report the unparseable-output rate (%) for STD-CoT and EKLAV under the same parser and decoding budget, with such failures scoring zero in the nDCG columns. See Appendix B.2 for the in-domain and OOD datasets.

Table 5 (truncation-budget ablation) doubles STD-CoT's normal 8,192-token output budget to 16,384 tokens, the only change, to test whether EKLAV's gains simply come from cases where STD-CoT was cut off mid-trace; bold marks the highest score per benchmark and Δ is EKLAV minus STD-CoT (extended).

B.2 Table reranking datasets

This paper uses eight datasets for table reranking: four in-domain benchmarks drawn from two prior benchmark papers, NQ-Tables Herzig et al. (2021) and MultiTableQA Zou et al. (2026), and four further datasets used exclusively for out-of-domain evaluation.

Training (in-domain). We use the NQ-Tables Herzig et al. (2021) training set for single-table, correct-type questions. For multi-table QA, we use only the HybridQA Chen et al. (2020b) portion of the MultiTableQA benchmark Zou et al. (2026), and only its training split. We keep the remaining datasets from the MultiTableQA benchmark for evaluation and still treat them as in-domain.

Testing (out-of-domain). For out-of-domain (OOD) evaluation, we draw on the IBM Research Table-Text IR Evaluation collection Doshi et al. (2026) and report results on AIT-QA Katsis et al. (2021), FeTaQA Nan et al. (2022), OTT-QA Chen et al. (2021), and Open-WikiTable Kweon et al. (2023) (Table 4); none of these datasets' examples appear in the fine-tuning corpus. OTT-QA is built on top of HybridQA's training split, and we evaluate on OTT-QA's own test split (Section 6).

Candidate pool construction. Both training and evaluation candidate pools are built with the same retrieval pipeline: BM25 Robertson et al. (1995), SPLADE-v3, and all-mpnet-base-v2 each retrieve a top-100 pool per query, and the three rankings are merged by reciprocal rank fusion (RRF, $k = 60$) into a single top-100 pool, with any missing gold table injected. Combining a lexical retriever (BM25) with two dense retrievers (SPLADE-v3, all-mpnet-base-v2) ensures candidates are not selected by semantic similarity alone, following the retrieval setup used by Rank1 Weller et al. (2025) for passage reranking.

B.3 PASSAGE RERANKING: PER-DOMAIN BREAKDOWN

The Qwen3-8B per-domain breakdown is given in the main text (Table 3), where the largest gains are on Earth Science (+6.1) and Psychology (+6.7), domains that require reasoning across diverse evidence types rather than lexical overlap, and substantial gains also appear on the coding domains LeetCode (+4.1) and Pony (+4.2). The single regression, TheoremQA-Theorems (-2.4), is a mathematical-proof domain where the teacher trace may carry procedural knowledge (proof structure, theorem application) that benefits from direct imitation.

The remaining three model tiers show similar profiles: EKLAV leads on the average and on the majority of domains. Regression domains only partially overlap across tiers: at 4B, the regressions are on Pony, AoPS, and TheoremQA-Theorems; at 9B, only Robotics regresses; at 14B, Pony and Sustainable Living regress. Pony regresses at both 4B and 14B but not at 8B or 9B. Regressions are few (one to three domains per tier) and no single domain regresses at every tier.

B.4 PASSAGE RETRIEVAL: FULL BREAKDOWN WITH BASELINES

We use the candidate pools released by Rank1 Weller et al. (2025): the BM25 Robertson et al. (1995) top-100 passages per BRIGHT query, exactly as published, rather than reconstructing them ourselves. All four training conditions rerank this same fixed pool, so any nDCG@10 difference reflects reranking quality rather than a difference in candidate recall. Our pointwise scoring formulation, generating a reasoning trace before a `true`/`false` verdict, is likewise inspired by Rank1's approach to test-time compute for reranking.

Table 9 gives the complete per-domain BRIGHT breakdown including published baselines. Table 11 gives the full consolidated view (training FLOPs and inference speed alongside accuracy).

