CAST: CREDIT ALIGNMENT FOR CROSS-TOKENIZER ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

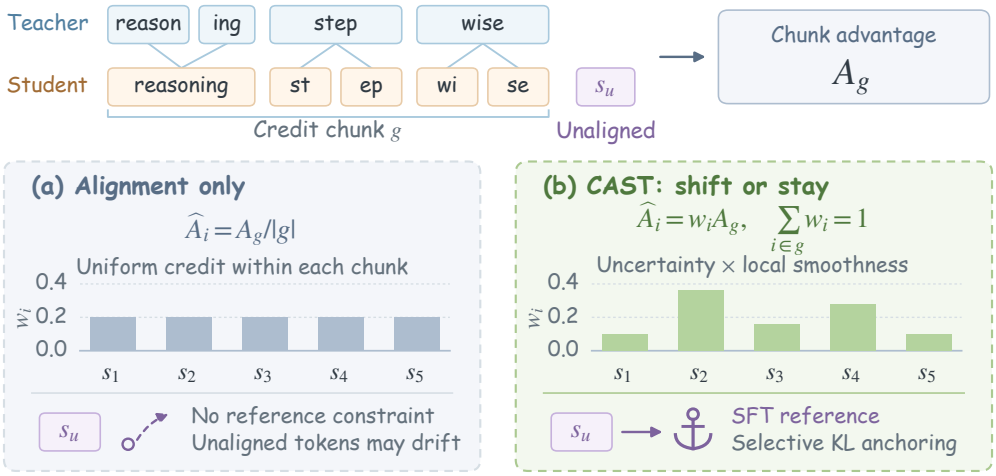

Figure 1: **Chunk-level alignment does not resolve credit alignment. (a)** Alignment-only propagation uniformly distributes each chunk's advantage across aligned student tokens, leaving unaligned tokens unconstrained. **(b)** CAST assigns credit using student uncertainty and local teacher-signal smoothness, while anchoring unaligned tokens to the supervised fine-tuning (SFT) reference with selective KL regularization.

ABSTRACT

Cross-tokenizer on-policy distillation (OPD) requires aligning teacher and student sequences before their scores can be compared. Decoded-text alignment provides such correspondence at the chunk level, but leaves a distinct problem unresolved: *credit alignment*. A chunk-level signal specifies how much a span should change, but not how that signal should be distributed across student tokens, while tokens without teacher correspondence remain unconstrained. We propose **CAST** (**C**hunk-aware credit **A**ssignment with **S**elective **T**oken Anchoring), a *shift-or-stay* framework that addresses both issues. For aligned chunks, CAST redistributes each chunk's advantage across student tokens using predictive uncertainty and local teacher-signal smoothness, emphasizing learning opportunities supported by locally consistent teacher feedback while preserving the chunk's total advantage. The resulting weights exactly solve a KL-regularized credit-allocation problem. For unaligned positions, CAST selectively anchors the student to its initial policy through a KL penalty, limiting drift where teacher supervision is unavailable. Because credit assignment is separated from chunk construction, CAST also applies to punctuation-defined or fixed-length chunks under shared tokenization. Across six mathematics, science, and code benchmarks, CAST improves the average over the strongest evaluated baseline by 2.45 and 1.32 points for cross-tokenizer transfer from Qwen3-8B to Mistral-7B and Llama-3.1-8B, respectively, and by 2.10 points for shared-tokenizer transfer to Qwen3-1.7B.

1 INTRODUCTION

On-policy distillation (OPD) trains a student on trajectories sampled from its own policy using feedback from a stronger teacher (Agarwal et al., 2024; Gu et al., 2024; Ko et al., 2024; Jang et al., 2026). Because supervision is applied to states the student actually visits, OPD can target student-

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.

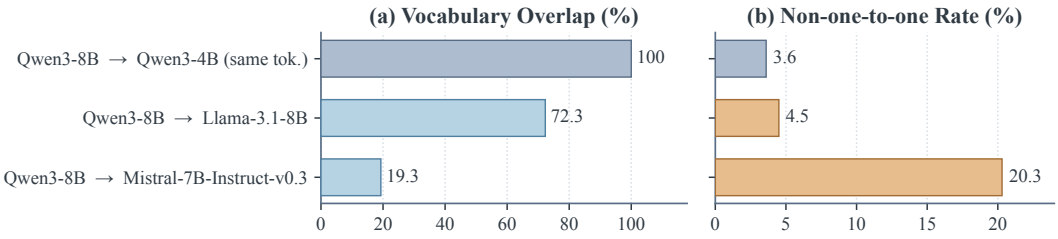

Figure 2: **Tokenizer mismatch persists despite vocabulary overlap. (a)** Teacher vocabulary with a decoded-content match in the student. **(b)** Student response tokens without one-to-one teacher correspondence under heterogeneous decoded-text alignment (Appendix A.1). The shared-tokenizer 3.6% residual results from decode/re-encode normalization and special/boundary-token handling; training instead uses identity correspondence.

specific errors and reduce the train–inference mismatch induced by learning only from fixed teacher-generated responses. It has therefore become an increasingly useful paradigm for language-model post-training, particularly for transferring reasoning ability (Yang et al., 2025; Bousselham et al., 2026).

A basic difficulty arises when teacher and student use different tokenizers. With a shared tokenizer, the token-wise log-density ratio $\log \pi_T(y_t \mid x, y_{<t}) - \log \pi_{\theta_{\text{old}}}(y_t \mid x, y_{<t})$ compares probabilities of the same next-token event. Across tokenizers, however, the same text can be segmented differently, so one student token may correspond to multiple teacher tokens, or vice versa. Teacher and student logits are also defined over different vocabularies, making direct token-wise comparison ill-defined. As shown in Figure 2, only 19.3% of the Qwen3-8B vocabulary has a decoded-content match in Mistral-7B-Instruct-v0.3, compared with 72.3% in Llama-3.1-8B. Moreover, vocabulary overlap alone does not imply token correspondence: identical IDs need not represent the same event, and identical text can still admit different segmentations.

Existing cross-tokenizer distillation methods address this mismatch by constructing a common space in which teacher and student scores can be compared (Boizard et al., 2025; Cui et al., 2025; Minixhofer et al., 2026; Singh et al., 2026). Dynamic Probabilistic Cross-tokenizer Alignment (DPCA), for example, extends decoded-text alignment to OPD by matching teacher and student sequences into text-equivalent chunks and aggregating their token probabilities within each chunk (Niu et al., 2026). Such methods establish *chunk-level alignment*: the teacher and student scores now refer to the same decoded text.

Our key observation is that *chunk alignment does not resolve credit alignment*. Alignment determines *what* teacher and student quantities are comparable, but OPD ultimately optimizes individual student-token probabilities. A chunk-level discrepancy therefore still has to be translated into token-level updates. Uniform propagation gives every student token in a chunk the same advantage, implicitly treating all positions as equally useful and equally reliable learning targets. This need not hold. Some positions are already predicted confidently by the student, whereas others are uncertain and offer greater opportunity for correction. At the same time, high student uncertainty alone is insufficient: it can coincide with abrupt local variation in the teacher-derived signal, where aggressively concentrating credit may be undesirable. A second issue appears outside aligned regions. Tokens without teacher correspondence receive no direct supervision, yet their predictions can still drift because they share parameters with supervised positions. Thus, decoded-text alignment leaves two coupled optimization questions: *where should aligned chunk credit be assigned, and how should the student behave where teacher evidence is absent?*

We address these questions with **CAST** (**C**hunk-aware credit **A**ssignment with **S**elective **T**oken Anchoring), a token-level *shift-or-stay* framework for OPD (Figure 1). At aligned positions, CAST selectively *shifts* the student toward the teacher by redistributing each chunk's advantage across its student tokens. Its credit rule combines two complementary signals: student predictive uncertainty identifies *where learning may be useful*, while local teacher-signal smoothness identifies *where that supervision is locally consistent*. Their product therefore favors uncertain student predictions supported by a stable local teacher signal. Normalization within each chunk preserves the chunk's total advantage, so CAST changes only *where* the credit is assigned, not *how much* aggregate credit the chunk receives. Moreover, the resulting weights exactly solve a KL-regularized allocation prob-

lem that trades off an uncertainty-based prior against local teacher-signal variation, with bounded relative reweighting from the smoothness term.

At unaligned positions, CAST instead encourages the student to *stay* close to its initial policy. Selective token anchoring applies a KL penalty toward the supervised fine-tuning (SFT) initialization only where aligned teacher supervision is unavailable, constraining incidental drift without directly competing with teacher-guided updates. Together, the two components separate the roles of teacher supervision and reference preservation: *shift where teacher evidence is available, stay where it is not*.

This perspective also makes CAST applicable beyond tokenizer mismatch. Credit assignment is conceptually separate from how chunks are constructed. Decoded-text alignment provides chunks under heterogeneous tokenization, but the same allocation rule can operate on punctuation-defined or fixed-length chunks when teacher and student share a tokenizer. CAST therefore addresses a more general problem: converting span-level learning signals into structured token-level updates. We additionally find that multi-rollout sampling can further improve performance by exposing the credit mechanism to multiple student trajectories per prompt.

We evaluate CAST on six benchmarks spanning mathematics, science, and code: AIME 2024, AIME 2025, AIME 2026 (Dekoninck et al., 2026), MATH-500 (Hendrycks et al., 2021), GPQA-Diamond (Rein et al., 2024), and LiveCodeBench (Jain et al., 2025). For cross-tokenizer distillation from Qwen3-8B to Mistral-7B and Llama-3.1-8B, CAST improves the six-benchmark average over the strongest evaluated baseline by 2.45 and 1.32 points, respectively. Under shared tokenization, CAST improves the corresponding average for Qwen3-1.7B by 2.10 points. On Mistral, its advantage over uniform and uncertainty-only credit controls persists across evaluated seeds, together with fewer observed training collapses (Table 5; Appendix E). On Llama, CAST also outperforms ULD across all three evaluated seeds (Table 6; Appendix E).

