LEARN FROM THE WHOLE CHAIN: ANSWER-ANCHORED REWEIGHTING FOR ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) is an important method for transferring reasoning capabilities from large language models to smaller models. Prior studies report that student-generated prefixes can drift from the teacher's typical trajectories, weakening supervision at later positions and motivating methods that shorten training rollouts. However, we argue that token position is only a coarse proxy for supervision utility. To account for each token's role in forming the student's answer, we propose Answer-Anchored Reweighting (AAR). AAR uses the generated answer as an anchor for trace-based attribution over the full reasoning trajectory. Based on the resulting within-response rankings, it upweights selected backbone tokens while retaining baseline teacher supervision at other valid positions. These weights are recomputed for each online batch, and the weighted loss is normalized within each response, with teacher distributions remaining the learning targets. Experiments on AIME2024, AIME2025, and AMC2023 show that AAR achieves higher macro-averaged Avg@32 than standard OPD across all four evaluated response lengths. At the longest evaluated response length (16k), AAR reaches 52.29%, exceeding the corresponding OPD baseline by 4.20 percentage points and the best evaluated OPD configuration by 1.60 percentage points. These results support the value of later supervision and show that answer-anchored reweighting can improve long-chain distillation without relying solely on shorter training trajectories.

1 INTRODUCTION

In on-policy distillation (OPD), a student generates trajectories and a frozen teacher provides token-level distributional supervision on the same prefixes (Agarwal et al., 2024; Song & Zheng, 2026). OPD provides denser learning signals than reinforcement learning based solely on outcome-level rewards. It also mitigates the state distribution mismatch between training and inference associated with sequence-level distillation on teacher-generated trajectories (Kim & Rush, 2016; Gu et al., 2024; Hübotter et al., 2026).

However, **student-generated prefixes may drift from the teacher's generation distribution** as reasoning progresses, weakening supervision at later positions and causing OPD gains to plateau or decline on long trajectories (Liu et al., 2026; Yang et al., 2026). Teacher-continuation experiments provide complementary evidence for this trend, showing that the teacher's accuracy advantage diminishes as student-generated prefixes grow longer (Li et al., 2026). Prior work often evaluates methods for improving OPD using relatively short responses or explicitly shortens rollouts to mitigate this issue, leaving open **how supervision should be allocated within a complete trajectory** (Ziheng et al., 2026). We argue that **token position is only a coarse proxy for supervision utility**, as it does not directly capture a token's role in forming the student's current answer. Some later positions may remain valuable for learning even when the average benefit of later supervision declines. We therefore look beyond response length alone and use each trajectory's generated answer as an anchor to reweight supervision throughout the reasoning process, including its later stages.

To this end, we propose Answer-Anchored Reweighting (AAR). Using **the student's generated answer span as an anchor**, AAR traces representation contributions from generated positions and

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.

ranks these positions by attribution score within each response. Selected backbone tokens receive higher relative weights, while other valid positions retain baseline supervision. The training objective is normalized by the sum of the weights. Attribution scores and weights are updated with each online batch to adapt the learning emphasis to the student's current trajectories. The student's answer serves only as an attribution reference, not as a ground-truth label; teacher distributions remain the learning targets. AAR thus combines **answer-anchored selective emphasis with dense teacher supervision** while preserving supervision at the remaining positions.

We evaluate AAR on AIME2024, AIME2025, and AMC2023 and examine key design choices through ablations and analysis. For the main comparison, we use the best-performing AAR hyperparameter configuration at each response length. AAR improves average accuracy across multiple response lengths, and its 16k configuration outperforms standard OPD at shorter response lengths. These results support improved supervision allocation as an alternative to shortening trajectories for better distillation.

Our main contributions are as follows:

- **Answer-Anchored Reweighting** We propose AAR, which converts attribution scores anchored to student answers into token-level distillation weights normalized within each response. It retains dense teacher supervision while adapting selective emphasis to the student's evolving trajectories.
- **Value of later supervision in long trajectories** We show that AAR retains later supervision while achieving higher average accuracy than standard OPD at shorter response lengths. This provides empirical support for the learning value of later supervision and establishes an alternative to shortening responses for improving long-chain distillation.
- **Accuracy gains in mathematical reasoning** AAR achieves higher macro-averaged Avg@32 than standard OPD across all four evaluated response lengths. At 16k, AAR reaches 52.29%, exceeding standard OPD at the same response length by 4.20 percentage points and the best OPD configuration in the main results table by 1.60 percentage points.

2 The Declining Utility of Teacher Supervision on Long Trajectories

In OPD, the current student $p_{\bar{\theta}}$ generates a response $y = (y_1, \ldots, y_T)$ to a problem $x \sim \mathcal{D}$. At each prefix $h_t = (x, y_{<t})$, the student distribution $p_{\theta,t}$ is matched to the fixed teacher distribution $q_{\phi,t}$ using a discrepancy measure D (Agarwal et al., 2024; Gu et al., 2024):

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim p_{\bar{\theta}}(\cdot|x)} \left[\frac{1}{T} \sum_{t=1}^{T} D(p_{\theta,t}, q_{\phi,t}) \right]. \tag{1}$$

OPD trajectories are on-policy for the student, but they may be atypical under the teacher's generation distribution. At the beginning of reasoning, both models primarily condition on the same problem. As generation proceeds, the teacher must also condition on intermediate assumptions, calculation results, and reasoning branches chosen by the student. These states may differ from those the teacher visits when solving the problem independently. Thus, even a teacher that can solve a problem correctly from the original prompt may provide less effective guidance on arbitrary student prefixes. Longer responses require the teacher to account for more of these accumulated student decisions (Li et al., 2026; Xing et al., 2026).

For a fixed problem and fixed student and teacher snapshots, let $P_S^{<t}$ and $P_T^{<t}$ denote their respective distributions over the first $t-1$ generated tokens. The chain rule for KL divergence gives:

$$D_{\mathrm{KL}}\big(P_S^{<t} \| P_T^{<t}\big) = \sum_{i=1}^{t-1} \mathbb{E}_{y_{<i}\sim P_S^{<i}} \left[D_{\mathrm{KL}}\big(p_{\bar{\theta},i} \| q_{\phi,i}\big) \right]. \tag{2}$$

Each term on the right measures the discrepancy between student and teacher next-token predictions on prefixes actually visited by the student. All terms are nonnegative, so even modest discrepancies

at individual steps can accumulate as generation proceeds, producing larger differences between the distributions of longer student and teacher prefixes. This identity characterizes the accumulation of distributional differences rather than proving that supervision quality declines with length. More broadly, strong overall teacher capability alone does not guarantee effective guidance throughout a long student trajectory.

Extending a trajectory therefore changes both the amount of supervision and the context in which the teacher provides it. Additional tokens can cover more of the reasoning process, but they also require the teacher to condition on a longer student-generated history. This history may include repeated trial and error, redundant derivations, or intermediate decisions that depart from an effective solution path. Even when all such positions receive teacher feedback, the resulting signals may differ in learning value (Li et al., 2025; Luo et al., 2026). Assessing the training value of a long response thus requires considering both the number of supervised tokens and the reasoning states in which their supervision is provided.

Our response-length experiments show that gains are nonmonotonic. Intuitively, the short 3k setting provides fewer supervised tokens and limited coverage of multistep reasoning; this setting also yields the lowest performance. Increasing response length includes more intermediate derivations and subsequent solution steps, giving the student an opportunity to learn from more extensive teacher guidance. However, in the comparison between 12k and 16k, OPD performs worse at the longer response length. These results show that very short responses can restrict the available supervision, while simply increasing the number of supervised tokens need not improve training.