Table 11: ***Consolidated results, full version.*** *Estimated training FLOPs and inference elapsed time.*

Model	Method	BRIGHT avg.	Estimated FLOPs	Inference time
Qwen3-4B	STD-COT	29.3	1.093×10^{19}	118.5 min (BRIGHT)
	EKLAV	31.5	7.553×10^{18}	79.8 min (BRIGHT)
Qwen3-8B	STD-COT	31.5	2.150×10^{19}	132.0 min (BRIGHT)
	EKLAV	34.2	1.470×10^{19}	95.5 min (BRIGHT)
GLM-Z1-9B	STD-COT	30.1	2.370×10^{19}	104.4 min (BRIGHT)
	EKLAV	33.2	1.661×10^{19}	88.3 min (BRIGHT)
Qwen3-14B	STD-COT	32.5	3.529×10^{19}	176.1 min (BRIGHT)
	EKLAV	34.7	2.689×10^{19}	139.0 min (BRIGHT)

Figure 2 shows the BRIGHT comparison by individual domain and model tier. EKLAV leads on the 12-domain average at every tier, though not on every domain within a tier; the regression domains and counts are listed in Section 5. Figure 3 plots the same averages against model size.

B.5 TABLE RERANKING: IN-DOMAIN COMPARISON

The in-domain comparison is given in the main text (Table 4). On the 0–100 nDCG@10 scale used throughout, $\bar{\Delta}_{\text{OOD}} = 3.05$ and $\bar{\Delta}_{\text{ID}} = 1.02$, giving $\bar{\Delta}_{\text{OOD}} - \bar{\Delta}_{\text{ID}} = 2.03$ (Section 6).

B.6 GENERATION LENGTH

On the analyzed Qwen3-8B passage-reranking sample (899 matched query-passage pairs across all twelve BRIGHT domains), EKLAV generates a mean of 198 words per trace against STD-COT's 312 words, a baseline-to-EKLAV word-count ratio of 1.58. The reduction holds in all twelve BRIGHT domains, with the largest ratio on AoPS ($2.3\times$) and the smallest on Biology ($1.4\times$).

Table 9: ***BRIGHT contextual baselines and paired results, StackExchange domains.*** *Scores are* 100 × nDCG@10. *Published rows provide context SU et al. (2025); Weller et al. (2025); our paired rows share the recipe in Section 4. Delta rows retain the supplied relative changes and can differ from ratios of rounded cells.*

Model	Bio	Earth	Econ	Psych	Rob	StackOv	Sus
BM25	19.2	27.1	14.9	12.5	13.5	16.5	15.2
BM25 on GPT-4o CoT	53.6	53.6	24.3	38.6	18.8	22.7	25.9
MonoT5-3B	16.0	24.0	17.7	19.5	8.0	10.5	19.5
RankLLaMA-7B	17.5	15.5	13.1	13.6	17.9	6.9	16.9
RankLLaMA-13B	21.6	19.1	16.3	14.0	15.7	7.7	18.5
Rank1-7B	48.8	36.7	20.8	35.0	22.0	18.7	36.2
Rank1-14B	49.3	37.7	22.6	35.2	22.5	20.8	33.6
Rank1-32B	49.7	35.8	22.0	37.5	22.5	21.7	35.0
Qwen3-4B (STD-CoT)	44.9	41.3	24.4	38.0	22.4	28.9	31.0
Qwen3-4B (EKLAV)	45.5	45.8	27.9	39.9	25.6	31.3	32.3
Δ vs. STD-CoT	+1.4%	+10.8%	+14.5%	+5.0%	+14.6%	+8.1%	+4.3%
Qwen3-8B (STD-CoT)	45.2	43.1	29.3	35.8	24.2	30.5	34.9
Qwen3-8B (EKLAV)	50.0	49.2	30.9	42.4	27.9	32.4	35.6
Δ vs. STD-CoT	+10.5%	+14.2%	+5.5%	+18.6%	+15.3%	+6.2%	+1.9%
GLM-Z1-9B (STD-CoT)	41.8	39.7	26.9	40.0	24.1	28.8	33.0
GLM-Z1-9B (EKLAV)	48.5	45.3	30.4	40.1	23.6	32.7	36.6
Δ vs. STD-CoT	+16.1%	+14.2%	+13.2%	+0.2%	−2.1%	+13.5%	+10.9%
Qwen3-14B (STD-CoT)	46.6	43.6	31.7	42.1	26.1	31.5	35.6
Qwen3-14B (EKLAV)	50.6	50.0	32.7	42.4	27.4	33.7	35.5
Δ vs. STD-CoT	+8.6%	+14.7%	+3.2%	+0.8%	+5.1%	+7.2%	−0.3%