Our contributions are:

- We distinguish *chunk-level alignment* from *credit alignment* in cross-tokenizer OPD. Chunk alignment establishes comparable teacher and student signals, but leaves their token-level allocation and the behavior of unaligned positions unresolved.
- We propose CAST, a *shift-or-stay* framework that allocates chunk credit according to student uncertainty and local teacher-signal smoothness while selectively anchoring unaligned positions. The allocation preserves each chunk's total advantage and admits an exact KL-regularized optimization interpretation.
- We show that credit alignment is useful beyond heterogeneous tokenization by applying the same rule to punctuation-defined and fixed-length chunks under shared tokenization. Across six benchmarks, CAST outperforms the strongest evaluated baseline for both heterogeneous students and shared-tokenizer Qwen3-1.7B, with additional studies of robustness, training dynamics, and component interactions.

2 Related Work

On-policy distillation. Generalized knowledge distillation (GKD) supervises student-generated sequences with teacher token distributions and serves as our shared-tokenizer baseline (Agarwal et al., 2024). MiniLLM adopts a reverse-KL objective, while DistiLLM and DistiLLM-2 develop streamlined and contrastive objectives (Gu et al., 2024; Ko et al., 2024; 2025). KDRL combines distillation with reinforcement learning for reasoning post-training (Xu et al., 2025). Training stability has also been studied through group-relative baselines from multiple responses per prompt and adaptive target reformulation (Guo et al., 2025; Jang et al., 2026). These methods assume a shared tokenizer. In this setting, CAST retains its token-level credit rule and uses punctuation-defined or fixed-length chunks.

Cross-tokenizer distillation and credit assignment. ULD compares sorted logit distributions through optimal transport, and MultiLevelOT extends transport across multiple granularities (Boizard et al., 2025; Cui et al., 2025). ALM matches likelihoods over shared text chunks, while byte-level distillation uses a tokenizer-independent byte interface (Minixhofer et al., 2026; Singh et al., 2026). Minimum edit-distance (MinED) token mapping supports knowledge fusion (Wan

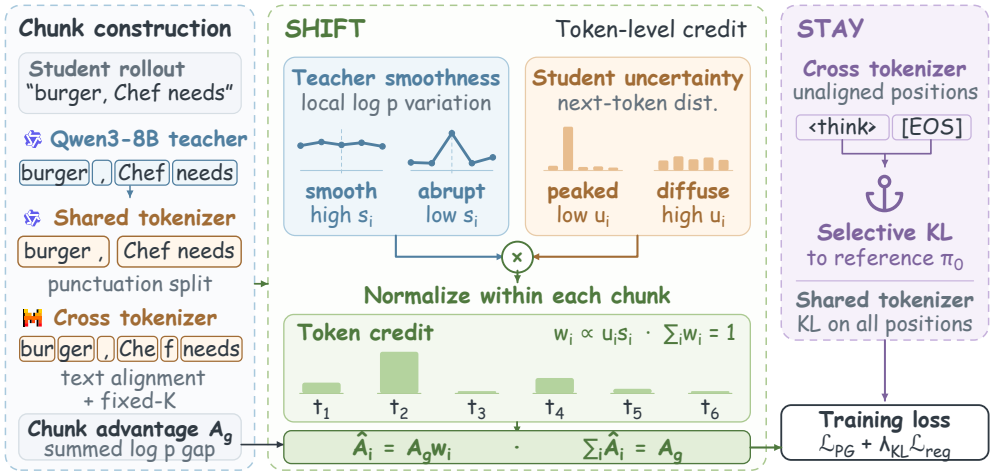

Figure 3: **CAST: token-level shift-or-stay optimization. Shift** normalizes uncertainty u_i times smoothness s_i to allocate token credit while preserving each chunk's advantage. **Stay** anchors unaligned predictions to the initial policy. Heterogeneous pairs use decoded-text alignment and fixed-length credit chunks; shared-tokenizer training uses punctuation-defined chunks here and regularizes all positions.

et al., 2024). These methods primarily determine *what* can be compared across tokenizers. DPCA extends decoded-text alignment to on-policy distillation by matching teacher and student sequences into text-equivalent chunks and propagating aggregate teacher scores (Niu et al., 2026). Shared-tokenizer methods use local divergence reweighting or adaptive policy distillation (Jung et al., 2025; Zhang et al., 2025). E-SCOT constrains offline optimal transport by character overlap and reweights token-level costs using teacher–student Rényi-entropy gaps (Liu et al., 2026). CAST addresses token-level *shift-or-stay* optimization after decoded-text alignment. It redistributes each chunk's on-policy advantage to individual tokens, using uncertainty to highlight potential learning opportunities and smoothness to temper credit where teacher scores vary abruptly, while preserving the total advantage. Selective KL anchoring keeps unaligned predictions close to the initial student policy, constraining drift from shared parameter updates.

3 Method

3.1 Preliminaries

We distill a teacher p_T into a student policy π_θ with vocabulary $\mathcal{V}_S$; the two models may use different tokenizers. For prompt x, the rollout policy samples $y = (a_1, \ldots, a_N) \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$ and records $\ell_i^S = \log \pi_{\theta_{\text{old}}}(a_i \mid x, a_{<i})$. The teacher retokenizes the decoded response as $(b_1, \ldots, b_M)$ and returns $\ell_j^T = \log p_T(b_j \mid x^T, b_{<j})$ under its own input format x^T.

Following DPCA (Niu et al., 2026), we match decoded text across tokenizers to form *alignment chunks* and obtain uniformly projected teacher scores $\widetilde{\ell}_i^T$ on aligned student positions $\mathcal{A}$, leaving $\mathcal{U}$ unaligned. We regroup aligned tokens into K-token *credit chunks* $g \in \mathcal{G}$ ($K = 6$ by default; Figure 3, left).

With shared tokenization, $M = N$, $\widetilde{\ell}_i^T = \ell_i^T$, and $\mathcal{U} = \varnothing$; $\mathcal{G}$ uses punctuation-defined or fixed-length chunks. Construction details are in Appendix A.1.

3.2 Uncertainty–Smoothness Credit Assignment

The *shift* component (Figure 3, center) redistributes each credit chunk's advantage to individual student tokens. For $g \in \mathcal{G}$, we define the chunk's teacher–student log-probability gap and token advantages as

$$A_g = \sum_{i \in g} (\widetilde{\ell}_i^T - \ell_i^S), \qquad \widehat{A}_i = A_g w_i, \quad w_i \geq 0, \quad \sum_{i \in g} w_i = 1. \tag{1}$$

Student uncertainty uses rollout-policy entropy to highlight potential learning opportunities:

$$H_i = -\sum_{v \in \mathcal{V}_S} \pi_{\theta_{\text{old}}}(v \mid x, a_{<i}) \log \pi_{\theta_{\text{old}}}(v \mid x, a_{<i}), \qquad u_i = \max(H_i, 10^{-8}). \tag{2}$$

Teacher-signal smoothness tempers credit where teacher scores vary abruptly. For ordered $g = (i_1, \ldots, i_{K_g})$ with $K_g = |g|$ and $q_k = \widetilde{\ell}^T_{i_k}$, define

$$\delta_k = \begin{cases} 0, & K_g = 1, \\ |q_2 - q_1|, & k = 1,\ K_g > 1, \\ |q_{K_g} - q_{K_g - 1}|, & k = K_g,\ K_g > 1, \\ (|q_k - q_{k-1}| + |q_{k+1} - q_k|)/2, & 1 < k < K_g. \end{cases} \tag{3}$$

For heterogeneous tokenizers, projected scores are constant within each alignment chunk (Eq. (12), Appendix A.1). Thus δ_k reflects transitions between alignment chunks within credit chunk g. With a shared tokenizer, Eq. (3) measures adjacent teacher-token score variation. We combine uncertainty with inverse local variation:

$$\epsilon_g = \max\left(\frac{1}{K_g}\sum_{k=1}^{K_g} \delta_k, 0.1\right), \qquad s_{i_k} = \frac{1}{\delta_k + \epsilon_g}, \tag{4}$$

$$w^{\text{US}}_{i_k} = \frac{u_{i_k} s_{i_k}}{\sum_{r=1}^{K_g} u_{i_r} s_{i_r}}, \qquad \widehat{A}_{i_k} = A_g w^{\text{US}}_{i_k}. \tag{5}$$

The adaptive floor limits inverse weights at small variations. Appendix B.3 illustrates the resulting token allocation on a recorded rollout (Figure 6). Smoothness is a structural prior, not an estimate of teacher correctness. All weighting quantities are stop-gradient. Constant u_i or δ_i recovers smoothness-only or uncertainty-only allocation, respectively; if both are constant, $w_i = 1/|g|$.

Variational characterization and bounded distortion. For credit chunk g, define the uncertainty prior $p^u_i = u_i / \sum_{j \in g} u_j$, the dimensionless local irregularity $\rho_i = 1 + \delta_i/\epsilon_g$, and the probability simplex $\Delta_g = \{w : w_i \geq 0,\ \sum_{i \in g} w_i = 1\}$.

