OPDARENA: UNDERSTANDING LEARNING AND FORGETTING IN ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) is increasingly used for language model post-training, yet how teacher supervision shapes learning and forgetting remains unclear. We present OPDArena, a framework for comparing supervised fine-tuning (SFT), reinforcement learning with verifiable rewards (RLVR), and OPD across five pretrained models and diverse teacher sources, jointly evaluating mathematical learning, general capability retention, and predictive distribution changes. We find that teacher choice strongly shapes retention: RLVR-trained teachers support task learning while preserving prior capabilities, whereas SFT-trained teachers induce greater capability loss. Out-of-domain (OOD) distribution change shows a more consistent positive association with forgetting than in-domain change. We then examine the additional information teachers provide relative to student Base. Matched teacher comparisons connect preferences learned through RLVR to lower relative OOD influence and stronger task learning with limited forgetting; calibrated local directions further support differences in cross-domain effects. Parameter and prediction trajectories show how students acquire teacher preferences through different parameter paths, with task gains and forgetting inherited at different rates.

1 INTRODUCTION

Post-training enables language models to acquire new skills, but can erode capabilities learned during pretraining (Wang et al., 2024; Luo et al., 2025). On-policy distillation (OPD) has become a prominent approach to post-training, combining student-generated responses with token-level teacher supervision (Gu et al., 2024; Agarwal et al., 2024; Li et al., 2026).

Compared with supervised fine-tuning (SFT) on fixed, off-policy responses, reinforcement learning with verifiable rewards (RLVR) often preserves prior capabilities better, a behavior linked to on-policy data and limited distributional change (Shenfeld et al., 2025; Zhang et al., 2026). OPD shares the on-policy setting, yet retention outcomes vary substantially across distillation settings (Ye et al., 2026; Wang et al., 2026). This raises the question of how teacher choice shapes learning and capability retention in OPD. Related work is reviewed in Appendix F.

We conduct a systematic comparison of SFT, RLVR, and OPD across Base models and teacher sources under a shared mathematical post-training setup. Across models, RLVR and Self-OPD (RLVR) show the strongest retention, followed by OPSD. OPD from larger teachers and Self-OPD (SFT) incur greater forgetting, while SFT has the largest average capability loss. Figure 1 places this hierarchy alongside in-domain learning: RLVR and Self-OPD (RLVR) combine learning gains with strong retention, whereas OPSD's low forgetting accompanies weaker in-domain performance.

Distribution measurements show a more consistent positive association between forgetting and OOD change than between forgetting and in-domain change. We measure update selectivity by the amount of OOD prediction change per unit of in-domain change, then examine how it varies with the information teachers provide relative to student Base.

OPD from larger Base teachers produces small in-domain changes and high relative OOD influence. At fixed teacher and student sizes, comparisons with RLVR-trained and constructed teachers connect task-trained preferences to more selective updates and stronger task learning with limited forgetting. Local direction comparisons support this relationship after calibrating in-domain change. Paired

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

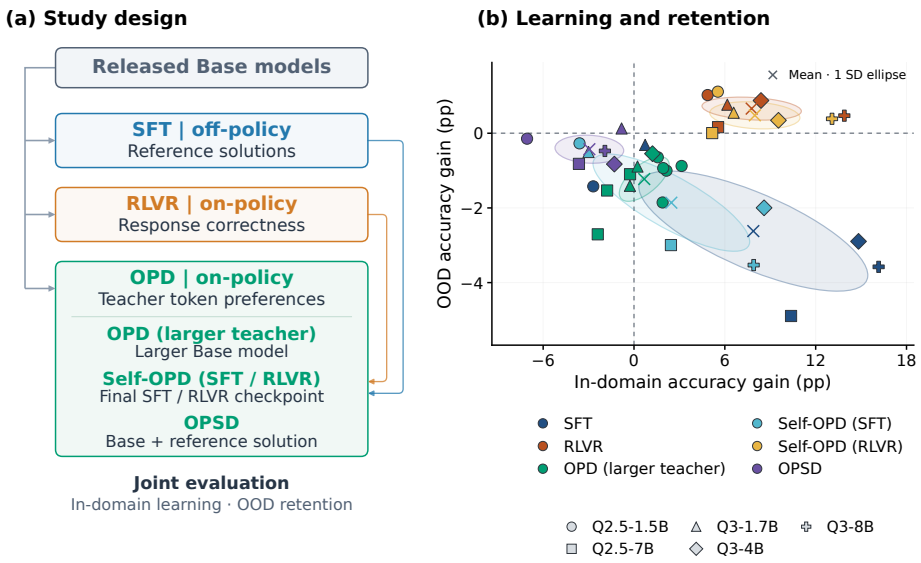


Figure 1: Training methods and joint learning–retention outcomes. (a) Training paths and feedback sources from a shared Base model. (b) In-domain accuracy gain versus OOD accuracy gain from Base; colors identify methods and shapes identify models. RLVR and Self-OPD (RLVR) combine task learning with low forgetting, while OPSD's strong retention accompanies weaker in-domain performance.

source-training and Self-OPD analyses further trace how teacher preferences are inherited through parameter and prediction trajectories.

Our contributions are:

- **A systematic evaluation framework.** OPDArena compares SFT, RLVR, and OPD teacher sources under a shared setup, jointly characterizing target-domain learning, general capability retention, and predictive distribution changes.
- **A refined view of distribution change and retention.** We show that OOD distribution change is more consistently associated with forgetting across student models and teacher configurations.
- **Teacher information and update selectivity.** Matched teacher comparisons and calibrated local directions connect information sources to relative OOD influence. Parameter, prediction, and prefix analyses further examine the different rates at which students inherit task gains and forgetting.

2 Method

We compare post-training methods applied to the same released Qwen Base models and measure their effects on mathematical learning and general capability retention. Our framework distinguishes how supervision enters the model from how its predictions change across domains.

2.1 Training Signals and Teachers

Let p_0 denote a Base model, p_θ the model being trained from it, and q a frozen distillation teacher. The training set $\mathcal{D}$ contains problems x and reference solutions c. SFT learns from these fixed, off-policy solutions; RLVR and OPD use on-policy responses y generated by the model being trained. At position t, the student prefix is $s_t = (x, y_{<t})$; s_t^{T} denotes the same problem and continuation with

the teacher's conditioning information. The training objectives are

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x,c)\sim\mathcal{D}}\left[\sum_t \log p_\theta(c_t \mid x, c_{<t})\right], \tag{1}$$

$$\mathcal{L}_{\mathrm{RLVR}}(\theta) = -\mathbb{E}_{x\sim\mathcal{D},\, y\sim p_\theta(\cdot|x)}\left[R(x,y)\right], \tag{2}$$

$$\mathcal{L}_{\mathrm{OPD}}(\theta;\bar{\theta}) = \mathbb{E}_{s\sim d_{\bar{\theta}}}\left[\mathrm{KL}\big(p_\theta(\cdot \mid s) \,\|\, q(\cdot \mid s^{\mathrm{T}})\big)\right]. \tag{3}$$

Here $R(x, y)$ is a verifiable correctness reward. For the OPD update, $d_{\bar{\theta}}$ is the empirical prefix distribution collected with the current rollout policy $p_{\bar{\theta}}$, held fixed during the actor update. Reverse KL transfers teacher preferences at these student-generated prefixes (Gu et al., 2024; Agarwal et al., 2024).

Training feedback. Our RLVR baseline uses GRPO (Shao et al., 2024) with verifiable correctness rewards. Centering and scaling rewards within each group of responses to the same problem gives a relative advantage shared by all tokens in a response. OPD assigns token-dependent advantages through the teacher–student log-probability ratio. We use a student-weighted Top-16 approximation (Li et al., 2026). Input formats and the actor update settings are given in Appendices A.2 and A.4.

Teacher source. We study self-distillation from post-trained checkpoints, distinguishing Self-OPD (SFT) and Self-OPD (RLVR) by how the teacher is trained. We reuse the final saved checkpoints of the corresponding SFT and RLVR experiments reported here as frozen teachers. Each student starts from the same released Base model used in its teacher's training. OPD from larger models uses frozen Base teachers with more parameters than the student, drawn from the same or a different model family. We use OPSD (Zhao et al., 2026) as a representative of self-distillation with privileged information, where the teacher receives additional context unavailable to the student. Within our shared OPD recipe, a frozen copy of the student's Base model receives the problem and a reference solution, while the student receives only the problem and learns from the teacher's probabilities on student-generated prefixes.

2.2 Capability and Distribution Measurements

For a task group $\mathcal{G}$, let $\mathrm{Acc}_{\mathcal{G}}$ be the equally weighted mean of its task accuracies. We measure learning by in-domain accuracy gain from Base and forgetting by OOD accuracy loss. To compare any two models p and q, we evaluate their next-token predictions on the same fixed prefixes, with ν_j denoting the prefix distribution for task j:

$$\begin{aligned} F_{\mathcal{G}}(p_\theta) &= \mathrm{Acc}_{\mathcal{G}}(p_0) - \mathrm{Acc}_{\mathcal{G}}(p_\theta), \\ D_{\mathcal{G}}(p,q) &= \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} \mathbb{E}_{s\sim\nu_j}\left[\mathrm{JS}\big(p(\cdot \mid s), q(\cdot \mid s)\big)\right]. \end{aligned} \tag{4}$$

$F_{\mathcal{G}}$ is expressed in percentage points; negative values indicate accuracy gains. We use natural logarithms and $\mathrm{JS}(p,q) = \frac{1}{2}\mathrm{KL}(p\|m) + \frac{1}{2}\mathrm{KL}(q\|m)$, where $m = (p+q)/2$. Distribution measurements use the full vocabulary and weight datasets equally. Prefix sampling is described in Appendix A.6.

3 Experiments

3.1 Experimental Setup

Models and methods. To study how post-training changes pretrained capabilities, we fine-tune five released Qwen Base models (Yang et al., 2024; 2025): Qwen2.5-1.5B, Qwen2.5-7B, Qwen3-1.7B, Qwen3-4B, and Qwen3-8B. We compare the SFT, RLVR, and OPD configurations defined in Section 2.1.

Training. We train the five Base models on OpenR1-50K for one epoch, following previous work (Huang et al., 2026; Li et al., 2026). The models being trained receive Raw prompts without chat templates, matching the Base models' text-completion format and keeping prompt formatting consistent across methods. Our Raw/Template baseline comparison supports this choice (Appendix A.1).

Table 1: Main results for four Base models: accuracy (%) with changes from Base in parentheses (percentage points). Negative OOD changes indicate forgetting. RLVR and Self-OPD (RLVR) combine in-domain gains with strong capability retention. Selected larger-teacher configurations are shown; complete five-model results are in Appendix B.

Method (Teacher)	**Qwen2.5-1.5B**		**Qwen2.5-7B**	
	In-domain	OOD	In-domain	OOD
Base	13.93	35.20	26.08	49.82
SFT	11.23 (−2.69)	33.78 (−1.42)	36.45 (+10.37)	44.93 (−4.89)
RLVR	18.79 (+4.86)	36.23 (+1.02)	31.63 (+5.55)	49.99 (+0.16)
OPD (Q2.5-7B)	15.50 (+1.57)	34.56 (−0.64)	–	–
OPD (Q3-8B)	15.82 (+1.89)	33.35 (−1.85)	23.69 (−2.39)	47.12 (−2.71)
OPD (Q3-14B)	17.07 (+3.14)	34.33 (−0.88)	25.82 (−0.26)	48.72 (−1.10)
Self-OPD (SFT)	10.33 (−3.60)	34.93 (−0.27)	28.52 (+2.44)	46.83 (−2.99)
Self-OPD (RLVR)	19.46 (+5.54)	36.32 (+1.11)	31.24 (+5.16)	49.83 (+0.00)

Method (Teacher)	**Qwen3-4B**		**Qwen3-8B**	
	In-domain	OOD	In-domain	OOD
Base	26.85	45.45	31.32	51.37
SFT	41.68 (+14.83)	42.56 (−2.90)	47.48 (+16.16)	47.79 (−3.58)
RLVR	35.23 (+8.39)	46.33 (+0.88)	45.21 (+13.89)	51.84 (+0.47)
OPD (Q3-8B)	28.07 (+1.22)	44.90 (−0.55)	–	–
Self-OPD (SFT)	35.44 (+8.59)	43.45 (−2.00)	39.23 (+7.90)	47.84 (−3.53)
Self-OPD (RLVR)	36.39 (+9.54)	45.80 (+0.35)	44.40 (+13.08)	51.76 (+0.39)

Evaluation. We evaluate in-domain mathematical reasoning on AMC23, AIME24, AIME25, HMMT25, and Math500. OOD evaluations use HellaSwag, ARC-Challenge, MMLU, IFEval, TriviaQA, BBH, and NQ-Open. For each group, we report the unweighted mean of the per-dataset accuracies. Full training and evaluation settings, including original-model baselines, are provided in Appendix A.

3.2 Main Results

Using the shared setup above, we systematically compare mathematical learning and OOD retention across the five Base models. Table 1 reports in-domain and OOD accuracy, averaged equally over datasets, with changes from Base in parentheses. Negative OOD changes indicate forgetting.

Forgetting varies with the method and teacher. RLVR and Self-OPD (RLVR) keep OOD accuracy close to Base across the five models while improving mathematical accuracy. SFT and Self-OPD (SFT) generally incur larger OOD losses. Thus, Self-OPD with SFT and RLVR teachers shows different retention patterns even though both variants learn from student-generated responses. Limited SFT gains and modest forgetting in the smallest models are consistent with prior observations of difficulty learning long reasoning traces (Li et al., 2025) and milder forgetting at smaller scales (Luo et al., 2025). OPSD likewise preserves OOD accuracy; its reduced mathematical accuracy may reflect Base teachers' difficulty in turning supplied solutions into useful supervision.

Training curves and individual-task results are in Appendix B; parameter and behavior measurements are in Appendix C. Section 4.3 examines learning and retention under weight scaling; Appendix D gives the complete sweeps.

