ENTROPY-AWARE ON-POLICY DISTILLATION OF LANGUAGE MODELS

Woogyeol Jin[1] **Taywon Min**[1] **Yongjin Yang**[2,3] **Swanand Ravindra Kadhe**[4] **Yi Zhou**[4]
Dennis Wei[4] **Nathalie Baracaldo**[4] **Kimin Lee**[1*]
[1]KAIST, [2]University of Toronto, [3]Vector Institute, [4]IBM Research

ABSTRACT

On-policy distillation is a promising approach for transferring knowledge between language models, where a student learns from dense token-level signals along its own trajectories. This framework typically uses reverse KL divergence, encouraging the student to match the teacher's high-confidence predictions. However, we show that the mode-seeking property of reverse KL reduces generation diversity and yields unstable learning signals when the teacher distribution has high entropy. To address this, we introduce Entropy-Aware On-Policy Distillation. Our key idea is augmenting the standard reverse KL objective with forward KL when teacher entropy is high, capturing the full range of plausible outputs while retaining precise imitation elsewhere. It balances mode-seeking precision with mode-covering robustness without sacrificing on-policy training efficiency. Experiments show that our method maintains generation diversity (sustained token-level entropy) and improves student–teacher alignment (lower forward KL on high-entropy tokens). Across six math reasoning benchmarks, this yields Pass@8 accuracy gains of $+1.37$ for Qwen3-0.6B-Base, $+2.39$ for Qwen3-1.7B-Base, and $+5.05$ for Qwen3-4B-Base compared to baseline on-policy distillation methods. These results demonstrate that accounting for teacher uncertainty is essential for maintaining diversity and achieving effective knowledge transfer.

1 INTRODUCTION

Knowledge distillation (Hinton et al., 2015) is a promising approach for transferring the capabilities of large language models (LLMs) to smaller, more efficient models with lower inference cost and improved deployability. Traditional distillation relies on off-policy teacher data, training students with supervised loss or forward KL divergence (Kim & Rush, 2016). However, this introduces a distribution mismatch between training sequences and those generated by the student at inference time.

On-policy distillation addresses this by having the student generate samples that are corrected by the teacher, typically via reverse KL divergence (Gu et al., 2023; Agarwal et al., 2024; Lu & Lab, 2025). As noted by (Lu & Lab, 2025), this objective can be seamlessly integrated into standard reinforcement learning (RL) pipelines. Furthermore, the approach is highly efficient: recent works (Lu & Lab, 2025; Yang et al., 2025) demonstrate that on-policy distillation can match RL-trained models on math reasoning benchmarks at $10\times$ lower compute cost than GRPO (Shao et al., 2024).

However, reverse KL is a *mode-seeking* objective: while it effectively captures the teacher's dominant modes, we find that it reduces student diversity and yields unstable learning signals at positions where the teacher distribution has high entropy. This limits the student's ability to preserve the teacher's distributional structure when probability mass is spread across multiple tokens. It is particularly problematic in reasoning tasks, where high-entropy tokens often represent key decision points with multiple valid paths (Wang et al., 2025; Cheng et al., 2025).

Contribution. To address this limitation, we propose Entropy-Aware On-Policy Distillation (EOPD), a distillation framework that balances efficiency and diversity. Our key insight is that reverse KL and forward KL are complementary: reverse KL enables efficient learning on confident

*Correspondence to: kiminlee@kaist.ac.kr

predictions, while forward KL's mode-covering property transfers uncertainty and global structure. Our specific contributions are as follows:

- **Analysis of Diversity Degradation and Training Instability.** We conduct a systematic analysis of token-level entropy (§3.1), revealing that standard on-policy distillation causes diversity collapse, retaining only 6.8% of high-entropy tokens compared to 18.5% in the teacher. Furthermore, through a controlled toy experiment (§3.2), we demonstrate that the reverse KL objective provides unstable gradient signals when the teacher is uncertain, preventing proper convergence.
- **Entropy-Aware On-Policy Distillation (EOPD).** We introduce an entropy-aware strategy that dynamically adapts the training objective. By selectively applying reverse KL in low-entropy regions for efficiency and forward KL in high-entropy regions to preserve diversity, EOPD effectively transfers the teacher's uncertainty without the computational overhead of naive forward KL.
- **Improvements on Reasoning Benchmarks.** Empirically, EOPD maintains substantially higher generation diversity, preserving the teacher's uncertainty by retaining more probability mass in high-entropy regions than standard on-policy distillation. This improvement translates into consistent downstream gains: averaged over six mathematical reasoning benchmarks, EOPD improves Avg@8 accuracy by +1.16 and Pass@8 by +1.37 for Qwen3-0.6B-Base, with larger gains of +0.99/+2.39 for the 1.7B model and +1.80/+5.05 for the 4B model.

2 PRELIMINARIES

2.1 KL-BASED DIVERGENCES

Let P and Q be probability distributions defined over the same sample space $\mathcal{X}$. The Kullback–Leibler (KL) divergence is a non-symmetric measure of difference between two distributions, quantifying how well Q approximates the reference distribution P.

$$\mathrm{KL}(P \,\|\, Q) = \mathbb{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right]. \tag{1}$$

Due to its asymmetry, $\mathrm{KL}(P \,\|\, Q)$ and $\mathrm{KL}(Q \,\|\, P)$ induce different optimization behaviors, depending on which distribution is used as the reference.

For auto-regressive sequence models like Large Language Models (LLMs), the probability distribution of a token x is conditioned on a context $\mathbf{c}$, where $\mathbf{c}$ is the sequence generated before x. In distillation, we have two models, a student model denoted by $\pi_\theta(\cdot \mid \mathbf{c})$, and a teacher model $\pi_{\mathtt{te}}(\cdot \mid \mathbf{c})$. We now define forward and reverse KL divergences in terms of these models.

Forward KL (Teacher-to-Student). The forward KL divergence is defined as an expectation over the teacher's distribution:

$$\mathrm{KL}(\pi_{\mathtt{te}}(\cdot \mid \mathbf{c}) \,\|\, \pi_\theta(\cdot \mid \mathbf{c})) = \mathbb{E}_{x\sim\pi_{\mathtt{te}}(\cdot|\mathbf{c})}\left[\log \frac{\pi_{\mathtt{te}}(x \mid \mathbf{c})}{\pi_\theta(x \mid \mathbf{c})}\right]. \tag{2}$$

Minimizing equation 2 is equivalent to standard supervised learning (maximizing likelihood on teacher samples). It penalizes the student for assigning low probability to any token the teacher considers likely. This induces *mode-covering* behavior, where the student attempts to match the entire support of the teacher, potentially leading to overly diffuse distributions if the student has limited capacity (Minka, 2005).

Reverse KL (Student-to-Teacher). The reverse KL divergence is defined as an expectation over the student's distribution:

$$\mathrm{KL}(\pi_\theta(\cdot \mid \mathbf{c}) \,\|\, \pi_{\mathtt{te}}(\cdot \mid \mathbf{c})) = \mathbb{E}_{x\sim\pi_\theta(\cdot|\mathbf{c})}\left[\log \frac{\pi_\theta(x \mid \mathbf{c})}{\pi_{\mathtt{te}}(x \mid \mathbf{c})}\right]. \tag{3}$$

Since the expectation is taken over student samples, equation 3 penalizes generated tokens that the teacher considers unlikely, but ignores teacher modes that the student does not visit. It induces *mode-seeking* behavior, encouraging the student to concentrate probability mass on a single high-likelihood mode of the teacher while ignoring others (Minka, 2005).