Proposition 1 (Credit allocation under a local-variation penalty)**.** *CAST uniquely minimizes*

$$w^{\text{US}} = \arg\min_{w \in \Delta_g} \left\{ \text{KL}(w \,\|\, p^u) + \sum_{i \in g} w_i \log \rho_i \right\}. \tag{6}$$

Moreover, the adaptive scale in Eq. (4) bounds the relative distortion due to smoothness alone:

$$1 \leq \rho_i \leq K_g + 1, \qquad \frac{\max_{i \in g} s_i}{\min_{i \in g} s_i} \leq K_g + 1. \tag{7}$$

Proof. Let $Z_g = \sum_{i \in g} p^u_i / \rho_i$ and $w^*_i = p^u_i/(\rho_i Z_g) = w^{\text{US}}_i$. The objective satisfies $F_g(w) = \text{KL}(w \,\|\, w^*) - \log Z_g$, uniquely minimized at w^* by nonnegativity of KL. Since $0 \leq \delta_i \leq \sum_j \delta_j \leq K_g \epsilon_g$, we have $1 \leq \rho_i \leq K_g + 1$. Using $s_i = (\epsilon_g \rho_i)^{-1}$ gives the smoothness-ratio bound. □

Equation (6) balances the uncertainty prior against local teacher-score variation. Optimality concerns this allocation objective; gradient and downstream effects are assessed in Figure 4 and Table 4. Smoothness reweighting is bounded, but uncertainty can still concentrate credit.

3.3 Selective Token Anchoring and Optimization

The *stay* component (Figure 3, right) constrains drift from shared parameter updates by keeping unaligned predictions close to the initial student policy π_0 (the SFT initialization for cross-tokenizer pairs). We apply a selective KL penalty:

$$z_i = \log \pi_0(a_i \mid x, a_{<i}) - \log \pi_\theta(a_i \mid x, a_{<i}), \qquad d_i = \exp(z_i) - z_i - 1, \tag{8}$$

$$\mathcal{L}_{\text{anchor}} = \frac{\sum_{i \in \mathcal{U}} \min(d_i, d_{\max})}{\max(|\mathcal{U}|, 1)}. \tag{9}$$

Here $d_i \geq 0$ vanishes when sampled-token probabilities match, and $d_{\max}$ clips outliers. Thus aligned positions remain free to absorb teacher credit, while $\mathcal{L}_{\text{anchor}} = 0$ if $\mathcal{U} = \varnothing$.

We use multi-rollout sampling with n independent student responses per prompt, scoring and chunking each separately before averaging over their tokens. With $r_i(\theta) = \pi_\theta(a_i \mid x, a_{<i}) / \pi_{\theta_{\text{old}}}(a_i \mid x, a_{<i})$, the aligned-token objective is

$$\mathcal{L}_{\text{PG}} = -\frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \min\Big(r_i \widehat{A}_i, \text{clip}(r_i, 1 - \varepsilon_{\text{low}}, 1 + \varepsilon_{\text{high}}) \widehat{A}_i \Big) . \tag{10}$$

Unaligned positions have zero distillation advantage. The full objective is

$$\mathcal{L} = \mathcal{L}_{\text{PG}} + \lambda_{\text{KL}} \mathcal{L}_{\text{reg}}. \tag{11}$$

For heterogeneous tokenizers, $\mathcal{L}_{\text{reg}} = \mathcal{L}_{\text{anchor}}$. Shared-tokenizer training retains the same credit rule and applies reference KL regularization toward π_0 over all response positions.

4 Experiments

4.1 Experimental Setup

Baselines. Cross-tokenizer experiments distill Qwen3-8B into Mistral-7B and Llama-3.1-8B (Grattafiori et al., 2024), both initialized by SFT on 400,000 OpenThoughts examples (Guha et al., 2026). Figure 2 characterizes tokenizer mismatch for these pairs. Heterogeneous baselines include the student initialization, teacher-reward GRPO, DPCA, minimum edit-distance alignment (MinED), Universal Logit Distillation (ULD), Approximate Likelihood Matching (ALM), and a matched uniform-credit control. DPCA is the decoded-text-alignment OPD baseline; the uniform control splits each chunk advantage equally while retaining CAST's clipping, selective anchoring, and two-rollout sampling to isolate its *shift* credit rule. Shared-tokenizer experiments pair Qwen3-8B or Qwen3-30B-A3B teachers with Qwen3-1.7B and Qwen3-4B students, using raw instruction models or the same 400K SFT initialization as marked in each table, with GRPO, GKD, MiniLLM, and DistiLLM as baselines. We also directly evaluate the official Hugging Face releases of Qwen3-8B, Mistral-7B-Instruct-v0.3, and Llama-3.1-8B-Instruct without additional training.

Training data. We randomly sample both v1 and v2 from OpenThoughts using seed 44. The prompt-only v1 set contains 20,000 prompts, predominantly competition mathematics, with estimated shares of 78.4% mathematics, 17.4% code, $\approx$ 3.2% science, and $\approx$ 1.0% commonsense or logic. Without domain labels, we infer mathematics and code from the `\boxed{` answer format and executable-Python-function template, respectively, and estimate other domains manually. v2 contains 15,000 OpenThoughts3-1.2M prompts, with 5,000 each from mathematics, code, and science. Both sets use responses generated on-policy by the current student and are reported separately.

Training and evaluation protocol. Hyperparameters (Table 9), compute budgets, checkpoint selection, and the shared six-benchmark pass@1 evaluation protocol are detailed in Appendix C.

4.2 Main Results

Cross-tokenizer distillation. Table 1 shows that CAST achieved the highest six-benchmark average for both student families: 47.22 for Mistral and 43.87 for Llama, improving over ULD by 2.45 and 1.32 points, respectively. CAST led ULD on five benchmarks for Mistral and four for Llama; MATH-500 was lower for both, as was GPQA for Llama. Relative to the SFT initialization, CAST improved five of six Mistral scores and raised the average from 44.33 to 47.22, while LCB declined slightly from 28.0 to 27.7. Llama improved on all three AIME benchmarks and MATH-500, while GPQA and LCB declined; its average increased from 42.08 to 43.87. Original DPCA reached averages of 41.40 and 41.75, respectively.

Shared-tokenizer distillation. With Qwen3-8B as teacher and v1 prompts, CAST improves all six benchmarks over raw Qwen3-1.7B, reaching 48.68 Avg-6 versus 46.58 for the strongest baseline, DistiLLM (Table 2). It exceeds uniform and shuffled credit by 2.26 and 3.20 points (Table 3).

Table 1: Cross-tokenizer pass@1 (%). Avg.: six-benchmark macro mean (two decimals). None: no additional training; SFT init: distillation initialization; DPCA: original DPCA+OPD. **Bold** marks the best scores per student; teacher scores are separate.

Student	Teacher	Method	A24	A25	A26	M500	GPQA	LCB	Avg.
–	Qwen3-8B	None	69.5	56.6	60.4	81.0	57.8	59.0	64.05
Mistral-7B	–	None	0.2	0.0	0.0	13.8	23.0	4.6	6.93
	–	SFT init	41.5	36.5	48.3	74.1	37.6	**28.0**	44.33
	Qwen3-8B	GRPO	19.4	23.1	25.4	65.5	31.2	21.2	30.97
	Qwen3-8B	MinED	44.7	36.8	47.9	74.4	34.6	25.7	44.02
	Qwen3-8B	ULD	45.5	36.8	47.3	**79.6**	37.2	22.2	44.77
	Qwen3-8B	ALM	22.3	18.9	26.2	63.5	15.7	12.5	26.52
	Qwen3-8B	DPCA	41.8	35.8	47.7	73.8	35.4	13.9	41.40
	Qwen3-8B	**CAST**	**45.7**	**40.9**	**51.0**	78.5	**39.5**	27.7	**47.22**
Llama-3.1-8B	–	None	3.3	0.4	1.7	45.5	26.5	9.8	14.53
	–	SFT init	39.0	35.8	39.2	73.9	**37.4**	**27.2**	42.08
	Qwen3-8B	GRPO	40.9	35.2	40.2	73.2	37.2	26.2	42.15
	Qwen3-8B	MinED	41.0	36.8	39.4	74.9	32.7	23.6	41.40
	Qwen3-8B	ULD	41.6	36.2	41.9	75.1	34.7	25.8	42.55
	Qwen3-8B	ALM	35.6	29.3	40.8	**76.2**	36.9	26.4	40.87
	Qwen3-8B	DPCA	39.7	37.6	41.9	73.9	33.3	24.1	41.75
	Qwen3-8B	**CAST**	**44.1**	**37.7**	**45.0**	75.0	34.5	26.9	**43.87**

Table 2: Shared-tokenizer pass@1 (%) with raw Qwen3-1.7B, a Qwen3-8B teacher, and v1.

Student	Teacher	Method	A24	A25	A26	M500	GPQA	LCB	Avg.
–	Qwen3-8B	None	69.5	56.6	60.4	81.0	57.8	59.0	64.05
Qwen3-1.7B	–	None	46.2	34.0	42.5	75.9	37.8	34.1	45.08
Qwen3-1.7B	Qwen3-8B	GRPO	44.5	33.2	41.2	75.5	36.4	35.5	44.38
Qwen3-1.7B	Qwen3-8B	GKD	50.3	35.4	42.7	76.2	36.6	**35.6**	46.13
Qwen3-1.7B	Qwen3-8B	MiniLLM	48.5	35.1	46.7	76.9	35.0	34.0	46.03
Qwen3-1.7B	Qwen3-8B	DistiLLM	49.4	**39.1**	44.2	76.4	37.4	33.0	46.58
Qwen3-1.7B	Qwen3-8B	**CAST**	**51.6**	37.0	**51.7**	**77.2**	**40.0**	34.6	**48.68**

Table 3: Shared-tokenizer CAST ablations (Qwen3-8B $\rightarrow$ raw Qwen3-1.7B, v1). Uniform uses equal weights; shuffled permutes CAST weights within chunks; fixed K=6 replaces punctuation-defined chunks.