4 ANALYSIS

The preceding experiments show that teacher sources lead to different learning and retention patterns under the same OPD objective. RLVR teachers support task gains with capability retention, whereas SFT teachers induce more forgetting; stronger teacher OOD capabilities also need not yield better student retention. We examine these differences through distribution change, teacher information sources, and the parameter and prediction trajectories through which students inherit teacher preferences.

4.1 DISTRIBUTION CHANGE AND FORGETTING

Motivated by analyses of learning and forgetting through supervision distributions (Shenfeld et al., 2025), we first compare in-domain and OOD prediction changes with forgetting. Using Equation (4), we write $D_{\text{in}} = D_{\text{in}}(p_0, p_t)$ and $D_{\text{OOD}} = D_{\text{OOD}}(p_0, p_t)$ for prediction changes from student Base p_0 to the current student p_t.

OOD prediction change is more consistently associated with forgetting. Figure 2 compares checkpoints across SFT, RLVR, and OPD configurations. OOD JS is positively correlated with forgetting for each model, with stronger associations for the larger students. The relationship with in-domain JS varies across models and is negative for the two smallest students (Appendix Figure 17).

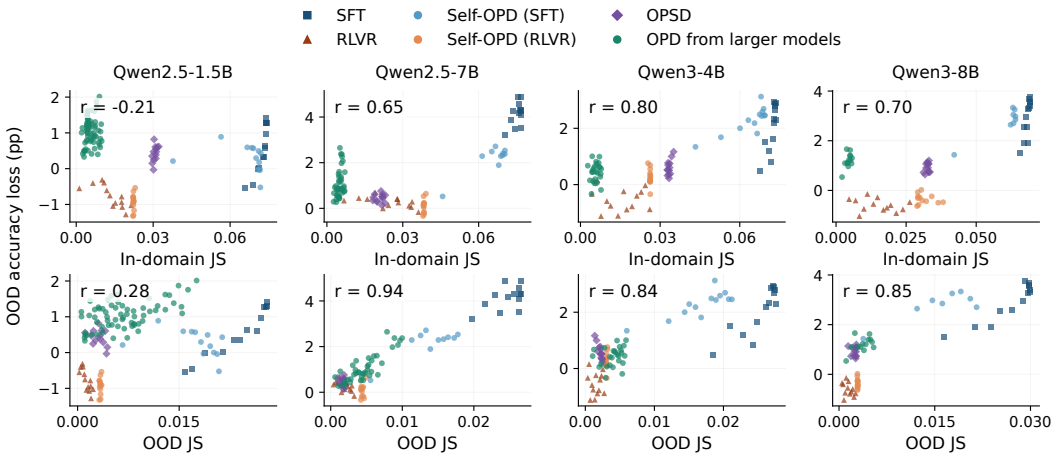


Figure 2: In-domain JS (top) and OOD JS (bottom) versus forgetting. Each point pairs distribution measurements with complete OOD accuracy at the same student checkpoint. The vertical axis is accuracy loss from Base; r is Pearson correlation. JS is in nats, with equal dataset weights within the five in-domain and seven OOD tasks. The complete five-model comparison is in Appendix Figure 17.

Relative OOD change measures update selectivity. This association motivates seeking task learning with less accompanying OOD change. Since methods also produce different amounts of in-domain change, we define

$$\rho = \frac{D_{\text{OOD}}}{D_{\text{in}}}, \qquad D_{\text{in}} > 0. \tag{5}$$

The ratio measures OOD change per unit of in-domain prediction change; lower ρ indicates changes more concentrated in the target domain.

In Figure 3, OPD from larger Base teachers produces small in-domain changes and high ρ, while post-trained teachers and supervision with reference solutions distribute changes differently despite sharing mathematical data and on-policy distillation. We next examine these differences through the additional information provided by each teacher.

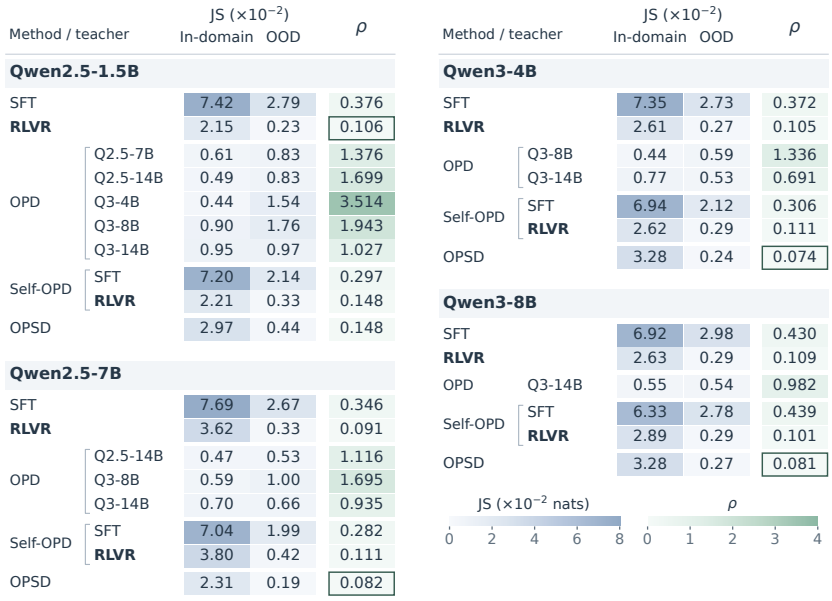


Figure 3: Prediction changes from student Base. The first two columns show in-domain and OOD JS in 10^{-2} nats; the last shows relative OOD influence ρ. Boxes mark the lowest ρ for each model.

4.2 Teacher Information Sources and Relative OOD Influence

To understand the high ρ of OPD from larger Base teachers, we examine the information teachers provide relative to student Base, using matched teacher comparisons to study its relationship with update selectivity.

Stronger teacher OOD capabilities can accompany greater student forgetting. Table 2 compares two teachers with similar mathematical accuracy for the same Qwen2.5-7B student. The Qwen3-8B Base teacher has higher OOD accuracy, yet its student obtains lower mathematical accuracy, a larger OOD decline, and higher ρ. This contrast motivates examining the prediction differences the teacher asks the student to fit and the information underlying them.

Teacher	Teacher accuracy (%)		Student accuracy (%)		
	Math	OOD	Math	OOD	ρ
Student Base			26.08	49.82	–
Qwen3-8B Base	31.32	51.37	23.69	47.12	1.695
Qwen2.5-7B RLVR	31.63	49.99	31.25	49.83	0.111

Table 2: Teacher capabilities and Qwen2.5-7B student outcomes. Accuracy averages datasets equally within each domain.

What additional information does each OPD method provide? We treat the teacher's prediction preferences relative to student Base as additional information. For larger Base teachers, this information comes from cross-model differences formed during pretraining. Self-OPD (SFT) adds preferences learned from offline demonstrations, while Self-OPD (RLVR) adds preferences learned through verifiable rewards on model-generated responses. OPSD supplies a reference solution through the teacher's context, keeping its weights identical to student Base (Section 2.1). Self-OPD (SFT) generally has lower ρ than OPD from larger Base teachers, but higher ρ than RLVR-based supervision and OPSD (Figure 3). Since both SFT and RLVR involve in-domain training, their difference further points to how information is acquired and which preferences it reinforces.

Information sources can affect the range of predictions influenced by feedback. We hypothesize that prediction differences formed by pretraining larger and smaller models span a broad range of behaviors, whereas preferences added by in-domain RLVR are more concentrated on the target task, leading to different update selectivity. Mathematical training data determine the prompts sampled by OPD, while teacher feedback can still reinforce preferences shared across tasks (Wang et al., 2026). Fitting these differences through shared parameters can therefore affect OOD predictions. Self-OPD removes cross-model pretraining differences through a shared Base, but its post-training preferences still differ: SFT fits complete demonstrations and may change both reasoning content and general response patterns, whereas RLVR supervision depends on the correctness of sampled responses.

Controlling teacher information to examine selectivity. To vary information sources at fixed model sizes, let p_0 be student Base, and q_B, q_R the Base and RLVR endpoint of the same larger model. The full RLVR teacher's preferences relative to student Base decompose as

$$\log \frac{q_R(v \mid s)}{p_0(v \mid s)} = \underbrace{\log \frac{q_B(v \mid s)}{p_0(v \mid s)}}_{\delta_{\mathrm{pre}}(v|s)} + \underbrace{\log \frac{q_R(v \mid s)}{q_B(v \mid s)}}_{\delta_{\mathrm{RL}}(v|s)}. \tag{6}$$

The two terms represent cross-model pretraining differences and the teacher's RLVR increment. Groups A and B use q_B and q_R, respectively. Following the idea of extracting log-probability increments (Heo et al., 2026), group C combines δ_{RL} with student Base:

$$q_C(v \mid s) = \frac{p_0(v \mid s)\, \exp\big[\delta_{\mathrm{RL}}(v \mid s)\big]}{Z(s)}. \tag{7}$$

Here $Z(s)$ normalizes the distribution. Group C tests whether student Base predictions combined with the teacher's RLVR information can transfer high in-domain selectivity. All groups use the same student initialization and ordinary OPD setup; construction details are in Appendix C.10. A/B changes the task-training information of the same larger teacher; B/C combines the same RLVR increment with either the larger teacher's or the student's Base predictions, testing the effect of cross-model pretraining differences.

RLVR information supports task learning with low relative OOD influence. Groups A/B in Table 3 have the same teacher and student sizes. After RLVR training of the larger teacher, both student divergences increase, but in-domain JS increases proportionally more, reducing ρ. Group C also produces larger in-domain changes and lower ρ. B/C further improve mathematical accuracy while maintaining OOD accuracy near or above student Base. At fixed model sizes, adding RLVR-derived information makes student prediction changes more concentrated in the target domain. This selectivity persists when the same information is combined with student Base, alongside substantial task gains.

Teacher supervision	Distribution change			Accuracy (%)	
	D_{in}	D_{OOD}	ρ	Math (Δ)	OOD (Δ)
Qwen2.5-7B→Qwen2.5-1.5B					
Student Base	–	–	–	13.93	35.20
A: Larger Base teacher	0.3696	0.1151	0.311	14.67 (+0.75)	34.82 (−0.39)
B: Larger RLVR teacher	4.1650	0.6873	0.165	19.17 (+5.25)	35.99 (+0.79)
C: Constructed teacher	4.0247	0.5658	0.141	19.22 (+5.29)	35.76 (+0.56)
Qwen3-8B→Qwen3-4B					
Student Base	–	–	–	26.85	45.45
A: Larger Base teacher	0.3556	0.2286	0.643	27.85 (+1.01)	45.76 (+0.31)
B: Larger RLVR teacher	3.1320	0.4312	0.138	39.34 (+12.49)	45.22 (−0.23)
C: Constructed teacher	2.8076	0.3995	0.142	41.54 (+14.70)	45.27 (−0.19)

Table 3: Student distribution changes and capabilities under different information sources. All A/B/C comparisons use student step 120. JS uses shared prefixes, is relative to Base, and is reported in 10^{-2} nats. Parentheses show accuracy changes from Base in percentage points. Datasets are weighted equally within each domain; ρ is undefined at Base.

Information sources also induce different local update directions. After extracting directions on the same student prefixes and calibrating their in-domain change, B/C still yield lower relative OOD influence than A on independent evaluation prefixes. This supports a difference in the cross-domain effects of directions induced by different feedback sources. The procedure and local JS analysis are in Appendix E.2; complete training curves are in Appendices C.8 and C.7.

Self-OPD also reproduces learning and retention patterns of the training that produced its teachers. We next test whether this functional similarity corresponds to similar parameter paths from a shared Base.

4.3 Parameter and Prediction Trajectories Behind Similar Behavior

Self-OPD reproduces source-dependent learning and retention patterns. We pair Self-OPD with the SFT/RLVR training that produced its frozen teacher, starting from the same Base. Students of RLVR teachers approach teacher task performance early with similar capability retention. Larger SFT students incur OOD declines approaching their teachers' while still lagging in task learning (Appendix C.3). We examine whether these shared behavioral patterns correspond to similar parameter paths.

Models along parameter lines provide a capability reference. One possible explanation is that students lie approximately on the parameter line between Base and teacher. We first interpolate and extrapolate the SFT, RLVR, and Self-OPD endpoints (Huang et al., 2026) to examine changes in update magnitude:

$$\theta_\lambda = \theta_0 + \lambda(\theta_{\text{end}} - \theta_0). \tag{8}$$

The source SFT/RLVR endpoints are the frozen teachers, so their sweeps provide a capability reference along the Base–teacher line; Self-OPD sweeps examine the student's own update direction.

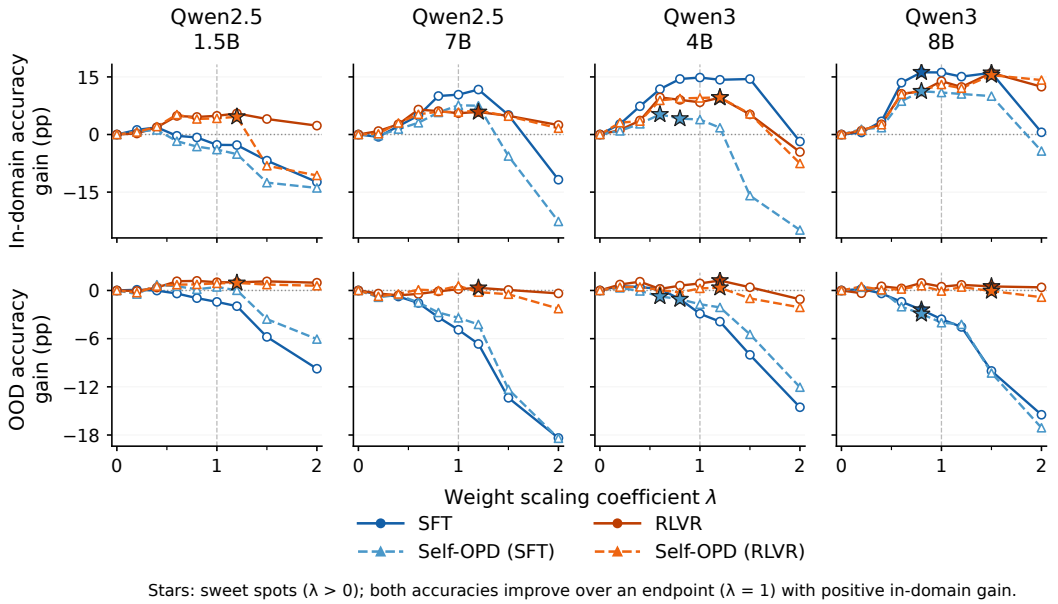


Figure 4: Accuracy changes from Base under weight scaling. Stars mark sweet-spot coefficients that improve both accuracies over the original endpoint, as defined in Appendix D.