2.2 On-Policy Distillation

On-policy distillation (OPD) (Agarwal et al., 2024) is a post-training method in which a student model learns by matching a teacher's token-level probability distribution on its own generated sequences. By training on on-policy rollouts rather than teacher-generated trajectories, OPD enables precise credit assignment and mitigates compounding errors inherent in off-policy imitation.

Recent OPD methods optimize the reverse KL divergence (Gu et al., 2023; Lu & Lab, 2025), encouraging mode-seeking behavior that helps the student focus on the teacher's dominant modes. Specifically, given inputs $\mathbf{q} \sim \mathcal{D}$, we denote $\mathbf{x} = (x_1, x_2, \ldots, x_{|\mathbf{x}|})$ as the student-generated token sequence. With $\mathbf{c}_t = (\mathbf{q}, x_{<t})$ as the context for token t, we have $x_t \sim \pi_\theta(\cdot \mid \mathbf{c}_t)$. The on-policy reverse-KL objective is then defined as:

$$\mathbb{E}_{\mathbf{q}\sim\mathcal{D}}\left[\mathbb{E}_{\mathbf{x}\sim\pi_\theta(\cdot|\mathbf{q})}\left[\frac{1}{|\mathbf{x}|}\sum_{t=1}^{|\mathbf{x}|}\mathcal{L}_t^{\mathrm{RKL}}(\theta;\mathbf{c}_t)\right]\right], \tag{4}$$

where $\mathcal{L}_t^{\mathrm{RKL}}(\theta;\mathbf{c}_t)$ is the per-token reverse KL from equation 3:

$$\mathcal{L}_t^{\mathrm{RKL}}(\theta;\mathbf{c}_t) = \mathrm{KL}(\pi_\theta(\cdot \mid \mathbf{c}_t) \,\|\, \pi_{\mathtt{te}}(\cdot \mid \mathbf{c}_t)). \tag{5}$$

In practice, Lu & Lab (2025) use a single-sample Monte-Carlo estimate of the expectation in equation 3 and plug it into a policy-gradient-style update. This is done by defining the token-level reward as the log-probability difference between the teacher and student:

$$A_t = \log \pi_{\mathtt{te}}(x_t \mid \mathbf{c}_t) - \log \pi_\theta(x_t \mid \mathbf{c}_t). \tag{6}$$

This reward measures how preferred the student-selected token is under the teacher distribution in the same context, assigning positive values when the teacher assigns a higher probability to the token than the student, and negative values otherwise. The student then essentially optimizes the objective $\max_\theta \mathbb{E}_{\pi_\theta}\left[\sum_t A_t\right]$. The result is an effective on-policy distillation method based on policy gradient optimization.

2.3 OPD with Clipped-Reverse KL

To stabilize training, the OPD objective can be implemented with standard PPO-style importance sampling and clipping (Schulman et al., 2017). We sample trajectories using a behavior policy $\pi_{\theta_{\text{old}}}$ instantiated from the student policy π_θ, query the teacher for log-probabilities of the sampled student tokens, and define a per-token advantage $\hat{A}_t = \log \pi_{\mathtt{te}}(x_t \mid \mathbf{c}_t) - \log \pi_{\theta_{\text{old}}}(x_t \mid \mathbf{c}_t)$, substituting the behavior policy in equation 6. We then update π_θ by minimizing the clipped PPO surrogate over generated tokens. To simplify notation, we omit the expectations over $\mathbf{q}$ and $\mathbf{x}$ in equation 4 and focus on single tokens. The surrogate objective is then

$$\mathcal{L}_t^{\mathrm{OPD}}(\theta;\mathbf{c}_t) = \mathbb{E}_{x_t\sim\pi_{\theta_{\text{old}}}(\cdot|\mathbf{c}_t)}\left[\widetilde{A}_t\right], \tag{7}$$

where $\widetilde{A}_t$ is the clipped reverse KL loss:

$$\widetilde{A}_t = \max\left(-r_t\hat{A}_t,\ -\mathrm{clip}(r_t, 1-\epsilon, 1+\epsilon)\,\hat{A}_t\right). \tag{8}$$

Here, $r_t = \frac{\pi_\theta(x_t|\mathbf{c}_t)}{\pi_{\theta_{\text{old}}}(x_t|\mathbf{c}_t)}$ corrects for sampling under $\pi_{\theta_{\text{old}}}$, while clipping limits overly large updates.

3 Diversity Degradation and Instability in On-Policy Distillation

In §3.1, we first analyze token-level entropy distributions to identify diversity degradation after on-policy distillation due to reverse KL. We then show that reverse KL produces unstable learning signals when teacher entropy is high, with the student's top-k predictions failing to converge in §3.2.

3.1 Token-Level Entropy Analysis

In domains that require complex reasoning, such as mathematical and multi-step reasoning tasks, high-entropy tokens[1] in the teacher model are not merely noise but encode important knowledge

[1] We use *high-entropy token* as shorthand for a token at a position where the teacher's conditional distribution has high entropy.

in the form of multiple plausible reasoning paths and meaningful uncertainty (Wang et al., 2025; Cheng et al., 2025). However, on-policy distillation may fail to properly capture this knowledge. As discussed in §2.2, it typically minimizes the reverse KL divergence, a mode-seeking objective that favors fast convergence at the cost of reduced exploration, thereby hindering the transfer of the teacher's uncertainty to the student.

To examine this effect, we generate responses from the teacher model (Qwen3-8B (Yang et al., 2025)) and an on-policy-distilled student (Qwen3-1.7B-Base) and evaluate on AIME24 and AIME25 prompts (MAA, 2025), and analyze their token entropy distributions (see §5.1 for more details on the experimental setup). We find that the distilled student retains fewer high-entropy tokens (entropy ≥ 1.0) than the teacher, specifically only 6.8% compared to 18.5% of the teacher (see Figure 2 in §5.4 for full histogram). This suggests that reverse KL drives aggressive mode-seeking behavior rather than preserving the teacher's inherent uncertainty.

3.2 Instability of Reverse KL-based Reward

To gain insight, we first conduct a simplified toy experiment to analyze how reverse KL-based reward optimization behaves under different levels of teacher uncertainty.

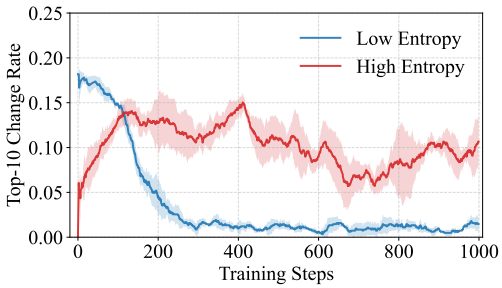


Figure 1: Top-10 change rate for Scenario A (blue), where the teacher distribution has low entropy, and Scenario B (red), where the teacher distribution has high entropy across 3 seeds. For Scenario B, a student optimized with reverse KL fails to capture the teacher's distribution, as evidenced by frequent Top-1 token changes and highly persistently high and fluctuating Top-10 change rates.