Setting	A24	A25	A26	M500	GPQA	LCB	Avg.
Uniform credit	46.6	36.7	46.5	76.5	38.3	33.9	46.42
Fixed chunks (K=6)	48.9	36.6	46.5	76.3	39.3	**35.1**	47.12
Shuffled credit	48.2	36.6	42.9	76.2	36.0	33.0	45.48
CAST	**51.6**	**37.0**	**51.7**	**77.2**	**40.0**	34.6	**48.68**

Shuffling preserves chunk weights but changes their token assignments, supporting the importance of credit placement. Punctuation-defined chunks outperform fixed-length chunks (K=6) by 1.56 points, although the latter score higher on LCB (35.1 versus 34.6). These controls support CAST's *shift* rule under exact token correspondence. Broader comparisons show that its gains vary with teacher–student scale and prompt composition (Tables 7 and 8 in Appendix B.1).

4.3 Discussions

Credit assignment beyond chunk-level alignment. In Table 4, the matched uniform control retains decoded-text alignment, selective anchoring, and two-rollout sampling but removes structured token credit, reducing the average from 47.22 to 45.07. Shuffling preserves CAST's within-chunk weights but breaks their token correspondence; CAST improves five of six benchmarks over this control, raising the average from 44.17 to 47.22, with gains of 4.8 on AIME-2026 and 6.0 on GPQA, while LCB is slightly lower (27.7 versus 27.9). These controls support CAST's token-specific *shift* credit. Appendix B.3 illustrates its allocation within a chunk and across token categories in one rollout (Figure 6).

Table 4: Mistral component ablations (Qwen3-8B teacher). Credit columns identify weighting signals; Sel./Global anchor unaligned/all positions. Shuffled permutes CAST weights within each chunk; rollouts count responses per prompt.

Setting	Credit assignment			KL anchor	Rollouts			Pass@1 (%)						
	Unif.	Uncert.	Smooth.		1	2	4	A24	A25	A26	M500	GPQA	LCB	Avg.
DPCA	✓			None	✓			41.8	35.8	47.7	73.8	35.4	13.9	41.40
Uniform credit	✓			Sel.		✓		42.3	37.2	46.7	78.9	38.5	26.8	45.07
Shuffled credit		✓	✓	Sel.		✓		45.5	36.4	46.2	75.5	33.5	27.9	44.17
Uncertainty only		✓		Sel.		✓		**48.3**	**41.6**	51.2	75.3	**40.2**	17.0	45.60
Smoothness only			✓	Sel.		✓		45.7	35.8	46.9	73.5	37.1	21.1	43.35
No anchoring		✓	✓	None		✓		47.5	37.1	52.5	73.8	35.6	17.5	44.00
Global anchoring		✓	✓	Global		✓		46.1	37.2	49.4	75.6	37.6	19.9	44.30
No anchor + single		✓	✓	None	✓			46.4	39.9	46.9	77.9	37.9	27.0	46.00
Single rollout		✓	✓	Sel.	✓			45.2	35.7	48.5	**81.5**	37.1	**28.4**	46.07
Four rollouts		✓	✓	Sel.			✓	47.2	37.2	**52.7**	75.2	38.1	19.9	45.05
CAST		✓	✓	Sel.		✓		45.7	40.9	51.0	78.5	39.5	27.7	**47.22**

Table 5: Three-seed Mistral pass@1 (%, lr 10^{-6}): mean ± sample SD of highest-Avg-6 evaluated checkpoints (one per seed). Per-seed results and selection: Appendix E.

Method	A24	A25	A26	M500	GPQA	LCB	Avg.
DPCA	40.4 ± 1.8	35.8 ± 1.3	45.7 ± 1.8	73.8 ± 0.1	35.1 ± 0.9	13.2 ± 3.2	40.66 ± 0.86
Uniform credit	44.4 ± 1.8	36.1 ± 1.0	46.2 ± 1.2	77.8 ± 1.2	37.2 ± 1.2	27.2 ± 1.0	44.81 ± 0.22
Uncertainty only	49.0 ± 1.5	40.7 ± 1.0	49.7 ± 1.3	74.7 ± 1.1	39.3 ± 0.8	17.8 ± 0.8	45.21 ± 0.64
CAST	45.6 ± 0.7	39.9 ± 0.9	50.1 ± 1.2	79.0 ± 1.2	39.5 ± 0.7	27.9 ± 0.2	47.01 ± 0.28

Table 6: Three-seed Llama-3.1-8B pass@1 (%, Qwen3-8B teacher): mean ± sample SD. Per-seed checkpoints and aggregation: Appendix E.

Method	A24	A25	A26	M500	GPQA	LCB	Avg.
ULD	41.4 ± 0.3	34.9 ± 1.1	41.3 ± 1.0	74.6 ± 0.8	34.5 ± 1.3	25.7 ± 0.1	42.08 ± 0.43
CAST	43.3 ± 0.7	38.0 ± 0.9	43.5 ± 1.7	74.8 ± 0.6	36.0 ± 1.3	26.7 ± 0.2	43.72 ± 0.16

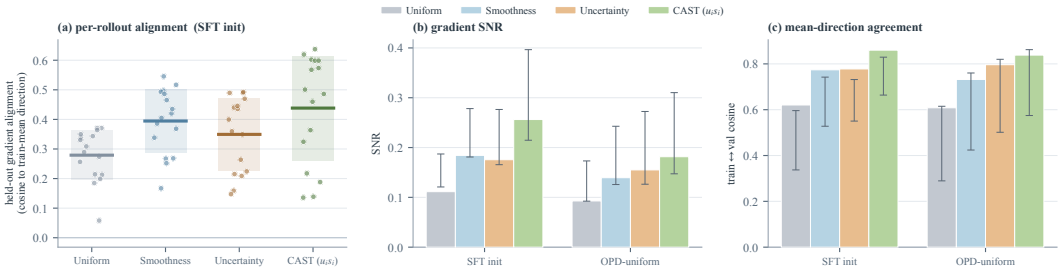

Figure 4: **Gradient consistency across four credit rules.** Sketched Mistral LM-head gradients use 48 shared rollouts per checkpoint (32 train, 16 held out). **(a)** Held-out cosine with each rule's training mean at SFT initialization (mean ± SD). **(b)** SNR. **(c)** Training–held-out mean cosine. Error bars in b,c: 95% bootstrap CIs. Protocol: Appendix B.5.

Performance across training seeds. Tables 5 and 6 summarize three seeds per configuration. CAST reaches 47.01 ± 0.28 Avg-6 on Mistral and 43.72 ± 0.16 on Llama, versus 42.08 ± 0.43 for Llama ULD. For both students, all three CAST scores exceed the corresponding comparators' highest Avg-6 (Tables 10–11 in Appendix E).

CAST yields more consistent gradient directions. Figure 4 compares four credit rules on shared rollouts at SFT initialization and an OPD-uniform checkpoint. CAST has the highest gradient signal-to-noise ratio (SNR) and training–held-out mean-direction agreement at both; SFT SNR rises from 0.111 under uniform credit to 0.256 under CAST (protocol in Appendix B.5). Complementary training traces in Figure 5 show higher teacher endorsement and policy entropy, smaller gradient

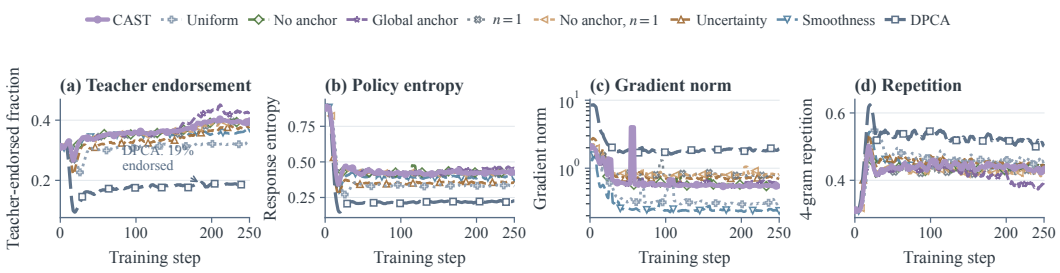

Figure 5: **Mistral training dynamics.** **(a)** Positive-teacher-advantage fraction; **(b)** policy entropy; **(c)** gradient norm; **(d)** 4-gram repetition. Curves: five-step moving averages. Details: Appendix B.4.

norms, and lower repetition than DPCA over the final 50 updates. Appendix B.4 details these runs; selected 2024 AIME I responses in Appendix D illustrate CAST reaching final answers where SFT or DPCA responses repeat or remain unfinished.

Smoothness helps mitigate training collapse. Smoothness produces a more balanced domain profile by moderating uncertainty-driven credit: with selective anchoring and two rollouts fixed, LCB rises from 17.0 to 27.7 and Avg-6 from 45.60 to 47.22, while the other five benchmarks change by -0.20 points on average (Table 4). This moderation discounts uncertain tokens when teacher scores vary sharply, redistributing credit toward more locally regular supervision. It primarily uses teacher structure: 79% of redistribution occurs at teacher-variation positions, and comparable variation persists without projection under shared tokenization (Figure 7 in Appendix B.6). Separate seed-44 diagnostics show slightly broader effective credit support for CAST (2.814 tokens versus 2.632 for uncertainty-only weighting; Figure 8, Appendix E). This broader, teacher-informed allocation offers a plausible explanation for CAST's fewer observed collapses than uncertainty-only credit across six seeds (Table 12, Appendix E).

The role of selective anchoring. With CAST credit and n=2 fixed, selective anchoring reaches 47.22 Avg-6, versus 44.30 with global anchoring and 44.00 without it; the largest selective–global gain is on LCB (+7.8; Table 4). With selective anchoring, two rollouts outperform one (+1.15 Avg-6) and four (Appendix B.2). Over the final 50 updates, global anchoring has higher entropy and lower repetition than CAST, yet weaker downstream performance (Figure 5b,d), consistent with overconstraint: global anchoring penalizes departures from SFT at aligned positions, potentially opposing teacher credit. Selective anchoring instead constrains drift at unaligned positions while leaving aligned positions to teacher supervision.