Table 10: ***BRIGHT coding and theorem domains, with overall averages.*** *This continues Table 9. Scores use the same scale; the delta rows are the supplied relative percentage changes. The Qwen3-8B AoPS improvement is visible only at higher precision (Table 3).*

Model	Leet	Pony	AoPS	TheoQ	TheoT	Avg
BM25	24.4	7.9	6.0	13.0	6.9	14.8
BM25 on GPT-4o CoT	19.3	17.7	3.9	18.9	20.2	26.5
MonoT5-3B	17.2	29.2	7.1	20.3	12.0	16.8
RankLLaMA-7B	8.4	46.8	2.2	4.5	3.5	13.9
RankLLaMA-13B	8.8	31.1	1.7	4.4	4.9	13.7
Rank1-7B	12.7	31.2	6.3	23.7	37.8	27.5
Rank1-14B	17.7	33.2	8.4	22.5	41.4	28.7
Rank1-32B	18.8	32.5	10.8	22.9	43.7	29.4
Qwen3-4B (STD-CoT)	14.2	33.6	8.4	24.6	40.4	29.3
Qwen3-4B (EKLAV)	26.3	29.9	7.4	26.8	39.9	31.5
Δ vs. STD-CoT	+85.9%	−11.0%	−12.6%	+8.9%	−1.1%	+7.6%
Qwen3-8B (STD-CoT)	21.6	33.4	9.7	27.0	43.4	31.5
Qwen3-8B (EKLAV)	25.7	37.6	9.7	28.5	41.0	34.2
Δ vs. STD-CoT	+19.0%	+12.4%	+0.3%	+5.6%	−5.6%	+8.7%
GLM-Z1-9B (STD-CoT)	17.2	35.8	7.5	25.9	40.7	30.1
GLM-Z1-9B (EKLAV)	24.9	36.1	7.8	28.8	43.6	33.2
Δ vs. STD-CoT	+45.0%	+1.0%	+4.0%	+11.2%	+7.2%	+10.3%
Qwen3-14B (STD-CoT)	19.4	37.1	9.2	25.7	41.3	32.5
Qwen3-14B (EKLAV)	24.7	36.1	9.7	28.3	44.7	34.7
Δ vs. STD-CoT	+27.6%	−2.7%	+5.5%	+10.1%	+8.2%	+6.7%



Figure 2: BRIGHT nDCG@10 by domain and model tier (4B, 8B, 9B, 14B), the full breakdown behind Table 9. STD-COT *vs.* EKLAV.

B.7 INFERENCE EFFICIENCY

EKLAV has a lower recorded inference time for each model pair in the reported BRIGHT workload. The records do not specify timed workload counts or repeated timing trials. Elapsed times are reported in Table 11 and summarized in Section 7.

B.8 TRAINING LOSS CURVES

Figures 5–10 give training and evaluation loss curves for the Qwen3-4B, Qwen3-8B, and GLM-Z1-9B conditions; we do not have corresponding curves for Qwen3-14B. In interpreting these curves, absolute loss values across methods are not directly comparable because the supervised token sets differ (Section 3.3). STD-COT computes loss over both reasoning and answer tokens; EKLAV computes loss over answer tokens only. Differences in supervised token composition and weighting prevent the absolute loss levels from serving as a direct comparison of convergence quality across methods.