Figure 4 shows broader ranges of retained task gains for RLVR-based updates, with a more consistent advantage in OOD retention: accuracy remains near Base even for $\lambda > 1$, while forgetting grows faster along SFT-based directions. Sweet spots occur in interpolation for SFT-based updates and extrapolation for RLVR-based updates. We next locate actual students relative to the Base–teacher line.

Parameter and prediction distances evolve differently. We measure parameter distances to the teacher, d_θ, and to the Base–teacher line, $d_\perp$, normalized by the initial parameter gap (Appendix C.2). Prediction distance to the frozen teacher q_T is likewise normalized by its initial value:

$$d_{\text{out},d}(t) = \frac{D_d(p_t, q_T)}{D_d(p_0, q_T)}, \qquad d \in \{\text{in}, \text{OOD}\}. \tag{9}$$

In Figure 5, the RLVR-teacher student approaches teacher predictions while remaining far from the teacher's parameters and the Base–teacher line. Both sources reduce OOD prediction gaps, while SFT-teacher students retain larger in-domain gaps. Teacher-like behavior thus emerges through different parameter paths, with task gains and forgetting inherited at different rates.

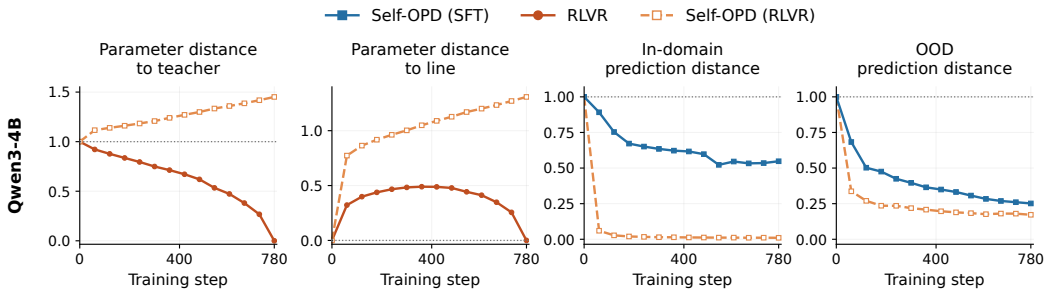


Figure 5: Qwen3-4B parameter distances for RLVR and its Self-OPD student (left) and student–teacher JS for SFT/RLVR teachers (right). All distances are normalized by the initial Base–teacher gap.

Student probabilities weight teacher feedback. Let d_{p_t} be the training-prefix distribution and $A_t(v \mid s) = \log q_T(v \mid s) - \log p_t(v \mid s)$ the feedback. For a full-vocabulary score-function model with fixed sampling distributions and stop-gradient feedback, the mean gradient is

$$\begin{aligned} \mu_t(v \mid s) &= p_t(v \mid s)\big(\overline{A}_t(s) - A_t(v \mid s)\big), \\ \mathbb{E}[\nabla_\theta \mathcal{L}] &= \mathbb{E}_{s \sim d_{p_t}}\left[\sum_v \mu_t(v \mid s) \nabla_\theta z_t(v \mid s)\right], \end{aligned} \tag{10}$$

where $\overline{A}_t(s) = \mathbb{E}_{a \sim p_t}[A_t(a \mid s)]$ and z_t denotes student logits. Student prefix and token probabilities weight teacher feedback; these weights and the logit Jacobians change along the student's path (Appendix E.1).

Prefix accessibility is associated with inheritance rates. RLVR-teacher responses have higher probability under the shared Base; this source difference persists during student training (Appendix C.4). We hypothesize that some SFT task preferences depend on less accessible prefixes, slowing task transfer relative to OOD changes, consistent with supervision-distribution accounts of learning and forgetting (Shenfeld et al., 2025; Zhang et al., 2026). An opening-token intervention provides further evidence of impaired task transfer from SFT response-opening preferences (Appendix C.5).

5 Conclusion

OPDArena compares learning and capability retention across post-training objectives and teacher sources. OOD distribution change correlates more consistently with forgetting than in-domain change. Matched teacher comparisons connect RLVR-acquired information to more selective updates and stronger task learning with limited forgetting. Teacher preferences can be inherited through different parameter paths; prefix accessibility helps explain why task gains and forgetting transfer at different rates. Weight scaling reveals broader retention ranges for RLVR-based updates. These results motivate selecting teachers by the information they add relative to student Base and its interaction with student predictions, alongside teacher capabilities.

AI USE STATEMENT

We used LLM-based assistants, including OpenAI Codex, for writing, literature discovery, research ideation and execution, manuscript drafting, and mathematical analysis. For writing, these tools helped organize the paper, draft sections, translate between Chinese and English, and revise and polish the text. For literature discovery, they helped search for, identify, and summarize related work.

For research ideation and execution, the tools helped develop and refine hypotheses and conceptual explanations, design experiments and diagnostic comparisons, implement methods and evaluation code, assist with experiment execution, prepare and visualize analysis results, and interpret findings. For mathematical analysis, they helped formulate claims, propose proof arguments, and derive, check, and write the gradients and local distribution-change analysis in Section 4 and Appendix E. The authors take responsibility for the final text, references, mathematical claims and proofs, code, analyses, and reported results, including content developed with AI assistance.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2306.13649.

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

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge Distillation of Large Language Models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2306.08543.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021b. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Byeongho Heo, Jaehui Hwang, Sangdoo Yun, and Dongyoon Han. On-Policy Delta Distillation. *arXiv preprint arXiv:2607.15161*, 2026. URL https://arxiv.org/abs/2607.15161.

Yangyi Huang, Ruotian Peng, Zeju Qiu, Jiale Kang, Yandong Wen, Bernhard Schölkopf, and Weiyang Liu. PEFT-Arena: Understanding Parameter-Efficient Finetuning from a Stability-Plasticity Perspective. *arXiv preprint arXiv:2605.28819*, 2026. URL https://arxiv.org/abs/2605.28819.

Hugging Face. Open R1: A Fully Open Reproduction of DeepSeek-R1, 2025. URL https://github.com/huggingface/open-r1.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147/.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav

Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6086–6096, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1612. URL https://aclanthology.org/P19-1612/.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang 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. URL https://arxiv.org/abs/2604.13016.

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 *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25366–25394, 2025. doi: 10.18653/v1/2025.findings-acl.1301. URL https://aclanthology.org/2025.findings-acl.1301/.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. URL https://arxiv.org/abs/2305.20050.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-Tuning. *IEEE Transactions on Audio, Speech and Language Processing*, 33:3776–3786, 2025. doi: 10.1109/TASLPRO.2025.3606231. URL https://doi.org/10.1109/TASLPRO.2025.3606231.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL's Razor: Why Online Reinforcement Learning Forgets Less. *arXiv preprint arXiv:2509.04259*, 2025. URL https://arxiv.org/abs/2509.04259.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.824. URL https://aclanthology.org/2023.findings-acl.824/.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A Comprehensive Survey of Continual Learning: Theory, Method and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329. URL https://doi.org/10.1109/TPAMI.2024.3367329.

Meng Wang, Haohan Zhao, Wenzhuo Liu, Lu Yang, Geng Liu, Haiyang Guo, Guo-Sen Xie, Gaofeng Meng, Hongbin Liu, and Fei Zhu. Denser $\neq$ Better: Limits of On-Policy Self-Distillation for Continual Post-Training. *arXiv preprint arXiv:2607.01763*, 2026. URL https://arxiv.org/abs/2607.01763.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 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.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-Policy Context Distillation for Language Models. *arXiv preprint arXiv:2602.12275*, 2026. URL https://arxiv.org/abs/2602.12275.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Zhihao Zhang, Qiaole Dong, Qi Zhang, Enyu Zhou, Jun Zhao, Zhiheng Xi, Senjie Jin, Xiaoran Fan, Yuhao Zhou, Mingqi Wu, Yanwei Fu, Tao Ji, Tao Gui, Xuanjing Huang, and Kai Chen. Why Reinforcement Fine-Tuning Preserves Prior Knowledge Better: A Data Perspective. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2506.23508.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. *arXiv preprint arXiv:2601.18734*, 2026. URL https://arxiv.org/abs/2601.18734.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Experimental Details

A.1 Base-Model Baselines

Tables 4 and 6 report standalone in-domain and OOD evaluations of the released Base models. Table 4 compares Raw inputs with the corresponding Qwen chat template; Table 5 reports the corresponding output lengths. Qwen's documentation describes Base releases as pretrained models that have not learned the predefined chat template.[1] Raw input yields higher mean in-domain accuracy than Template for every model in the paired comparison, supporting the use of the Base models' text-completion format. We use Raw input for the models being trained and their evaluations. Accuracy values are percentages. For in-domain and OOD, training curves use a shared original-model baseline for each model and input configuration at the start of training.

For in-domain evaluation of both the original models and training checkpoints, we sample eight responses per problem at temperature 0.7 and top-p 0.95. The input prompt is limited to 1,024 tokens. For AMC23, AIME24, AIME25, and HMMT25, each response is limited to 31,744 tokens, within a total context budget of 32,768 tokens. Math500 uses all 500 test problems with a response limit of 16,384 tokens and a total context budget of 17,408 tokens. Its evaluations use the Base-model Raw and Template configurations in Table 4, as well as every available saved checkpoint of the five Base models' Raw training runs. The Raw/Raw OPD runs use four Math500 responses per problem; the other mathematical datasets retain eight responses. We report acc@8 as the fraction of correct responses averaged over the eight samples and all problems in each dataset. Dataset sizes and scoring rules are listed in Appendix A.5.

Table 4: In-domain acc@8 (%) of the released Base models. The Mean column weights the five datasets equally. Raw and Template denote inputs without and with the corresponding Qwen chat template.

Model	Input	AMC23	AIME24	AIME25	HMMT25	Math500	Mean
Qwen2.5-1.5B Base	Raw	20.00	1.25	0.83	0.42	47.13	13.93
Qwen2.5-1.5B Base	Template	1.88	0.42	0.00	0.00	4.58	1.37
Qwen2.5-7B Base	Raw	46.88	9.17	5.00	2.08	67.28	26.08
Qwen2.5-7B Base	Template	31.25	6.67	3.33	0.42	57.03	19.74
Qwen2.5-14B Base	Raw	51.25	10.42	7.08	2.50	73.23	28.90
Qwen3-1.7B Base	Raw	32.50	4.17	4.58	0.00	59.58	20.17
Qwen3-1.7B Base	Template	14.38	2.08	0.42	0.42	25.75	8.61
Qwen3-4B Base	Raw	45.00	9.58	6.25	1.67	71.73	26.85
Qwen3-4B Base	Template	30.00	8.75	4.17	1.67	37.13	16.34
Qwen3-8B Base	Raw	57.50	12.08	11.25	0.00	75.78	31.32
Qwen3-8B Base	Template	48.75	12.08	10.00	2.50	67.50	28.17
Qwen3-14B Base	Raw	60.00	14.58	12.08	2.50	79.43	33.72

A.2 Training Input Formats

All methods train on OpenR1-50K.

Rollout input format. All five Base models use Raw input for their rollout-based methods. The model receives only the original problem text:

```
{ORIGINAL_PROBLEM}
```

No system message, user/assistant delimiters, chat generation prefix, or empty thinking block is inserted. Student rollouts and in-domain validation use the same Raw protocol. For Self-OPD and OPD from larger models, teachers also use Raw input when scoring student-generated responses.

[1] Qwen documentation, "Key Concepts": https://qwen.readthedocs.io/en/latest/getting_started/concepts.html.

Table 5: Mean in-domain output length in tokens for the same Base-model evaluations. The Mean column weights the five datasets equally. Math500 uses a 16,384-token response limit; the other datasets use 31,744 tokens.

Model	Input	AMC23	AIME24	AIME25	HMMT25	Math500	Mean
Qwen2.5-1.5B Base	Raw	2359.4	5101.0	3827.6	3450.3	1074.3	3162.5
Qwen2.5-1.5B Base	Template	25725.3	25467.4	25473.3	22321.9	13228.8	22443.4
Qwen2.5-7B Base	Raw	1160.0	3131.1	3121.3	2364.9	787.3	2112.9
Qwen2.5-7B Base	Template	1541.7	1985.4	2171.4	2038.7	696.8	1686.8
Qwen2.5-14B Base	Raw	912.9	1214.1	1483.7	1317.1	614.7	1108.5
Qwen3-1.7B Base	Raw	1807.0	5101.9	4117.2	4692.0	1073.2	3358.3
Qwen3-1.7B Base	Template	11554.2	14284.5	14597.9	12014.1	5884.6	11667.1
Qwen3-4B Base	Raw	1399.7	2901.5	1803.1	1610.1	719.4	1686.8
Qwen3-4B Base	Template	11628.7	11135.5	11435.0	10028.8	6385.6	10122.7
Qwen3-8B Base	Raw	1082.4	2600.4	2191.7	1612.8	711.8	1639.8
Qwen3-8B Base	Template	2077.9	4310.8	4274.0	3950.0	1068.1	3136.2
Qwen3-14B Base	Raw	1205.0	2291.0	1386.7	1060.6	663.0	1321.2

Table 6: OOD accuracy (%) of the released Base models with Raw input. The macro is an unweighted mean over the seven datasets. HellaSwag, ARC-Challenge, and MMLU use candidate likelihood; the other tasks score one generated response. Response budgets are listed in Table 8.