Overall, the quantity of supervision in a long trajectory should be distinguished from its utility. Longer responses provide more training positions, but the learning value of these positions also depends on their contexts and roles in the current reasoning process. Expanding the supervision range alone may therefore yield limited additional benefit. Token position can capture broad trends, while determining the learning value of individual tokens requires considering these contextual and functional differences.

3 METHODOLOGY

AAR traces representation contributions to the answer along trajectories generated by the current student and selects backbone tokens based on within-response attribution rankings. We adjust their distillation weights relative to the remaining positions while retaining supervision at all valid generated positions. The teacher provides target distributions on the same student prefixes.

3.1 TOKEN SCORING VIA ANSWER-ANCHORED TRACING

For problem x and student response y, fix $z = (x, y)$ and student snapshot parameters $\bar{\theta}$. Let $\mathcal{A}$ denote the boxed-answer window near the response end and $\mathcal{G}$ generated positions excluding the final EOS token. We use span-based recursive attribution (Pan et al., 2026b) with target positions $\mathcal{S}$ and aggregation weights β:

$$s(\mathcal{S}, \beta) = \mathrm{Attr}_{\bar{\theta}}(z; \mathcal{S}, \beta). \tag{3}$$

We aggregate target representations and per-head source contributions with β. Proximity measures the target's L_1-norm decrease after subtracting each contribution, clipped at zero (Ferrando & Voita, 2024). Scores are summed per source across heads, normalized within each layer by total source and residual scores, then summed across layers before applying the stop-token mask (Appendix A).

Hop0 assigns equal weights $\beta^{(0)}$ to non-stop-token positions in $\mathcal{A}$. Subsequent hops target $\mathcal{G}$ using normalized preceding scores:

$$
\begin{aligned}
s^{(0)} &= \mathrm{Attr}_{\bar{\theta}}(z; \mathcal{A}, \beta^{(0)}), \\
\beta_j^{(h+1)} &= \frac{s_j^{(h)}}{\sum_{k \in \mathcal{G}} s_k^{(h)} + \varepsilon}, \qquad j \in \mathcal{G}, \\
s^{(h+1)} &= \mathrm{Attr}_{\bar{\theta}}(z; \mathcal{G}, \beta^{(h+1)}).
\end{aligned} \tag{4}
$$

Here, ε ensures numerical stability. After H recursive passes, generated-position scores in $s^{(H)}$ determine backbone selection.

On the selected trajectory (Figure 1), masking high-attribution tokens reduces the likelihood of a fixed answer suffix faster than masking equally many random or low-attribution tokens.

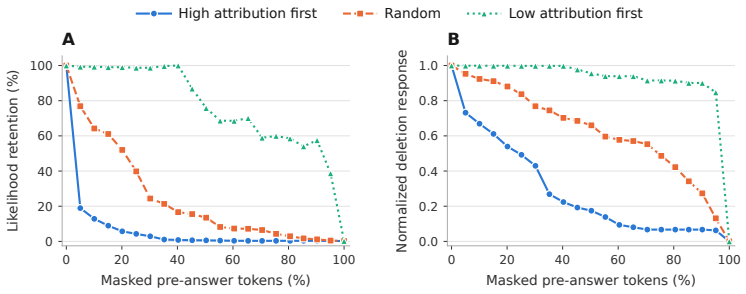

Figure 1: Hop1 token-masking comparison on a selected trajectory. **(A)** Geometric-mean target-token probability relative to the unmasked baseline (100%). **(B)** Target log-likelihood normalized to 1 before masking and 0 at full masking, clipped to $[0, 1]$, and made non-increasing by a cumulative minimum.

3.2 Backbone-Based Distillation Reweighting

For response $y_j = (y_{j1}, \ldots, y_{jT_j})$ to x_j, T_j excludes padding but includes the final EOS in selection and loss computation. Let $s_{jt}^{(H)}$ be the aligned attribution score at position t. With selection ratio $\rho \in (0, 1]$, relative multiplier $\alpha > 0$, and $n_j = \lceil \rho T_j \rceil$, we define the backbone set and weights as:

$$
\begin{aligned}
\mathcal{I}_j &= \mathrm{TopIdx}_{n_j}\left(s_{j1}^{(H)}, \ldots, s_{jT_j}^{(H)}\right), \\
w_{jt} &= 1 + (\alpha - 1)\mathbf{1}[t \in \mathcal{I}_j], \qquad t = 1, \ldots, T_j.
\end{aligned} \tag{5}
$$

Here, TopIdx_{n_j} returns the n_j highest-scoring position indices, with ties resolved by position (Appendix A); $\mathbf{1}[\cdot]$ is the indicator. Selected and unselected positions receive positive weights α and 1, respectively; answer positions follow the same ranking rule. Figure 2 illustrates attribution and selection.

At prefix $h_{jt} = (x_j, y_{j,<t})$, we renormalize the student and teacher distributions $p_{\theta,jt}$ and q_{jt} over the teacher's top-K vocabulary set $\mathcal{S}_{jt}^K$ (Fu et al., 2026):

$$
\begin{aligned}
p_{\theta,jt}^K(v) &= \frac{p_{\theta,jt}(v)}{\sum_{u \in \mathcal{S}_{jt}^K} p_{\theta,jt}(u)}, \\
q_{jt}^K(v) &= \frac{q_{jt}(v)}{\sum_{u \in \mathcal{S}_{jt}^K} q_{jt}(u)}, \qquad v \in \mathcal{S}_{jt}^K.
\end{aligned} \tag{6}
$$

For B responses, we weight each distribution-matching term for predicting y_{jt} by w_{jt}:



Figure 2: Answer-anchored attribution and backbone selection on a reasoning excerpt. Left: darker blue indicates higher attribution scores. Right: blue and gray mark selected and unselected positions, weighted by α and 1, respectively. Color intensity does not encode weights.

$$\mathcal{L}_{\mathrm{AAR}}(\theta) = \frac{1}{B}\sum_{j=1}^{B} \frac{\sum_{t=1}^{T_j} w_{jt} D\left(p^K_{\theta,jt}, q^K_{jt}\right)}{\sum_{t=1}^{T_j} w_{jt}}. \tag{7}$$

Here, D is instantiated as the student-to-teacher reverse Kullback–Leibler divergence (RKL) in the main experiments:

$$D\left(p^K_{\theta,jt}, q^K_{jt}\right) = D_{\mathrm{KL}}\left(p^K_{\theta,jt} \| q^K_{jt}\right) = \sum_{v \in \mathcal{S}^K_{jt}} p^K_{\theta,jt}(v) \log \frac{p^K_{\theta,jt}(v)}{q^K_{jt}(v)}.$$

The loss is weight-normalized within responses, then averaged equally. The multiplier α remains the selected-to-unselected coefficient ratio; $\alpha = 1$ recovers uniform OPD with other settings fixed.

Each iteration generates fresh student responses, computes attribution weights, and queries the teacher on the same prefixes. Responses, teacher targets, attribution scores, and weights are fixed during updates, with gradients only through $p^K_{\theta,jt}$. Updated student parameters are synchronized before the next rollout. Truncated responses or those with unlocatable answers or attribution failures receive unit weights and remain in training with unchanged D and K.