At the epoch boundary (step ≈3,000 of 6,000), EKLAV's training loss drops roughly an order of magnitude while its evaluation loss barely moves (Figure 8), a pattern consistent with the model memorizing answers for its specific training examples rather than improving its general answer-prediction ability. STD-COT's training loss shows a comparable but visibly smaller drop at the same boundary (Figure 7). This is consistent with the supervised answer being recoverable from the

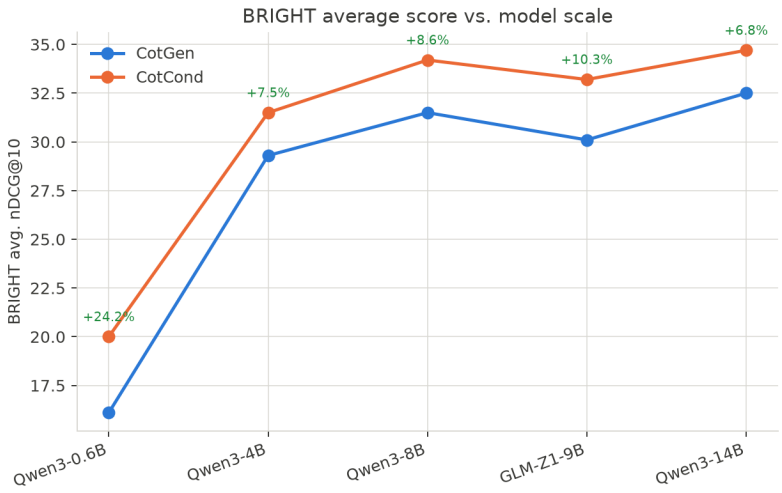


Figure 3: BRIGHT average nDCG@10 vs. model scale (4B, 8B, 9B, 14B). EKLAV *is above* STD-COT *at every size.*

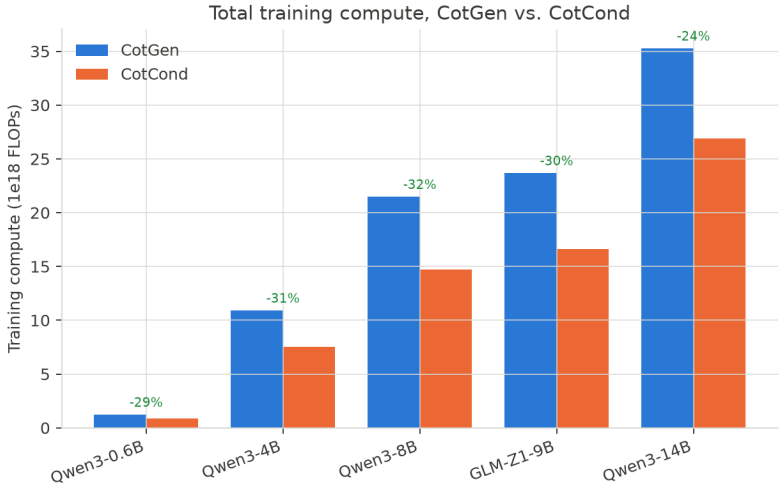


Figure 4: Reported training-compute estimates by tier (4B, 8B, 9B, 14B). The paired records show lower estimated compute and higher average BRIGHT scores under EKLAV. *These estimates use the* $6ND$ *proxy discussed in Section 7.*

training context under EKLAV (Section 3.2), and it means the near-zero training loss should not be read as evidence that the model has learned a general answer-prediction skill from the second epoch onward.

B.9 GENERATED OUTPUT LENGTH

Table 12 reports the average number of generated tokens per query for table reranking. Across all four benchmarks, EKLAV produces substantially shorter outputs than STD-COT, with reductions ranging from $1.50\times$ to $1.77\times$. The largest reduction occurs on NQ-Tables, where the average output length decreases from 2,403.9 to 1,361.7 tokens. These results show that the shorter-generation pattern observed in passage reranking also extends to table reranking.

