Three Regimes of Chain-of-Thought Utility for Student Models

Anonymous ACL submission

Abstract

Teacher chain-of-thought (CoT) is increasingly supplied to a smaller model as *input context*—in inference-time model cascades, retrieved-rationale prompting, and reasoning-compression pipelines—yet it is unclear how much of a teacher rationale, and which parts, a downstream model actually needs. We study this directly: we fine-tune a student to produce the answer *given* a teacher's CoT as input, then truncate that CoT with controlled strategies and measure how accuracy responds. The student does not generate its own CoT and the teacher rationale is present at inference, which isolates the utility of a *provided* rationale from the student's own reasoning. Across four student configurations and seven reasoning benchmarks, this utility follows a consistent three-regime structure: under a tight input budget the last CoT sentences matter far more than the first (a recency gap up to 72.5 points on GSM8K); at moderate in-domain budgets the strategies converge, revealing heavy redundancy; and under domain shift the full teacher CoT can *hurt*, with moderate truncation beating it by up to 7.1 points. Controls (numeral masking, answer-sentence removal) show the student exploits the late *computation* in the rationale rather than a copied answer, and an oracle leave-one-out analysis locates per-sentence utility at the end of the chain. The practical message is simple: before compressing a teacher rationale for a downstream model, characterize where its usable information sits—keeping just the last few sentences is already a strong baseline.

1 Introduction

Chain-of-thought (CoT) prompting has transformed the reasoning capabilities of large language models (LLMs). By eliciting step-by-step reasoning traces, CoT unlocks dramatic accuracy gains on mathematical, logical, and multi-hop reasoning tasks (Wei et al., 2022; Kojima et al., 2022). However, this reasoning ability comes at a steep cost: a single CoT trace can span hundreds to thousands of tokens, inflating inference latency and memory by an order of magnitude over direct prediction. As reasoning models are deployed at scale, the question of how to compress these verbose traces—without sacrificing accuracy—has become a pressing practical and scientific challenge.

Yet our understanding of teacher CoT as *input to a downstream model* remains remarkably shallow. A growing set of systems do not let a small model reason on its own but instead hand it a teacher's rationale as context—model cascades that route a question to a cheap answerer, retrieved or cached rationales, and prompt-compression pipelines that shorten reasoning before passing it on. Most such work either keeps the teacher rationale intact or proposes increasingly elaborate ways to compress or select it. But before asking which compression mechanism is best, there is a more basic question: *which parts of a teacher CoT does a downstream model actually benefit from?* Without an answer, compression risks optimizing the wrong object entirely—preserving redundant reasoning, or passing on source-domain patterns the consumer cannot use.

In this paper, we study teacher-CoT utility through controlled truncation. Rather than proposing a new compression module, we vary the retained budget K and the selection strategy, and measure how a student answerer responds when the truncated teacher CoT is supplied to it as input. Crucially, the student is fine-tuned to predict the answer *given* this CoT and never produces a chain of its own, so we measure how usable a *provided* rationale is, not how well a student learns to reason unaided. Across in-domain and cross-domain settings, this CoT utility follows a three-regime structure (Figure 1). Under a *tight budget* (the recency regime), position dominates: keeping the *last* K sentences vastly outperforms keeping the *first* K, by as much as 72.5 points on GSM8K. Un-

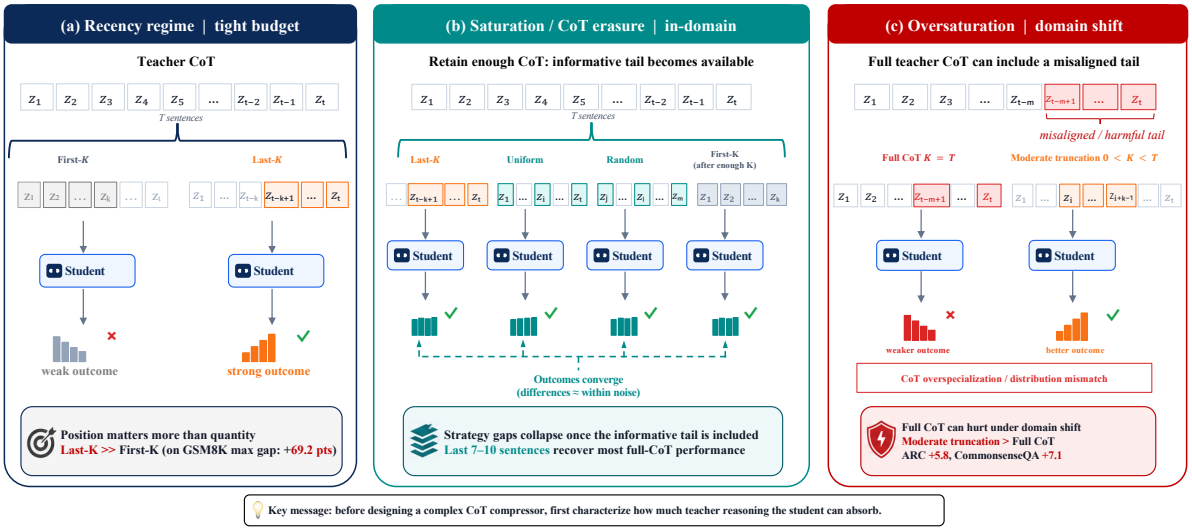

Figure 1: Three regimes of student-facing teacher-CoT utility as the retained budget K grows: **(a)** under a tight budget position dominates (last-$K \gg$ first-K); **(b)** at a moderate in-domain budget strategies converge and the remainder is redundant; **(c)** under domain shift full CoT can hurt, so moderate truncation wins.

der a *moderate in-domain budget* (the saturation regime), this advantage erodes and the strategies converge, revealing substantial redundancy in the retained chain from the student's perspective. Under *domain shift* (the oversaturation regime), the relationship inverts—full teacher CoT can actively *hurt* the student, and moderate truncation outperforms it by up to 7.1 points on CommonsenseQA and 5.8 on ARC, because domain-specific CoT fine-tuning impairs cross-domain reasoning.

The obvious objection is that recency is trivial—the answer simply sits at the end. We treat this as the central threat to validity and rule it out directly: masking the numerals in the retained sentences is devastating (-56 to -73 points), whereas deleting the sentence that contains the gold answer changes nothing, so the student relies on the late *computation* rather than a copied answer. The effect is not an artifact of one dataset, model, or granularity: it replicates across all four student configurations, across math, science, and commonsense benchmarks, and at token granularity. What weakens it is not a different model but a different kind of task—it fades precisely where the teacher chain stops being a single coherent computation (Section 7.4).

To organize these findings, we introduce a lightweight information-theoretic lens, Cumulative Information Accumulation (CIA; Section 5)—an explanatory tool, not a law of reasoning—that accounts for why later steps dominate at low budgets, why strategy gaps collapse once enough context is retained, and why domain mismatch can make additional teacher reasoning counterproductive.

The practical implication is simple: before investing in complex step-ranking or importance-scoring mechanisms to compress a rationale passed between models, first characterize the consumer's utility curve with respect to the teacher CoT—in domain the last 7–10 sentences already recover most of full-CoT accuracy, while under transfer the full CoT may be the wrong thing to pass on. We establish this across four student configurations (Qwen2.5-3B/1.5B, Llama-3.2-3B, and a Qwen2.5-7B→3B teacher–student pair) and seven benchmarks spanning math, science, STEM, commonsense, and symbolic reasoning.

Our contributions are threefold:

- We show that *a full teacher CoT is not the best rationale to hand a downstream model*: its utility as input follows three regimes—recency, saturation, and oversaturation—in which providing more teacher reasoning is sometimes strictly worse.
- We argue that compressing a rationale for a consumer model should first estimate *what the consumer can use* rather than teacher-side informativeness, and find that sentence position is a surprisingly strong proxy for it when reasoning is coherent.
- We establish these claims as robust—across four student configurations, seven benchmarks, and a control battery against answer leakage—and show that simple positional truncation matches learned compression in domain and beats full

CoT under transfer.

2 Related Work

Chain-of-thought reasoning and its analysis. Scratchpads (Nye et al., 2021) and CoT prompting (Wei et al., 2022; Kojima et al., 2022) showed that intermediate steps sharply improve LLM reasoning, with later extensions to self-consistency (Wang et al., 2023b) and tree search (Yao et al., 2023), and modern reasoning models trained via RL or test-time scaling (DeepSeek-AI, 2025; Muennighoff et al., 2025) now emit very long traces. A complementary strand asks what makes CoT work: empirically, which parts of a demonstration matter (Wang et al., 2023a) and when CoT helps at all (Sprague et al., 2025); and theoretically, that CoT extends transformer expressivity by an amount growing with the length of intermediate generation (Feng et al., 2023; Merrill and Sabharwal, 2024), echoing the broader compression–prediction link (Delétang et al., 2024; Tishby and Zaslavsky, 2015). We ask the orthogonal, student-facing question of *what fraction of a completed trace a student can use*, and organize the answer with our CIA framework.