Model	HellaSwag	ARC-C	MMLU	IFEval	TriviaQA	BBH	NQ-Open	Macro
Qwen2.5-1.5B Base	49.22	39.45	61.40	20.70	22.00	42.59	11.05	35.20
Qwen2.5-7B Base	61.33	44.53	74.21	31.79	44.40	72.96	19.53	49.82
Qwen2.5-14B Base	62.11	50.78	78.07	31.79	55.20	79.63	–	–
Qwen3-1.7B Base	46.09	37.11	65.26	24.03	20.60	50.37	9.17	36.09
Qwen3-4B Base	53.13	41.02	75.09	31.98	34.00	66.67	16.29	45.45
Qwen3-8B Base	59.77	44.53	76.49	37.52	47.20	72.59	21.50	51.37
Qwen3-14B Base	61.72	48.44	80.70	43.25	54.80	79.26	–	–

SFT input format. Raw SFT tokenizes the original problem as the prompt and its complete worked solution as the target, without chat-control tokens. An end-of-sequence token is appended to the solution. Cross-entropy loss is applied only to solution and end-of-sequence tokens; prompt tokens are masked. This convention applies to both Qwen2.5 and Qwen3 Base students.

OPSD teacher augmentation. The teacher receives the original problem with the following private block; the student receives only the original user content:

```
Here is a privileged reference solution to the original problem:
=== Reference Solution Begin ===

{FULL_REFERENCE_SOLUTION}

=== Reference Solution End ===

Use the reference solution as private guidance. Independently solve the
original problem, and produce the answer in the format requested in the
original prompt.
```

For OPSD, the problem and private block are wrapped in the teacher's chat template, while the student continues to receive the Raw problem. The assistant-start marker separates the reference material from the student response being scored. This follows the original OPSD prompt's use of an explicit transition to a new answer after the reference solution (Zhao et al., 2026); the teacher only scores student responses. Appendix C.6 reports the corresponding answer-boundary diagnostics.

A.3 Evaluation Input Formats

In-domain. Raw tokenizes the dataset user content directly without chat control tokens. For example, the complete Raw input for the first AIME25 validation item is:

```
Find the sum of all integer bases $b>9$ for which $17_{b}$ is a divisor
of $97_{b}$. Please reason step by step, and put your final answer
within \boxed{}.
```

The in-domain evaluations along the training curves use this Raw format. Standalone baseline input modes are identified in the baseline tables.

OOD. All seven OOD tasks use task-native Raw input without a chat template.

HellaSwag scores candidate endings appended to the activity and context. ARC-Challenge scores each answer after the question and choices. MMLU uses zero-shot subject prompts and scores answer letters. IFEval checks the official instruction constraints; TriviaQA uses flexible answer containment. BBH uses three chain-of-thought examples and exact-matches the extracted final answer, stopping at EOS or a newline followed by `Q:`; paragraph breaks are allowed. We extract the last explicit conclusion, accepting capitalization, colon, "correct/final answer," and boxed-answer variants. For multiple-choice tasks we retain the option label; for other tasks we extract the short answer in the required format, removing quotes and accompanying explanations. Otherwise, we use the text extracted after the first literal "the answer is" cue.

A.4 Training Settings

Table 7 summarizes the training configurations. We use separate learning rates for SFT and on-policy methods, following prior training setups (Li et al., 2026; Huang et al., 2026). We train all methods on OpenR1-50K, a subset of the Open R1 data (Hugging Face, 2025), and evaluate on five in-domain and seven OOD tasks, sampling eight responses per in-domain problem.

Our student Top-16 setting follows the ablation of Li et al. (2026): increasing k beyond 4 brings negligible additional accuracy gains, while Top-16 and Top-64 maintain smooth training. Top-16 retains this stability with lower candidate-processing overhead than Top-64.

For RLVR, correctness is computed by the mathematical rule scorer and GRPO advantages are formed within each eight-response group. For OPD and OPSD, we evaluate the teacher on student-generated

Table 7: Current OpenR1-50K training settings. OPD, Self-OPD, and OPSD share the listed optimization settings; the teacher input or token signal differs.

Setting	SFT	RLVR	OPD variants / OPSD
Train records	49,993	49,993	49,993
Epochs	1	1	1
Global batch	64	64	64
Learning rate	10^{-5}	10^{-6}	10^{-6}
Optimizer	AdamW	AdamW	AdamW
Adam (β_1, β_2)	$(0.9, 0.999)$	$(0.9, 0.999)$	$(0.9, 0.999)$
Adam ϵ	10^{-8}	10^{-8}	10^{-8}
Weight decay	0	0.01	0.01
Global gradient-norm clip	1.0	1.0	1.0
Schedule	cosine; 5% warmup	constant; no warmup	constant; no warmup
Training precision	BF16	FP32 FSDP model dtype	FP32 FSDP model dtype
Sequence limits	14,336 total	1,024 prompt; 7,168 response	1,024 prompt; 7,168 response
Train responses per prompt	–	8	4 student responses
Train sampling	–	temp. 1.0; top-p 1.0	student temp. 1.0; top-p 1.0
Objective	assistant-token CE	token-mean GRPO; KL 0	token-mean Top-16 teacher signal; KL 0
Teacher support	–	–	student top-16; student-p weighting
Validation / save interval	20 / 20 steps	20 / 20 steps	20 / 20 steps
Hardware	8 A100 GPUs	8 A100 GPUs	8 A100 GPUs

prefixes and use the student's Top-16 candidate tokens. Student probabilities normalized on that support weight the teacher–student log-probability difference; both log probabilities are evaluated under the full-vocabulary distributions. The prepared weights and coefficients are held fixed during the actor's probability-ratio update. Self-OPD uses the final saved checkpoints of the corresponding SFT and RLVR training; teacher parameters remain frozen throughout distillation.

Each configuration uses a single training seed. For Qwen3-4B, the main RLVR results use seed 43, with the corresponding Self-OPD student using that teacher. The capability, parameter, and behavior study in Appendix C uses the seed-42 teacher and its matched students.

A.5 Validation Data and Scoring

AMC23 and AIME24/AIME25 use problems from the MAA mathematics competitions,[2] and HMMT25 uses the February HMMT competition.[3] Math500 is the evaluation subset of MATH (Hendrycks et al., 2021b) released by Lightman et al. (2023). The OOD suite comprises HellaSwag (Zellers et al., 2019), ARC-Challenge (Clark et al., 2018), MMLU (Hendrycks et al., 2021a), IFEval (Zhou et al., 2023), TriviaQA (Joshi et al., 2017), BBH (Suzgun et al., 2023), and the open-domain split of Natural Questions (Kwiatkowski et al., 2019; Lee et al., 2019). We compute each group mean only when all five in-domain or all seven OOD dataset accuracies are available, weighting datasets equally.

For IFEval, we retain the first 512 tokens of each generated response and apply the official prompt-level strict checks. This budget is shared by all methods, checkpoints, and original-model baselines, including the weight interpolation and extrapolation experiments. Answer lengths are measured on the retained text. All four generative OOD tasks sample one response per problem at temperature 0.7 and top-p 0.95, with the budgets in Table 8. HellaSwag, ARC-Challenge, and MMLU score the provided candidate answers without generating a response.

IFEval and NQ-Open use their full evaluation splits. The other OOD tasks use fixed subsets, with per-subject or per-task sampling where specified in Table 8. NQ-Open uses five fixed question–answer examples from its training split. Generation stops at a newline, period, or comma. We report exact match against any official answer alias after lowercasing, removing punctuation and English articles, and normalizing whitespace.

Math evaluation prompts end with "Please reason step by step, and put your final answer within `\boxed{}`." OpenR1-50K training prompts end with "Let's think step by step and output the final answer within `\boxed{}`."

[2] https://maa.org/math-competitions/

[3] https://www.hmmt.org/www/archive/

Table 8: Fixed validation suite and response budgets in tokens. OOD category subsamples use seed 42 and are reused at every checkpoint. A dash denotes likelihood scoring without response generation. The IFEval budget applies to the response prefix used for scoring.

Domain	Dataset	Evaluation	Examples	Response budget (tokens)	Primary score
In-domain	AMC23	eight generated responses	40	31,744	acc@8
	AIME24	eight generated responses	30	31,744	acc@8
	AIME25	eight generated responses	30	31,744	acc@8
	HMMT25	eight generated responses	30	31,744	acc@8
	Math500	eight generated responses	500	16,384	acc@8
OOD	HellaSwag	candidate likelihood	256	–	normalized accuracy
	ARC-Challenge	candidate likelihood	256	–	normalized accuracy
	MMLU	candidate likelihood; 10 per subject	570	–	accuracy
	IFEval	generation	541	512	prompt-strict accuracy
	TriviaQA	generation	500	256	flexible answer containment
	BBH	three-shot chain-of-thought; 10 per task	270	1,024	final-answer exact match
	NQ-Open	five-shot generation	3,610	256	normalized exact match

A.6 Predictive-Distribution Measurements

For each Base student, we construct a fixed prefix bank for all twelve datasets: five in-domain mathematics benchmarks and seven OOD benchmarks. We select 16 prompts per dataset and continue each with Base and an RLVR checkpoint for up to 128 tokens. Up to 16 positions per continuation are selected; the prompt and preceding response tokens at each position form a text prefix. All compared methods are evaluated on these same prefixes.

At each prefix, we compute JS divergence between full-vocabulary next-token distributions, using natural logarithms. We average over the fixed prefixes within each dataset and then weight datasets equally within each task group, as defined in Section 2.2. Base comparisons measure the change from initialization; comparisons with the respective SFT or RLVR teacher measure the distance between Self-OPD students and their teachers. The in-domain and OOD JS means require measurements for all five and all seven tasks, respectively.

For each model and method, Table 1 and Figure 1 report step-780 results. The table gives accuracies and changes from Base; the figure pairs in-domain gains with OOD accuracy gains. The overall forgetting hierarchy weights models equally, averaging larger-teacher configurations within each model before averaging across models. Figure 3 reports step-780 JS. Main-run distribution measurements are taken at saved checkpoints at intervals of 60 steps; correlations pair JS and accuracy at the same checkpoint.

B Complete Raw Results

Figure 6 gives the training trajectories for all methods and teacher configurations, using the same task groups and Base references as Table 1. OPD and Self-OPD use Raw teachers; OPSD retains its Template teacher.

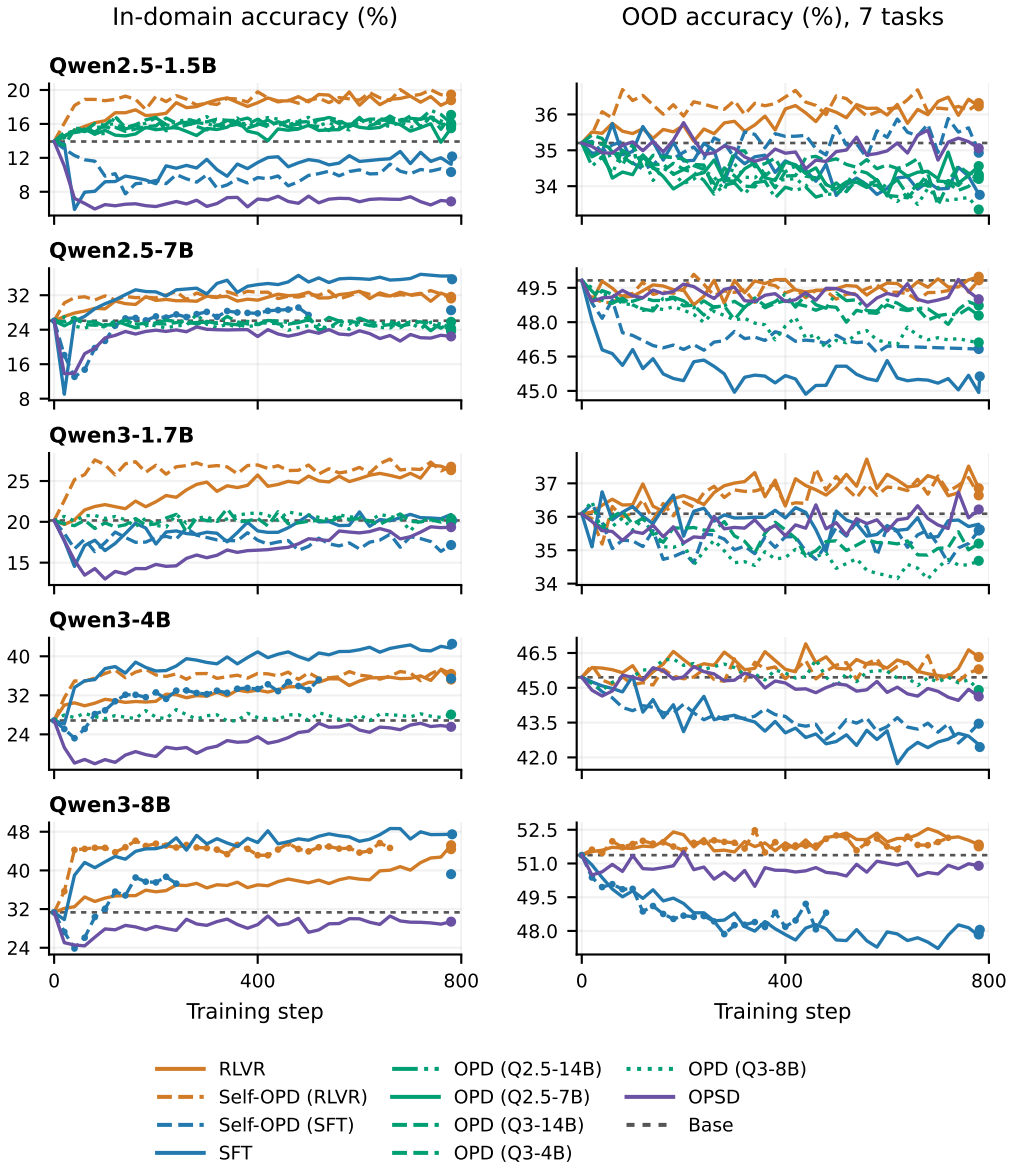


Figure 6: In-domain learning and OOD retention throughout training. Curves show task-group mean accuracy; dashed horizontal lines mark Base. RLVR and Self-OPD (RLVR) combine in-domain learning with sustained capability retention.