On passage reranking, the same $1.58\times$ average reduction (Section 7) holds across all twelve BRIGHT domains, with the largest ratio on AoPS ($2.3\times$) and the smallest on Biology ($1.4\times$).

Table 12: ***Generated output length per query*** *(tokens), table reranking. Both methods generate reasoning before answering.* EKLAV *consistently produces shorter outputs.*

Benchmark	STD-COT	EKLAV	Ratio
SQA	3,318.5	2,145.3	1.55×
TAT-QA	3,704.2	2,473.4	1.50×
TabFact	3,716.1	2,454.3	1.51×
NQ-Tables	2,403.9	1,361.7	1.77×

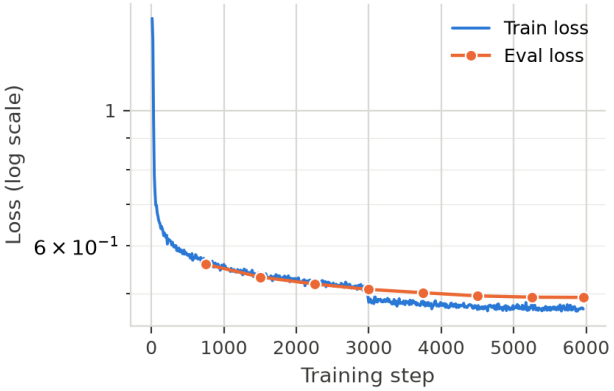


Figure 5: Training/eval loss – Qwen3-4B, STD-COT.

C ADDITIONAL ANALYSIS

The surface-style and voice-proximity analyses are given below (Appendix C.2).

C.1 EXPLORATORY GENERATIVE REASONING

The available exploratory summary covers five model families and seven mathematical-reasoning benchmarks, with 29 reported model–benchmark comparisons. EKLAV generates fewer reasoning tokens in 25 of 29 comparisons, with median baseline-to-EKLAV length ratio 1.5. It has lower pass@1 in 23 of 29 comparisons, with mean accuracy difference -5.9 percentage points.

This summary shows that shorter outputs do not imply better accuracy outside the reranking tasks. The supplied source does not include the 29 individual score cells, model and benchmark identities, or the complete training and evaluation configurations. We therefore retain these as an aggregate exploratory finding and do not treat them as a fully specified cross-domain experiment.

C.2 SURFACE STYLE AND VOICE

One might ask whether EKLAV's improvement arises from more varied or creative phrasing. On the contrary, EKLAV is *more* formulaic. On the 899 matched passage pairs, EKLAV reuses the same six-word opening in 94% of generations versus 45% for STD-COT, and its within-domain trigram Jaccard similarity is also higher (0.049 vs. 0.040). The improvements therefore cannot be attributed to greater surface diversity; EKLAV's gains come despite more stereotyped phrasing.

On table reranking, we measure lexical proximity to the base model using length-matched Jensen–Shannon divergence over 1,370 matched questions. EKLAV is 8% closer to the base model's token distribution than STD-COT (JSD 0.232 vs. 0.252, Wilcoxon $p = 1.9 \times 10^{-22}$), with higher bigram overlap (0.298 vs. 0.263). This suggests that removing trace imitation allows the student to retain more of its pretrained voice rather than adopting the teacher's stylistic patterns. NAIVE-SFT shows a much larger divergence from base (JSD 0.373), indicating that the proximity is specific to EKLAV's training with teacher context, not a general property of answer-only objectives.

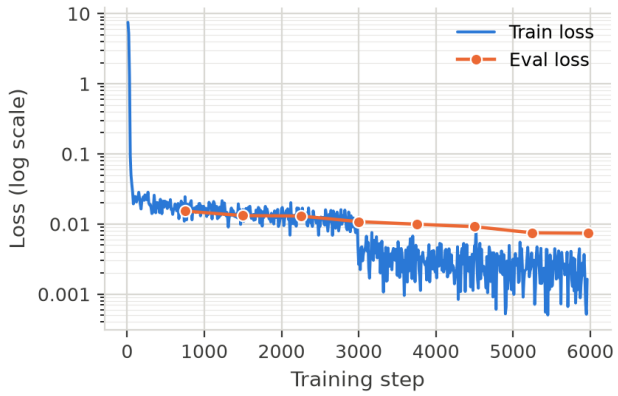


Figure 6: Training/eval loss – Qwen3-4B, EKLAV.