5 Conclusion

We introduced CAST for token-level *shift-or-stay* cross-tokenizer OPD: uncertainty–smoothness credit redistributes aligned chunk advantages, while selective anchoring keeps unaligned predictions close to the initial policy. The advantage-preserving credit rule admits an exact KL-regularized characterization and extends to shared-tokenizer punctuation-defined or fixed-length chunks. CAST improves the six-benchmark average over the strongest evaluated baseline by 1.32–2.45 points for heterogeneous students and 2.10 for Qwen3-1.7B. Controls support token-specific credit placement, more consistent LM-head gradients, and gains across Mistral and Llama seeds. Smoothness recovers code performance and broadens effective credit support, alongside fewer observed collapses; selective anchoring improves the two-rollout configuration.

Limitations

Our evaluation covers a limited set of model families and mainly reasoning-oriented training data, so domain gains may change with student capacity or prompt composition. Training compute varies across methods, and repeated-seed performance covers four Mistral and two Llama configurations. Broader model and data coverage, matched-compute evaluation, and direct tests of how credit allocation affects full-model optimization and training collapse remain future work.

Reproducibility Statement

The method is described in §3, with credit weights in Eq. (5), selective anchoring in Eqs. (8)–(9), and the training objective in Eq. (11). The experimental setup is in §4.1; training and evaluation protocols are in Appendix C. Tables 5 and 6 report cross-seed statistics, with individual checkpoints and collapse records in Appendix E. Protocols for the main-text gradient and training diagnostics (Figures 4 and 5) are in Appendices B.5 and B.4. The token-level example is provided in Appendix B.3. Figure-generation scripts use reported results, recorded token-level traces, gradient summaries, and training logs.

AI Use Disclosure

Generative AI tools were used for manuscript polishing.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024. 1, 3

Nicolas Boizard, Kevin El Haddad, Céline Hudelot, and Pierre Colombo. Towards cross-tokenizer distillation: the universal logit distillation loss for LLMs. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. 2, 3

Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. Vold: Reasoning transfer from llms to vision-language models via on-policy distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26209–26218, 2026. 2

Xiao Cui, Mo Zhu, Yulei Qin, Liang Xie, Wengang Zhou, and Houqiang Li. Multi-level optimal transport for universal cross-tokenizer knowledge distillation on language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 23724–23732, 2025. 2, 3

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: MathArena as an evaluation platform for mathematics with LLMs. arXiv preprint arXiv:2605.00674, 2026. 3, 17

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 6

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024. 1, 3

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, et al. Openthoughts: Data recipes for reasoning models. In *International Conference on Learning Representations*, volume 2026, pp. 108059–108130, 2026. 6

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. 3

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe. 3, 17

Naman Jain, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pp. 58791–58831, 2025. 3, 17

Ijun Jang, Jewon Yeom, Juan Yeo, Hyunggyu Lim, and Taesup Kim. Stable on-policy distillation through adaptive target reformulation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 42217–42227, 2026. 1, 3

Seongryong Jung, Suwan Yoon, DongGeon Kim, and Hwanhee Lee. Todi: Token-wise distillation via fine-grained divergence control. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8089–8102, 2025. 4

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 24872–24895. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/ko24c.html. 1, 3

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. Distillm-2: a contrastive approach boosts the distillation of llms. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025. 3

Zhi-Ping Liu, Simiao Li, Wei Li, Hanting Chen, Jie Hu, Hua-Lei Yin, and Xinghao Chen. Entropy-aware span-constrained optimal transport for robust cross-tokenizer knowledge distillation. In *Forty-third International Conference on Machine Learning*, 2026. 4

Benjamin Minixhofer, Ivan Vulić, and Edoardo Maria Ponti. Universal cross-tokenizer distillation via approximate likelihood matching. *Advances in Neural Information Processing Systems*, 38: 79297–79326, 2026. 2, 3

Yifan Niu, Han Xiao, Dongyi Liu, Zelong Wang, Dihong Gong, Yasheng Wang, and Jia Li. Breaking the tokenizer barrier: On-policy distillation across model families. arXiv preprint arXiv:2606.09456, 2026. 2, 4, 12

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98. 3, 17

Avyav Kumar Singh, Yen-Chen Wu, Alexandru Cioba, Alberto Bernacchia, and Davide Buffelli. Cross-tokenizer llm distillation through a byte-level interface. In *Proceedings of the Second Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual (CustomNLP4U)*, pp. 84–96, 2026. 2, 3

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. Knowledge fusion of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 18303–18322, 2024. 3

Hongling Xu, Qi Zhu, Heyuan Deng, Jinpeng Li, Lu Hou, Yasheng Wang, Lifeng Shang, Ruifeng Xu, and Fei Mi. Kdrl: Post-training reasoning llms via unified knowledge distillation and reinforcement learning. *arXiv preprint arXiv:2506.02208*, 2025. 3

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. 2

Songming Zhang, Xue Zhang, Tong Zhang, Bojie Hu, Yufeng Chen, and Jinan Xu. Alignditil: Token-level language model alignment as adaptive policy distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19791–19807, 2025. 4

A Additional Method Details

A.1 Chunk Construction

Heterogeneous tokenizers. We use DPCA's decoded-text chunk alignment (Niu et al., 2026): a greedy two-pointer scan extends the shorter side until the decoded student and teacher strings match after Unicode and tokenizer-specific whitespace normalization. Each matched pair of contiguous token groups (I_c, J_c) forms an *alignment chunk*. We discard excessively large or unreliable matches; retained student positions form $\mathcal{A}$ and the remainder $\mathcal{U}$. Each retained alignment chunk's teacher score is projected onto the student grid as

$$\widetilde{\ell}_i^T = \frac{1}{|I_c|} \sum_{j \in J_c} \ell_j^T, \qquad i \in I_c. \tag{12}$$

After the projection in Eq. (12), we regroup contiguous aligned student tokens into fixed-length K-token *credit chunks* (plus a shorter residual chunk), collected in $\mathcal{G}$. These define the units for advantage computation and token-level credit allocation; their boundaries need not coincide with alignment-chunk boundaries. We use $K = 6$ unless stated otherwise.

Shared tokenizer. The training path uses identity correspondence between response tokens, so $M = N$, $\widetilde{\ell}_i^T = \ell_i^T$, and $\mathcal{U} = \varnothing$. We split responses after commas, periods, exclamation marks, and question marks (with any trailing suffix as the final chunk); these punctuation-defined credit chunks directly form $\mathcal{G}$ without fixed-K regrouping. The fixed-length variant in Table 3 instead partitions the response into consecutive $K = 6$ token chunks, with a shorter final chunk when needed.

B Additional Experiments and Diagnostics

B.1 Teacher Scale and Training Distribution

Table 7: Shared-tokenizer teacher–student scaling results from raw Qwen3 instruction models. The method column identifies results trained with CAST.

Student	Teacher	Method	A24	A25	A26	M500	GPQA	LCB	Avg.
Qwen3-1.7B	None (raw)	–	46.2	34.0	42.5	75.9	37.8	34.1	45.08
	Qwen3-8B	**CAST**	**51.6**	37.0	**51.7**	**77.2**	**40.0**	34.6	**48.68**
	Qwen3-30B-A3B	**CAST**	50.2	**39.4**	49.0	76.5	37.9	**35.3**	48.05
Qwen3-4B	None (raw)	–	66.5	58.6	58.8	**81.7**	**54.5**	**54.4**	62.42
	Qwen3-8B	**CAST**	66.4	59.3	60.6	81.2	54.3	52.2	62.33
	Qwen3-30B-A3B	**CAST**	**69.2**	**63.6**	**66.9**	81.5	53.7	52.6	**64.58**

Teacher–student scaling. Table 7 shows that the benefit of a larger teacher depends on student capacity. For Qwen3-1.7B, the 8B teacher reaches 48.68, above the 30B-A3B teacher's 48.05; both exceed the raw student's 45.08. For Qwen3-4B, the 8B teacher leaves the average nearly unchanged (62.33 versus 62.42), whereas the 30B-A3B teacher reaches 64.58 and improves all three AIME benchmarks. Thus the larger teacher benefits the 4B student but provides no additional average gain for the 1.7B student in these comparisons.

Training distribution and teacher scale. Table 8 compares prompt sets and teacher scale across raw Qwen3 students. For the 1.7B student with an 8B teacher, CAST reaches averages of 48.68 on v1 and 47.48 on v2. For the 4B student, v2 raises the average from 62.33 to 64.05 with the 8B teacher and from 64.58 to 64.93 with the 30B-A3B teacher. The larger teacher obtains the higher 4B average under both prompt sets, with the gains concentrated on the three AIME benchmarks.

Prompt distribution further affects code retention in the shared-tokenizer setting (Table 8). With an 8B teacher, Qwen3-1.7B retains LCB near the raw student's 34.1, reaching 34.6/34.4 on v1/v2. For Qwen3-4B, v2 reduces the LCB loss relative to the raw model under both teachers: scores increase from 52.2 to 53.9 with the 8B teacher and from 52.6 to 53.4 with the 30B-A3B teacher,

Table 8: Student, teacher, and prompt-set comparison from raw Qwen3 instruction models. All trained rows use CAST with n=2.