Reasoning distillation. Building on knowledge distillation (Hinton et al., 2015), a line of work transfers reasoning to smaller models through multi-task rationale distillation (Hsieh et al., 2023), Fine-tune-CoT (Ho et al., 2023), and specialization or decomposition of student training (Magister et al., 2023; Fu et al., 2023; Shridhar et al., 2023). These methods train the student to *generate* reasoning and discard the teacher at test time; we instead keep the teacher's rationale as input and ask how much of it a downstream answerer can use—a question about inference-time rationale provision rather than about internalizing reasoning into the student's weights.

CoT compression, efficiency, and latent reasoning. Many methods shorten reasoning—concise prompting (Xu et al., 2025), self-generated rationales (Zelikman et al., 2022), learned token skipping or shortening (Xia et al., 2025; Kang et al., 2025), and perplexity-based prompt compression (Jiang et al., 2023, 2024), with Jin et al. (2024) showing step length itself affects accuracy—while a parallel line replaces explicit text with continuous or implicit computation (Hao et al., 2024; Deng et al., 2023; Cheng and Van Durme, 2024; Goyal et al., 2024). These ask *how much* to compress, or *how* to represent reasoning; the prompt-compression line is closest to our setting, since it too shortens context fed to a model, but it optimizes a teacher-side objective. We instead ask *what a downstream consumer actually uses*, and find that simple positional truncation (last-K at $K^* \approx 7$) is a strong baseline without any learned scoring.

3 Experimental Setup

Our pipeline has four stages (Figure 2): the teacher generates and caches a CoT per training question; six independent students are trained, one per selection strategy, on *mixed-budget* truncations of that CoT; each student is then evaluated over a sweep of test-time budgets K; and all are tested in-, near-, and cross-domain, yielding the accuracy-vs-K curves that expose the three regimes.

3.1 Models and Training

Primary configuration. Our primary configuration uses Qwen2.5-3B-Instruct (Team, 2024) as both teacher and student model. The teacher generates chain-of-thought reasoning with greedy decoding (`max_new_tokens` $= 512$).

Cross-model configurations. To test the generality of the regime structure, we additionally evaluate three student configurations that vary scale, architecture, and the teacher–student gap: Qwen2.5-1.5B-Instruct (a smaller same-family student), Llama-3.2-3B-Instruct (Llama Team, AI @ Meta, 2024) (a different architecture), and the primary 3B student paired with a larger Qwen2.5-7B-Instruct teacher. All use the same per-strategy training and evaluation pipeline described below.

Per-strategy training. We train six independent student models—one per strategy (No-CoT, Full-CoT, First-K, Last-K, Uniform, Random)—each initialized from the same base model and fine-tuned via LoRA (Hu et al., 2022) with rank r=16, α=32, targeting $\{q_{\text{proj}}, v_{\text{proj}}\}$ modules, for 3 epochs. The No-CoT student is trained on question $\rightarrow$ answer pairs and the Full-CoT student on question + full teacher CoT $\rightarrow$ answer; the four truncation students are each trained on question + strategy-truncated CoT. Crucially, every truncation student sees *mixed-K data*—for each example the CoT is truncated at a random $K \sim \text{Unif}\{3, 5, 7, 10, 15, 20\}$ under the student's assigned strategy—so that a single adapter generalizes across the full budget sweep at test time. In every case the teacher CoT is part of the student's *input* and the loss falls only on the

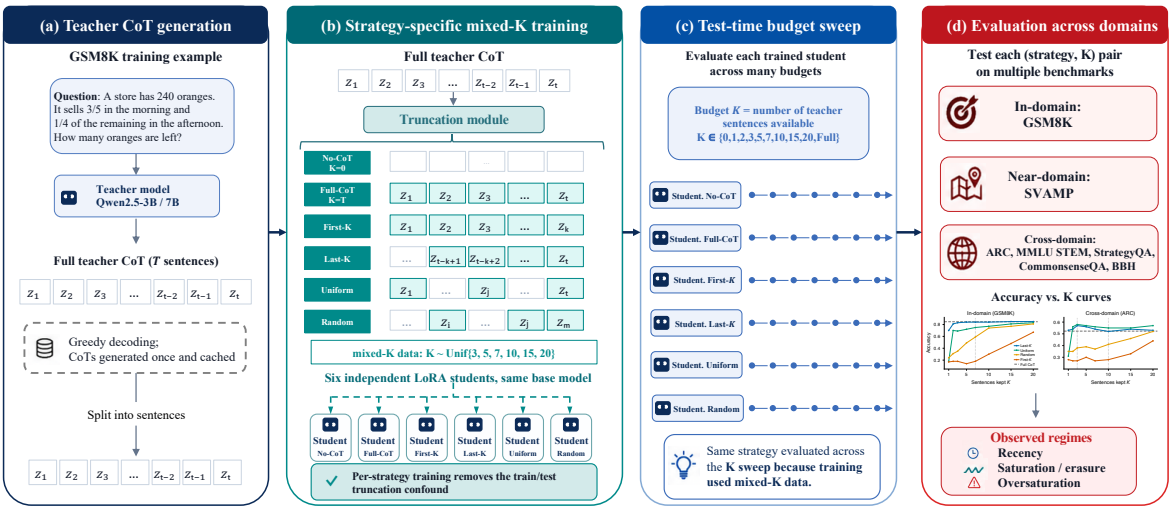

Figure 2: Experimental pipeline. **(a)** The teacher (Qwen2.5-3B/7B) generates a full CoT per GSM8K question with greedy decoding; CoTs are cached and sentence-segmented. **(b)** A truncation module produces strategy-specific data, and six LoRA students (No-/Full-CoT, First-/Last-K, Uniform, Random) are trained with *mixed-K* truncations ($K \sim \mathrm{Unif}\{3, 5, 7, 10, 15, 20\}$), which removes the train/test truncation confound. **(c)** Each student is swept over budgets $K \in \{0, \ldots, 20, \mathrm{Full}\}$. **(d)** Evaluation spans in-domain (GSM8K), near-domain (SVAMP), and cross-domain (ARC, MMLU STEM, StrategyQA, CommonsenseQA, BBH) tasks; the resulting accuracy-vs-K curves reveal recency, saturation, and oversaturation.

answer tokens: the student learns to answer *conditioned on* a provided rationale, not to generate one, and the teacher CoT is supplied at test time as well. Our setting is thus inference-time rationale provision (cascades, cached or compressed rationales), not CoT distillation; this isolates the utility of a *provided* rationale at the cost of requiring the teacher at inference (see Limitations).

This design avoids the confound of training on full CoT but testing on truncated CoT, while remaining tractable (6 LoRA adapters rather than $6 \times 9 = 54$).

Training uses all 7,473 GSM8K training examples (the complete training split). A 10% hold-out validation set (stratified across all strategies) is used for early stopping. Training hyperparameters: batch size 2, gradient accumulation 8, learning rate 2×10^{-4}, max sequence length 768, FP16.

3.2 Truncation Strategies

Given a CoT with T sentences, a selection strategy S_K retains exactly $K \leq T$ sentences. We evaluate four strategies: *First-K* keeps the leading sentences $\{z_1, \ldots, z_K\}$; *Last-K* keeps the trailing sentences $\{z_{T-K+1}, \ldots, z_T\}$; *Uniform* keeps K evenly spaced sentences (via $\texttt{linspace}(0, T-1, K)$); and *Random* keeps K sentences drawn uniformly at random (seed 42). For each strategy we sweep $K \in \{0, 1, 2, 3, 5, 7, 10, 15, 20\}$ and also evaluate the full CoT ($K{=}T$) and no-CoT ($K{=}0$) baselines.

3.3 Datasets

We evaluate on seven benchmarks spanning in-domain, near-domain, and cross-domain settings:

- **GSM8K** (Cobbe et al., 2021): Math word problems ($n{=}1319$ test examples). *In-domain*: the student is trained on GSM8K CoT pairs.
- **SVAMP** (Patel et al., 2021): Arithmetic word problems ($n{=}300$). *Near-domain*: similar reasoning to GSM8K.
- **ARC-Challenge** (Clark et al., 2018): Science QA ($n{=}1172$). *Cross-domain*: zero-shot transfer from GSM8K training.
- **MMLU STEM** (Hendrycks et al., 2021): 15 STEM subjects ($n{=}2511$). *Cross-domain*: zero-shot transfer.
- **StrategyQA** (Geva et al., 2021): Implicit multi-hop strategic reasoning ($n{=}1370$). *Cross-domain*: tests non-math reasoning.
- **CommonsenseQA** (Talmor et al., 2019): Multiple-choice commonsense reasoning ($n{=}2448$). *Cross-domain*: tests commonsense transfer.
- **BBH** (Suzgun et al., 2023): BIG-Bench Hard, a mixture of 24 challenging reasoning subtasks ($n{=}750$). *Cross-domain*: tests multi-task robustness.