4 Experiments

4.1 Experimental Setup

Models and Baseline We use DeepSeek-R1-Distill-Qwen-1.5B as the student and Skywork-OR1-Math-7B as the teacher, performing distillation on student-generated reasoning trajectories (Guo et al., 2025; He et al., 2025). Our main baseline is standard OPD, which assigns uniform weights to valid tokens within the specified supervision range. AAR reweights these positions using the answer-anchored attribution scores defined in Section 3.

Training Protocol We train on DAPO-Math-17K using 8 H-100(80G) GPUs (Yu et al., 2025). Each training step processes 16 problems, with 1 student response sampled per problem. The learning rate is 2×10^{-6}, and the maximum prompt length is 2,048 tokens. Both standard OPD and AAR use RKL with a teacher support size of $K = 20$ in the main experiments. Throughout this paper, the response-length labels for AAR indicate the maximum prefix length covered by the distillation loss; all AAR configurations use a fixed rollout generation cap of 16k tokens.

Benchmarks and Metrics We evaluate on AIME2024, AIME2025, and AMC2023 with 32 samples per problem (Maxwell-Jia, 2024; Zhang & Math-AI Team, 2025; zwhe99, 2024), using the settings in Appendix Table 5. Mean denotes unweighted averages across benchmarks. For each training run, we select the checkpoint at peak performance during training and report Avg@32 and Pass@32 from the same checkpoint.

4.2 Main Results

Table 1 summarizes the mathematical reasoning performance of standard OPD and AAR under the RKL objective at different response lengths.

Table 1: Performance of OPD and AAR across response lengths.

Response Length	Method	AIME2024	AIME2025	AMC2023	Mean
3k	OPD	33.96 / 70.00	27.19 / 50.00	71.02 / 95.00	44.05 / 71.67
	AAR	**37.29 / 76.67**	**27.71** / 50.00	**72.03 / 97.50**	**45.68 / 74.72**
7k	OPD	42.50 / 73.33	29.58 / 43.33	80.00 / 95.00	50.69 / 70.56
	AAR	42.50 / **76.67**	**33.75 / 53.33**	**82.19** / 95.00	**52.81*** **/ 75.00**
12k	OPD	37.50 / 70.00	32.50 / 46.67	76.88 / 95.00	48.96 / 70.56
	AAR	**42.92 / 73.33**	**33.33 / 50.00**	**77.81** / 95.00	**51.35 / 72.78**
16k	OPD	33.75 / 70.00	32.08 / 46.67	78.44 / 92.50	48.09 / 69.72
	AAR	**42.92 / 73.33**	**33.33 / 50.00**	**80.63 / 95.00**	**52.29 / 72.78**

Note. Scores are Avg@32 / Pass@32 (%). Bold marks AAR scores strictly above OPD at the same response length. $\star$ marks the highest Mean Avg@32 in the table.

AAR Improves Accuracy across Response Lengths At 3k, 7k, 12k, and 16k, AAR achieves macro-averaged Avg@32 scores of 45.68%, 52.81%, 51.35%, and 52.29%, respectively. These exceed the corresponding OPD baselines by 1.62, 2.12, 2.40, and 4.20 percentage points. Notably, **the 16k result also surpasses the best OPD configuration in Table 1**, which achieves 50.69% at 7k, by 1.60 percentage points. AAR thus improves average accuracy across the evaluated lengths, with its longest setting outperforming every OPD configuration in the table on this metric.

AAR and OPD exhibit different performance rankings at longer response lengths Standard OPD's macro-averaged Avg@32 decreases from 48.96% at 12k to 48.09% at 16k, a drop of 0.87 percentage points. In contrast, AAR's scores at these response lengths increase from 51.35% to 52.29%, a gain of 0.94 percentage points. AAR therefore achieves better reported performance at the longer response length of 16k than at 12k. This observation provides empirical support for the learning value of later supervision under appropriate supervision allocation.

AAR at 7k achieves the highest macro-averaged Avg@32 in the table, 52.81%, exceeding its 16k result by 0.52 percentage points. These results support the value of later supervision while showing that performance need not improve monotonically with response length. Overall, the disadvantage of longer responses under uniform OPD does not consistently carry over to AAR. Answer-anchored reweighting offers a viable way to learn from longer responses and provides empirical grounds for reassessing the training value of later supervision.

5 Ablation Studies

We examine the effects of recursive depth, backbone selection ratio, and weight multiplier on AAR's average accuracy at the longest evaluated response length (16k). Our common reference configuration uses $H = 1$, $\rho = 40\%$, and $\alpha = 1.5$. In each comparison, we vary one core parameter at a time while holding the remaining training and evaluation settings fixed.

Recursive Depth To assess how recursive attribution affects supervision allocation, we fix $\rho = 40\%$ and $\alpha = 1.5$ and compare $H \in \{0, 1, 2\}$ (Figure 3). Full Avg@32 / Pass@32 results are

reported in Appendix Table 7. At $H = 0$, attribution directly targets the answer window, while $H = 1$ and $H = 2$ add one and two recursive passes, respectively. Among the three depths, $H = 1$ achieves the highest Avg@32 on each of AIME2024, AIME2025, and AMC2023. The macro-averaged Avg@32 scores for $H = 0, 1, 2$ are 50.48%, 52.29%, and 50.30%, respectively. A second recursive pass yields no further improvement.

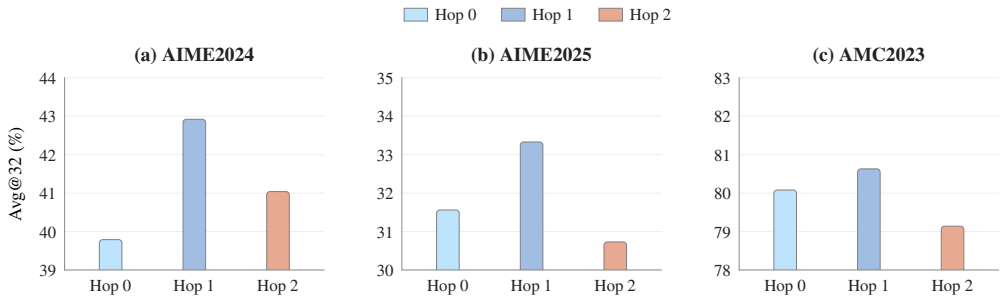

Figure 3: Effect of recursive attribution depth at a response length of 16k. Complete Avg@32/Pass@32 results are given in Appendix Table 7.

Selection Ratio The selection ratio ρ controls the fraction of positions receiving additional emphasis, while all valid response positions retain baseline supervision. We fix $H = 1$ and $\alpha = 1.5$ and compare selection ratios of 20%, 40%, and 60% (Table 2). At $\rho = 40\%$, macro-averaged Avg@32 reaches 52.29%, exceeding the 20% and 60% configurations by 1.83 and 1.92 percentage points, respectively. Under the current attribution depth and weight multiplier, this intermediate selection ratio yields higher average accuracy than either smaller or larger selections.

Table 2: Effect of backbone selection ratio at a response length of 16k.

Selection ratio ρ	AIME2024		AIME2025		AMC2023		Mean	
	Avg	Pass	Avg	Pass	Avg	Pass	Avg	Pass
20%	40.73	80.00	31.67	63.33	78.98	95.00	50.46	79.44
40%†	**42.92**	73.33	**33.33**	50.00	80.63	**95.00**	**52.29**	72.78
60%	39.69	76.67	30.73	53.33	80.70	95.00	50.37	75.00

Note. † marks the reference configuration; bold marks its best or tied-best scores in each column.