Toy setup. We study how a student learns from a teacher via reverse KL policy gradients, in a setting that retains the essential characteristics of having a teacher distribution with multiple modes and a student with limited coverage. To simplify matters, we remove the autoregressive conditioning on context $\mathbf{c}_t$. This reduces the distributions to categorical distributions over V indices (here $V = 80$) and no longer requires complex LLMs.

The teacher distribution $P_{\texttt{te}}$ is constructed as follows. We sample logits $\mathbf{z} \in \mathbb{R}^V$ i.i.d. from $\mathcal{N}(0, 1)$, then overwrite five randomly chosen entries with larger values $\{1.7, 1.9, 2.1, 2.3, 2.5\}$ to create five modes. We then apply temperature scaling:

$$P_{\texttt{te}}(x) = \mathrm{softmax}(\mathbf{z}/T)_x.$$

We consider two scenarios: (A) $T = 0.3$, yielding a low-entropy (peaked) teacher, and (B) $T = 1.0$, yielding a high-entropy (diverse) teacher. The teacher is fixed throughout training (see Appendix B for a detailed summary and visualization of the distribution).

The student distribution P_S is parameterized by learnable logits $\mathbf{s} \in \mathbb{R}^V$ initialized i.i.d. from $\mathcal{N}(0, 1)$. To mimic limited model capacity, we restrict sampling to the student's top-10 indices. We denote this capacity-limited (top-10) student distribution as $P_{S^{10}}$.

Optimization. At each step, we sample an index $x \sim P_{S^{10}}$ and compute the reward

$$r(x) = \log P_{\texttt{te}}(x) - \log P_{S^{10}}(x),$$

corresponding to a sample-based estimate of the negative reverse KL. We then update the sampled logit via $s_x \leftarrow s_x + \eta r(x)$ where η denotes the learning rate.

Metrics. We track two metrics during training: (1) *the top-10 change rate*, defined as the Jaccard distance (Jaccard, 1901) $\left(1 - \frac{|\mathcal{S}_t \cap \mathcal{S}_{t-1}|}{|\mathcal{S}_t \cup \mathcal{S}_{t-1}|}\right)$ between the sets of top-10 tokens $\mathcal{S}_{t-1}$, $\mathcal{S}_t$ at steps $t-1$ and t; and (2) *the top-1 change count*, the number of times the student's most probable index changes between updates.

Results. As shown in Figure 1 and Table 1, under the low-entropy teacher (Scenario A), the top-10 change rate decreases steadily and top-1 changes are rare. In contrast, under the high-entropy teacher (Scenario B), training exhibits persistent instability: the top-1 index changes frequently and the top-10 set fails to converge. These results demonstrate that reverse-KL rewards provide unstable learning signals when the teacher distribution is uncertain. This motivates training objectives that more directly transmit the teacher's distributional structure to the student.

Table 1: Top-1 Change Count for low teacher entropy scenario and a high teacher entropy scenario. We observe that when the teacher entropy is high, the top-1 index frequently changes.

Teacher Entropy	Temp.	Top-1 Change Count
(A) Low	0.3	7.3 ± 1.6
(B) High	1.0	84.0 ± 16.7

4 Entropy-Aware On-Policy Distillation

To address the limitations of reverse KL in on-policy distillation, we propose Entropy-Aware On-Policy Distillation (EOPD). Our key insight is that reverse KL and forward KL offer complementary strengths: reverse KL enables efficient, stable learning on confident teacher predictions, while forward KL's mode-covering property transfers the teacher's uncertainty and global structure. However, naively applying forward KL forces the student to cover the teacher's full distribution, including low-probability tails. This can degrade training efficiency, especially for students with limited capacity (Gu et al., 2023; Cha & Cho, 2025).

To leverage the best of both objectives, we propose an *entropy-aware* strategy that selectively applies forward KL based on the teacher's token-level uncertainty. Specifically, we define our token-level objective as:

$$\mathcal{L}_t^{\text{EOPD}}(\theta; \mathbf{c}_t) = \mathcal{L}_t^{\text{OPD}}(\theta; \mathbf{c}_t) + \mathbb{I}\big[H_t^{\text{te}} > \tau\big]\, \mathcal{L}_t^{\text{FKL}}(\theta; \mathbf{c}_t), \tag{9}$$

where $H_t^{\texttt{te}} = -\sum_{x \in \mathcal{V}} \pi_{\texttt{te}}(x|\mathbf{c}_t) \log \pi_{\texttt{te}}(x|\mathbf{c}_t)$ denotes the teacher's token-level entropy at position t, $\mathcal{V}$ denotes the vocabulary, and the forward KL divergence is

$$\mathcal{L}_t^{\text{FKL}}(\theta; \mathbf{c}_t) = \text{KL}(\pi_{\texttt{te}}(\cdot \mid \mathbf{c}_t) \,\|\, \pi_\theta(\cdot \mid \mathbf{c}_t)).$$

The first term $\mathcal{L}_t^{\text{OPD}}(\theta)$ in equation 9 corresponds to the clipped reverse KL loss defined in equation 8. The second term $\mathcal{L}_t^{\text{FKL}}(\theta; \mathbf{c}_t)$ is activated only when $H_t^{\texttt{te}} > \tau$, encouraging the student to preserve probability mass over multiple plausible continuations. In low-entropy regions where the teacher is confident, the objective reduces to standard reverse KL, retaining its efficiency and fast convergence. In high-entropy regions, forward KL prevents mode collapse and preserves the teacher's distributional diversity. The hyperparameter τ controls this transition; we study its effect in Appendix F

The full objective function for EOPD is like equation 4, where we bring back the expectations over prompts $\mathbf{q}$ and generated tokens $\mathbf{x}$, but with $\mathcal{L}_t^{\text{RKL}}(\theta; \mathbf{c}_t)$ replaced by $\mathcal{L}_t^{\text{EOPD}}(\theta; \mathbf{c}_t)$ in equation 9. We use Algorithm 1 to optimize this objective (See Appendix E). The expectations over $\mathbf{q}$ and $\mathbf{x}$ are approximated by sampling batches $\mathcal{B}$ (line 3 in Algorithm 1) and generating rollouts using the behavior policy π_{old} (line 6). The teacher is then queried to collect quantities needed to compute the objective (line 9). For the $\mathcal{L}_t^{\text{OPD}}(\theta; \mathbf{c}_t)$ term in equation 9, the expectation in equation 7 is approximated by a single-sample Monte Carlo estimate as discussed previously. In effect, the clipped reverse KL loss equation 8 is evaluated at the token x_t sampled during the rollouts. The forward KL $\mathcal{L}_t^{\text{FKL}}(\theta; \mathbf{c}_t)$ is approximated not by sampling, but as an expectation computed over the teacher's top-k tokens $\mathcal{S}_t^k$:

$$\mathcal{L}_t^{\text{FKL}}(\theta; \mathbf{c}_t) \approx \sum_{x \in \mathcal{S}_t^k} \tilde{\pi}_{\text{te}}(x \mid \mathbf{c}_t) \log \frac{\tilde{\pi}_{\text{te}}(x \mid \mathbf{c}_t)}{\pi_\theta(x \mid \mathbf{c}_t)}, \tag{10}$$

where $\tilde{\pi}_{\texttt{te}}(\cdot \mid c_t)$ denotes the teacher distribution renormalized over the top-k tokens $\mathcal{S}_t^k$, defined as

$$\tilde{\pi}_{\texttt{te}}(x \mid \mathbf{c}_t) = \frac{\pi_{\texttt{te}}(x \mid \mathbf{c}_t)}{\sum_{x' \in \mathcal{S}_t^k} \pi_{\texttt{te}}(x' \mid \mathbf{c}_t)}, \qquad x \in \mathcal{S}_t^k.$$

We restrict to the teacher's top-k tokens to ensure that the student does not have to learn from the low-probability tails of the teacher, along with improving computational efficiency (Shum et al., 2024; Peng et al., 2025).