Table 1: Recency effect on GSM8K (n=1319): the Last-K − First-K accuracy gap. Identical budgets at opposite ends of the chain differ by up to 69 points.

K	1	2	3	5	7	10
First-K	.17	.18	.18	.14	.18	.30
Last-K	.70	.81	.83	**.84**	.84	.84
Gap	+.52	+.63	+.65	**+.69**	+.66	+.55

The student is trained only on GSM8K; cross-task evaluation tests zero-shot generalization of the compression phenomenon.

3.4 Evaluation Protocol

We use exact-match accuracy against gold answers. For GSM8K, we extract the final numerical answer via regex matching on #### delimiters. For ARC and MMLU (multiple choice with options A–D), we extract the predicted option letter from the first line of the student's generation via regex matching (`[A-D]`); if no letter is found, we fall back to scanning subsequent lines. All evaluations use greedy decoding with `max_new_tokens`=64 for the student.

Cross-domain prompt alignment. To avoid confounding from prompt mismatch, we use task-specific prompt templates for each dataset (full templates in Appendix A.16). Batched inference (batch size 16) is used; teacher CoTs are generated once and cached, then shared across all truncation strategies to ensure fair comparison.

4 The Recency Effect and CoT Erasure

Before asking whether a sophisticated selector can identify the "important" steps of a teacher rationale, we characterize the utility curve of the rationale itself. We do so in the cleanest possible setting—in-domain GSM8K—using six independently trained Qwen2.5-3B-Instruct students, one per strategy, so that any difference we observe reflects what the student can *learn* to use rather than a train/test truncation mismatch (Section 3).

Recency. Under tight budgets, *where* a sentence sits in the chain matters far more than how many sentences are kept. Retaining the last five sentences yields 83.6%; retaining the first five yields only 14.4%—a 69.2-point gap from the same budget spent at opposite ends of the chain (Table 1; full K-sweep matrix and curves in Appendix Table 23 and Figure 2d). The asymmetry is steep even at K=1, where a single trailing sentence (69.5%) already recovers most of full-CoT accuracy while a single leading sentence (17.4%) barely clears the no-CoT floor of 15.0%. This does not mean early reasoning is worthless in absolute terms; it means that, as *provided context under a budget*, the steps nearest the answer dominate those that set the problem up.

Erasure. The mirror image of recency is that, once enough of the chain is retained, the identity of the retained subset stops mattering—an effect we call *CoT erasure*. The spread between strategies falls from 69 points at K=5 to 1.3 points once the full chain is used. Convergence is nonetheless slow and asymmetric: at K=15 the strategies still span 37 points because First-K has yet to reach the answer-bearing tail, whereas Last-K is already saturated by K=5. Practically, this means aggressive compression is nearly free in domain—last-7 reaches 83.9%, within 0.9 points of full CoT at roughly 1.5× fewer input tokens—provided the retained budget is taken from the end. We verify that this reflects genuine use of the late computation rather than answer copying through seed, significance, leakage, and answer-proximity controls (Section 7, Appendix A.16).

5 An Interpretive Framework for CoT Utility Regimes

We interpret our findings through a lightweight framework, *Cumulative Information Accumulation* (CIA). The goal is explanatory, not theoretical: we organize the observed regimes and state the assumptions that would make them plausible, labeling key statements as *modeling hypotheses* rather than theorems.

Let X be the question, Y the answer, $Z = (z_1, \ldots, z_T)$ the teacher CoT, and S_K a strategy retaining $K \leq T$ sentences; student accuracy is $\mathcal{A}(S_K) \triangleq \mathbb{E}[\mathbf{1}\{f_S(X, Z_{S_K}) = Y\}]$. Since the teacher generates autoregressively, define the per-sentence causal contribution $\psi_t \triangleq I(z_t; Y \mid X, z_{<t})$.

Hypothesis 1 (CIA Property)**.** *The CoT distribution satisfies CIA if ψ_t is non-decreasing: $\psi_1 \leq \cdots \leq \psi_T$.*

This is plausible when early sentences restate the problem and later ones carry the decisive computation; it is a property of the teacher-CoT distribution that may vary across tasks.

Recency. If ψ_t increases, then $\sum_{t=T-K+1}^{T} \psi_t > \sum_{t=1}^{K} \psi_t$ (each $\psi_{T-K+i} > \psi_i$), explaining why Last-K beats First-K at low budgets: with few

steps affordable, the student benefits most from steps closest to the answer.

Saturation. Beyond a saturation point K^*, accuracy becomes insensitive to which sentences are kept: once any strategy includes all high-ψ_t sentences, residual differences vanish. On GSM8K this occurs near K=Full (range 1.3%), though slowly—at K=15 First-K still trails because it has not yet accumulated the high-information tail.

Oversaturation. The in-domain framework assumes $\psi_t \geq 0$. Under mismatch (cross-domain transfer or teacher–student style mismatch, Section A.20) this can fail: define signed utility $\phi_t^{\times} \triangleq \mathcal{A}(X, z_{\leq t}; f_S) - \mathcal{A}(X, z_{<t}; f_S)$, which—unlike ψ_t—can be negative. If $\phi_t^{\times}$ has a negative tail at large t, truncating before that tail beats full CoT, explaining oversaturation.

Oracle validation. Our leave-one-out analysis (Section 6.1, Figure 3) directly supports CIA: per-sentence utility is concentrated at the end of the CoT, with early sentences contributing near-zero marginal utility.

6 Mechanism Analysis

The preceding sections establish the three-regime structure behaviorally; we now ask whether it is visible in the *per-sentence* information content of the chain itself. If recency is real, the marginal value of a sentence should rise toward the end of the chain. We test this directly with a leave-one-out (LOO) oracle, and find that it does.

6.1 Per-Sentence Utility Concentrates at the Tail

For each test example with CoT $Z = (z_1, \ldots, z_T)$, we evaluate the student's accuracy with all sentences present, then with each sentence z_t individually removed. The *per-sentence utility* is defined as:

$$u_t = \mathcal{A}(Z) - \mathcal{A}(Z \setminus \{z_t\}), \qquad (1)$$

where $\mathcal{A}$ denotes correctness on that example. Positive u_t indicates that removing z_t hurts accuracy; negative u_t indicates that z_t is actively harmful.

Figure 3 shows the average utility profile across all GSM8K test examples (Full-CoT student, n=1319). The profile is consistent with the CIA hypothesis (Section 5): per-sentence utility is low for early sentences (problem restatement) and rises toward the end of the CoT (computation and answer derivation). The mean leave-one-out utility of the second half of the CoT is 1.67× that of the first half, and the Spearman rank correlation between sentence position and utility is positive and significant (ρ=0.02, p<0.001 over 33,224 sentence removals); it is small because, with a high-accuracy student, removing any single sentence rarely flips the answer, though the direction robustly favors late sentences. This provides mechanistic evidence that the recency regime reflects information concentration in late segments rather than positional bias or answer leakage.

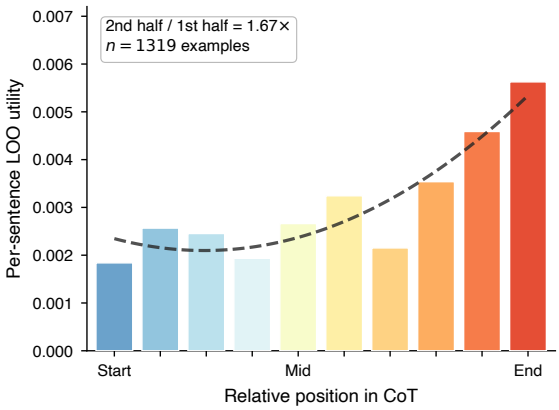

Figure 3: Oracle leave-one-out per-sentence utility profile on GSM8K. Each point shows the average accuracy drop when sentence at position t is removed. Utility is concentrated at the end of the CoT, providing mechanistic grounding for the recency regime.