Weight Multiplier Once attribution rankings determine the backbone positions, the weight multiplier α controls their emphasis relative to the remaining positions. We fix $\rho = 40\%$ and $H = 1$ and compare $\alpha \in \{0.5, 1.5, 2.0, 2.5\}$ (Table 3). The highest macro-averaged Avg@32 is 52.29% at $\alpha = 1.5$. Increasing the multiplier to 2.0 and 2.5 yields 50.17% and 50.84%, respectively, both below the result at 1.5. Downweighting backbone positions with $\alpha = 0.5$ also yields a lower score of 48.47%. Thus, under the current selection ratio and attribution depth, $\alpha = 1.5$ achieves higher average accuracy than both the tested downweighting setting and stronger emphasis settings, providing empirical support for moderate backbone weighting.

Together, these ablations show that greater recursive depth, a larger backbone set, and stronger relative weighting do not necessarily improve AAR's performance. These findings are specific to the 16k setting and the evaluated experimental conditions. The next section analyzes how attribution rankings and weight normalization jointly shape effective supervision allocation.

6 More Analysis

A Unified Objective for Dense Supervision and Selective Emphasis For a fixed response, let U be the valid supervised positions, $S \subseteq U$ a nonempty backbone set, and $q = |S|/|U|$. Let ℓ_U and ℓ_S be their mean distribution-matching losses. For $\alpha \geq 1$, the normalized objective is equivalently

Table 3: Effect of relative token-weight multiplier at a response length of 16k.

Weight ratio α	AIME2024		AIME2025		AMC2023		Mean	
	Avg	Pass	Avg	Pass	Avg	Pass	Avg	Pass
0.5	37.50	76.67	29.48	56.67	78.44	95.00	48.47	76.11
1.5†	**42.92**	73.33	**33.33**	50.00	**80.63**	**95.00**	**52.29**	72.78
2.0	39.38	76.67	30.83	50.00	80.31	95.00	50.17	73.89
2.5	39.79	76.67	32.19	53.33	80.55	95.00	50.84	75.00

Note. † marks the reference configuration; bold marks its best or tied-best scores in each column.

$$\ell_{\mathrm{AAR}} = (1-\lambda)\ell_U + \lambda\ell_S, \qquad \lambda = \frac{q(\alpha-1)}{1+q(\alpha-1)}. \tag{8}$$

Figure 4 shows how q and α jointly determine the backbone set's extra coefficient mass relative to uniform supervision (left) and the mixture coefficient λ (right). At the starred setting $q = 0.4$ and $\alpha = 1.5$, the selected 40% of valid positions receive 50% of the total normalized loss coefficient mass. When $S = U$, the objective reduces to uniform supervision, so expanding the selected set does not always strengthen selective emphasis.

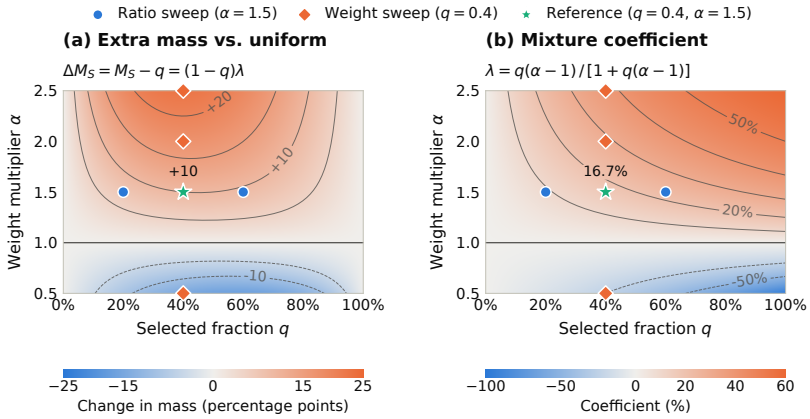

Figure 4: Analytic redistribution of normalized loss coefficients: (a) extra coefficient mass on the selected set relative to uniform supervision, with $M_S = q\alpha/[1+q(\alpha-1)]$; (b) mixture coefficient λ. Markers illustrate the table-listed settings with $q = \rho$; for $\alpha < 1$, λ is a signed coefficient rather than a convex-mixture weight.

The Role of the Supervision Set The parameters H and ρ determine the composition of the backbone set, and α controls its relative emphasis. Changes to the valid supervision set U can alter the learning signals and update directions associated with ℓ_U and ℓ_S, even when q, α, and hence the mixture coefficient λ remain fixed.

The Value of Later Supervision Depends on How It Is Used Answer-anchored attribution provides answer-relevance cues rather than direct estimates of distillation gains, while the teacher supplies the learning targets. Both the 12k and 16k AAR settings use a generation limit of 16k and perform attribution using the generated answer. Their macro-averaged Avg@32 scores are 51.35% and 52.29%, respectively. With both settings using generated-answer information for attribution, the configuration that retains direct supervision at later positions achieves the higher score. This supports the learning value of later supervision under appropriate reweighting. The difference reflects the effects of the complete training configurations rather than the isolated contribution of later tokens. AAR at 7k remains the best result in the table, showing that gains need not increase monotonically with response length. Thus, the disadvantage of longer responses under uniform OPD alone is insufficient to dismiss later supervision; its value depends on how it is used.

7 Related Work

On-Policy Distillation Knowledge distillation transfers information through teachers' soft distributions, while sequence-level methods train students on complete teacher-generated sequences (Hinton et al., 2015; Kim & Rush, 2016). To reduce training–inference mismatch, OPD provides teacher feedback on student-generated trajectories rather than relying on fixed teacher-generated trajectories (Xu et al., 2025; Peng et al., 2026). Research focuses on trajectory sampling, distribution-matching objectives, and optimization strategies (Wu et al., 2025; Jin et al., 2026). GKD explores divergences for capacity-limited students, while MiniLLM optimizes reverse KL using policy gradients, mixed sampling, and length normalization (Agarwal et al., 2024; Gu et al., 2024). Further work combines distillation with reinforcement learning fine-tuning, integrating dense teacher feedback with task-level objectives (Hübotter et al., 2026; Ding, 2026; Zhang et al., 2026b).

Supervision Utility Dense teacher feedback varies in learning value across token positions in student-generated trajectories (Jang et al., 2026; Pan et al., 2026a). Teacher capability alone does not determine supervision utility: reasoning-pattern compatibility, local distributional overlap, and prefix drift also matter, and corrective signals may weaken as reasoning progresses (Li et al., 2026; Liu et al., 2026). Proposed remedies shorten generation, restrict supervision to shorter prefixes, or construct auxiliary feedback from teacher-confidence changes induced by candidate continuations (Ziheng et al., 2026; Zhang et al., 2026a; Yang et al., 2026; Liu et al., 2026). At the token level, TIP combines student uncertainty with teacher-student disagreement to identify important positions, while TA-OPD incorporates local support compatibility to assess whether corrective signals are learnable (Xu et al., 2026; Wang et al., 2026). These approaches address supervision utility through range selection, feedback construction, and token-level learnability.