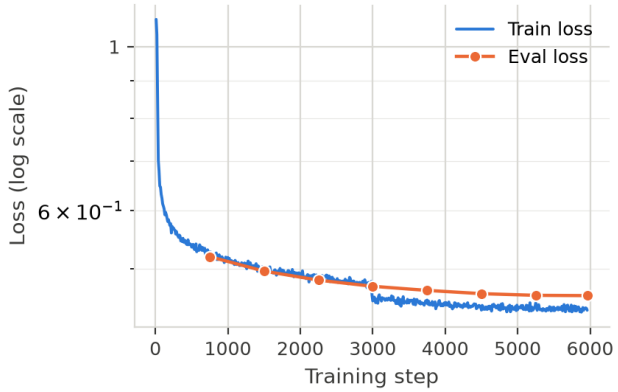


Figure 7: Training/eval loss – Qwen3-8B, STD-COT.

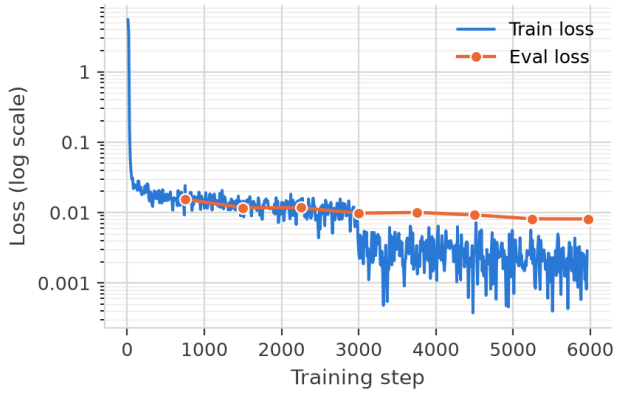


Figure 8: Training/eval loss – Qwen3-8B, EKLAV.

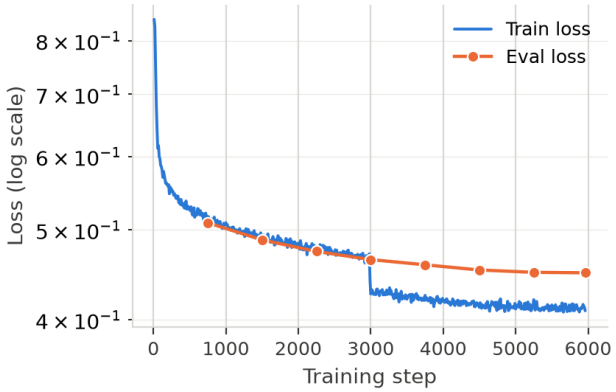


Figure 9: Training/eval loss – GLM-Z1-9B, STD-COT.

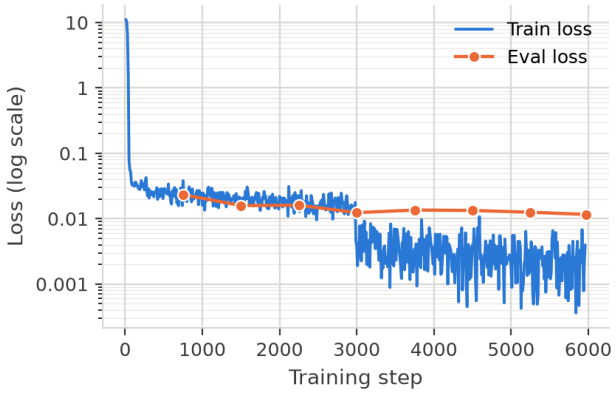


Figure 10: Training/eval loss – GLM-Z1-9B, EKLAV.