7 Experiments

Section 4 established recency and erasure in domain. We now ask how far that structure carries. After confirming that the in-domain effect is robust rather than an artifact, we follow the teacher chain out of its training domain—where a third regime appears—then test the structure across four student configurations and across non-mathematical tasks, and finally compare positional truncation against learned alternatives. Full per-K matrices and the complete control battery are in Appendix A.16.

The in-domain effect is not an artifact. A battery of controls confirms that recency reflects genuine use of late reasoning. The gap is stable across three random seeds and significant under McNemar's test (p<0.001, Cohen's h≥1.17). Masking the numerals in the retained sentences collapses accuracy by 56–73 points, yet deleting the sentence that contains the gold answer changes nothing—so the student exploits the late *computation*, not a visible answer. A summary baseline and a perplexity baseline (below) further rule out trivial explana-

tions.

7.1 Overspecialization: CoT Training, Not LoRA

Before turning to transfer, we isolate a confound that shapes the cross-domain story. A GSM8K-tuned student could fail out of domain either because chain-of-thought fine-tuning overfits source-domain reasoning, or simply because LoRA fine-tuning erodes the base model's general competence. Comparing three models with full CoT separates the two. The unadapted base model scores 58.0% on ARC and 46.8% on MMLU; a student LoRA-tuned *without* CoT actually improves on it (74.7% and 58.4%), ruling out generic LoRA degradation. The Full-CoT student, by contrast, falls to 52.3% and 36.9%—below even the base model. The damage is therefore specific to training on chain-of-thought: optimizing for GSM8K rationales teaches arithmetic-verification habits that actively misfire on science and STEM questions. We refer to this effect as *CoT overspecialization* and return to it throughout.

7.2 Cross-Task Transfer and Oversaturation

We next apply each GSM8K-trained student, unchanged, to ARC-Challenge (n=1172) and MMLU STEM (n=2511)—a pure zero-shot transfer in which the teacher supplies CoT in the same GSM8K style. Two patterns recur. The recency ordering survives transfer: on ARC, Last-K again dominates First-K, peaking at $+30.0$ points at K=3 (p<0.001). But a new regime appears that was absent in domain. Moderate truncation now *beats* full reasoning—Uniform at K=3 reaches 58.1% versus the Full-CoT student's 52.3% (p<0.001)—because the later, domain-specific portions of the teacher chain carry patterns the student cannot transfer. This oversaturation compounds with overspecialization: the No-CoT student given full CoT reaches 74.7%, far above any CoT-trained student. MMLU shows the same qualitative picture with attenuated magnitude (the recency gap peaks at only $+12.5$ points, Cohen's h=0.28), and the more even Uniform strategy slightly outperforms Last-K, suggesting that multi-subject reasoning rewards spreading the budget rather than concentrating it at the tail.

Table 2 draws these together: recency is universal, but oversaturation emerges only under domain shift and fades as the target task approaches the training domain. A near-domain check on SVAMP (Appendix A.16) makes the point from the other side—recency replicates, convergence is faster, and no oversaturation appears, exactly as expected when source and target reasoning coincide.

Table 2: The three regimes across tasks. No-CoT is the no_cot student at K=0; Full CoT is the full_cot student at K=Full. Oversaturation appears only under domain shift and weakens with domain distance.

Property	GSM8K	ARC	MMLU
Full CoT accuracy	84.7%	52.3%	36.9%
No-CoT baseline	15.0%	31.4%	26.6%
Max recency gap	$+69.2\%$	$+30.0\%$	$+12.5\%$
Convergence K^*	Full	$\sim$10	$\sim$7
Oversaturation?	No	Yes ($+5.8\%$)	Marg. ($+2.9\%$)

7.3 Robustness Across Models

To test whether the structure is specific to one architecture, we rerun the full pipeline on three further configurations: a smaller same-family student (Qwen2.5-1.5B), a different architecture (Llama-3.2-3B), and a scale-mismatched pair in which a Qwen2.5-7B teacher supplies CoT to the 3B student.

The recency ordering holds in every configuration at all budgets up to K=10 (Appendix Table 27), and is in fact largest for the cross-architecture Llama student ($+72.5$ points at K=3), confirming that it is not a Qwen idiosyncrasy. Scale modulates the effect in interpretable ways. The 1.5B student shows the same shape but a muted gap ($+26.5$ at K=5) and lower absolute accuracy, consistent with limited capacity to exploit any reasoning. The scale-mismatched Qwen7B→3B pair is the most revealing: the stronger teacher lifts in-domain accuracy (90.4% full CoT, versus 84.7% same-scale), yet the teacher–student gap also produces clear cross-domain oversaturation on ARC (last-3 at 75.7% versus 73.0% for full CoT). Architecture changes absolute numbers but not the regime structure—Llama reaches 83.1% in-domain and exhibits the same overspecialization out of domain. Full per-strategy matrices appear in Appendix A.17.

7.4 Where the Structure Weakens: Non-Math Reasoning

Does the structure depend on the highly sequential nature of arithmetic? We probe three non-mathematical benchmarks using the GSM8K-trained students: StrategyQA (n=1370), CommonsenseQA (n=2448), and BBH (n=750).

Table 3: Non-math transfer (GSM8K-trained students). Recency gap (Last-K − First-K) at K=5 and oversaturation (best truncated − full CoT). The structure is strongest where reasoning is most coherent and computation-like.

Dataset	Last-K (5)	First-K (5)	Recency	Oversat.
StrategyQA	.616	.498	+11.8	−0.1
CommonsenseQA	.524	.242	+28.2	+7.1
BBH	.345	.339	+0.6	+2.4

The answer is that coherence, not arithmetic per se, drives the effect (Table 3). CommonsenseQA, whose questions admit a single coherent reasoning line, shows a strong recency gap (+28.2 points) and the largest non-math oversaturation (+7.1 points), indicating that GSM8K's arithmetic rationales transfer especially poorly here. StrategyQA, which mixes logical inference with world knowledge, shows a moderate gap (+11.8). BBH, an aggregate of 24 heterogeneous subtasks, shows almost no structure at all (+0.6): averaging over disparate reasoning styles washes out any consistent positional signal. The three-regime structure is therefore sharpest precisely where the teacher chain is a single coherent computation and dissolves as that coherence breaks down—a boundary condition we make explicit rather than hide.

7.5 Comparison to Learned and Summary Baselines

Finally, we ask whether more sophisticated selection beats simple positional truncation. It does not, in any of three comparisons (full tables in Appendix A.16). Teacher-generated summaries, evaluated on the Full-CoT student, reach 72–79% on GSM8K—well above naive truncation of the same student (15–47%), but still 4–11 points below the per-strategy Last-K student (83–85%), so specialization to the truncated distribution matters more than information-preserving summarization. Perplexity-based compression fares no better than position: at ratio 0.2, LLMLingua's highest-perplexity sentences yield 17.5% versus 82.6% for last-N, and sentence-level perplexity selection is statistically indistinguishable from First-N. In short, what predicts a sentence's usefulness to the consumer is its position, not its estimated informativeness.

8 Discussion

A natural reading of our results is discouraging for elaborate selectors, and we think it is correct. Above the erasure point, accuracy is nearly indifferent to *which* sentences survive, so little remains to optimize, and neither summaries nor perplexity-based selection beat last-K in our comparisons. The operative variable is position, and position is free. We do not claim no selector can help—only that demonstrating a gain first requires placing the budget below the erasure point, where selection still has purchase.

The cross-domain picture is more surprising. A student fine-tuned on GSM8K rationales becomes *worse* at using teacher CoT elsewhere than one that never saw CoT (52.3% vs. 74.7% on ARC); since the No-CoT student also beats the base model, the cause is CoT fine-tuning installing domain-specific habits, not LoRA eroding competence. This oversaturation is governed by *distribution alignment*, not domain labels: substituting the dataset's human-written GSM8K rationales for self-generated ones reintroduces oversaturation in domain (+7.2 vs. −0.8 points), and the effect should arise whenever teacher and student diverge in style or scale—as it does in our 7B→3B pair. It is sharpest where the chain is a single coherent computation and dissolves where it is not (strong on GSM8K, weak on commonsense, absent on the BBH mixture), exactly as the CIA reading predicts.

Practically, last-K is a strong in-domain default (last-7: 83.9% at ∼1.5×) whose exact budget barely matters above erasure, while under domain shift one should prefer moderate truncation ($K \approx 3$–10) and be wary of committing domain-specific CoT to the weights at all.

9 Conclusion

We asked which parts of a teacher rationale a student answerer can use when the rationale is supplied to it as input, and found a structured answer: the tail dominates at tight budgets, the strategies converge at moderate in-domain budgets, and under domain shift the full CoT can be *worse* than moderate truncation. The lesson is that a full teacher rationale is not automatically the best thing to pass downstream; before building a more elaborate compressor for reasoning shared between models, one should first chart this *rationale utility curve*—simple positional truncation is already a strong baseline to beat.