Trace-Based Attribution Trace-based attribution estimates how context tokens contribute to output representations or predictions by tracking internal information flow. Early methods aggregate cross-layer attention, while subsequent approaches incorporate representation transformations, residual connections, normalization, and relevance propagation (Abnar & Zuidema, 2020; Ferrando et al., 2022; Ferrando & Voita, 2024; Achtibat et al., 2024). For extended reasoning, span aggregation and recursive tracing expand attribution from individual predictions to multi-token targets and indirect dependencies through intermediate steps (Pan et al., 2026b). Beyond interpretation, attention-induced information flow has also been used for token-level credit assignment in reinforcement learning, connecting dependency tracing with fine-grained training feedback (Dong et al., 2026).

8 Conclusion

We propose AAR, which uses attribution anchored to student answers to reweight token-level distillation supervision. This introduces trajectory-specific learning emphasis while retaining baseline teacher supervision. At the longest evaluated response length (16k), AAR achieves a macro-averaged Avg@32 of 52.29%, exceeding OPD at the same response length by 4.20 percentage points and the best OPD configuration in the main table by 1.60 percentage points. These results provide empirical support for the learning value of later supervision under appropriate reweighting and show that improved supervision allocation offers a viable alternative to reducing response length. We discuss limitations concerning experimental scope, attribution, and computational cost in Appendix F.

AI Use Statement

Generative AI tools were used to aid and polish writing, including translation and language editing; to support retrieval and discovery, including finding and organizing related work; to assist research ideation and execution, including discussing methodological choices and reviewing implementation code; and to draft sections of the manuscript. The authors are responsible for verifying AI-assisted content and for the originality, accuracy, and final presentation of the work. AI-generated text is not treated as independent experimental evidence; the reported results are based on training and evaluation outputs.

Ethics Statement

This work studies distillation for mathematical reasoning. Distilled models can inherit errors and biases from their teachers, and gains on mathematical benchmarks do not establish reliability for open-ended tasks or high-stakes applications. Use beyond the evaluated setting should be accompanied by domain-specific validation and appropriate human oversight. The use and redistribution of models, datasets, and third-party code should respect their licenses and access conditions.

Reproducibility Statement

To support reproducibility, Section 3 defines the attribution procedure, backbone selection, and normalized distillation objective, and Appendix A provides the detailed attribution computations. Section 4.1 describes the models, training data, key training and evaluation settings, and metrics. The configurations used for the main results are listed in Appendix Table 6, while Section 5 specifies the parameter values and controlled comparisons used in the ablation studies.

References

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.385. URL https://aclanthology.org/2020.acl-main.385/.

Reduan Achtibat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. AttnLRP: Attention-aware layer-wise relevance propagation for transformers. 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. 135–168. PMLR, July 2024. URL https://proceedings.mlr.press/v235/achtibat24a.html.

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 B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5be69a584901a26c521c2b51e40a4c20-Paper-Conference.pdf.

Ken Ding. HDPO: Hybrid distillation policy optimization via privileged self-distillation. *arXiv preprint arXiv:2603.23871*, 2026. doi: 10.48550/arXiv.2603.23871. URL https://arxiv.org/abs/2603.23871.

Zhichen Dong, Yang Li, Yuhan Sun, Weixun Wang, Yijia Luo, Zinian Peng, Taiheng Ye, Chao Yang, Wenbo Su, Yu Cheng, Bo Zheng, and Junchi Yan. How does reasoning flow? tracing attention-induced information flow for targeted RL in LLMs. *arXiv preprint arXiv:2606.10646*, 2026. doi: 10.48550/arXiv.2606.10646. URL https://arxiv.org/abs/2606.10646v1. Version 1; accepted at ICML 2026.

Javier Ferrando and Elena Voita. Information flow routes: Automatically interpreting language models at scale. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17432–17445, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.965. URL https://aclanthology.org/2024.emnlp-main.965/.

Javier Ferrando, Gerard I. Gállego, and Marta R. Costa-jussà. Measuring the mixing of contextual information in the transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8698–8714, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.595. URL https://aclanthology.org/2022.emnlp-main.595/.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026. doi: 10.48550/arXiv.2603.25562. URL https://arxiv.org/abs/2603.25562.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/8ac015d409635f196f9e3e9dcfb9a94e-Paper-Conference.pdf.

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

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. Skywork Open Reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*, 2025. doi: 10.48550/arXiv.2505.22312. URL https://arxiv.org/abs/2505.22312.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. doi: 10.48550/arXiv.1503.02531. URL https://arxiv.org/abs/1503.02531.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026. doi: 10.48550/arXiv.2601.20802. URL https://arxiv.org/abs/2601.20802.

Ijun Jang, Jewon Yeom, Juan Yeo, Hyunggyu Lim, and Taesup Kim. Stable on-policy distillation through adaptive target reformulation. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 42217–42227, San Diego, California, United States, July 2026. Association for Computational

Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.2094. URL https://aclanthology.org/2026.findings-acl.2094/.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026. doi: 10.48550/arXiv.2603.07079. URL https://arxiv.org/abs/2603.07079.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, March 1951. doi: 10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan ang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026. doi: 10.48550/arXiv.2604.13016. URL https://arxiv.org/abs/2604.13016v2. Version 2.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25366–25394, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1301. URL https://aclanthology.org/2025.findings-acl.1301/.

J. Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. doi: 10.1109/18.61115. URL https://doi.org/10.1109/18.61115.

Yanjiang Liu, Jie Lou, Xinyan Guan, Yuqiu Ji, Hongyu Lin, Ben He, Xianpei Han, Le Sun, Xing Yu, and Yaojie Lu. Your teacher can't help you here: Combating supervision fidelity decay in on-policy distillation. *arXiv preprint arXiv:2605.30833*, 2026. doi: 10.48550/arXiv.2605.30833. URL https://arxiv.org/abs/2605.30833v1. Version 1.

Feng Luo, Yu-Neng Chuang, Guanchu Wang, Zicheng Xu, Xiaotian Han, Tianyi Zhang, and Vladimir Braverman. Demystifying OPD: Length inflation and stabilization strategies for large language models. *arXiv preprint arXiv:2604.08527*, 2026. doi: 10.48550/arXiv.2604.08527. URL https://arxiv.org/abs/2604.08527.

Maxwell-Jia. AIME 2024 dataset. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/Maxwell-Jia/AIME_2024. Dataset mirror of AIME 2024 I and II, 30 problems. Accessed 2026-09-25.

Leyi Pan, Shuchang Tao, Yunpeng Zhai, Lingzhe Zhang, Zhaoyang Liu, Bolin Ding, Aiwei Liu, and Lijie Wen. RLCSD: Reinforcement learning with contrastive on-policy self-distillation. *arXiv preprint arXiv:2606.11709*, 2026a. doi: 10.48550/arXiv.2606.11709. URL https://arxiv.org/abs/2606.11709.

Wenbo Pan, Zhichao Liu, Xianlong Wang, Haining Yu, and Xiaohua Jia. Towards long-horizon interpretability: Efficient and faithful multi-token attribution for reasoning LLMs. *arXiv preprint arXiv:2602.01914*, 2026b. doi: 10.48550/arXiv.2602.01914. URL https://arxiv.org/abs/2602.01914v4. Version 4; accepted as an oral paper at ICML 2026.

Jingyu Peng, Maolin Wang, Hengyi Cai, Yuchen Li, Kai Zhang, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. AdaSwitch: Balancing exploration and guidance in knowledge distillation via adaptive switching. *arXiv preprint arXiv:2510.07842*, 2026. doi: 10.48550/arXiv.2510.07842. URL https://arxiv.org/abs/2510.07842.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026. doi: 10.48550/arXiv.2604.00626. URL https://arxiv.org/abs/2604.00626.