By adapting to the teacher's local uncertainty, EOPD balances training stability with diversity preservation, enabling effective knowledge transfer across both confident and ambiguous regions of the output distribution.

5 EXPERIMENTS

We address the following research questions:

RQ1: Does EOPD improve mathematical reasoning performance compared to existing baselines? (§5.2)

RQ2: Does EOPD improve out-of-domain performance compared to existing baselines? (§5.3)

RQ3: How does EOPD affect the transfer of token-level uncertainty from the teacher to the student? (§5.4)

RQ4: How does EOPD differ from other entropy-promoting methods in transferring teacher uncertainty? (§5.5)

Beyond the above research questions, we present additional ablation studies in Appendix F.

5.1 EXPERIMENTAL SETTINGS

Models and Training Datasets. We conduct experiments using three Qwen3 (Yang et al., 2025) student models of different sizes, Qwen3-0.6B-Base, Qwen3-1.7B-Base, and Qwen3-4B-Base. We use Qwen3-8B as the teacher model, without enabling thinking mode. For the 0.6B and 1.7B student model, training was performed using the MATH (Hendrycks et al., 2021) dataset, while for the 4B student model, the more challenging DAPO (Yu et al., 2025) dataset is used.

Evaluation Benchmarks and Metrics. We evaluate our models on MATH500 (Hendrycks et al., 2021), AIME24/25 (MAA, 2025), AMC23 (MAA, 2023), Minerva (Lewkowycz et al., 2022), and OlympiadBench (He et al., 2024), using a rollout temperature of 1.0, top-p sampling with $p = 0.8$, and a maximum response length of 8192 tokens. We sample 8 responses per question and report the average accuracy (Avg@8) and pass rate (Pass@8).

Baselines. We compare our method with several baselines:

1. **Knowledge Distillation** (Hinton et al., 2015; Kim & Rush, 2016): KD trains the student model by minimizing forward KL divergence with respect to the teacher's distribution, along with a cross-entropy loss on hard labels from an off-policy dataset generated by the teacher.
2. **On-Policy Distillation** (Lu & Lab, 2025): The student is trained using on-policy trajectories, following the formulation described in §2.3.
3. **Group Relative Policy Optimization (GRPO)** (Shao et al., 2024): GRPO optimizes the policy by comparing verifiable rewards across multiple sampled outputs for the same input.

Full implementation details are provided in Appendix A.

5.2 MAIN RESULTS

Performance on Mathematical Reasoning. As shown in Table 2, EOPD demonstrates stable performance improvements across six mathematical reasoning benchmarks, achieving improved or competitive results in terms of Avg@8 and Pass@8. In particular, compared to OPD, EOPD improves Avg@8 by $+1.16$ and Pass@8 by $+1.37$ on average across the six benchmarks for the Qwen3-0.6B-Base model, Avg@8 by $+0.99$ and Pass@8 by $+2.39$ for the Qwen3-1.7B-Base model, and $+1.80$ in Avg@8 and $+5.05$ in Pass@8 for the Qwen3-4B-Base model. These improvements highlight the effectiveness of EOPD, particularly its ability to better transfer the teacher's local uncertainty.

Pass@k Performance. Pass@k measures the probability of obtaining at least one correct solution among k sampled rollouts. While Avg@k reflects average reasoning quality, Pass@k more directly captures the model's problem-solving capability by approximating best-case performance under multiple samples. In Appendix D, we report Pass@k for AIME24 and AIME25 with k ranging from 8 to 128, and for AMC23 with k ranging from 4 to 64. EOPD achieves consistently higher Pass@k compared to OPD. Notably, on harder benchmarks such as AIME, the gap between the

Table 2: Main results (accuracy %) on six mathematical reasoning benchmarks. EOPD demonstrates consistent improvements across benchmarks and model scales. Parentheses indicate training data. **Bold** indicates best performance and underline indicates second-best.

Method	MATH500		AMC23		Minerva		OlympiadBench		AIME24		AIME25	
	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8
Qwen3-0.6B-Base (MATH)												
KD	47.80	69.60	23.43	52.50	18.61	36.03	18.15	36.59	2.19	6.67	0.83	6.67
GRPO	51.83	74.40	25.25	55.00	20.08	35.66	18.91	34.96	**4.58**	10.00	0.83	**10.00**
OPD	50.09	73.20	24.69	**57.50**	19.98	34.93	**20.15**	37.19	2.50	10.00	**1.25**	6.67
EOPD	**52.02**	**76.00**	**27.81**	55.00	**20.82**	**37.13**	19.52	**39.56**	4.17	**13.33**	**1.25**	6.67
Qwen3-1.7B-Base (MATH)												
KD	63.80	84.20	38.11	70.00	28.14	44.10	27.66	48.40	10.06	20.00	3.36	16.67
GRPO	**68.83**	84.60	40.62	70.00	29.46	48.16	29.89	50.81	9.17	20.00	4.58	16.67
OPD	67.76	84.80	39.06	70.00	29.83	47.06	30.09	**51.56**	8.33	20.00	**6.25**	16.67
EOPD	68.73	**87.60**	**41.88**	**75.00**	**30.15**	**50.74**	**30.28**	51.11	**10.42**	**23.33**	5.83	16.67
Qwen3-4B-Base (DAPO-Math-14k)												
KD	74.73	92.20	48.13	85.00	34.65	55.15	37.33	62.52	12.50	30.00	12.08	23.33
GRPO	**80.47**	91.00	58.44	75.00	**40.81**	55.51	**43.37**	60.74	14.85	30.00	12.66	26.67
OPD	78.81	90.80	57.33	80.06	40.08	54.00	42.10	58.80	**18.33**	26.67	12.08	30.00
EOPD	80.20	**93.00**	**60.94**	**87.50**	39.71	**56.99**	43.24	**63.11**	17.92	**36.67**	**17.50**	**33.33**

two methods widens as k increases. This suggests that EOPD more effectively explores diverse reasoning trajectories, thereby increasing the likelihood of reaching a correct solution.

5.3 Out-of-domain Evaluation

Table 3 reports performance on out-of-domain benchmarks for the Qwen3-1.7B-Base student: GPQA-Diamond (Rein et al., 2024), MMLU-Pro (Wang et al., 2024), and AlpacaEval 2.0 (Dubois et al., 2024), which evaluate general reasoning and instruction-following abilities. Although trained exclusively on math data, OPD and EOPD exhibit stable performance increases over KD and GRPO across these benchmarks, indicating that on-policy distillation transfers useful reasoning behaviors beyond the training distribution.