Student	Teacher	Data	Method	A24	A25	A26	M500	GPQA	LCB	Avg.
Qwen3-1.7B	–	–	None	46.2	34.0	42.5	75.9	37.8	34.1	45.08
	Qwen3-8B	v1	CAST	**51.6**	37.0	**51.7**	**77.2**	**40.0**	34.6	**48.68**
	Qwen3-8B	v2	CAST	49.1	37.9	48.8	76.8	37.9	34.4	47.48
Qwen3-4B	–	–	None	66.5	58.6	58.8	81.7	54.5	**54.4**	62.42
	Qwen3-8B	v1	CAST	66.4	59.3	60.6	81.2	54.3	52.2	62.33
	Qwen3-8B	v2	CAST	67.7	62.8	63.3	**82.1**	54.5	53.9	64.05
	Qwen3-30B-A3B	v1	CAST	69.2	**63.6**	**66.9**	81.5	53.7	52.6	64.58
	Qwen3-30B-A3B	v2	CAST	**70.3**	62.6	**66.9**	81.0	**55.4**	53.4	**64.93**

alongside higher Avg-6. These results locate the observed trade-offs in both credit design and prompt distribution.

B.2 Additional Sampling Analysis

Sampling multiplicity and selective anchoring. Table 4 compares $n \in \{1, 2, 4\}$ independent responses per prompt, generating 64, 128, or 256 responses per update at a fixed batch of 64 prompts. At the reported checkpoints, with selective anchoring fixed, two rollouts improve four of six benchmarks over one, including gains of 0.5, 5.2, and 2.5 points on AIME-2024, AIME-2025, and AIME-2026, and raise Avg-6 from 46.07 to 47.22. Four rollouts achieve the strongest AIME-2024 and AIME-2026 scores among these settings, but reduce LCB by 7.8 points relative to n=2 and lower Avg-6 to 45.05. Without anchoring, increasing from one to two rollouts instead lowers Avg-6 from 46.00 to 44.00. Thus, selective anchoring contributes +3.22 points at n=2 but only +0.07 at n=1, a difference of 3.15 points in these estimates. Anchor scope also matters: at n=2, selective anchoring improves five of six benchmarks over global anchoring (47.22 versus 44.30 Avg-6), with the largest gap on LCB (27.7 versus 19.9). These comparisons favor CAST's two-rollout configuration with selective anchoring among the tested settings; the benefit of additional responses depends on the accompanying regularization.

B.3 Token-Level Credit Analysis

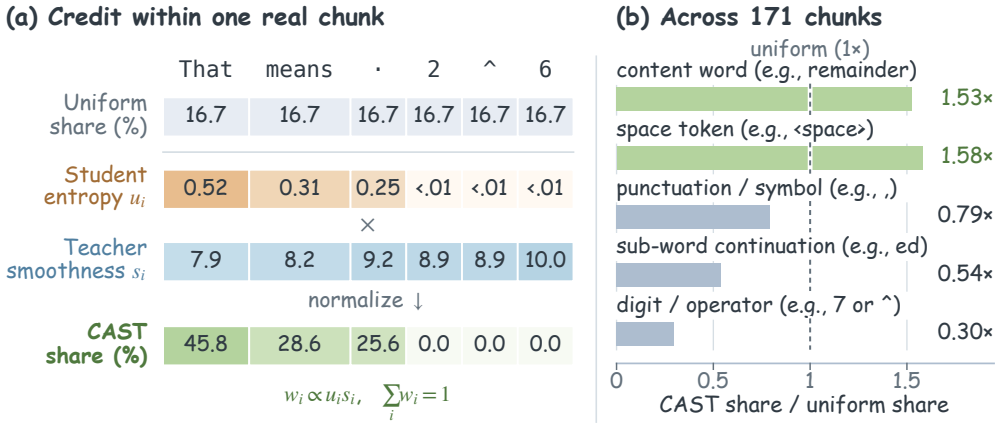

Figure 6: **CAST redistributes credit to individual tokens.** **(a)** Uniform shares, uncertainty u_i, smoothness s_i, and CAST weights w_i for a six-token chunk from a Qwen3-8B → Mistral-7B rollout. **(b)** CAST-to-uniform credit-share ratios by token category across all 171 chunks of the same rollout; ratios above 1 indicate upweighting.

Token-level analysis of credit assignment. Figure 6 illustrates the credit rule in Eq. (5) on a recorded training rollout. In panel (a), uniform credit assigns 16.7% to each token, whereas CAST assigns 45.8%, 28.6%, and 25.6% to the three high-entropy positions. The near-deterministic digit and operator tokens in 2^6 receive effectively no share. Panel (b) aggregates the complete rollout: content words and space tokens receive 1.53× and 1.58× their uniform shares, while punctuation, sub-word continuations, and digit/operator tokens receive 0.79×, 0.54×, and 0.30×. The controlled downstream comparison is in Table 4.

B.4 Training Dynamics: Protocol and Analysis

Diagnostic protocol. The main-text training curves in Figure 5 compare Qwen3-8B → Mistral-7B CAST with seven pipeline variants and DPCA. Global anchor regularizes all response positions; No anchor removes reference regularization. Pipeline variants use two rollouts unless marked n=1. DPCA uses the plain OPD loss with uniform within-chunk credit and one rollout. The Uniform diagnostic uses learning rate 5×10^{-7}; other runs use 10^{-6}. These runs are separate from the seed evaluation in Table 5. The teacher-endorsed fraction is the fraction of rollout tokens for which the teacher assigns a higher log-probability than the student. The other panels show mean per-token

response-policy entropy, gradient norm on a log scale, and rollout 4-gram repetition. Curves show five-step moving averages.

Training dynamics and stability. Credit concentration, averaged over training, shows that CAST assigns each six-token chunk's budget to an effective 2.8 tokens, whereas smoothness-only credit remains diffuse over 5.3 tokens. This agrees with the uncertainty-sensitive token-level behavior in Figure 6. The training curves show how these configurations evolve. Over the final 50 updates, DPCA's teacher-endorsed fraction falls to 0.19, compared with 0.40 for CAST, so the teacher would downweight roughly 80% of the sampled tokens (Figure 5a). At the same time, DPCA reaches a response entropy of 0.22 versus 0.43 for CAST, a mean gradient norm of 1.86 versus 0.55, and a 4-gram repetition rate of 0.51 versus 0.44. Together, these trends describe a low-entropy policy whose sampled tokens often receive lower teacher scores, with larger gradients and more repetitive generation. The uniform-credit diagnostic run lies between CAST and DPCA in teacher endorsement, response entropy, and repetition, while having a smaller gradient norm.

The added anchor controls show that these diagnostics need to be interpreted alongside downstream performance. Over the same final 50 updates, global anchoring reaches teacher endorsement 0.43, response entropy 0.45, and repetition 0.39, compared with 0.40, 0.43, and 0.44 for CAST. Despite these favorable entropy and repetition statistics, global anchoring has a lower reported six-benchmark average (44.30 versus 47.22; Table 4). With one rollout, removing anchoring leaves endorsement, entropy, and repetition close to the selectively anchored $n{=}1$ run, while increasing the mean gradient norm from 0.79 to 0.85. Together with the anchor–rollout comparison in Table 4, these curves show that anchoring affects optimization behavior and downstream transfer differently across settings.

B.5 Gradient Diagnostic Protocol

Figure 4 compares four credit rules at the Mistral-7B SFT initialization and an OPD-uniform (DPCA) checkpoint at step 150; neither policy was trained with CAST's product rule. At each checkpoint, the rules reuse the same 48 on-policy rollouts, generated with the training response limit of 16,384 tokens. We normalize weights proportional to 1, s_i, u_i, or $u_i s_i$ within each chunk and form token advantages $A_g w_i$. We measure the resulting policy gradients on the language-model head and compress them with a fixed CountSketch shared by all rules.

Let g denote a sketched per-rollout head gradient. Gradient SNR is $\|\mathbb{E}[g]\|^2/\mathbb{E}\|g-\mathbb{E}[g]\|^2$, which measures the squared mean-gradient magnitude relative to variation across rollouts. For direction comparisons, we split the rollouts into 32 for mean-gradient estimation and 16 held out. Figure 4a shows each held-out rollout gradient's cosine with its rule's training-set mean gradient; Figure 4c compares the training- and held-out-set mean gradients. Bars in panels b and c show point estimates with 95% bootstrap intervals from 2,000 resamples. Each rule is compared with its own mean direction, so these metrics measure consistency across sampled rollouts.

CAST has the highest SNR and training–held-out mean-gradient cosine at both checkpoints, with the single-factor rules between CAST and uniform credit. Relative to uniform credit, SNR increases from 0.111 to 0.256 at SFT initialization and from 0.093 to 0.182 at the DPCA checkpoint. At SFT initialization, mean held-out rollout alignment also increases from 0.279 to 0.438. These observations concern sampled head gradients at the two fixed policies; downstream effects are assessed by the credit-rule controls in Table 4.