Each figure below summarizes one Base student, combining in-domain and OOD evaluations. Panels show the five mathematical and seven OOD tasks; aggregate accuracies appear in Figure 6. Colors and line styles follow the overall method comparison.

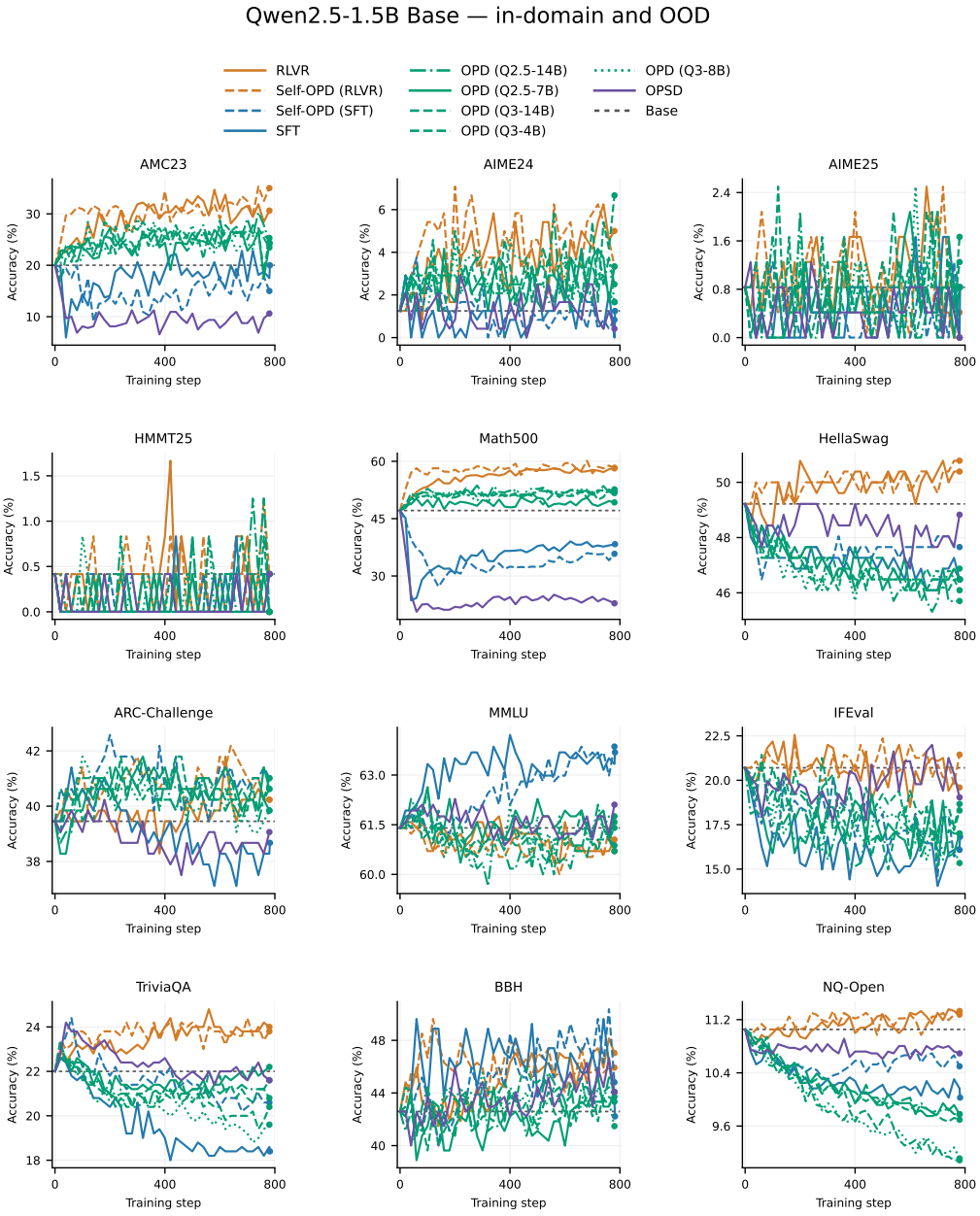


Figure 7: Task accuracies during training for Qwen2.5-1.5B Base.

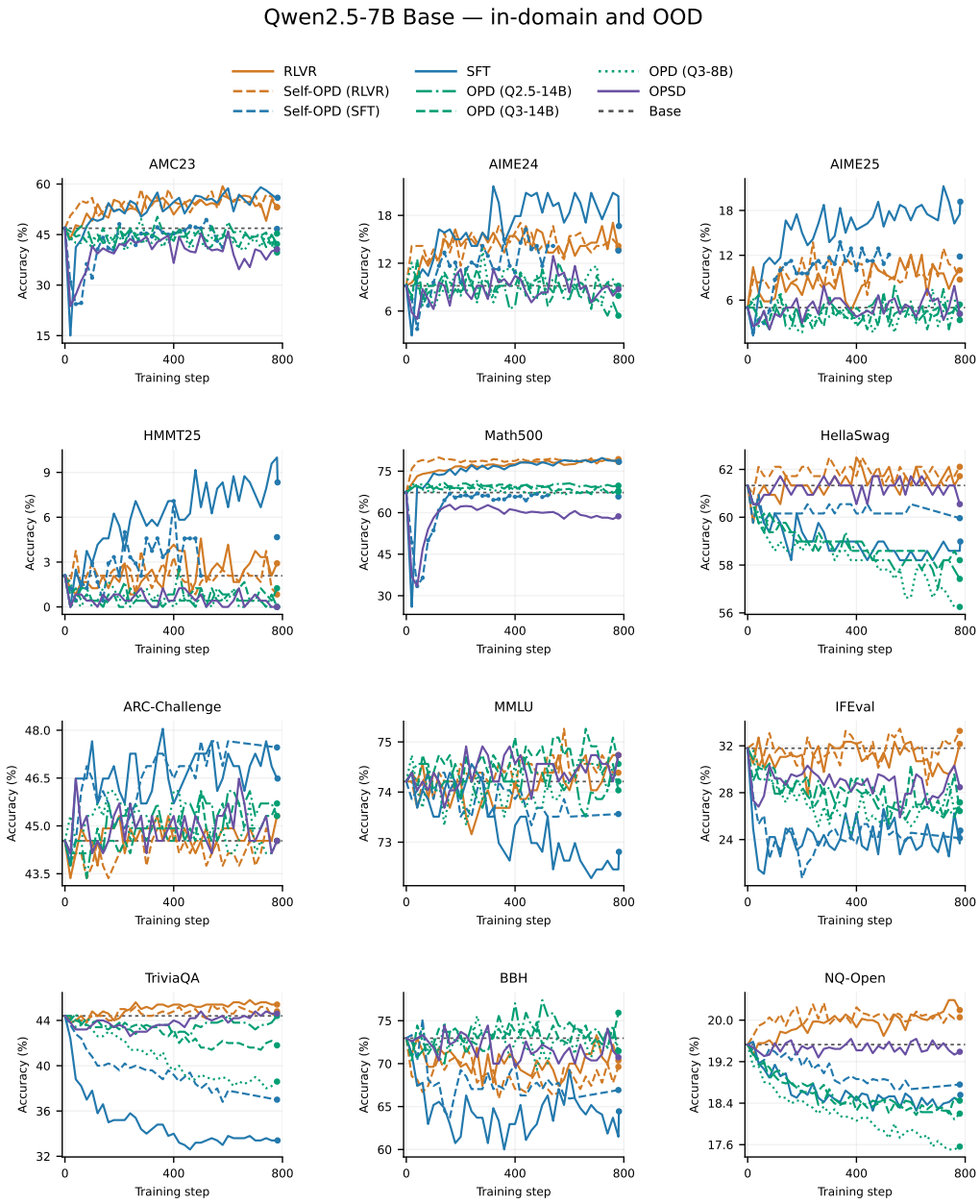


Figure 8: Task accuracies during training for Qwen2.5-7B Base.

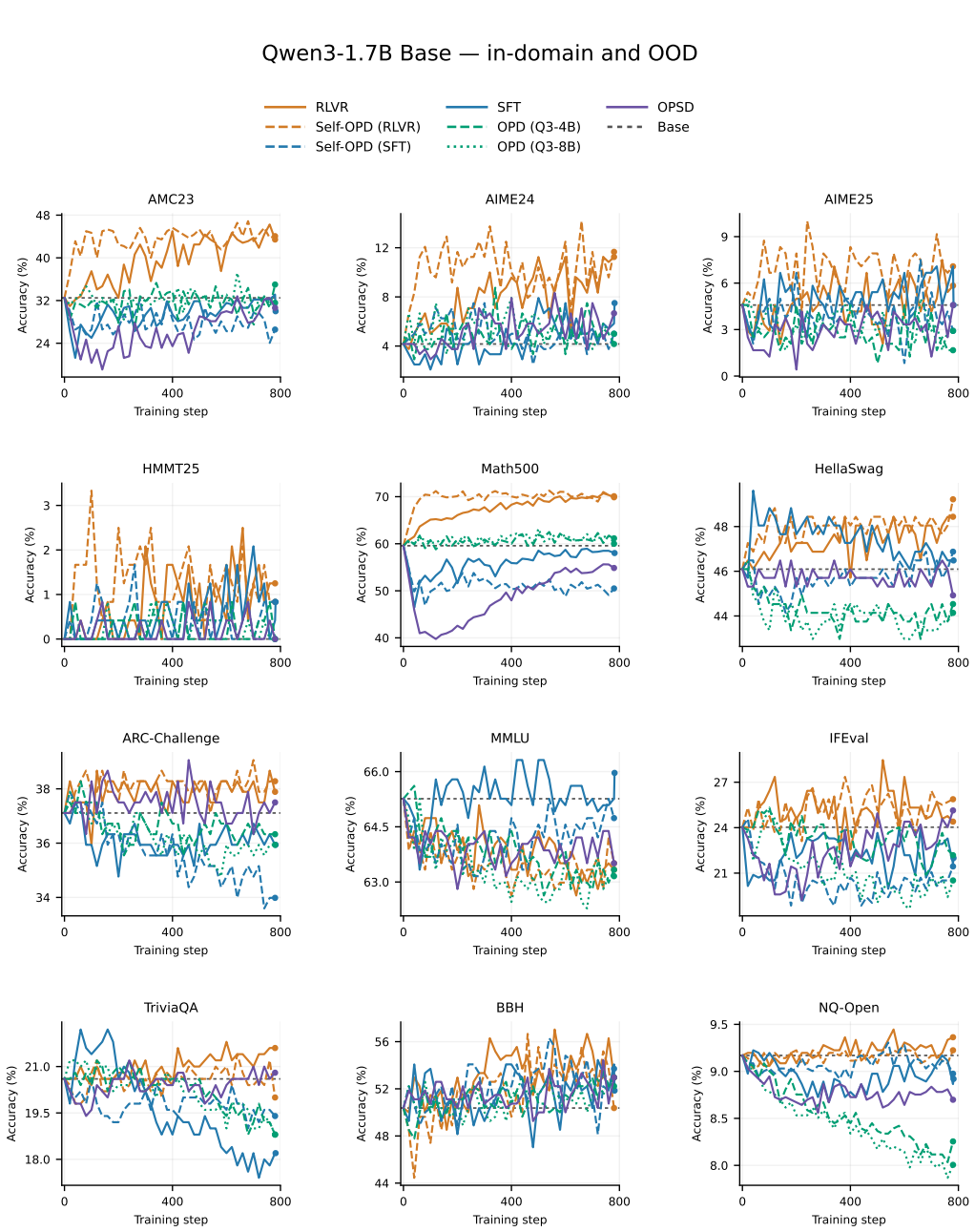


Figure 9: Task accuracies during training for Qwen3-1.7B Base.

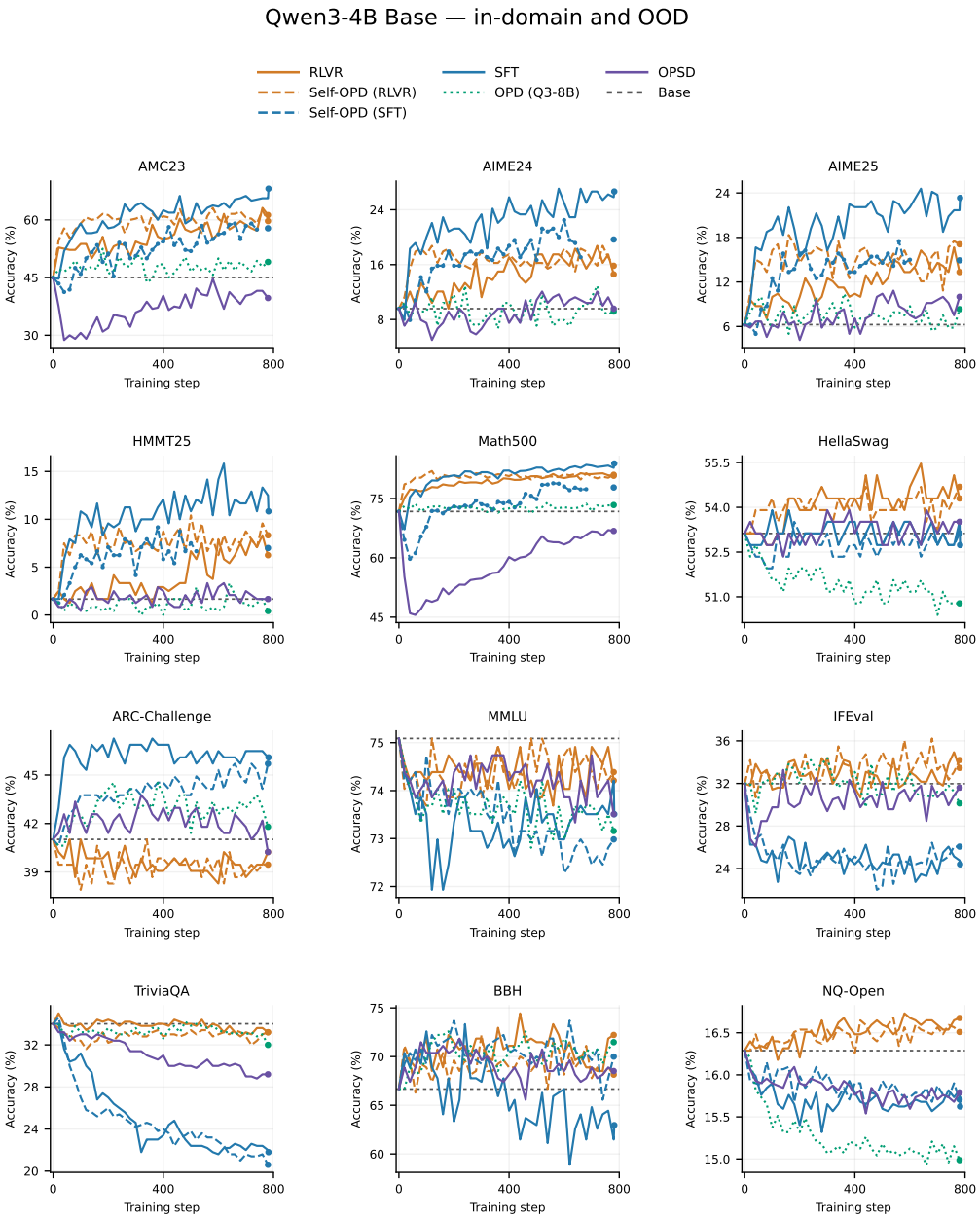


Figure 10: Task accuracies during training for Qwen3-4B Base.