Yuanyi Wang, Su Lu, Yanggan Gu, Pengkai Wang, Yifan Yang, Zhaoyi Yan, Congkai Xie, Jianmin Wu, and Hongxia Yang. Not all disagreement is learnable: Token teachability in on-policy distillation. *arXiv preprint arXiv:2605.26844*, 2026. doi: 10.48550/arXiv.2605.26844. URL https://arxiv.org/abs/2605.26844.

Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. Rethinking Kullback-Leibler divergence in knowledge distillation for large language models. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5737–5755, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.383/.

Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation. *arXiv preprint arXiv:2606.01249*, 2026. doi: 10.48550/arXiv.2606.01249. URL https://arxiv.org/abs/2606.01249.

Wenda Xu, Rujun Han, Zifeng Wang, Long Le, Dhruv Madeka, Lei Li, William Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 64616–64646, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a2747a3844ca1e4667fbff3f558eb39b-Paper-Conference.pdf.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. TIP: Token importance in on-policy distillation. *arXiv preprint arXiv:2604.14084*, 2026. doi: 10.48550/arXiv.2604.14084. URL https://arxiv.org/abs/2604.14084.

Zhicheng Yang, Zhijiang Guo, Yifan Song, Minrui Xu, Yongxin Wang, Yiwei Wang, Xiaodan Liang, and Jing Tang. Prune-OPD: Efficient and reliable on-policy distillation for long-horizon reasoning. *arXiv preprint arXiv:2605.07804*, 2026. doi: 10.48550/arXiv.2605.07804. URL https://arxiv.org/abs/2605.07804.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. doi: 10.48550/arXiv.2503.14476. URL https://arxiv.org/abs/2503.14476.

Dongxu Zhang, Zhichao Yang, Sepehr Janghorbani, Jun Han, Andrew Ressler, II, Qian Qian, Gregory D. Lyng, Sanjit Singh Batra, and Robert E. Tillman. Fast and effective on-policy distillation from reasoning prefixes. *arXiv preprint arXiv:2602.15260*, 2026a. doi: 10.48550/arXiv.2602.15260. URL https://arxiv.org/abs/2602.15260.

Yifan Zhang and Math-AI Team. American Invitational Mathematics Examination (AIME) 2025. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/math-ai/aime25. Dataset mirror, 30 problems. Accessed 2026-09-25.

Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. Reinforcement-aware knowledge distillation for LLM reasoning. *arXiv preprint arXiv:2602.22495*, 2026b. doi: 10.48550/arXiv.2602.22495. URL https://arxiv.org/abs/2602.22495.

Zhou Ziheng, Jiaqi Li, Huacong Tang, Ying Nian Wu, and Demetri Terzopoulos. Less is more: Early stopping rollout for on-policy distillation. *arXiv preprint arXiv:2605.27028*, 2026. doi: 10.48550/arXiv.2605.27028. URL https://arxiv.org/abs/2605.27028v1. Version 1.

zwhe99. AMC23: AMC 2023 evaluation dataset. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/zwhe99/amc23. 40-problem subset of AMC 12A and AMC 12B 2023; source problem URLs are included in the dataset. Accessed 2026-09-25.

A Computational Details of Answer-Anchored Attribution Scores

This appendix details the attribution operator in Section 3.1 and specifies the implementation conventions for the answer window, stop tokens, and backbone selection. The operator uses attribution based on span-level representation contributions and recursive propagation; AAR uses its scores at generated positions for within-response backbone selection and distillation reweighting (Pan et al., 2026b).

Answer Window and Original Token Alignment Attribution uses the token IDs actually generated by the student and maps the results back to the same response positions. Display text is used only to identify answer commands and stop-token positions; the original generated sequence is not replaced by a retokenized sequence. While preserving the original token indices, the answer locator checks for `\boxed` across BPE tokens, allows whitespace between the command and the opening brace, and selects the last valid command that is unescaped, has balanced braces, and contains nonempty content. Let s be the original token position containing the command's first letter, `b`, and let T_{trace} denote the response length after removing the terminal EOS. The implementation uses the closed interval $[s, \min(s+32, T_{\mathrm{trace}})]$ as the attribution window, containing at most 33 tokens. Checking command completeness and extracting the window are separate steps: the window is not trimmed to the actual length of the answer inside the braces.

Stop Tokens and Termination Markers For token display strings in the user's question and the response, we first strip leading and trailing whitespace. A position is treated as a stop token if the resulting string is empty, exactly a comma, or exactly a period. The remaining positions in the chat template retain their default masks. Positions whose display value is `None` retain their original indices and are not automatically classified as stop-token positions. This is a lexical filtering rule, not a semantic stopword list or a list of EOS tokens. Stop-token positions in the initial answer window do not participate in target aggregation, and their corresponding attribution scores are set to zero after aggregation across layers; neither operation directly removes tokens from the training sequence. The terminal EOS is used separately to determine whether a response is complete and is excluded from attribution targets. During training, its original position is still handled according to the candidate set and loss mask in Section 3.2.

Representation Aggregation over the Target Span Consider attention head b in layer ℓ. Let $a_{ji}^{\ell,b}$ be the attention weight assigned by target position j to source position i, $v_i^{\ell,b}$ the value vector at the source position, and $W_O^{\ell,b}$ the corresponding output projection. Let r_j^ℓ and h_j^ℓ denote the residual representations before and after the attention computation, respectively; h_j^ℓ here does not include the subsequent feedforward sublayer update. For a target position set $\mathcal{S}$ and weights β, we separately aggregate the target representations, residuals, and the contributions of each source position in each attention head:

$$\begin{aligned} u_{\mathcal{S}}^{\ell} &= \sum_{j\in\mathcal{S}} \beta_j h_j^{\ell}, \qquad r_{\mathcal{S}}^{\ell} = \sum_{j\in\mathcal{S}} \beta_j r_j^{\ell}, \\ z_{i\to\mathcal{S}}^{\ell,b} &= \sum_{j\in\mathcal{S}} \beta_j a_{ji}^{\ell,b} W_O^{\ell,b} v_i^{\ell,b}. \end{aligned} \tag{9}$$

The same target weights β are used to aggregate both the target representations and the source contributions. The computation first aggregates representations over multiple target tokens as a whole and then evaluates the relationship between source positions and the aggregated target, rather than independently attributing each target token and averaging the results. Stop-token positions in the initial answer window do not participate in target aggregation, and $\beta^{(0)}$ assigns equal weights to the remaining positions.

Proximity Scoring of Per-Head Contributions For a target representation u and a contribution vector z on a fixed trajectory, we define nonnegative proximity through the change in the L_1 norm after subtracting that contribution (Ferrando et al., 2022):

$$\mathrm{Prox}(z, u) = \max\{0,\ \|u\|_1 - \|u - z\|_1\}\,. \tag{10}$$

Here, z denotes a contribution vector rather than the token sequence $z = (x, y)$ used in the main text. The computation uses forward representations of a fixed input trajectory and does not involve deleting text tokens and rerunning inference. For each source position, we first compute a scalar proximity score separately for each attention head and then sum these scores, rather than combining contribution vectors across heads before computing proximity.