10 Limitations

Setting: provided rationale, not distillation. Throughout, the student is trained and evaluated to answer *given* a teacher CoT as input; it does not generate its own reasoning, and the teacher is therefore required at inference. Our findings concern inference-time rationale provision and compression—model cascades, retrieved or cached rationales, and prompt-compression pipelines—and do *not* directly speak to classic CoT distillation, where the student must reason unaided at test time. Whether the same positional structure governs what a student should be trained to *generate* is an open question we do not address.

Truncation granularity. Our primary analysis uses sentence-level truncation. Token-level truncation confirms that the recency effect persists at sub-sentence granularity, but selection by finer semantic units (e.g., individual reasoning steps, named predicates, or equation spans) may behave differently and is not evaluated here.

Training distribution. All students are trained on GSM8K CoT only; cross-task evaluation is zero-shot. The CoT overspecialization effect may therefore partly reflect this specific single-domain training distribution rather than a universal property of rationale transfer. We do not evaluate students trained on mixed-domain CoT, which could attenuate overspecialization.

Model scale. We replicate the regime structure across four student configurations (Qwen2.5-3B, Qwen2.5-1.5B, Llama-3.2-3B, and a Qwen2.5-7B teacher with a 3B student), all in the 1.5B–7B range. Whether the same regimes hold for substantially larger students (e.g., 13B–70B) or for models trained with reinforcement learning on long reasoning traces remains open.

Interpretive framework, not a theory. The Cumulative Information Accumulation (CIA) hypothesis is an organizational tool, not a proved property of teacher-CoT distributions. It rationalizes the observed regimes but does not predict the saturation point K^* or the magnitude of oversaturation a priori. Our oracle leave-one-out analysis is directionally consistent with CIA (second-half per-sentence utility is $1.67\times$ the first half; Spearman $\rho{=}0.02$, $p{<}0.001$), but the effect size is small because, with a high-accuracy student, removing any single sentence rarely flips the answer—useful information is distributed and redundant across the late CoT rather than concentrated in a few individually critical sentences.

Baseline coverage. We compare positional truncation against teacher-summary compression and perplexity-based selection (LLMLingua), but not against learned importance scorers or step-level rationale rankers. Stronger learned selectors could outperform Last-K truncation in regimes we do not study. Our claim is therefore that simple truncation is a *strong and underrated baseline*, not that it is optimal.

Task scope. Our seven benchmarks emphasize math, science, STEM multiple-choice, commonsense, and strategic reasoning. The three-regime structure is most pronounced for tasks with coherent, structured reasoning (math) and attenuates on diverse-subtask mixtures (BBH). Free-form generation, long-context, and multilingual reasoning are not covered.

References

Jeffrey Cheng and Benjamin Van Durme. 2024. Compressed chain of thought: Efficient reasoning through dense representations. *arXiv preprint arXiv:2412.13171*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI. 2025. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*.

Grégoire Delétang, Anian Ruoss, Paul-Ambroise Duquenne, Elliot Catt, Tim Genewein, Christopher Mattern, Jordi Grau-Moya, Li Kevin Wenliang, Matthew Aitchison, Laurent Orseau, Marcus Hutter, and Joel Veness. 2024. Language modeling is compression. *ICLR*.

Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. 2023. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*.

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. 2023. Towards revealing the mystery behind chain of thought: A theoretical perspective. *NeurIPS*.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. *ICML*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did Aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *TACL*.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. 2024. Think before you speak: Training language models with pause tokens. *ICLR*.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *ICLR*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. *ACL*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *Findings of ACL*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. *ICLR*.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023. LLMLingua: Compressing prompts for accelerated inference of large language models. *EMNLP*.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. *ACL*.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. 2024. The impact of reasoning step length on large language models. *Findings of ACL*.

Yu Kang, Xianghui Sun, Liangyu Chen, and Wei Zou. 2025. C3oT: Generating shorter chain-of-thought without compromising effectiveness. *AAAI*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *NeurIPS*.

Llama Team, AI @ Meta. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. *ACL*.

William Merrill and Ashish Sabharwal. 2024. The expressive power of transformers with chain of thought. *ICLR*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*.

Maxwell Nye, Anders Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. 2021. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? *NAACL*.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2023. Distilling reasoning capabilities into smaller language models. *Findings of ACL*.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2025. To CoT or not to CoT? chain-of-thought helps mainly on math and symbolic reasoning. *ICLR*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. *Findings of ACL*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. *NAACL*.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Naftali Tishby and Noga Zaslavsky. 2015. Deep learning and the information bottleneck principle. *IEEE Information Theory Workshop*.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. 2023a. Towards understanding chain-of-thought prompting: An empirical study of what matters. *ACL*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. *ICLR*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *NeurIPS*.

Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. 2025. TokenSkip: Controllable chain-of-thought compression in LLMs. *EMNLP*.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. 2025. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *NeurIPS*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. STaR: Bootstrapping reasoning with reasoning. *NeurIPS*.

A Appendix

A.1 Retained Information Decomposition

For a selection $S_K \subset [T]$ with $|S_K| = K$, the retained information satisfies:

$$I(Z_{S_K}; Y \mid X) = \sum_{t \in S_K} I(z_t; Y \mid X, z_{S_K \cap [t-1]}) + R(S_K), \tag{2}$$

where $R(S_K) \geq 0$ captures interaction effects between selected sentences. Under the autoregressive teacher assumption, $R(S_K)$ is small when S_K selects contiguous or near-contiguous sentences, and vanishes for the full CoT.

A.2 Complete Training Details

Per-strategy training. We train six independent student models (No-CoT, Full-CoT, First-K, Last-K, Uniform, Random), each initialized from the same base model (Qwen2.5-3B-Instruct) and fine-tuned separately. This yields 6 LoRA adapters total.

- **No-CoT**: Training data consists of question → answer pairs without any CoT. Prompt format: "Question: X\n\nThe answer is:"
- **Full-CoT**: Training data consists of question + full teacher CoT → answer.
- **First-K / Last-K / Uniform / Random**: Each training example is truncated at a *random* K drawn uniformly from $\{3, 5, 7, 10, 15, 20\}$ using the student's assigned strategy. This mixed-K training allows each student to generalize across the K sweep at test time, avoiding the need for separate per-K adapters.

Training data generation. For all 7,473 GSM8K training examples, the teacher (Qwen2.5-3B-Instruct) generates a CoT with greedy decoding. For each truncation strategy, we pre-generate truncated versions at all standard K values ($\{1, 2, 3, 5, 7, 10, 15, 20\}$). During training, one K is sampled uniformly per example per epoch, creating mixed-K training data.

Training hyperparameters.

- LoRA: rank r=16, α=32, dropout 0.05, targeting $\{q_{\text{proj}}, v_{\text{proj}}\}$
- Epochs: 3
- Batch size: 2 (per device), gradient accumulation: 8 (effective batch: 16)
- Learning rate: 2×10^{-4}
- Max sequence length: 768
- FP16 precision
- Best checkpoint selected by validation loss (10% stratified holdout)

Hardware. All experiments run on A800 GPUs (local) or RTX 3090/4090 (remote). Training one strategy on 7,473 examples takes ∼30–60 minutes. Full evaluation pipeline (6 strategies $imes$ 3 datasets) takes ∼12–24 hours.

Sentence segmentation. We split CoT by newlines as primary delimiter, falling back to sentence boundaries (. ! ?) when only one segment is found.

A.3 Statistical Testing Methods

Confidence intervals. We report Clopper-Pearson exact confidence intervals for binomial proportions:

$$\text{CI}_{95\%} = [\text{Beta}(\alpha/2; k, n-k+1), \text{Beta}(1-\alpha/2; k+1, n-k)], \tag{3}$$

where k is the number of correct predictions and n is the total number of examples.

McNemar's test. For comparing two strategies on the same examples, we use McNemar's test for paired binary outcomes. The test statistic is:

$$\chi^2 = \frac{(n_{01} - n_{10})^2}{n_{01} + n_{10}}, \tag{4}$$

where n_{01} is the count of examples where strategy A is correct and B is wrong, and n_{10} is the reverse. This tests whether the disagreement between strategies is symmetric (null: $n_{01} = n_{10}$).

Effect sizes. We report Cohen's h for comparing two proportions:

$$h = 2\left(\arcsin\sqrt{p_1} - \arcsin\sqrt{p_2}\right), \qquad (5)$$

with interpretation: $|h| > 0.8$ (large), > 0.5 (medium), > 0.2 (small).