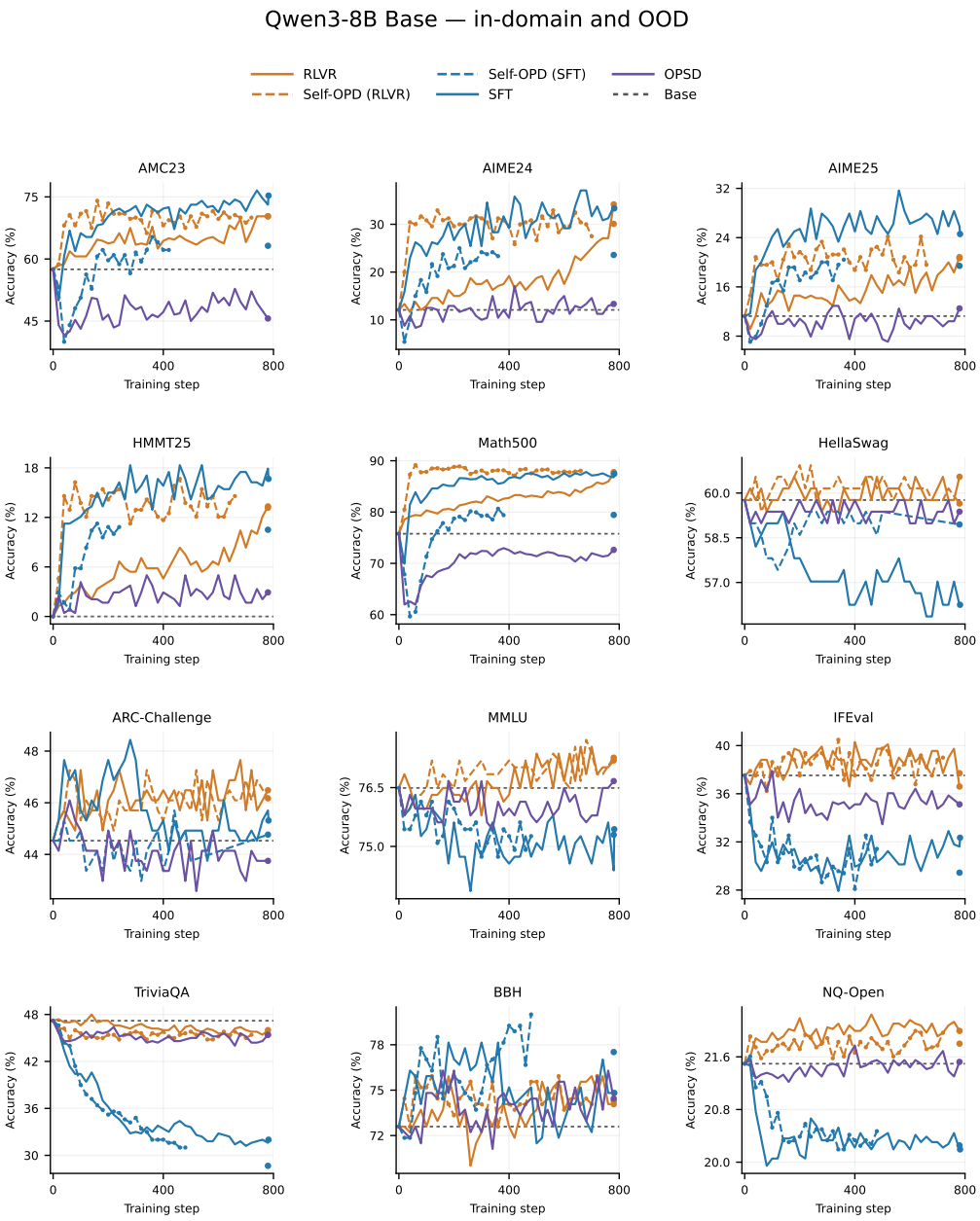


Figure 11: Task accuracies during training for Qwen3-8B Base.

C Supplementary Measurements for Analysis

C.1 Student–Teacher Prediction Gaps

Let B, T, S_t denote student Base, the frozen teacher, and the current student. The right panels of Figure 5 track normalized prediction distances. We express the reduction in the initial prediction gap as

$$G_d(t) = 1 - \frac{D_d(S_t, T)}{D_d(B, T)}, \qquad d \in \{\mathrm{in}, \mathrm{OOD}\}. \tag{11}$$

Higher G_d means closer teacher predictions relative to the initial gap. The complete trajectories below report the remaining fraction $1 - G_d$; changes from Base are separately measured by $D_d(B, S_t)$.

On shared evaluation prefixes, the square root of JS satisfies the triangle inequality, giving

$$D_{\mathrm{in}}(B, S_t) \geq \left[\sqrt{D_{\mathrm{in}}(B, T)} - \sqrt{D_{\mathrm{in}}(S_t, T)}\right]_+^2. \tag{12}$$

A teacher far from Base and a student close to that teacher therefore imply substantial student change. The bound also holds for mean conditional JS: the joint distribution $P_X(s, v) = w_s p_X(v \mid s)$ uses the same prefix weights for all models, and its JS equals the weighted mean of conditional JS.

C.2 Parameter and Prediction Trajectories

Figure 5 presents Qwen3-4B. Figures 12 and 13 give the complete four-model results. Parameter and prediction measurements each use the fixed student–teacher pairs in their corresponding studies. Parameter trajectories use the original Raw/Template Self-OPD runs. They compare RLVR training and its Self-OPD student with the same architecture and Base initialization, using Euclidean distances across all model parameters normalized by the Base–teacher endpoint distance. Let $d_T = \theta_T - \theta_0$ and $d_S(t) = \theta_{S_t} - \theta_0$ be the teacher and student displacements. The normalized distances are

$$d_\theta(t) = \frac{\|\theta_{S_t} - \theta_T\|_2}{\|d_T\|_2}, \qquad d_\perp(t) = \frac{\|d_S(t) - \alpha_t d_T\|_2}{\|d_T\|_2}, \quad \alpha_t = \frac{\langle d_S(t), d_T\rangle}{\|d_T\|_2^2}. \tag{13}$$

Here α_t projects the student displacement onto the teacher direction; $d_\perp$ includes the residual in all orthogonal directions and is zero for models on the Base–teacher line. The teacher endpoint is fixed throughout each trajectory, including the reference trajectory of the source training.

Prediction trajectories compare students of SFT and RLVR teachers with their respective teachers on fixed evaluation prefixes. In-domain tasks are AMC23, AIME24, AIME25, HMMT25, and Math500; OOD tasks are HellaSwag, ARC-Challenge, MMLU, IFEval, TriviaQA, BBH, and NQ. We average JS equally across datasets within each domain, then divide by the initial Base–teacher JS on the same prefixes. The normalized distance is one at Base. Each curve keeps its teacher and prefixes fixed.

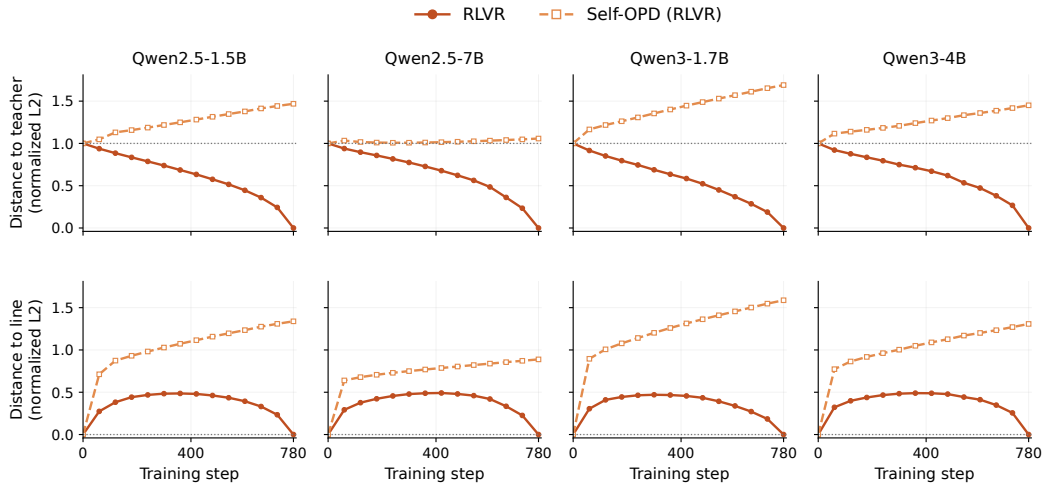


Figure 12: Complete parameter trajectories for RLVR and its Self-OPD student. Top: distance to the fixed teacher endpoint d_θ. Bottom: distance from the Base–teacher line $d_\perp$. Both use the initial Base–teacher distance for normalization, with each student compared to its paired teacher.

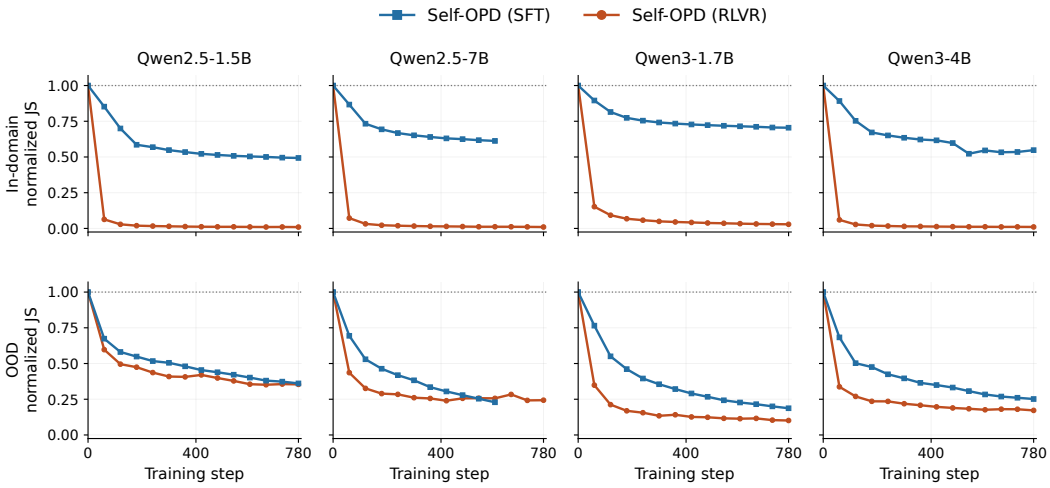


Figure 13: Self-OPD prediction distances to the respective frozen teachers throughout training. In-domain and OOD JS average five and seven datasets equally, respectively, and are normalized by the corresponding initial Base–teacher JS. Lower is closer.

C.3 Paired Capability Curves and Evaluation

Each Self-OPD student is paired with the SFT or RLVR run that produced its frozen teacher. Figure 14 uses completed in-domain and seven-task OOD evaluations.

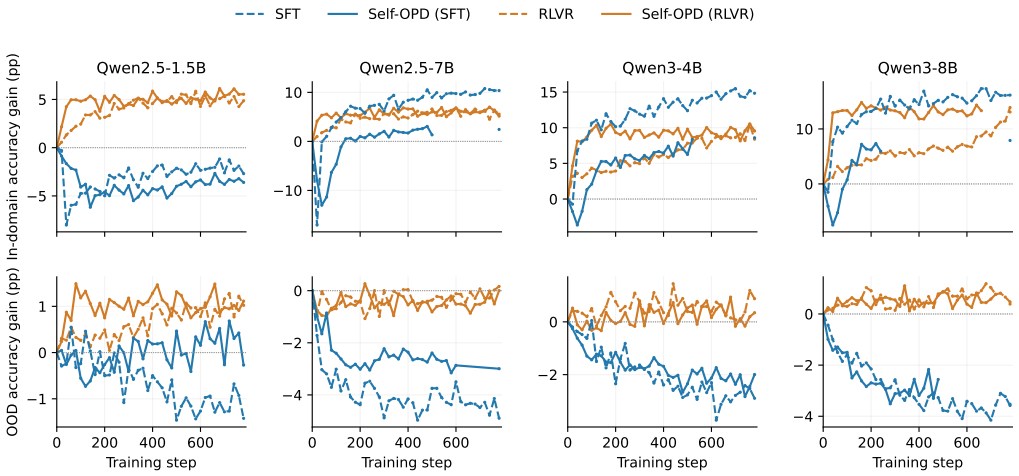


Figure 14: Accuracy changes from Base for source training (dashed) and the corresponding Self-OPD student (solid). Students use a fixed endpoint teacher.