Furthermore, EOPD outperforms OPD on all out-of-domain benchmarks except for AlpacaEval 2.0 win rate. This suggests that selectively incorporating teacher guidance on high-uncertainty tokens also provides additional benefits for general reasoning and instruction-following as well.

Table 3: Out-of-domain benchmark results on Qwen3-1.7B-Base student, covering general reasoning and instruction following. Our method achieves higher average accuracy and pass@8 on general reasoning, and is competitive or superior in win rate (*WR*) and length-controlled win rate (*LC-WR*). Best and second-best results are shown in **bold** and underline, respectively.

Benchmark	Metric	KD	GRPO	OPD	EOPD
GPQA-Diamond	Avg@8	27.01	27.86	30.08	**31.50**
	Pass@8	75.20	77.83	77.78	**81.31**
MMLU-Pro	Pass@1	37.54	41.46	42.26	**43.20**
AlpacaEval 2.0	LC-WR	19.30	16.86	22.86	**23.83**
	WR	23.63	20.55	**29.92**	29.54

Table 4: Comparison with entropy-driven exploration baselines for the Qwen3-1.7B-Base student. EOPD achieves higher Avg@8 and Pass@8 compared to the other baselines. **Bold** indicates best performance and underline indicates second-best.

Benchmark	Metric	Entropy Bonus	Advantage Shaping	EOPD
MATH500	Avg@8	66.87	67.90	**68.73**
	Pass@8	86.00	85.20	**87.60**
AMC23	Avg@8	39.69	36.56	**41.88**
	Pass@8	**75.00**	**75.00**	**75.00**
AIME24	Avg@8	9.58	8.75	**10.42**
	Pass@8	20.00	**23.33**	**23.33**
AIME25	Avg@8	4.58	**5.83**	**5.83**
	Pass@8	16.67	**20.00**	16.67

5.4 Token-Level Entropy Analysis

To support the hypothesis that EOPD more effectively explores diverse reasoning trajectories, we conduct a token-level analysis of uncertainty transfer from the Qwen3-8B teacher to the Qwen3-1.7B-Base student. Following §3.1, we compare students trained with OPD and EOPD on the AIME24 and AIME25 benchmarks.

For each generated token, we compute the entropy of the model's predicted distribution conditioned on the preceding context. Figure 2 aggregates these values into a histogram. In the mid-entropy range (approximately 0.1–1.0), OPD and EOPD exhibit similar distributions and remain close to the teacher. The key difference emerges at higher en-

tropy values (entropy $\geq$ 1.0): EOPD retains substantially more probability mass in this region and stays much closer to the teacher, whereas OPD markedly under-represents it.

High-entropy tokens correspond to intrinsically ambiguous decision points where the teacher assigns non-negligible probability to multiple plausible continuations. Therefore, preserving a teacher-like distribution at these positions is likely important. We hypothesize that EOPD's improved preservation of high-entropy tokens mitigates premature overconfidence and mode collapse, contributing to the downstream performance gains observed in §5.2.

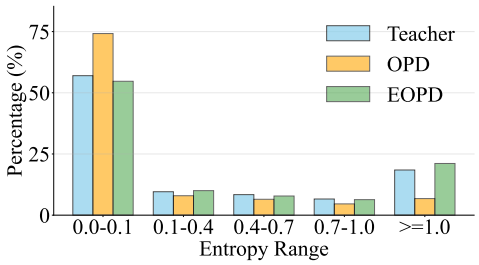


Figure 2: Token-level entropy histograms comparing the Qwen3-8B teacher with Qwen3-1.7B-Base trained using OPD and EOPD on the AIME 2024 and 2025 benchmarks. While both methods exhibit similar distributions to the teacher in the mid-entropy range, EOPD preserves more probability mass in the high-entropy region, staying closer to the teacher than OPD.

5.5 Comparison with Entropy-Driven Baselines

We compare our method with entropy-based approaches commonly used to encourage exploration in reinforcement learning. Specifically, we evaluate the Qwen3-1.7B-Base student using two strategies: an entropy bonus (Schulman et al., 2017), which explicitly regularizes the policy toward higher entropy, and advantage shaping (Cheng et al., 2025), which augments the advantage with an entropy-dependent term to bias updates toward high-uncertainty actions. Detailed implementation details are provided in Appendix A.

As shown in Table 4, EOPD outperforms both baselines across several benchmarks. To explain these gains, we analyze entropy dynamics (Figure 7) and forward KL at high-entropy positions (Figure 8). OPD + Advantage Shaping exhibits substantially lower training entropy than other methods, indicating limited diversity and explaining its poor performance. However, EOPD and OPD + Entropy Bonus maintain similar entropy levels, so entropy alone does not explain EOPD's advantage. Looking at the forward KL at positions where the teacher's entropy exceeds $\tau = 0.8$, EOPD achieves consistently lower values than OPD + Entropy Bonus, indicating better alignment with the teacher in uncertain regions. Overall, these results show that preserving entropy alone is insufficient for the student to match the teacher's diversity, especially in high-entropy regions.

6 Related Work

Knowledge Distillation (Buciluǎ et al., 2006; Hinton et al., 2015) trains a smaller model to approximate a larger model's output distribution. For auto-regressive models, various approaches have been proposed, including matching token-level distributions (Sanh et al., 2019), teacher-generated sequences (Kim & Rush, 2016), attention scores (Wang et al., 2020), and alternative divergence objectives (Wen et al., 2023; Ko et al., 2024; Shing et al., 2025) to overcome limitations of forward and reverse KL. To address exposure bias from the mismatch between teacher-generated training data and self-generated sequences at inference (Bengio et al., 2015; Ranzato et al., 2015), several on-policy methods have been proposed, including combining off-policy and on-policy data (Lin et al., 2020; Agarwal et al., 2024), reverse KL over student-generated contexts (Gu et al., 2023; Lu & Lab, 2025), and interleaved sampling (Xu et al., 2024). In this paper, we build upon on-policy distillation (Lu & Lab, 2025) by leveraging dense teacher supervision and effectively transferring the teacher's uncertainty based on its entropy.

Reasoning Abilities of Language Models have advanced through prompting (Wei et al., 2022; Yao et al., 2023), test-time scaling (Muennighoff et al., 2025; Snell et al., 2024), and distillation from larger models (Guha et al., 2025; He et al., 2025; Guo et al., 2025). Recently, RL methods have received particular attention, including methods that optimize verifiable rewards (Shao et al., 2024; Yu et al., 2025; Liu et al., 2025) or apply step-level supervision using process reward models (Uesato et al., 2022; Lightman et al., 2023). In addition, Cheng et al. (2025); Wang et al. (2025) have proposed entropy-based methods to encourage exploration during training. Instead of relying on intrinsic entropy for exploration, our method uses teacher-guided KL selection to transfer uncertainty and distributional structure.