A.4 Cross-Domain Prompt Templates

To avoid confounding from prompt mismatch across domains, we use task-specific prompt templates:

- **GSM8K**: "Let's solve step by step." / "Therefore, the answer is:"
- **ARC-Challenge**: "Let's think through this carefully." / "The answer is:"
- **MMLU STEM**: "Let's analyze this step by step." / "The answer is:"

These templates are applied consistently across all strategies and K values during evaluation.

A.5 LoRA Configuration Ablation

We ablate the LoRA configuration to verify that the three-regime structure is robust to the choice of adapter parameters. Using the Full-CoT student, we sweep rank ($r \in \{8, 16, 32\}$) and target modules ($\{q, v\}$ vs. $\{q, k, v, o\}$). All configurations exhibit the same broad regime pattern: a strong recency effect at low K, gradual convergence toward full CoT accuracy, and identical No-CoT baselines (15.0%). Increasing rank or adding target modules does not qualitatively change the picture, suggesting that the observed utility structure is not a trivial artifact of a single LoRA setting.

A.6 Full Cross-Task Results

See Tables 24 and 25 in the main paper for the complete accuracy matrices. Table 19 reports results on the SVAMP dataset (n=300).

A.7 Answer Leakage Control Details

For each (strategy, K) pair, we check whether the gold answer string (normalized) appears in any retained CoT sentence. If found, the matching substring is replaced with `[MASKED]`. The student is then re-evaluated on the masked inputs. A nonzero accuracy drop would indicate answer leakage. In our experiments, the masked accuracy equals the original accuracy at all (strategy, K) pairs, confirming that the recency effect is not an artifact of answer exposure.

A.8 Base Model Baseline Details

The base model is the same Qwen2.5-3B-Instruct used to initialize all LoRA students, evaluated *without any LoRA adapters*. Teacher CoTs are generated by the same model with greedy decoding. Four conditions are tested: no CoT (direct prediction), full CoT, last-3 sentences, and last-7 sentences. The same evaluation protocol (batch inference, answer extraction) is used as for the LoRA student evaluation.

A.9 LLMLingua Compression Details

We use the Full-CoT student as the evaluation model. LLMLingua computes token-level perplexity using a small auxiliary model, then retains sentences or tokens by their perplexity ranking. We test five compression methods at ratios $\{0.2, 0.4, 0.6, 0.8\}$:

- **Sent. PPL (highest)**: retain sentences with the highest average perplexity.
- **First-N / Last-N sentences**: positional baselines at matched ratios.
- **Token PPL (highest / lowest)**: retain tokens with the highest / lowest perplexity.

Sentence-level methods retain complete sentences to maintain coherence; token-level methods may break sentence boundaries.

A.10 Answer Proximity Control: Full Results

Table 4 reports the complete answer proximity control experiment across all K values and all seven conditions.

A.11 Fixed Student Comparison: Full Results

Tables 5 and 6 report the full fixed-student comparison across all strategies and K values.

A.12 Cross-Domain Confound Dissection: Full Results

Table 7 reports the three-model comparison (base, No-CoT student, Full-CoT student) across all K values on ARC and MMLU.

A.13 Auto-Select K^*: Full Results

Table 8 reports the full validation sweep grid, and Table 9 reports the cross-task transfer results.

A.14 Cross-Model Generalization: Full Results

Tables 10–13 report the complete accuracy matrices for all three additional model configurations.

Table 4: Full answer proximity control on GSM8K (n=1319), Last-K student. All 7 conditions $\times$ 8 K values.

Condition	K=1	2	3	5	7	10	15	20
Original	.695	.813	.829	.836	.839	.844	.845	.847
Remove final sent.	.695	.813	.829	.836	.839	.844	.845	.847
Mask numbers	.136	.121	.132	.128	.125	.110	.118	.112
Shuffle sentences	.695	.813	.817	.807	.825	.807	.816	.822
Reverse order	.695	.813	.829	.819	.821	.826	.824	.823
Final equation only	.695	.699	.661	.670	.746	.746	.747	.747
Last-K excl. answer	.695	.813	.829	.836	.839	.844	.845	.847

Table 5: Full-CoT student evaluated with different test-time truncation strategies on GSM8K (n=1319). The student was trained on full CoT and tested with truncated inputs. Last-n truncation transfers well; first-n does not.

Test-time strategy	K=0	1	3	5	7	10	15	20	Full
first_n	.083	.177	.172	.150	.180	.282	.466	.647	.847
last_n	.083	.691	.811	.832	.836	.840	.846	.848	.847
uniform	.083	.177	.667	.701	.734	.757	.803	.827	.847
random	.083	.253	.338	.474	.573	.724	.755	.799	.847

A.15 Qualitative Examples

We present illustrative examples demonstrating the regime structure.

Example: Last-K succeeds / First-K fails (K=3). Question: "Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day?" (Gold: 18)

The full CoT contains 28 sentences. The first 3 sentences contain only the problem restatement and "please solve this step by step," providing no useful computation. The last 3 sentences contain the final computation ($9 $\times$ 2 = $18) and the boxed answer. First-K predicts 8.0 (wrong); Last-K predicts 18.0 (correct). This exemplifies the **recency regime**: later sentences carry the critical computation while earlier ones restate the problem setup.

Example: Last-K also fails (K=3). Question: "John drives for 3 hours at 60 mph and then turns around…" (Gold: 45)

Here, the CoT itself contains a reasoning error (negative time). The last 3 sentences only contain "Revising the problem: / Correct total time: 4 hours / Time already spent:"—an incomplete correction. Last-K predicts 150.0 (wrong). This shows that truncation cannot compensate for flawed teacher reasoning.

Example: Oversaturation on ARC. Question: "Petrified palm trees are found in sedimentary rock near glaciers. The presence of the petrified palm trees most likely provides evidence for which statement?" (Gold: C)

The teacher generates a 21-sentence CoT with detailed step-by-step scientific reasoning. The Full-CoT student (trained on GSM8K arithmetic patterns) fails on this science question, while the same student with truncated CoT succeeds by avoiding the misleading source-domain reasoning patterns. This exemplifies the **oversaturation regime**.

Example: Shuffle collapse. In the answer proximity control experiment (Section A.16), shuffling the order of retained sentences causes accuracy to drop by 1–3 pp at moderate K values. This demonstrates that *sentence order*—not just content—contributes to the recency effect, consistent with multi-step reasoning requiring sequential computation.

Table 6: Base model (no LoRA) evaluated with different truncation strategies on GSM8K (n=1319). Without LoRA calibration, truncated CoT provides limited benefit except through last-n at small K.

Truncation strategy	K=0	1	3	5	7	10	15	20	Full
first_n	.017	.155	.146	.070	.092	.149	.343	.563	.816
last_n	.017	.309	.330	.371	.372	.373	.412	.531	.816
uniform	.017	.155	.498	.446	.477	.481	.528	.627	.816
random	.017	.173	.215	.243	.249	.363	.507	.656	.816

Table 7: Cross-domain confound dissection. Three models evaluated with Last-K truncation on ARC (n=1172) and MMLU STEM (n=2511). The No-CoT student (LoRA without CoT training) consistently outperforms both the base model and the Full-CoT student, confirming that CoT fine-tuning—not LoRA itself—causes cross-domain degradation.

	K=0	1	2	3	5	7	10	15	20	Full
				ARC						
Base model	.326	.580	.590	.641	.626	.617	.593	.604	.586	.580
No-CoT student	.314	.578	.596	.659	.667	.657	.625	.686	.719	.747
Full-CoT student	.306	.514	.499	.544	.543	.537	.515	.536	.526	.523
				MMLU						
Base model	.276	.391	.397	.408	.425	.427	.423	.425	.432	.468
No-CoT student	.266	.389	.399	.431	.451	.463	.462	.497	.517	.584
Full-CoT student	.258	.341	.339	.346	.358	.359	.347	.348	.355	.369

A.16 Additional Controls, Robustness, and Baselines

This appendix collects the in-domain controls, robustness checks, and baseline comparisons summarized in Section 7.

Statistical robustness (multi-seed). We assess variance using three independent seeds ($\{42, 123, 456\}$); each (strategy, seed) pair is an independently trained adapter.

Statistical significance (McNemar). All recency comparisons are significant at $p<0.001$ with large effect sizes; no truncated version significantly exceeds full CoT in-domain (χ^2=0.07, p=0.80).

Answer leakage control. We mask all occurrences of the gold answer string in the truncated CoT with `[MASKED]` and re-evaluate all strategies at all K on GSM8K. Masked accuracy is identical to the original at every (strategy, K) pair (max Δ=0.0%), ruling out answer leakage.