Within-Layer Normalization and Cross-Layer Aggregation Within each layer, a source position's score is normalized relative to the proximity scores for all source positions and all attention heads, together with the proximity of the aggregated residual. The scores are then summed across layers, and the stop-token mask is applied:

$$\begin{aligned} c_i^{\ell}(\mathcal{S}, \beta) &= \frac{\sum_b \mathrm{Prox}\left(z_{i\to\mathcal{S}}^{\ell,b},\, u_{\mathcal{S}}^{\ell}\right)}{\sum_{k,b} \mathrm{Prox}\left(z_{k\to\mathcal{S}}^{\ell,b},\, u_{\mathcal{S}}^{\ell}\right) + \mathrm{Prox}\left(r_{\mathcal{S}}^{\ell},\, u_{\mathcal{S}}^{\ell}\right) + \varepsilon}, \\ s_i(\mathcal{S}, \beta) &= m_i \sum_{\ell=1}^{L} c_i^{\ell}(\mathcal{S}, \beta). \end{aligned} \tag{11}$$

Here, L is the number of layers, and the numerical stabilizer in this denominator is $\varepsilon = 10^{-12}$. Source positions k cover both the input and generated sequences. The target aggregation weights and the recursive weight normalization in Equation (4) also use 10^{-12}, but these are distinct normalization steps; this value does not replace the stabilizers in the model's own RMSNorm or LayerNorm. Non-finite attribution values are set to zero before constructing recursive weights.

m_i is the stop-token mask described above, which sets the output scores at the corresponding positions to zero after aggregation across layers. This is distinct from filtering target positions in the initial answer window and is not equivalent to the distillation loss mask. Within-layer normalization makes each layer's scores reflect the contributions of source positions relative to other sources and the residual. The vector formed by the scores $s_i(\mathcal{S}, \beta)$ is the output of the attribution operator in Equation (3); AAR then reads the generated-position scores from the specified pass for distillation reweighting.

Implementation Conventions for Backbone Selection Backbone positions are ranked by attribution score in descending order, with ties resolved in favor of earlier response positions. After stop-token attribution scores are set to zero, their positions remain candidates in the ranking.

B Training and Evaluation Configurations

B.1 Training Configuration

B.2 Evaluation Configuration

Table 4: Training setup and default hyperparameters.

Setting	Value
Student	DeepSeek-R1-Distill-Qwen-1.5B
Teacher	Skywork-OR1-Math-7B
Training data	DAPO-Math-17K
GPUs	$8\times$ H100 80G
Training epochs	1
Global prompt batch size	16
Default responses per prompt	1
Microbatch size per GPU	1
Maximum prompt length	2,048 tokens
Rollout temperature / top-p	1.0 / 1.0
Learning rate	2×10^{-6}
Teacher support and loss	$K = 20$ and RKL for the main experiments

Table 5: Evaluation settings and metric definitions.

Setting	Value
Benchmarks	AIME2024, AIME2025, and AMC2023
Sampling temperature	0.6
Top-p	0.95
Maximum generation length	32,768 tokens
Samples per problem, k	32

B.3 Selected Configurations for the Main Results Table

Table 6: Summary of the configurations reported in the main results table.

Method	Loss coverage	Weight α	Hops H	Selection ratio ρ	Teacher top-K
OPD	3K	1	–	–	20
AAR	3K	1.5	1	40%	20
OPD	7K	1	–	–	20
AAR	7K	2	1	40%	20
OPD	12K	1	–	–	20
AAR	12K	1.5	1	40%	20
OPD	16K	1	–	–	20
AAR	16K	1.5	1	40%	20

Additional 7k Results We additionally evaluate AAR at 7k with $H = 1$, $\rho = 40\%$, $\alpha = 1.5$, and $K = 20$, using the same values of these four factors as the 3k, 12k, and 16k AAR configurations in Table 6. This configuration achieves Average / Pass scores of 42.92% / 66.67% on AIME2024, 32.92% / 53.33% on AIME2025, and 77.50% / 95.00% on AMC2023. The macro-averaged Average and Pass scores are 51.11% and 71.67%, respectively. Both numerically exceed the corresponding 7k OPD scores in Table 1, which are 50.69% and 70.56%, respectively.

B.4 Attribution Coverage of Training Rollouts

Figure 5 summarizes the adjusted attribution coverage across the four loss-prefix settings. The displayed values for 3K, 7K, 12K, and 16K are 83.09%, 89.87%, 91.74%, and 90.75%, respectively.

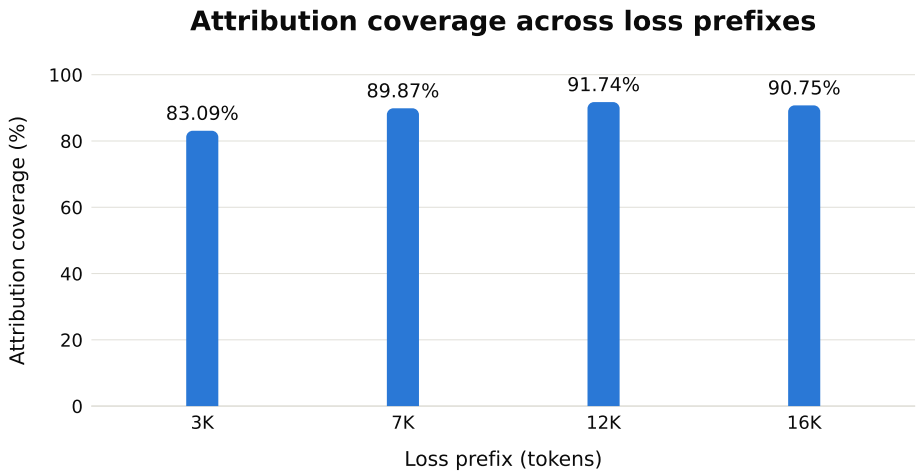

Figure 5: All plotted AAR runs use a maximum rollout length of 16K (16,384 tokens), with the horizontal axis specifying the loss prefix rather than the generation limit. Recorded attribution coverage is the fraction of training rollouts for which usable attribution weights are returned. Uncovered rollouts in the original counts include fallback caused by truncation and other factors, such as missing or malformed boxed answers.

B.5 Complete Results for Recursive Attribution Depth

Table 7: Effect of recursive attribution depth at a response length of 16k.

Depth H	AIME2024	AIME2025	AMC2023	Mean
0	39.79 / **80.00**	31.56 / **53.33**	80.08 / 95.00	50.48 / **76.11**
1	**42.92** / 73.33	**33.33** / 50.00	**80.63** / 95.00	**52.29** / 72.78
2	41.04 / 76.67	30.73 / 50.00	79.14 / **97.50**	50.30 / 74.72