7 CONCLUSION

We study on-policy distillation for language models and identify a key limitation of the reverse-KL objective: its mode-seeking nature can collapse generation diversity and destabilize learning at token positions where the teacher distribution has high entropy. To address this, we introduce EOPD, which adapts the on-policy distillation objective and selectively applies forward-KL for high-entropy teacher tokens. Empirically, EOPD better preserves the teacher-like distribution while retaining on-policy training efficiency, yielding consistent gains over standard on-policy distillation on six mathematical reasoning benchmarks Overall, our results demonstrate that explicitly modeling teacher uncertainty is essential for stable, diverse, and effective knowledge transfer.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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*, 2024.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in neural information processing systems*, 2015.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006.

Sungmin Cha and Kyunghyun Cho. Why knowledge distillation works in generative models: A minimal working explanation. *arXiv preprint arXiv:2505.13111*, 2025.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Wayne Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. *arXiv preprint arXiv:2506.14758*, 2025.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Charles Goddard and Lucas Atkins. Distillkit: Flexible knowledge distillation for large language models, 2024. URL `https://github.com/arcee-ai/distillkit`.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

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. *arXiv preprint arXiv:2506.04178*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Annual Meeting of the Association for Computational Linguistics*, 2024.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025.

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. *arXiv preprint arXiv:2103.03874*, 2021.

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

Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 1901.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Conference on Empirical Methods in Natural Language Processing*, 2016.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. In *Advances in neural information processing systems*, 2022.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2023.

Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. Autoregressive knowledge distillation through imitation learning. *arXiv preprint arXiv:2009.07253*, 2020.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *URL https://arxiv.org/abs/2503.20783*, 2025.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

MAA. American mathematics competitions - amc. `https://maa.org/`, 2023.

MAA. American invitational mathematics examination - aime. `https://maa.org/`, 2025.

Thomas P. Minka. Divergence measures and message passing. *arXiv preprint arXiv:0507022*, 2005.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. s1: Simple test-time scaling. In *Conference on Empirical Methods in Natural Language Processing*, 2025.

Hao Peng, Xin Lv, Yushi Bai, Zijun Yao, Jiajie Zhang, Lei Hou, and Juanzi Li. Pre-training distillation for large language models: A design space exploration. In *Annual Meeting of the Association for Computational Linguistics*, 2025.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

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 *Conference on Language Modeling*, 2024.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Makoto Shing, Kou Misaki, Han Bao, Sho Yokoi, and Takuya Akiba. Taid: Temporally adaptive interpolated distillation for efficient knowledge transfer in language models. *arXiv preprint arXiv:2501.16937*, 2025.

KaShun Shum, Minrui Xu, Jianshu Zhang, Zixin Chen, Shizhe Diao, Hanze Dong, Jipeng Zhang, and Muhammad Omer Raza. First: Teach a reliable large language model through efficient trustworthy distillation. *arXiv preprint arXiv:2408.12168*, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process-and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, et al. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*, 2025.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in neural information processing systems*, 2020.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in neural information processing systems*, 2022.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. F-divergence minimization for sequence-level knowledge distillation. *arXiv preprint arXiv:2307.15190*, 2023.

Wenda Xu, Rujun Han, Zifeng Wang, Long T Le, Dhruv Madeka, Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. *arXiv preprint arXiv:2410.11325*, 2024.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

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.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in neural information processing systems*, 2023.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

A Implementation Details

In this section, we provide implementation details of EOPD and other baselines in our experiments.

Off-policy Training. We perform off-policy distillation using DistillKit Goddard & Atkins (2024). The hyperparameters for all off-policy distillation experiments are summarized in Table 5. In this setting, we first sample a response from the teacher model for each question, and then train the student model on the resulting teacher context using a combination of cross-entropy loss and forward KL loss.

Table 5: Hyperparameters used for Off-policy distillation.

Hyperparameter	Value
Learning rate	1e-5
LR scheduler type	cosine
Optimizer	AdamW
CE loss weight	0.5
KL loss weight	0.5
Training Batch size	128
Training epoch	3
Cutoff length	4096
Top-k	16

On-policy Training. On-policy distillation and GRPO are implemented using the `verl` Sheng et al. (2024) framework. As shown in Table 6, we use a batch size of $B = 128$ and a mini-batch size of $B_{\text{mini}} = 32$, which results in four gradient update steps per training iteration. For on-policy distillation, we generate a single rollout per problem during training. In contrast, for GRPO, we generate eight rollouts per problem to enable relative comparison among trajectories.

Table 6: Hyperparameters used for On-policy distillation and GRPO.

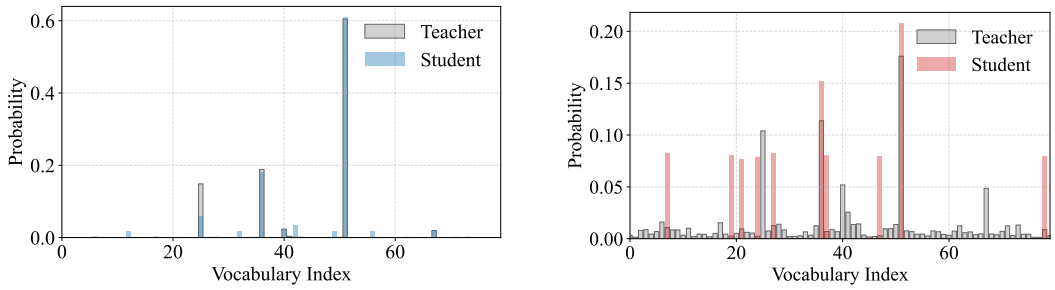
Hyperparameter	OPD, EOPD	GRPO
Learning rate	3e-6	3e-6
LR scheduler type	cosine	cosine
Optimizer	AdamW	AdamW
Training batch size	128	128
Mini batch size	32	32
Samples per prompt	1	8
Max response length	4096	4096
Top-k (for FKL)	16	-
Training temperature	1.0	1.0
Training epoch	3 (MATH), 2 (DAPO)	3 (MATH), 2 (DAPO)

Chat Template. During model training, for knowledge distillation, we used the same chat template as the teacher model to effectively learn the teacher distribution. Since the teacher model used in our experiments was the Qwen3-Instruct Yang et al. (2025) model in non-thinking mode, the chat template was defined as follows:

```
<|im_start|>user\n{query}<|im_end|>\n<|im_start|>assistant\n<think>
\n\n</think>
```

For GRPO training, we used the default chat template of the Qwen3-Base model. The template used is as follows:

```
<|im_start|>user\n{query}<|im_end|>\n<|im_start|>assistant\n
```



(a) Low Entropy Teacher Distribution

(b) High Entropy Teacher Distribution

Figure 3: Teacher–student distributions under low and high-entropy scenarios. When the teacher distribution has low entropy, the student model accurately converges to the teacher. In contrast, when the teacher distribution has high entropy, optimization with reverse KL based reward leads the student to align with a few dominant indices of the teacher distribution, while the global structure is not fully approximated.

Entropy Bonus. Entropy Bonus Schulman et al. (2017) adds the policy entropy directly to the optimization objective, discouraging premature policy convergence and encouraging stochastic action selection during training.