Answer proximity control. Masking numbers collapses accuracy (69.5%→13.6% at K=1), confirming dependence on numerical computation; removing the answer-containing sentence ("Last-K excl. answer") matches the original at all K, proving the recency advantage is not answer proximity. The full 7×8 grid is in Table 4.

Fixed-student comparison. The Full-CoT student with test-time last-n truncation (81.1% at K=3) trails the per-strategy Last-K student (82.9%) by 1.8 pp, confirming that per-strategy training calibrates to the truncated input distribution. Per-K matrices for the fixed Full-CoT and base-model students are in Tables 5–6.

Auto-selecting K^*. The validation oracle selects last_n, K=20, matching full CoT (84.7%) on the held-out half. Transferring GSM8K's K^* to ARC/MMLU *underperforms* the locally optimal truncation (ARC −4.2 pp, MMLU −1.4 pp; Tables 8–9), confirming the oversaturation regime.

SVAMP near-domain validation.

Summary-based compression baseline.

Table 8: Auto-select validation sweep on GSM8K (first 50%, n=659). Per-strategy trained students. Best result in **bold**. Last-n dominates at all K values.

Strategy	K=1	3	5	7	10	15	20
first_n	.176	.173	.144	.162	.311	.481	.660
last_n	**.687**	**.827**	**.833**	**.836**	**.842**	**.847**	**.848**
uniform	.165	.690	.733	.760	.774	.810	.819
random	.247	.351	.492	.596	.745	.765	.813

Table 9: Cross-task transfer of GSM8K-selected K^* (last_n, K=20). Transfer gap is *negative*, confirming that the in-domain optimal budget is too large under domain shift.

Method	ARC (n=1172)	MMLU (n=2511)
Auto-selected (GSM8K K^*)	.528	.350
Best local last_n (K=3/5)	.570	.364
Transfer gap	−.042	−.014

Base-model baseline. On ARC, the base model with only last-3 CoT sentences (64.1%) exceeds every LoRA-trained student, confirming that domain-specific fine-tuning can hurt under domain shift.

Perplexity-based compression (LLMLingua). At ratio 0.2, Last-N sentences achieve 82.6% vs 17.5% for high-perplexity sentences—a 65.1 pp gap. Sentence-level PPL selection performs identically to First-N truncation, confirming position (not estimated informativeness) drives CoT utility.

A.17 Full Result Matrices (MMLU and Cross-Model)

A.18 Empirical Discovery: Token-Level Truncation

To disentangle position from sentence count, we truncate by token budget rather than sentence count, evaluating the independently trained First-K and Last-K students.

The recency gap peaks at 100 tokens (+66.6 pp) and decreases monotonically. Since token budgets are equal, this rules out a sentence-length explanation: the last 100 tokens already achieve 84.2%, nearly matching full CoT.

A.19 Interpretive Framework: Alternative Explanations and Predictions

Alternative explanations. (1) *Answer proximity*: the answer-proximity control (Table 16) shows removing the answer-containing sentence has zero effect while masking numbers is devastating, confirming genuine reasoning utility. (2) *LoRA overspecialization*: the confound dissection (Section 7.1) shows the No-CoT student exceeds the base model cross-domain, ruling out general LoRA degradation. (3) *Distribution mismatch*: per-strategy training uses mixed K; residual mismatch at extreme K may inflate the recency gap.

Checkable predictions. (i) recency at low K in-domain; (ii) strategy convergence as $K \to T$; (iii) cross-domain oversaturation ($K_{\text{opt}} < T$); (iv) CoT overspecialization under domain-specific fine-tuning. Patterns (i)–(ii) are confirmed on GSM8K; (iii)–(iv) on ARC/MMLU.

A.20 Mechanism: Window Sweep and Distribution-Mismatch Oversaturation

Oracle window sweep. The optimal contiguous window of any width W is positioned near the end of the CoT, ruling out the possibility that the recency effect is an artifact of non-contiguous selection.

Per-sentence selection does not help. The LOO utility profile (Section 6.1) is sparse—with a high-accuracy student, removing almost any single sentence leaves the answer unchanged. An oracle that keeps the top-K sentences by LOO utility therefore has no signal to exploit and collapses toward First-K (14–28% on GSM8K, far below Last-K). Useful information is distributed and redundant across the late CoT rather than concentrated in a

Table 10: Qwen2.5-1.5B on GSM8K (n=1319), per-strategy students. Full CoT: 65.6%; No-CoT: 8.2%.

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.11	.11	.11	.15	.17	.31	.47	.62	.65
Uniform	.10	.20	.28	.37	.43	.53	.60	.63	.65
Random	.13	.18	.23	.33	.40	.52	.58	.62	.65
Last-K	**.18**	**.25**	**.33**	**.42**	**.48**	**.57**	**.61**	**.65**	.66
Range	.08	.14	.21	.27	.31	.26	.21	.06	.02

Table 11: Llama-3.2-3B on GSM8K (n=1319), per-strategy students. Full CoT: 83.1%; No-CoT: 11.0%.

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.12	.10	.09	.20	.41	.70	.80	.83	.83
Uniform	.11	.55	.75	.77	.80	.80	.83	.83	.83
Random	.46	.63	.71	.78	.80	.82	.83	.83	.83
Last-K	**.80**	**.81**	**.81**	**.82**	**.83**	**.83**	**.83**	**.83**	.83
Range	.69	.71	.73	.62	.42	.12	.03	.00	.00

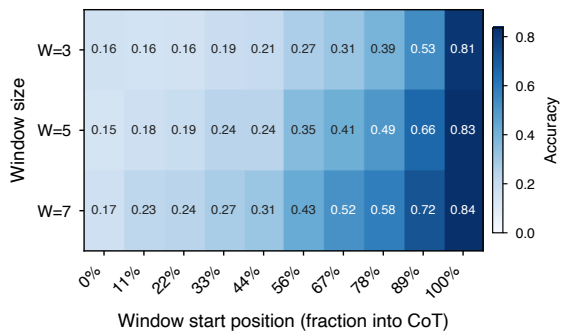

Figure 4: Oracle window sweep on GSM8K: accuracy when retaining a contiguous window of W sentences starting at position p. The optimal window is concentrated in the final positions.

few critical sentences, explaining why Last-K is hard to beat with per-sentence scoring.

Oversaturation depends on CoT source, not task domain. Replacing self-generated CoT with the GSM8K dataset's human-annotated traces, for the same Qwen2.5-3B student on the same test set, produces in-domain oversaturation.

Random selection (retaining early sentences) suffers most (+13.0 pp), Last-K (discarding early content) least (+7.2 pp). This defines a *distribution alignment spectrum*: aligned (self-generated, in-domain) $\rightarrow$ partially misaligned (human CoT, in-domain) $\rightarrow$ strongly misaligned (cross-domain). Cross-domain ARC confirms the spectrum:

With self-generated CoT the No-CoT student reaches 74.7% on ARC (strong overspecialization); with human CoT both students perform similarly (50.6% vs 50.3%), because human-CoT distribution mismatch already limits cross-domain utilization. Oversaturation risk thus increases whenever teacher and student differ in architecture, scale, or reasoning style—the setting most teacher–student pipelines face.

Table 12: Qwen7B teacher → 3B student on GSM8K (n=1319), per-strategy students. Full CoT: 90.4%; No-CoT: 15.2%. Larger teacher improves absolute accuracy but preserves the regime structure.

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.19	.19	.18	.19	.25	.34	.54	.72	.91
Uniform	.17	.57	.79	.82	.85	.86	.89	.90	.91
Random	.27	.33	.38	.53	.67	.80	.87	.89	.90
Last-K	**.77**	**.84**	**.87**	**.89**	**.90**	**.90**	**.90**	**.90**	.90
Range	.60	.65	.69	.70	.65	.56	.35	.18	.02

Table 13: Qwen7B teacher → 3B student on ARC (n=1172) and MMLU STEM (n=2511). Cross-domain oversaturation: on ARC, Last-K=3 (75.7%) exceeds Full CoT (73.0%); on MMLU, Last-K=5 (53.4%) approaches Full CoT (54.6%).

	ARC								MMLU STEM							
Strategy	K=1	3	5	7	10	15	20	Full	K=1	3	5	7	10	15	20	Full
First-K	.29	.30	.31	.31	.35	.47	.61	.73	.26	.25	.27	.26	.27	.33	.40	.56
Uniform	.29	.70	.67	.66	.66	.68	.70	.70	.25	.47	.49	.48	.48	.50	.50	.54
Random	.36	.36	.38	.43	.51	.60	.68	.70	.28	.29	.31	.33	.35	.39	.47	.53
Last-K	**.72**	**.76**	**.75**	**.72**	**.71**	**.72**	**.73**	.73	**.49**	**.51**	**.53**	**.52**	**.52**	**.53**	**.54**	.55

Table 14: Mean accuracy ± std across 3 seeds on GSM8K (n=1319). Variance peaks at low K and drops below 2.0 pp at $K \geq 15$.