Source SFT/RLVR evaluations use eight Math500 responses per problem; Self-OPD uses four. Both use eight responses for each other mathematical problem, the same OOD scoring rules, and equal dataset weights. RLVR students approach teacher task performance early, while larger SFT students inherit substantial OOD losses before matching task gains.

C.4 Teacher-Prefix Probabilities Under the Student

We fix 64 OpenR1 mathematical problems and generate four responses per problem from each SFT and RLVR endpoint teacher, using Raw inputs, temperature 1, top-$p = 1$, and a maximum response length of 1024 tokens. For the first L tokens of a teacher response, Base, both Self-OPD students, and the generating teacher each compute

$$\ell_L(p; y \mid x) = -\frac{1}{L}\sum_{t=1}^{L} \log p(y_t \mid x, y_{<t}). \tag{14}$$

Lower scores mean higher probability for the same fixed-length prefix. We first average responses within a problem, then weight problems equally.

Table 9 gives cross-scoring results for two models at the same student checkpoints, 60 and 240. Both students increase the probability of their teachers' prefixes. RLVR prefixes already have higher probability under Base, and their students approach teacher self-scores faster.

Correcting for teacher self-probability. Prefix scores also reflect how concentrated the teacher's own generation distribution is. We therefore subtract teacher self-scored negative log probability from the Base score:

$$g_L(q, p_0; y \mid x) = \frac{1}{L}\sum_{t=1}^{L} \log \frac{q(y_t \mid x, y_{<t})}{p_0(y_t \mid x, y_{<t})}. \tag{15}$$

Table 10 shows a larger teacher–Base scoring gap for SFT responses. This relationship persists when scoring only tokens 33–128, extending the source difference to later parts of the response.

Prefix source	Base	SFT@60	SFT@240	RLVR@60	RLVR@240	Teacher
Qwen2.5-7B						
SFT	1.057	0.871	0.580	1.563	1.567	0.379
RLVR	0.230	0.330	0.436	0.106	0.096	0.094
Qwen3-1.7B						
SFT	0.967	0.839	0.665	1.855	1.822	0.458
RLVR	0.300	0.332	0.444	0.145	0.136	0.135

Table 9: Mean negative log probability over the first 128 tokens of teacher responses, in nats/token. SFT/RLVR columns denote students of the corresponding teacher source; @ denotes student training step. Each teacher contributes all 256 responses; the final column is scored by the teacher that generated them.

Prefix source	Tokens 1–128	Tokens 33–128
Qwen2.5-7B		
SFT	0.678	0.514
RLVR	0.136	0.122
Qwen3-1.7B		
SFT	0.509	0.353
RLVR	0.165	0.132

Table 10: Scoring gaps after correcting for teacher self-probability, in nats/token. Larger values indicate a higher predictive cost under Base relative to the generating teacher.

C.5 SFT Teacher Openings and First-Position Feedback

Opening preferences and candidate coverage. Of the 49,993 complete OpenR1 training solutions, 99.54% begin with `Okay` or `Alright` after stripping whitespace and ignoring case. SFT retains these openings in its training targets. On 64 fixed mathematical problems with four responses per problem, all 256 responses from each of the Qwen2.5-7B and Qwen3-1.7B SFT teachers begin with these words; none from their RLVR counterparts do. At each response position, OPD computes feedback on the student's 16 highest-probability tokens. On the same Base-student responses, all positive probability mass added by either SFT teacher relative to Base at the first position falls outside this candidate set. At later sampled positions, the excluded fractions are only 7.02% and 8.87%, respectively, weighted by added probability mass. Thus, the teacher assigns low probability to the student's usual openings while its preferred openings receive no direct candidate-level positive feedback.

Removing first-position feedback improves task transfer. We compare the first 40 updates of paired Qwen2.5-7B runs with the same frozen SFT teacher, Base initialization, data order, and sampling seed. The intervention zeros all 16 candidate feedback values at the first valid response position; later feedback, response masks, and the loss denominator are unchanged. Both runs use Top-16 feedback, AdamW, a learning rate of 10^{-6}, and 64 problems with four responses per problem in each batch. Evaluation uses ordinary generation and scoring, averaging the five mathematical datasets equally. AMC23, AIME24, AIME25, and HMMT25 use eight responses per problem with a 31,744-token limit; Math500 uses four responses with a 16,384-token limit.

Training step	Ordinary feedback	First-position feedback zeroed	Gain (pp)
20	18.28	27.92	+9.64
40	14.47	27.53	+13.06

Table 11: Mathematical accuracy (%) in the paired opening-feedback intervention. Evaluation leaves all output positions unchanged.

Table 11 shows that first-position feedback impairs mathematical learning in this comparison. Together with the opening and coverage measurements, this supports an explanation in which fixed

response-opening preferences divert feedback from task transfer when they fall outside the student's candidate set.

C.6 OPSD Teacher Context and the Answer Boundary

OPSD teachers receive a complete reference solution before scoring the student's response. Holding 64 mathematical problems and student responses fixed, we compare direct Raw concatenation, chat-template packaging, and Raw input followed by `Answer to the original problem:`. Table 12 reports the teacher's mean probability of the end-of-sequence token before the response begins. Both an assistant-start marker and an explicit textual answer boundary sharply reduce this initial stopping preference.

Teacher	Raw	Chat template	Raw + answer boundary
Qwen2.5-7B	24.12	0.0100	0.0028
Qwen3-1.7B	15.36	0.0385	0.0034

Table 12: Teacher end-of-sequence probability (%) at the first response position for fixed student responses.

In paired short Qwen2.5-7B training runs, adding only the textual answer boundary also reduces empty student responses. Both runs begin with 12.11% empty responses in the first batch; by the fourth batch, the rates are 21.09% with direct concatenation and 1.95% with the boundary. Each batch contains the same 64 problems with four responses per problem. These results motivate explicitly separating the reference solution from the new response to be scored. Our OPSD teacher uses the chat template's assistant-start boundary, while the student sees only the Raw problem (Appendix A.2). The original OPSD prompt similarly introduces a new answer after the reference solution (Zhao et al., 2026).

C.7 Complete Capability Comparisons for Teacher Sources

Table 13 reports all four teacher–student pairs at student step 120, using the same A/B/C definitions as Table 3. This fixed early-training comparison uses complete in-domain accuracy, OOD accuracy, and JS measurements for every group. All three teachers within a group share the student initialization, update settings, and evaluation prefixes.

Teacher supervision	Distribution change			Accuracy (%)	
	D_{in}	D_{OOD}	ρ	Math (Δ)	OOD (Δ)
Qwen2.5-7B→Qwen2.5-1.5B					
Student Base	–	–	–	13.93	35.20
A: Larger Base teacher	0.3696	0.1151	0.311	14.67 (+0.75)	34.82 (−0.39)
B: Larger RLVR teacher	4.1650	0.6873	0.165	19.17 (+5.25)	35.99 (+0.79)
C: Constructed teacher	4.0247	0.5658	0.141	19.22 (+5.29)	35.76 (+0.56)
Qwen3-4B→Qwen3-1.7B					
Student Base	–	–	–	20.17	36.09
A: Larger Base teacher	0.2692	0.1772	0.658	19.38 (−0.78)	35.96 (−0.13)
B: Larger RLVR teacher	4.1255	0.3643	0.088	24.21 (+4.04)	36.19 (+0.10)
C: Constructed teacher	4.1056	0.3302	0.080	24.08 (+3.92)	36.05 (−0.04)
Qwen3-8B→Qwen3-1.7B					
Student Base	–	–	–	20.17	36.09
A: Larger Base teacher	0.3272	0.2875	0.879	20.65 (+0.49)	35.94 (−0.15)
B: Larger RLVR teacher	4.2713	0.3747	0.088	26.72 (+6.55)	37.04 (+0.94)
C: Constructed teacher	4.1650	0.3205	0.077	27.02 (+6.85)	36.66 (+0.57)
Qwen3-8B→Qwen3-4B					
Student Base	–	–	–	26.85	45.45
A: Larger Base teacher	0.3556	0.2286	0.643	27.85 (+1.01)	45.76 (+0.31)
B: Larger RLVR teacher	3.1320	0.4312	0.138	39.34 (+12.49)	45.22 (−0.23)
C: Constructed teacher	2.8076	0.3995	0.142	41.54 (+14.70)	45.27 (−0.19)

Table 13: All A/B/C comparisons at student step 120. JS is relative to student Base in 10^{-2} nats. Accuracies are percentages, with changes from Base in parentheses. Domains use equal dataset weights.

C.8 Distribution Changes Throughout Training

Figures 15 and 16 show training trajectories across methods and teacher configurations. Each row presents one student model, with in-domain JS, OOD JS, and ρ in successive columns. OPD uses Raw inputs for both student and teacher; the corresponding standard SFT and RLVR runs are references. We measure JS from each model's Base on fixed Base/RLVR prefix banks, averaging datasets equally within each domain. The higher relative OOD influence of larger Base teachers is visible at multiple training stages, while source differences also evolve during training. Both divergences are zero at Base, where ρ is undefined.

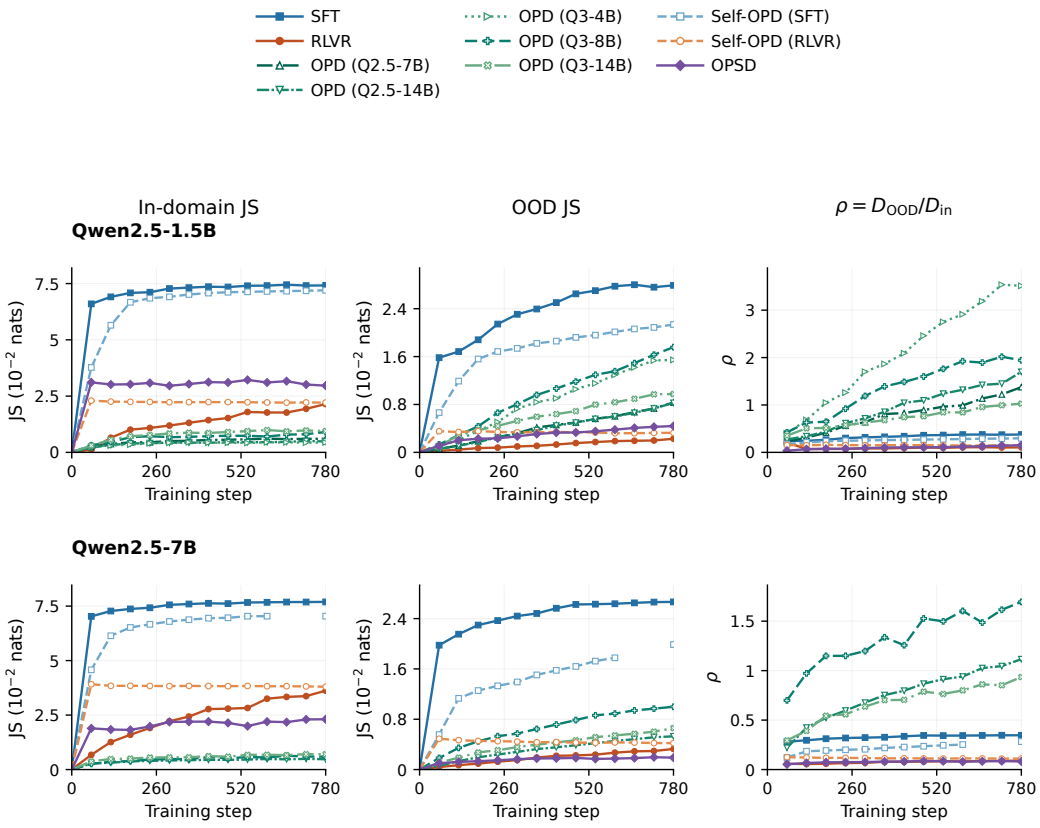


Figure 15: Distribution-change trajectories for the two Qwen2.5 students. The first two columns show JS from Base in 10^{-2} nats; the last shows relative OOD influence ρ. Curves distinguish methods and teacher configurations, connecting adjacent points.

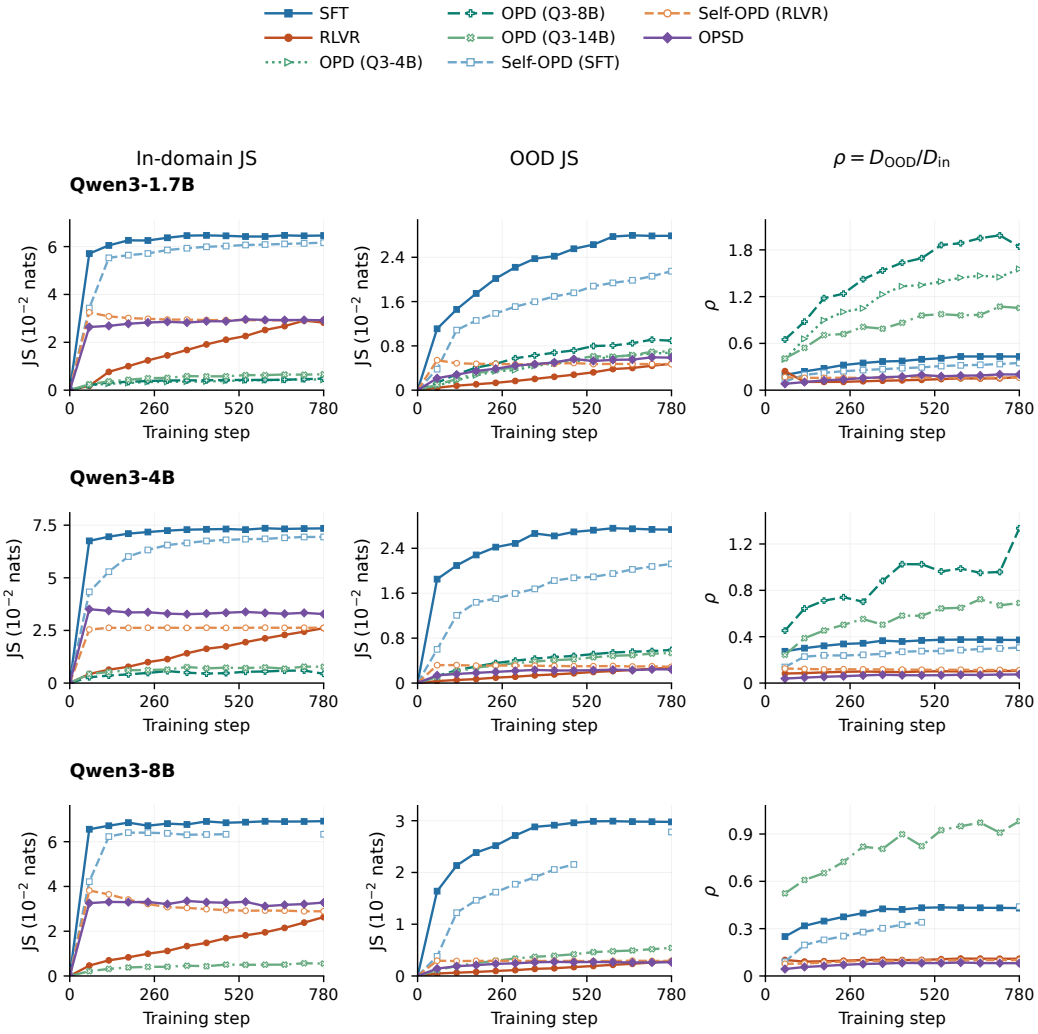


Figure 16: Distribution-change trajectories for the three Qwen3 students, with the same metrics as Figure 15.