Advantage Shaping. Advantage Shaping Cheng et al. (2025) augments the advantage function by adding a gradient-detached entropy term shaped by $\psi(\cdot)$:

$$A_t \leftarrow A_t + \psi(\mathcal{H}_t).$$

This mechanism encourages the model to reinforce uncertain decisions when $A_t > 0$ by providing additional reward, while reducing the penalty for uncertain decisions when $A_t < 0$, thereby promoting more exploratory reasoning behavior.

Here, the shaping function $\psi(\cdot)$ is defined as

$$\psi\left(\mathcal{H}_t\right) = \min\left(\alpha \mathcal{H}_t^{\text{detach}}, \frac{|A_t|}{\kappa}\right), \tag{11}$$

where $\alpha > 0$ and $\kappa > 1$ are hyperparameters. In the experiment described in subsection 5.5, we follow the setup of Cheng et al. (2025) and set $\alpha = 0.1$ and $\kappa = 2$.

B Toy Experiment Visualizations

In this section, we present visualizations of the toy experiment under each scenario. As shown in Figure 3, when the teacher distribution exhibits low entropy, the student model converges closely to the teacher distribution. However, when the teacher distribution has high entropy, optimization using a reverse-KL-based reward causes the student to concentrate on a small number of dominant indices. As a result, the global structure of the teacher distribution is not fully captured.

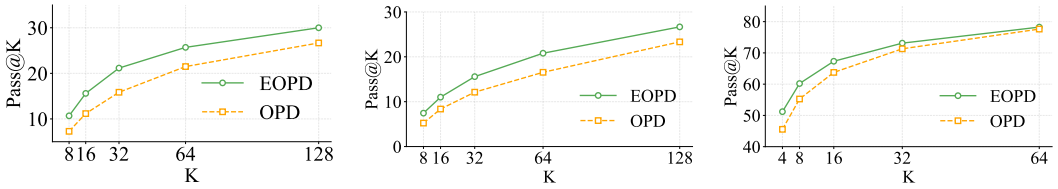
C Evaluation Details

Math reasoning benchmarks (six datasets) and the GPQA-Diamond Rein et al. (2024) evaluation were conducted using the evaluation pipeline released with Qwen2.5-Math Yang et al. (2024). All experiments were performed in a zero-shot setup. During sampling, the maximum sequence length was set to 8192, with temperature = 1.0 and top-p = 0.8. We report both average@k and pass@k as evaluation metrics. For all problems, the same prompt template was appended at the end: ``Please reason step by step, and put your final answer within \boxed{}.''

For out-of-distribution (OOD) evaluation, MMLU-Pro Wang et al. (2024) was evaluated under the default 5-shot setting (examples are sampled from the validation set), while all other configurations were kept identical to those used in the math reasoning benchmarks. The prompt template for MMLU-Pro is specified below. For AlpacaEval Dubois et al. (2024), we followed the default evaluation protocol and used the annotator model weighted_alpaca_eval_gpt4_turbo to report both win rate and length-controlled win rate against gpt4_turbo Achiam et al. (2023).

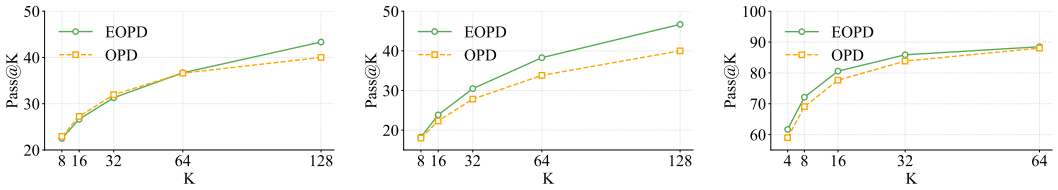
```
 The following are multiple choice questions (with answers)
about {$}.  Think step by step and then finish your answer
with ``the answer is (X)'' where X is the correct letter
choice.
```

D PASS@k EXPERIMENTS



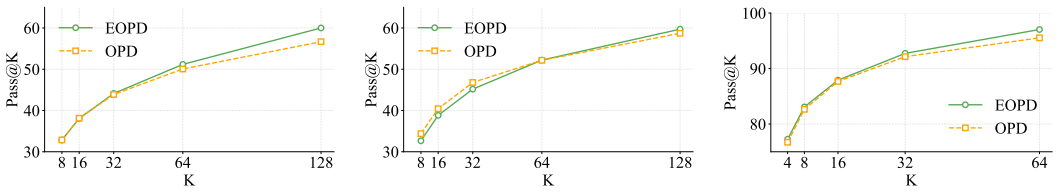
(a) AIME24 (b) AIME25 (c) AMC23

Figure 4: Pass@k performance comparison between OPD and EOPD on the AIME and AMC benchmarks of the Qwen3-0.6B Base student.



(a) AIME24 (b) AIME25 (c) AMC23

Figure 5: Pass@k performance comparison between OPD and EOPD on the AIME and AMC benchmarks of the Qwen3-1.7B Base student.



(a) AIME24 (b) AIME25 (c) AMC23

Figure 6: Pass@k performance comparison between OPD and EOPD on the AIME and AMC benchmarks of the Qwen3-4B Base student.

Figure 4, Figure 5, and Figure 6 show the Pass@k performance across different values of k for the Qwen3-0.6B-Base, Qwen3-1.7B-Base, and Qwen3-4B-Base models, respectively. Compared to OPD, EOPD achieves better performance on AIME24/25 MAA (2025) and AMC23 MAA (2023). On harder benchmarks such as AIME, the performance gap between the two methods widens as k increases. This suggests that EOPD more effectively explores diverse reasoning trajectories, thereby increasing the likelihood of reaching a correct solution.

E MAIN ALGORITHM

Algorithm 1 Entropy-Aware On-Policy Distillation

Require: Teacher policy $\pi_{\texttt{te}}$, student policy π_θ
Require: Entropy threshold τ, Top-k size k
Require: PPO clip ϵ, learning rate η
Require: Training dataset $\mathcal{D}$
for each training iteration **do**
  Set $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$
  Sample a prompt batch $\mathcal{B} = \{\mathbf{q}_i\}_{i=1}^{B} \subset \mathcal{D}$
  Rollout buffer $\mathcal{R} \leftarrow \emptyset$
  for each prompt $\mathbf{q} \in \mathcal{B}$ **do**
    Sample $\mathbf{x} = (x_1, \ldots, x_{|\mathbf{x}|}) \sim \pi_{\theta_{\text{old}}}(\cdot \mid \mathbf{q})$
    for $t = 1$ to $|\ \mathbf{x}\ |$ **do**
      Context $\mathbf{c}_t = (\mathbf{q}, x_{<t})$
      Query teacher Q_t: $(\log \pi_{\texttt{te}}(x_t \mid \mathbf{c}_t), H_t^{\texttt{te}}, \mathcal{S}_t^k)$
    end for
    Store trajectory-level data:

$$\left(\mathbf{q}, \mathbf{x}, \{Q_t\}_{t=1}^{|\mathbf{x}|}\right) \rightarrow \mathcal{R}$$

  end for
  for each mini-batch gradient step **do**
    Sample a mini-batch of prompts $\mathcal{B}_{\text{mini}} \subset \mathcal{R}$
    $\mathcal{L}(\theta) = \dfrac{1}{\sum_{(\mathbf{q},\mathbf{x})\in\mathcal{B}_{\text{mini}}} |\mathbf{x}|} \displaystyle\sum_{(\mathbf{q},\mathbf{x})\in\mathcal{B}_{\text{mini}}} \sum_{t=1}^{|\mathbf{x}|} \mathcal{L}_t^{\text{EOPD}}(\theta; \mathbf{c}_t)$ using equation 9
    Update parameters: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta)$
  end for
end for

Algorithm 1 implements the entropy-aware on-policy distillation objective introduced in Section 4, combining clipped reverse KL with a selectively activated forward KL term based on the teacher's token-level entropy. At each iteration, the student generates rollouts using a behavior policy, queries the teacher to obtain token-level log probability, entropy, and top-k distributions, and updates its parameters by optimizing the EOPD objective. This procedure enables efficient and stable knowledge transfer while preserving the teacher's uncertainty in high-entropy regions.