B.6 Smoothness Attribution

Teacher-signal variation and projection. Equation (12) in Appendix A.1 assigns each alignment chunk's aggregate teacher score uniformly to its student tokens. Although a credit chunk can cross alignment-chunk boundaries, the projected signal is constant within each alignment chunk. The smoothness term could therefore reward alignment geometry rather than genuine teacher structure. We test this explanation with two analyses that require no retraining (Figure 7). First, alignment chunks containing multiple student tokens cover 19% of student tokens in the cross-tokenizer rollouts. Defining smoothness-only credit as $w_i^S = s_i / \sum_{j\in g} s_j$, 79% of its re-weighting mass $|w_i^S - 1/K_g|$ falls on teacher-variation positions (alignment-chunk boundaries and 1:1 positions),

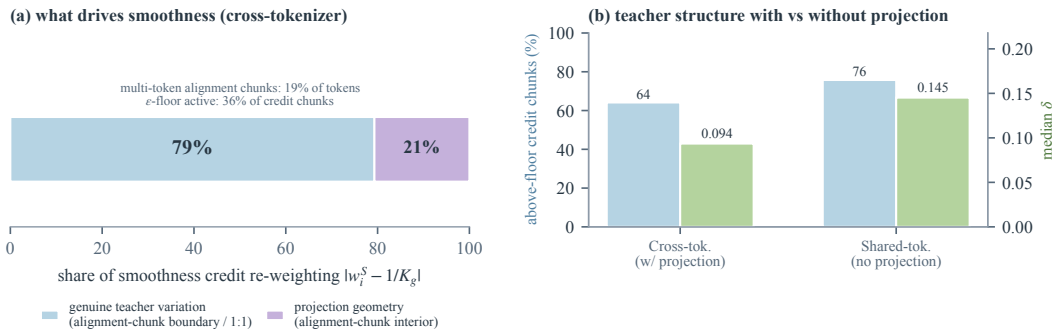

Figure 7: **Smoothness primarily reflects teacher structure rather than projection geometry.** **(a)** On cross-tokenizer rollouts (Mistral ← Qwen3-8B), 79% of the credit re-weighting induced by smoothness-only allocation, $|w_i^S - 1/K_g|$, falls on teacher-variation positions (alignment-chunk boundaries and 1:1 positions), while 21% falls at projection-flat alignment-chunk interiors. Alignment chunks containing multiple student tokens cover 19% of tokens; the ϵ floor is active in 36% of credit chunks. **(b)** Cross-tokenizer projection is compared with shared-tokenizer Qwen3-8B → Qwen3-4B, where $M = N$ and no projection is used. The shared-tokenizer setting has more credit chunks whose mean variation exceeds the floor and a larger median δ. Both panels are computed offline from saved rollouts and teacher scores, without additional training.

compared with 21% at projection-flat alignment-chunk interiors. The ϵ floor is active in 36% of credit chunks, limiting the influence of small local variations.

Second, the shared-tokenizer setting removes projection entirely because $M = N$. Its teacher signal retains at least comparable local structure: within credit chunks, median local variation is 0.145, compared with 0.094 cross-tokenizer, and mean variation exceeds $\epsilon_{\min}$ in 76% of credit chunks, compared with 64%. The median within-chunk smoothness-weight spread is also similar ($s_{\max}/s_{\min} = 2.68$ versus 2.60). Together, these analyses indicate that smoothness responds primarily to teacher-signal variation, while projection geometry contributes a minority of its observed re-weighting.

C Training and Evaluation Details

Table 9 lists the main cross-tokenizer training configuration. Tables 1–4 report checkpoint-level point estimates. Tables 5 and 6 summarize three evaluated seeds for four Mistral and two Llama configurations; Appendix E provides the individual checkpoints, scores, and collapse records. Training compute varies across methods, which share the evaluation protocol described below.

Training protocol. Unless otherwise specified, training uses a prompt batch size of 64, learning rate 10^{-6}, and maximum prompt and response lengths of 1,024 and 16,384 tokens, respectively. Heterogeneous pairs use regroup size K=6. By default, CAST uses multi-rollout sampling (n=2): both independently sampled responses per prompt contribute to training, without best-response selection or cross-response advantage normalization. The teacher only provides log-probabilities on student responses.

Compute and token budget. We implement training in the `verl` RL framework. Each run uses one node with eight H100 GPUs: four score responses with the frozen Qwen3-8B teacher using tensor parallelism, and four perform student rollout generation and policy updates. Training runs for 250 optimizer steps, with 64 prompts and n rollouts per prompt at each step (Table 4). The maximum prompt and response lengths are 1,024 and 16,384 tokens, respectively, and the learning rate is 10^{-6}.

For the default CAST configuration (n=2), the mean response length is 11.1k tokens. Each step therefore generates $\approx 64 \times 2 \times 11.1\text{k} \approx 1.42\text{M}$ response tokens, or $\approx 1.48\text{M}$ actor tokens including prompts. Over training, these totals are $\approx 355\text{M}$ generated tokens and $\approx 369\text{M}$ actor tokens. The teacher scores each rollout once, giving $64 \times 2 \times 250 = 32{,}000$ scoring forwards over $\approx 355\text{M}$

Table 9: Training hyperparameters for CAST (cross-tokenizer setting).

Hyperparameter	Value
Teacher / students	Qwen3-8B / {Mistral-7B, Llama-3.1-8B}
Student initialization	SFT on OpenThoughts 400K
Prompt-set sampling seed (v1 / v2)	44
Optimizer learning rate	1×10^{-6}
LR warmup	10 steps
Weight decay	0.01
Gradient clipping	1.0
PPO clip (ε_{low} / $\varepsilon_{\text{high}}$)	0.2 / 0.28
Advantage clamp	± 10.0
Rollouts per prompt n	2
Prompt batch size	64
Max prompt / response length	1024 / 16384 tokens
Regroup size K	6
Entropy floor η	10^{-8}
Smoothness floor $\epsilon_{\min}$	0.1
KL anchor coefficient λ_{KL}	0.2
Loss aggregation	token-mean
Rollout temperature (train / eval)	1.0 / 0.6
Total steps	250

teacher tokens. Wall-clock time is $\approx$ 30 h per run, or $\approx$ 437 s per step. The n=1 configurations (DPCA, single rollout, and no anchor+single) use half this generation and teacher-scoring budget: $\approx$ 178M tokens and 16,000 teacher forwards. The n=4 configuration uses twice the n=2 budget. At a given n, ULD, MinED, and ALM share the generation and teacher-scoring volumes but differ in alignment and loss computation.

Checkpoint selection. We save checkpoints every 25 steps and evaluate them on all six benchmarks. The point-estimate comparisons use one selected checkpoint per configuration. For the four repeated Mistral configurations, Tables 1 and 4 use the highest Avg-6 across evaluated seed–checkpoint pairs. Table 5 instead averages each seed's highest-Avg-6 evaluated checkpoint (details in Appendix E). For Llama CAST and ULD, Table 1 uses the highest Avg-6 among the three per-seed checkpoints in Table 11; Table 6 summarizes all three. Benchmark-based selection can make the reported averages optimistic.

Evaluation protocol and run variability. We evaluate AIME 2024, 2025, and 2026 (Dekoninck et al., 2026), MATH-500 (Hendrycks et al., 2021), GPQA-Diamond (Rein et al., 2024), and LiveCodeBench (Jain et al., 2025) at temperature 0.6 and top-p 0.95. We estimate pass@1 using 32 samples per AIME problem and four per problem on other benchmarks. Avg-6 is the unweighted mean of the six displayed one-decimal scores, rounded to two decimals. We also inspect individual benchmarks to distinguish aggregate gains from domain-specific effects. The annual AIME sets contain 30, 30, and 15 problems for 2024, 2025, and 2026, respectively. Each problem accounts for 3.3, 3.3, and 6.7 percentage points in the corresponding score. MATH-500 (500 problems), GPQA-Diamond (198), and LiveCodeBench (960) provide considerably more stable estimates. We therefore emphasize patterns that repeat across AIME years, the other benchmarks, both students (Mistral and Llama), and the component ablations (Table 4), together with the mechanism analysis. Tables 5 and 6 report means and sample standard deviations across three evaluated seeds for four Mistral and two Llama configurations; Tables 10 and 11 list their individual scores. Other comparisons use single runs. Figure 5 gives separate training diagnostics.

D CASE STUDY

The following cases compare the SFT initialization, DPCA, ULD, and CAST on four selected problems from the 2024 AIME I. Each panel presents a condensed question, excerpts from the final 240 characters of each recorded response, the extracted-answer outcome, and an analysis of the visible behavior. Ellipses indicate omitted text, with mathematical formatting normalized; – denotes no extracted final answer. The examples illustrate repetition and answer completion, complementing the training traces in Figure 5 and aggregate results in Table 1.

Case study 1: Grid paths and repeated direction sequences

Question. How many length-16 paths from the lower-left to the upper-right corner of an 8×8 grid change direction exactly four times?
Reference answer: 294. *2024 AIME I, Problem 6*

Method	Response excerpts	Result
CAST	Total ways for this case: $21 \times 7 = 147$ [...] Adding both cases: $147 + 147 = 294$	✓ Correct 294
SFT	Suppose the path starts with R. Then, the first move is R. [...] If it's U, that's	– No final answer
DPCA	up, right, up, right, up, right, up, right [...]	– No final answer
ULD	then up, then right, then up, then right, then up, then right [...]	– No final answer

Analysis. The DPCA and ULD response tails repeat alternating directions without returning a count. SFT continues reconsidering individual moves. CAST completes the count by adding two symmetric cases, $147 + 147 = 294$, illustrating the contrast between repeated local enumeration and answer completion.

Case study 2: Enclosing sphere and unfinished calculations

Question. Rectangular boxes have surface area 54 and volume 23. Let r be the radius of the smallest sphere that can contain each box. If $r^2 = p/q$ in lowest terms, find $p + q$.
Reference answer: 721. *2024 AIME I, Problem 15*

Method	Response excerpts	Result
CAST	We have $r^2 = \frac{657}{64}$ [...] $p + q = 657 + 64 = \boxed{721}$	✓ Correct 721
SFT	Thus numerator is $16{,}583 + 4{,}600\sqrt{6}$, Therefore	– No final answer
DPCA	Therefore, $f(x) = 2x^2 + \frac{529}{27x^2 - 46x}$. [...] But	– No final answer
ULD	Multiply all terms by 4: $16 - 31\lambda - 132.25 + 46\lambda = 0$ [...]	– No final answer

Analysis. The baseline tails remain in intermediate algebra: SFT expands a radical expression, DPCA rewrites a function of x, and ULD manipulates an equation in λ. CAST reaches $r^2 = 657/64$ and returns $p + q = 721$. This case contrasts ongoing intermediate calculations with completion of the requested answer.