Strategy	K=3	5	7	10	15
First-K	.177±.005	.159±.006	.192±.010	.307±.008	.482±.002
Last-K	.823±.008	.832±.002	.835±.004	.839±.002	.840±.002
Uniform	.710±.010	.735±.010	.763±.011	.778±.005	.809±.004
Random	.364±.023	.505±.048	.600±.046	.717±.036	.788±.019
Full CoT	.847±.002	.847±.002	.847±.002	.847±.002	.847±.002

Table 15: McNemar's test on GSM8K. $^{***}p<0.001$, ns = not significant.

Comparison	K	ΔAcc	χ^2	Cohen's h
Last-K vs First-K	3	+65.4%	826.6***	1.43
Last-K vs First-K	5	+69.2%	887.7***	1.53
Last-K vs First-K	7	+66.4%	847.0***	1.45
Last-K vs First-K	10	+54.5%	674.0***	1.17
Last-K vs Full CoT	20	+0.1%	0.07 (ns)	–

Table 16: Answer proximity control on GSM8K (n=1319), Last-K student. Masking numbers is devastating; removing the answer sentence has no effect.

Condition	K=1	3	7	20
Original	.695	.829	.839	.847
Remove final sent.	.695	.829	.839	.847
Mask numbers	.136	.132	.125	.112
Shuffle sentences	.695	.817	.825	.822
Reverse order	.695	.829	.821	.823
Final equation only	.695	.661	.746	.747
Last-K excl. answer	.695	.829	.839	.847

Table 17: Fixed-student comparison on GSM8K. Per-strategy training improves over fixed-student baselines at truncated budgets.

Configuration	K=3	7	Full
Per-strategy Last-K	.829	.839	–
Fixed (Full-CoT student + last_n)	.811	.836	.847
Fixed (base model + last_n)	.330	.372	.816

Table 18: Auto-select K^*. Validation sweep on GSM8K (50%) selects last_n, K=20; test matches full CoT. Cross-task transfer of GSM8K's K^* is *negative*.

Method	GSM8K Test	Config
Auto-select (val-tuned)	.847	last_n, K=20
Fixed Last-K=7	.842	last_n, K=7
Fixed Last-K=10	.845	last_n, K=10
Full CoT	.847	full_cot

Table 19: SVAMP accuracy (n=300), GSM8K-trained per-strategy students. Recency replicates; convergence faster than GSM8K; no oversaturation.

Strategy	K=1	3	5	7	10	15	20	Full
First-K	.55	.58	.65	.73	.80	.85	.87	.87
Uniform	.54	.76	.78	**.83**	**.85**	.86	.87	**.87**
Random	.61	.71	.77	.80	.83	.85	.86	.86
Last-K	**.65**	**.77**	**.79**	.80	.82	**.86**	.87	.87

Table 20: Summary baseline vs. truncation on GSM8K. Summary/Truncation use the Full-CoT student; Last-K uses the Last-K student.

Method	K=3	5	7	10	15
Summary (Full-CoT student)	.72	.76	.79	.79	.78
Truncation (Full-CoT student)	.17	.15	.18	.28	.47
Last-K (Last-K student)	.83	.84	.84	.84	.85

Table 21: Base model (no LoRA) vs. LoRA students. On cross-domain tasks the base model with CoT matches or exceeds fine-tuned students.

Condition	GSM8K	ARC	MMLU
Base: No CoT	.017	.326	.276
Base: Full CoT	.816	.580	.468
Base: Last-K=3	.330	.641	.408
Base: Last-K=7	.372	.617	.427
No-CoT student (K=0)	.150	.314	.266
Full-CoT student (Full)	.847	.523	.369
Best truncated student	.839	.581	.434

Table 22: LLMLingua vs. positional truncation on GSM8K (n=1319, Full-CoT student). Position dominates perplexity. Full CoT: 84.7%.

Method	Ratio 0.2	0.4	0.8
Sent. PPL (highest)	.175	.177	.643
First-N sentences	.175	.173	.666
Last-N sentences	**.826**	**.840**	**.847**
Token PPL (highest)	.202	.416	.820
Token PPL (lowest)	.829	.839	.849

Table 23: GSM8K accuracy (n=1319) with per-strategy trained students. No-CoT: 15.0%; Full CoT: 84.7%. Bold marks the best strategy at each K.

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.17	.18	.18	.14	.18	.30	.48	.67	**.85**
Uniform	.18	.70	.69	.72	.75	.77	.81	.83	.85
Random	.24	.31	.35	.49	.59	.74	.77	.81	.85
Last-K	**.70**	**.81**	**.83**	**.84**	**.84**	**.84**	**.85**	**.85**	.84
Range	.53	.63	.65	**.69**	.66	.55	.37	.18	.01

Table 24: ARC-Challenge accuracy (n=1172), per-strategy students. Full-CoT student: 52.3%; No-CoT student: 31.4% (K=0), 74.7% (K=Full). Uniform K=3 significantly exceeds full CoT (p<0.001).

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.28	.27	.27	.30	.27	.28	.33	.44	.53
Uniform	.31	.56	**.58**	.57	.56	.55	.55	.57	.56
Random	.35	.35	.38	.39	.37	.41	.46	.52	.53
Last-K	**.53**	**.54**	.57	**.56**	**.54**	.52	.54	.53	.52
Range	.25	.29	.31	.27	.29	.28	.22	.13	.04

Table 25: MMLU STEM accuracy (n=2511), per-strategy students. Full-CoT student: 36.9%; No-CoT student: 26.6% (K=0), 58.4% (K=Full).

Strategy	K=1	2	3	5	7	10	15	20	Full
First-K	.24	.24	.25	.25	.24	.25	.27	.32	.37
Uniform	.26	.37	.37	**.38**	**.37**	.37	**.38**	**.40**	**.43**
Random	.27	.28	.29	.28	.28	.30	.33	.36	.38
Last-K	**.35**	**.36**	**.36**	.36	.36	**.35**	.35	.35	.35

Table 26: Cross-model full accuracy: best truncated vs. Full CoT on GSM8K and ARC.

	GSM8K		ARC	
Model	Best tr.	Full	Best tr.	Full
Qwen2.5-3B	.839	.847	.581	.523
Qwen2.5-1.5B	.418	.656	.561	.562
Llama-3.2-3B	.821	.831	.520	.377
Qwen7B→3B	.885	.904	.757	.730

Table 27: Recency gap (Last-K − First-K, in points) on GSM8K across student configurations. The effect is large and consistent at low budgets everywhere.

Model	K=3	K=5	K=10
Qwen2.5-3B (primary)	+65.0	+69.2	+54.5
Qwen2.5-1.5B	+21.4	+26.5	+26.1
Llama-3.2-3B	+72.5	+62.2	+12.7
Qwen7B→3B	+69.2	+69.5	+56.1

Table 28: Token-level truncation on GSM8K (n=1319). The recency effect is robust at sub-sentence granularity: last-N exceeds first-N by up to $+66.6$ pp.

Tokens	50	100	150	200	300	400
First-N	.18	.18	.25	.38	.66	.81
Last-N	**.83**	**.84**	**.85**	**.85**	**.85**	**.85**
Gap	$+65.0$	$+66.6$	$+59.8$	$+46.4$	$+18.0$	$+3.8$

Table 29: In-domain oversaturation depends on CoT source (same model, task, test set).

CoT source	Last-K=7	Full CoT	Δ	Oversat.?
Self-generated	.839	.847	-0.8	No
Human-annotated	.837	.765	$+\mathbf{7.2}$	Yes

Table 30: In-domain oversaturation by strategy, human-annotated CoT on GSM8K. Strategies retaining more early content show larger oversaturation.

Strategy	Peak (trunc.)	Full CoT	Oversat.
Last-K	.837 (K=7)	.765	$+7.2$
Uniform	.787 (K=15)	.707	$+8.0$
Random	.736 (K=15)	.606	$+13.0$

Table 31: ARC by CoT training source (Qwen2.5-3B student). The overspecialization gap narrows from $+22.4$ to $+0.3$ pp under human CoT.

	Self-gen. CoT	Human CoT
Best truncated (Last-K)	.581 (K=3)	.538 (K=5)
Full-CoT student	.523	.503
No-CoT student + full CoT	.747	.506
Oversaturation	$+5.8$	$+3.8$