F Additional Ablations

Impact of Entropy Threshold τ. The entropy threshold τ is a hyperparameter that controls when the forward KL objective is activated based on the teacher model's token-level uncertainty. A lower τ activates forward KL at more tokens, thereby increasing its overall influence during training, while a higher τ restricts forward KL to only a small subset of tokens where the teacher exhibits the highest uncertainty. As shown in Table 7, EOPD demonstrates stable performance across a wide range of τ values, indicating that the method is not highly sensitive to this hyperparameter. Also, we observe an overall trend where Pass@8 performance decreases as τ increases. This suggests that restricting the application of forward KL can inhibit the transfer of the teacher's uncertainty and diverse reasoning trajectories. Overall, the best performance is obtained at $\tau = 0.8$, and we use this value for all other experiments in this paper.

τ	MATH500		AMC23	
	Avg@8	Pass@8	Avg@8	Pass@8
0.6	68.24	86.80	39.69	75.00
0.8	68.73	87.60	41.88	75.00
1.0	67.58	84.00	41.53	72.50
1.2	64.82	84.80	39.69	75.00
1.4	67.61	83.80	38.72	72.50
1.6	68.10	84.00	39.62	70.00

Table 7: Performance with respect to the threshold τ for the Qwen3-1.7B-Base student. EOPD is not highly sensitive to τ, although Pass@8 accuracy generally decreases as τ increases.

Impact of Forward KL Placement. To study the impact of token selection for application of forward KL, we compare EOPD against two alternative forward KL augmentation strategies for the Qwen3-1.7B-Base student: full forward KL, applied at all token positions[2] , and random forward KL, applied to a randomly selected 20% of positions[3].

As shown in Table 8, EOPD shows competitive performance compared to full and random forward KL. Notably, random forward KL underperforms the other two approaches across all benchmarks, suggesting that the placement of forward KL on appropriate positions, like EOPD is important.

Method	**MATH500**		**AMC23**		**AIME24**		**AIME25**	
	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8
Full	<u>67.58</u>	<u>86.20</u>	<u>39.39</u>	**75.00**	<u>8.75</u>	**23.33**	5.00	**20.00**
Random	67.33	84.80	36.88	<u>67.50</u>	7.92	<u>20.00</u>	<u>5.42</u>	13.33
EOPD	**68.73**	**87.60**	**41.88**	**75.00**	**10.42**	**23.33**	**5.83**	<u>16.67</u>

Table 8: Comparison with different forward KL placement strategies for the Qwen3-1.7B-Base student. EOPD shows competitive performance against full and random forward KL, while random forward KL underperforms the other two approaches. **Bold** indicates the best performance and <u>underline</u> indicates second-best.

[2]This can be viewed as a variant of GKD (Agarwal et al., 2024) that jointly optimizes the forward and reverse KL divergences.

[3]This choice is motivated by experiments with EOPD at $\tau = 0.8$, where, as shown in Figure 9, forward KL is applied to approximately 15–20% of tokens on average.

G Training Dynamics Compared with Entropy-Driven Baselines

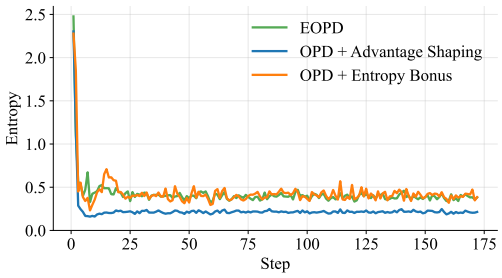


Figure 7: Average policy entropy during training for the Qwen3-1.7B-Base student trained with OPD + Entropy Bonus, OPD + Advantage Shaping, and EOPD. Advantage Shaping converges to a lower entropy regime, while Entropy Bonus maintains entropy levels comparable to EOPD.

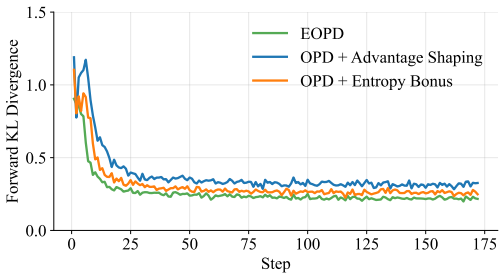


Figure 8: Average forward KL divergence measured during training at token positions where the teacher distribution exhibits high entropy (entropy ≥ 0.8) for the Qwen3-1.7B-Base student. Compared to other baselines, EOPD maintains lower forward KL values throughout training.

Figure Figure 7 and Figure Figure 8 summarize the evolution of training behavior for entropy-driven baselines and EOPD in the Qwen3-1.7B-Base setting. Figure Figure 7 tracks how the policy entropy changes over the course of training, showing that Advantage Shaping leads to a progressively more deterministic policy, whereas OPD + Entropy Bonus and EOPD retain broader action distributions. Figure Figure 8 depicts the forward KL divergence evaluated at token positions where the teacher distribution has high entropy, indicating the degree to which each method follows the teacher in regions of elevated uncertainty. These training dynamics complement the performance comparisons reported in subsection 5.5.

H HIGH ENTROPY TOKEN RATIO

Figure 9 shows the ratio of high-entropy tokens measured on rollouts generated by the student model during EOPD training, where entropy is computed from the teacher model. High-entropy tokens are defined as token positions whose teacher entropy exceeds a threshold τ $(= 0.8)$, indicating regions where the teacher exhibits higher uncertainty.

As shown in the figure, the proportion of high-entropy tokens is relatively high during the early stages of training and decreases as training progresses. After convergence, the ratio stabilizes at approximately 15% to 20%.

This observation motivates the random forward KL setting in Appendix F, where forward KL is applied to a randomly selected 20% of token positions. In this setting, the ratio of tokens receiving forward KL is matched to EOPD , while the token selection is independent of the teacher's uncertainty, serving as a controlled baseline.

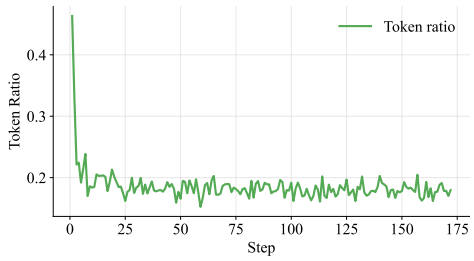


Figure 9: Ratio of tokens with high teacher entropy in rollouts produced by the student model during EOPD training.