B.6 Backbone-Set Turnover Across Attribution Hops

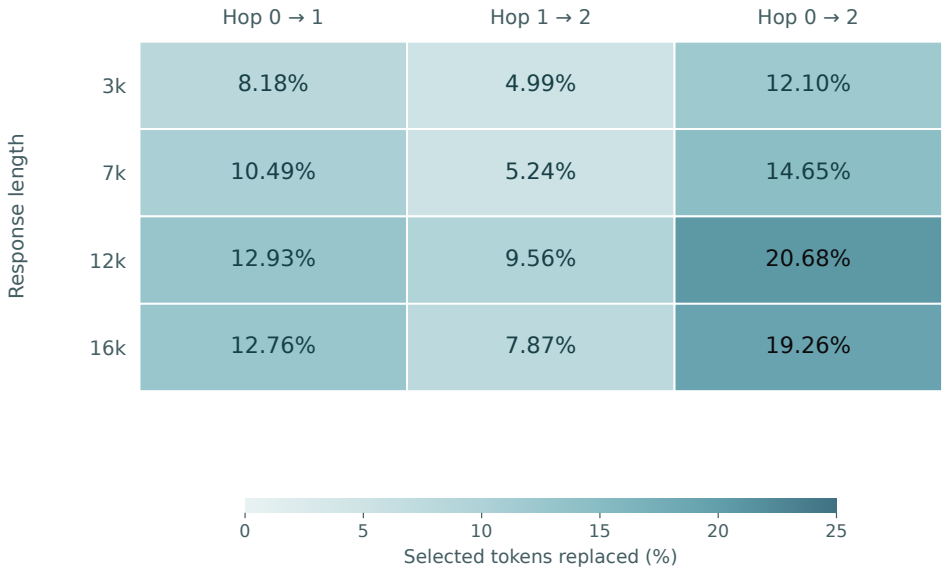

Figure 6: Backbone-set turnover across attribution hops on fixed responses replayed with the base student. Each cell shows $100|\mathcal{S}_i \setminus \mathcal{S}_j|/|\mathcal{S}_i|$ for top-40% sets.

C GPU Memory Footprint and Runtime of Attribution Computation

Figures 7 and 8 show the GPU memory footprint and runtime of attribution with one recursive pass ($H = 1$) for the DeepSeek-R1-Distill-Qwen-1.5B base model in BF16 precision on an NVIDIA H100 80G GPU. The response lengths are 3K, 7K, 12K, and 16K, and each point is the mean of measurements over 10 samples.

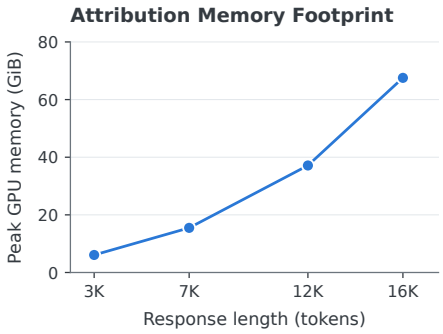

Figure 7: Attribution memory footprint across response lengths.

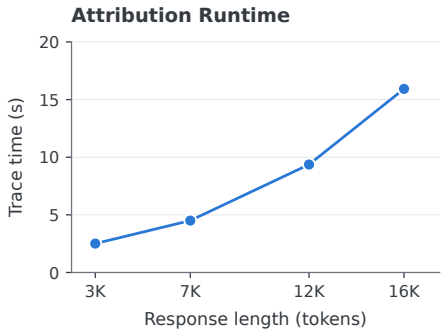

Figure 8: Attribution runtime across response lengths.

D END-TO-END TRAINING COST

Figure 9 compares the observed costs of the 16K-response-cap RKL runs over 869 logged steps on eight GPUs. Generation time is nearly unchanged, while the additional time is concentrated in the aggregate training stage. Mean total step time is 53.1 s for OPD and 84.6 s for AAR (+59.1%), giving cumulative logged-step costs of 102.6 and 163.3 GPU-hours, respectively. The nearly linear cumulative curves reflect stable per-step runtimes.

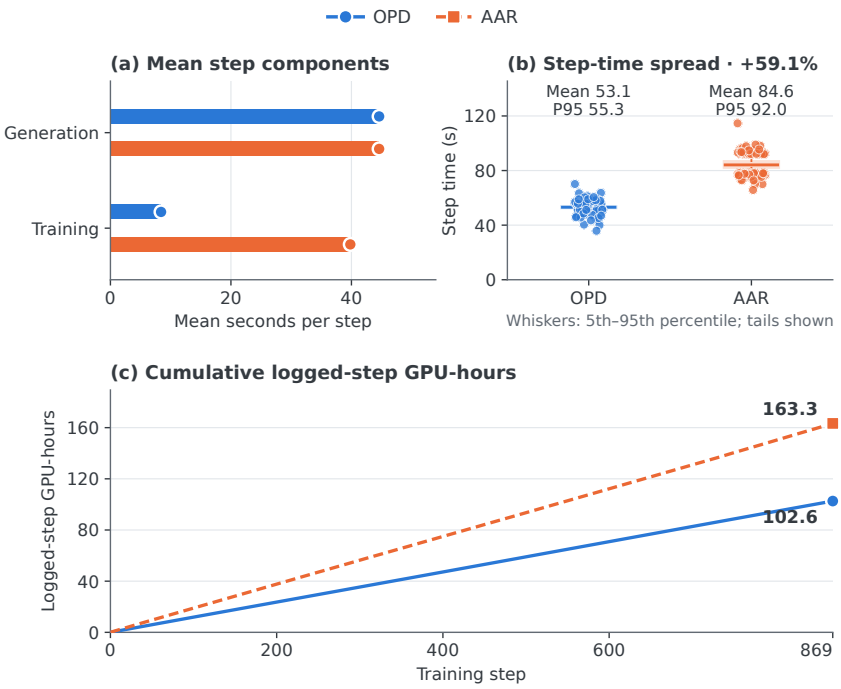

Figure 9: Training cost for OPD and AAR. (a) Mean stage times. (b) Boxes show the interquartile range and median; whiskers span the 5th–95th percentiles, with points marking steps outside that range. (c) Cumulative logged-step GPU-hours, $8\sum_{i=1}^{s} t_i/3600$, where t_i is the step duration in seconds. Totals include checkpoint saving within steps but exclude initialization and work outside the logged intervals.

E Teacher Support and Distillation Objective

We fix $H = 1$, $\rho = 40\%$, and $\alpha = 1.5$ and compare RKL with JSD at $K \in \{10, 20, 30\}$ (Table 8) (Kullback & Leibler, 1951; Lin, 1991). RKL achieves higher macro-averaged Avg@32 than JSD at all three support sizes. RKL with $K = 20$ reaches 52.29%, the highest score among the six configurations, and exceeds JSD at the same support size by 2.95 percentage points. JSD achieves its best score of 49.97% at $K = 10$. These comparisons support using RKL with $K = 20$ in this setting and indicate that the preferred support size depends on the distillation objective.

Table 8: Teacher support size and distillation objective at a response length of 16k.

Teacher top-K	RKL		JSD		RKL − JSD	
	Avg	Pass	Avg	Pass	ΔAvg	ΔPass
10	**50.25**	**78.33**	49.97	78.06	+0.29	+0.28
20†	**52.29**⋆	72.78	49.34	75.00	+2.95	-2.22
30	**49.81**	**76.11**	49.18	75.00	+0.63	+1.11

Note. † marks reference ($K = 20$, RKL). Bold marks RKL scores that exceed JSD at the same K; ⋆ marks the highest Mean Avg@32 across all six settings.

F Limitations

Experimental Scope Our experiments cover one teacher-student pair and three mathematical reasoning benchmarks. Future work can extend the evaluation to more teacher-student pairs, explore different model families, and incorporate a broader range of benchmarks to assess the method's applicability across model and task settings. We can also conduct additional independent training runs for each configuration to further evaluate the stability and reproducibility of the results.

Attribution and Computational Cost The current method relies on an identifiable answer window and successful attribution computation. Tasks without a clear final answer require a more general attribution reference. Attribution reflects answer relevance but does not directly measure the reliability of teacher supervision or the actual learning gains at individual positions. Response generation and online attribution also incur training costs, and the trade-off between performance and computation requires evaluation under matched budgets.