Case study 3: Lottery probability and shared success

Question. Jen picks four distinct numbers from $\{1, \ldots, 10\}$, and four numbers are drawn uniformly. Conditional on matching at least two, the probability of matching all four is m/n in lowest terms. Find $m + n$.
Reference answer: 116. *2024 AIME I, Problem 4*

Method	Response excerpts	Result
CAST	The conditional probability is $\frac{1}{115}$ [...] $m + n = 1 + 115 = \boxed{116}$	✓ Correct 116
SFT	So that's correct. Alternatively, maybe the problem is that the lottery is selecting	– No final answer
DPCA	But the problem is about the probability that her numbers are a subset of the lottery	– No final answer
ULD	The fraction $\frac{1}{115}$ is already in its simplest form [...] $m + n = 1 + 115 = 116$	✓ Correct 116

Analysis. ULD and CAST both finish with $1/115$ and the correct sum 116. The SFT and DPCA tails return to interpreting the lottery setup without an extracted answer. This case shows a success shared with ULD, alongside the unfinished responses of SFT and DPCA.

Case study 4: Logarithmic system and repeated reconsideration

Question. For real $x, y > 1$ satisfying $\log_x(y^x) = \log_y(x^{4y}) = 10$, find xy.
Reference answer: 25. *2024 AIME I, Problem 2*

Method	Response excerpts	Result
CAST	since the problem only asks for the value of xy [...] $\boxed{25}$	✓ Correct 25
SFT	Let me try t = 0. Let me see, but that's not allowed. Hmm, maybe t = 0. Let me try t = 0. [...]	– No final answer
DPCA	Try $k = 0.5$: [...] $\ln(10) \approx 2.3026$; $\ln(0.5) \approx -0.6931$ [...]	– No final answer
ULD	the product $xy = 25$ is derived directly from the relationships between x and y. [...] $\boxed{25}$	✓ Correct 25

Analysis. SFT repeatedly proposes $t = 0$ while acknowledging that it is not allowed, illustrating a loop of self-correction that does not resolve the problem. DPCA continues numerical substitutions without returning xy. ULD and CAST both conclude with 25, completing the requested calculation.

E Performance Across Training Seeds

Table 10 provides the per-seed checkpoints and scores underlying the main-text summary in Table 5. The four Qwen3-8B → Mistral-7B configurations are DPCA, uniform credit, uncertainty-only credit, and full CAST (Table 4). All use the same SFT initialization and training data, with learning rate 10^{-6}. CAST and both credit controls use n=2 and selective anchoring ($\lambda_{\mathrm{KL}} = 0.2$); original DPCA uses n=1 without anchoring. For each seed, we select the evaluated checkpoint with the highest Avg-6, using the unrounded mean of the six displayed scores. This yields three checkpoints per configuration, one per seed. Table 5 reports their mean and sample standard deviation; Tables 1 and 4 report the checkpoint with the highest Avg-6 among these three. These performance summaries exclude unevaluated early-terminated runs; Table 12 records training outcomes for all six listed seeds.

Table 10: Performance across three evaluated training seeds per configuration for Qwen3-8B → Mistral-7B at learning rate 10^{-6}. Per-seed pass@1 (%); Step identifies each seed's highest-Avg-6 evaluated checkpoint. Aggregate means and sample standard deviations appear in Table 5.

Method	Seed	Step	A24	A25	A26	M500	GPQA	LCB	Avg.
DPCA	44	150	38.3	37.1	44.4	73.7	35.7	16.0	40.87
	653512071	175	41.8	35.8	47.7	73.8	35.4	13.9	41.40
	893671858	175	41.0	34.6	45.0	73.8	34.1	9.8	39.72
Uniform credit	44	150	45.0	35.2	47.1	78.0	36.5	26.5	44.72
	653512071	175	42.3	37.2	46.7	78.9	38.5	26.8	45.07
	893671858	200	45.8	35.9	44.8	76.6	36.5	28.3	44.65
Uncertainty only	44	200	48.3	41.6	51.2	75.3	40.2	17.0	45.60
	653512071	200	47.9	39.6	49.2	73.4	38.8	17.9	44.47
	893671858	175	50.7	40.9	48.8	75.4	39.0	18.5	45.55
CAST	44	200	45.7	40.9	51.0	78.5	39.5	27.7	47.22
	653512071	175	46.2	39.5	50.6	78.1	40.2	28.1	47.12
	893671858	150	44.8	39.3	48.8	80.4	38.8	28.0	46.68

CAST's selected Avg-6 ranges from 46.68 to 47.22, above the highest selected control score (45.60). Its mean Avg-6 exceeds uniform credit, uncertainty-only credit, and DPCA by 2.19, 1.80, and 6.34 points, respectively, with differences computed from unrounded seed means. The per-benchmark summaries also retain the code-performance trade-off: CAST reaches 27.9 ± 0.2 on LCB, compared with 17.8 ± 0.8 for uncertainty-only credit, while the latter has the higher AIME-2024 mean.

Llama CAST and ULD. Table 11 gives the recorded checkpoints for seeds 44, 653512071, and 893671858 with Qwen3-8B as teacher. Table 6 reports the mean and sample standard deviation across these three runs, computing Avg-6 statistics from each seed's unrounded mean of the six displayed benchmark scores. The highest Avg-6 is 43.87 for CAST (seed 653512071, step 250) and 42.55 for ULD (the same seed, step 225); these checkpoints supply the corresponding complete rows in Table 1.

Table 11: Per-seed Llama-3.1-8B pass@1 (%) with Qwen3-8B as teacher. Step identifies the reported checkpoint; summary statistics appear in Table 6.

Method	Seed	Step	A24	A25	A26	M500	GPQA	LCB	Avg.
ULD	44	200	41.6	34.5	41.9	75.1	33.1	25.6	41.97
	653512071	225	41.6	36.2	41.9	75.1	34.7	25.8	42.55
	893671858	175	41.0	34.1	40.2	73.7	35.7	25.6	41.72
CAST	44	250	42.7	37.2	43.8	74.1	36.9	26.6	43.55
	653512071	250	44.1	37.7	45.0	75.0	34.5	26.9	43.87
	893671858	200	43.1	39.0	41.7	75.2	36.7	26.7	43.73

Table 12: Training collapse across six seeds (Qwen3-8B → Mistral-7B, lr 10^{-6}). ✓: completed 250 steps with healthy scores; ×: repetition collapse (RC) or entropy collapse (EC). Rates count collapsed runs out of six.

Method	Seed						Collapse rate
	44	45	46	653512071	893671858	417698746	
DPCA (vanilla)	✓	✓	✓	✓	✓	✓	0/6 (0.0%)
Uniform credit	✓	× (RC)	× (EC)	✓	✓	× (EC)	3/6 (50.0%)
Uncertainty only	✓	× (RC)	× (EC)	✓	✓	✓	2/6 (33.3%)
CAST	✓	✓	× (RC)	✓	✓	✓	1/6 (16.7%)

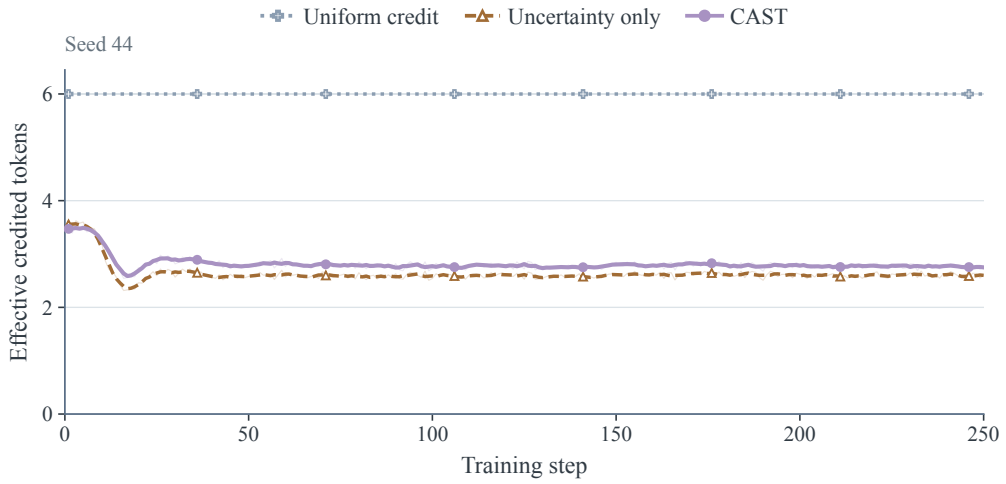

Figure 8: **CAST has slightly broader effective credit support than uncertainty-only weighting in seed-44 diagnostic runs.** Using the seed-44 diagnostic trajectories from Figure 5, we compare uniform credit, uncertainty-only weighting, and CAST. For each chunk, the effective token count is the inverse participation ratio $N_{\text{eff}}(g) = 1/\sum_{i\in g} w_i^2$, with normalized weights $w_i = \widehat{A}_i/A_g$. It equals K_g under uniform allocation and approaches 1 as credit concentrates on one token, rather than counting tokens above a threshold. The monitor averages this quantity over chunks with $K_g > 1$ and $|A_g| > 10^{-8}$; faint traces show the exported training-step values, and the overlaid curves show five-step moving averages.

Credit concentration and training stability. Over steps 1–250 of the seed-44 diagnostic runs, CAST averages 2.814 effective tokens versus 2.632 for uncertainty-only weighting, a 6.9% increase, while uniform credit averages 5.999 (Figure 8). Alongside the lower observed collapse count across six seeds (1/6 versus 2/6; Table 12), this is consistent with smoothness moderating uncertainty-driven credit concentration. The bound in Eq. (7) controls the relative reweighting introduced by smoothness; it does not guarantee diffuse product weights. Uniform credit remains more diffuse yet collapses in 3/6 runs, indicating that credit dispersion alone does not explain training stability.