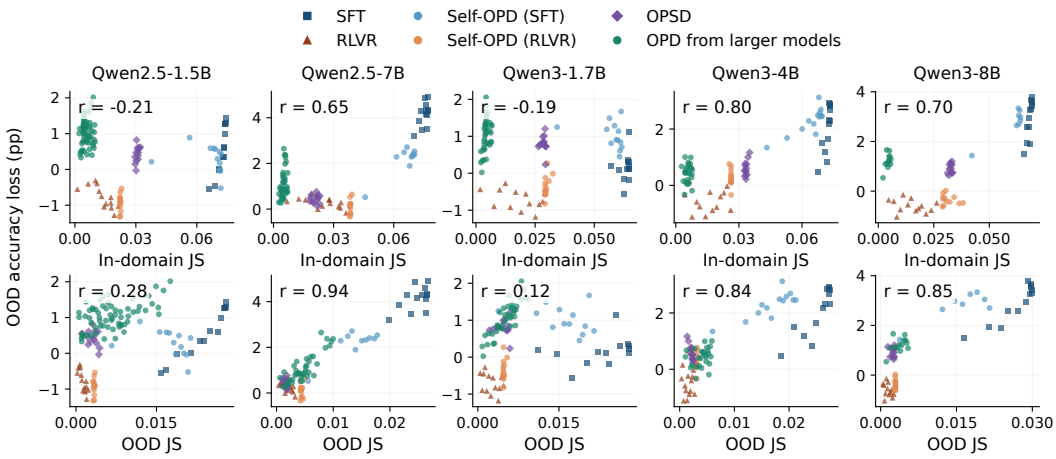


Figure 17: Complete five-model comparison of distribution change and forgetting, including Qwen3-1.7B. In-domain JS is shown above OOD JS; every point pairs complete measurements at the same checkpoint. r denotes Pearson correlation.

C.9 Training and Measurement Settings

Teacher-source and constructed-teacher comparisons start from the same public student Base and use OpenR1-50K with the same Top-k feedback and optimizer as the main OPD training. Students sample new on-policy responses before each update; both student and teacher use Raw inputs. Within each comparison, training data, batches, sampling rules, and optimization settings are held fixed while the teacher varies. Table 14 summarizes the setup.

Setting	Value
Candidate feedback / optimizer	Top-16 / AdamW
Student / teacher input	Raw / Raw
Problems per batch / responses per problem	64 / 4
Student steps for all A/B/C comparisons	120

Table 14: Training settings for teacher-source comparisons.

JS uses fixed evaluation prefixes and the same equally weighted in-domain and OOD dataset groups as the main evaluation. Base- and RLVR-generated prefixes receive equal weights. All A/B/C comparisons use the same student training step.

C.10 Constructed-Teacher Normalization and Current-Student Feedback

The models used in group C share a vocabulary. At each prefix, we combine logits from the frozen smaller Base, larger Base, and larger RLVR models:

$$q_C = \mathrm{softmax}(z_{p_0} + z_{q_R} - z_{q_B}). \tag{16}$$

The resulting ordinary OPD feedback is

$$\log \frac{q_C(v \mid s)}{p_t(v \mid s)} = \log \frac{q_R(v \mid s)}{q_B(v \mid s)} - \log \frac{p_t(v \mid s)}{p_0(v \mid s)} - \log Z(s). \tag{17}$$

The student's current change from Base is subtracted as training proceeds, while the teacher's source models remain frozen.

D WEIGHT INTERPOLATION AND EXTRAPOLATION

Section 4.3 compares SFT, RLVR, and their Self-OPD students under weight scaling. We retain these four methods for all five student models and use their final training endpoints. These sweeps use the original Raw/Template Self-OPD runs. The curves share the complete measured interval $0 \leq \lambda \leq 2$. We rescale the weight difference between Base (A) and the trained endpoint (B):

$$\theta_\lambda = (1-\lambda)\theta_A + \lambda\theta_B, \qquad \lambda \in \{0, 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.5, 2\}.$$

Values $0 \leq \lambda \leq 1$ interpolate between the endpoints; $\lambda > 1$ extrapolates beyond B. Evaluation follows Appendix A.

Sweet-spot coefficients. Let $\Delta\mathrm{Acc}_{\mathrm{in}}(\lambda)$ and $\Delta\mathrm{Acc}_{\mathrm{OOD}}(\lambda)$ be accuracy changes from Base, in percentage points. We mark sweet spots only for runs whose trained endpoint improves in-domain accuracy over Base, $\Delta\mathrm{Acc}_{\mathrm{in}}(1) > 0$. Stars in Figure 4 identify measured coefficients satisfying

$$\lambda > 0, \qquad \begin{aligned} \Delta\mathrm{Acc}_{\mathrm{in}}(\lambda) &> \Delta\mathrm{Acc}_{\mathrm{in}}(1), \\ \Delta\mathrm{Acc}_{\mathrm{OOD}}(\lambda) &> \Delta\mathrm{Acc}_{\mathrm{OOD}}(1). \end{aligned}$$

Both task means must be available at the candidate coefficient and at $\lambda = 1$. OOD accuracy is the equally weighted mean over the same tasks as the main evaluation.

The identified sweet spots lie within the interpolation range for SFT-based updates and the extrapolation range for RLVR-based updates. Learning rates and training progress can also affect the endpoint and the locations of these coefficients.

Learning and retention along a training direction. Interpolation toward Base can retain much of the in-domain gain while recovering OOD accuracy (Figure 18). Further extrapolation often reduces in-domain accuracy and accentuates SFT-family forgetting. RLVR directions preserve OOD accuracy over a broader range of weight scaling.

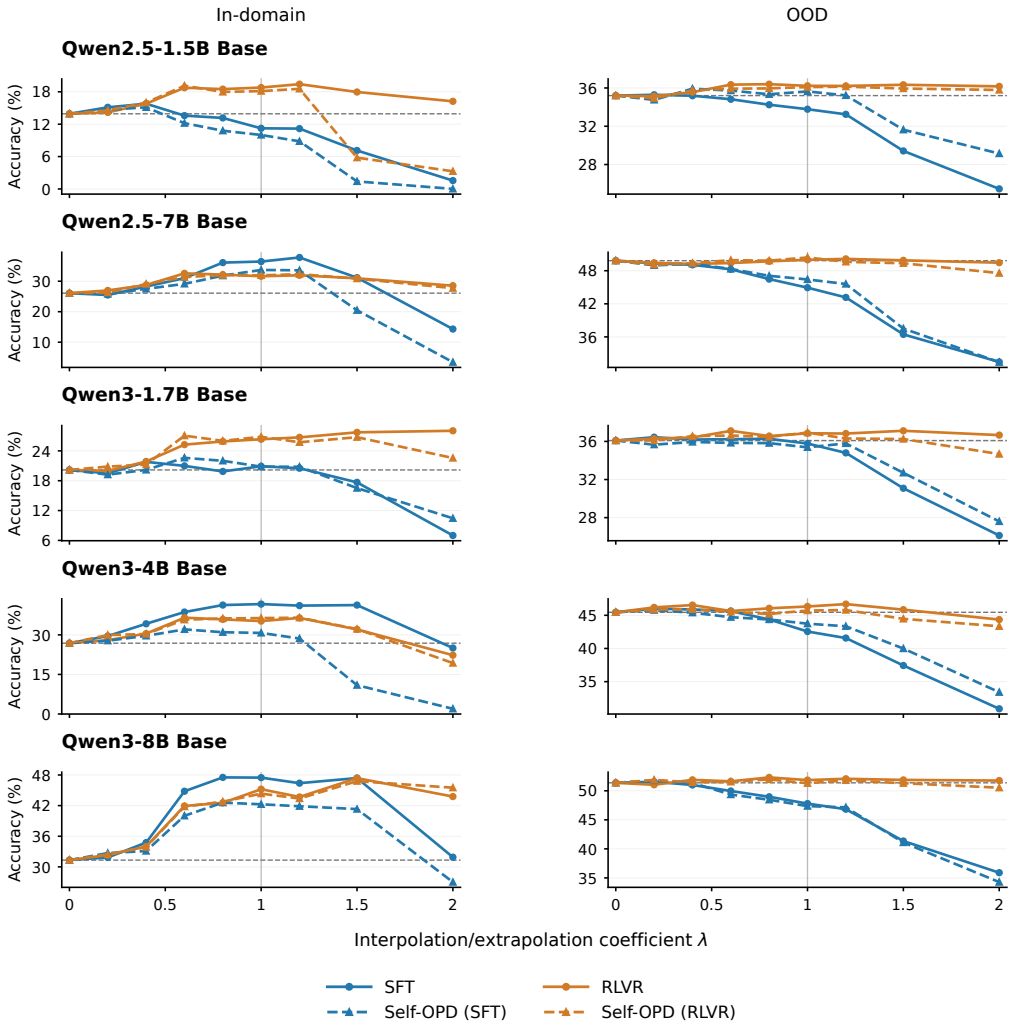


Figure 18: Accuracy sweeps over the common measured interval $0 \leq \lambda \leq 2$, from Base through trained endpoints ($\lambda = 1$). OOD averages all tasks equally. RLVR-based directions preserve general capabilities over a broader scaling range than SFT-based directions.

E SUPPORTING ANALYSIS

E.1 HOW STUDENTS WEIGHT TEACHER FEEDBACK

At a fixed prefix s, let $p = \mathrm{softmax}(z)$ and sample $a \sim p$. For the score-function surrogate $L = -\,\mathrm{sg}(A)\log p(a)$, define $m_v = \mathbb{E}[A \mid a = v, s]$. Then

$$g_v = -A\big(\mathbf{1}\{a = v\} - p(v)\big), \tag{18}$$

$$\mu_v(s) = \mathbb{E}[g_v \mid s] = -p(v)m_v + p(v)\sum_u p(u)m_u = p(v)\big(\mathbb{E}[A \mid s] - m_v\big). \tag{19}$$

For ordinary OPD, $m_v = \log q_T(v \mid s) - \log p(v \mid s)$. The student's current token probabilities weight differences in teacher preferences across candidates. Applying the chain rule and averaging over student-generated prefixes d_p gives

$$\mathbb{E}[\nabla_\theta L] = \mathbb{E}_{s\sim d_p}\left[\sum_v \mu_v(s)\nabla_\theta z_v(s)\right]. \tag{20}$$

Less frequently visited prefixes and lower-probability candidates receive smaller weights in this mean update. This full-vocabulary score-function model motivates the prefix analysis; the implemented candidate feedback and optimizer are listed in Table 14.

Token-dependent feedback means that m_v varies with candidate v at a fixed prefix. Random per-token feedback with the same conditional mean for every candidate still yields zero mean logit gradient. As student predictions approach the frozen teacher, ordinary OPD's log-probability feedback and its mean gradient shrink.

E.2 LOCAL DIRECTIONS AFTER CALIBRATING IN-DOMAIN CHANGE

A directional view of local JS. Let $p_0 = p_{\theta_0}$ be student Base and consider a small update $\theta_\epsilon = \theta_0 + \epsilon g$ along a fixed parameter direction g. For domain $d \in \{\mathrm{in}, \mathrm{OOD}\}$, fix evaluation prefixes and their task weights, represented by $\mathcal{S}_d$. Define the score vector at Base and the domain's predictive Fisher matrix:

$$\psi_s(v) = \nabla_\theta \log p_\theta(v \mid s)|_{\theta=\theta_0}, \qquad F_d = \mathbb{E}_{s\sim\mathcal{S}_d}\left[\sum_v p_0(v \mid s)\psi_s(v)\psi_s(v)^\top\right]. \tag{21}$$

F_d captures the local sensitivity of that domain's predictions to parameter changes. Using natural logarithms, smooth conditional distributions, and positive probabilities, mean JS satisfies

$$D_d(\theta_0, \theta_\epsilon) = \frac{\epsilon^2}{8} g^\top F_d g + \mathcal{O}(\epsilon^3). \tag{22}$$

The per-prefix quadratic term of JS is $\frac{1}{8}\sum_v (p_{\theta_\epsilon} - p_0)^2/p_0$. Substituting $p_{\theta_\epsilon}(v \mid s) - p_0(v \mid s) = \epsilon p_0(v \mid s)\psi_s(v)^\top g + \mathcal{O}(\epsilon^2)$ and averaging over fixed prefixes yields the result.

When $g^\top F_{\mathrm{in}} g > 0$,

$$\rho(\epsilon, g) = \frac{D_{\mathrm{OOD}}(\theta_0, \theta_\epsilon)}{D_{\mathrm{in}}(\theta_0, \theta_\epsilon)} = \frac{g^\top F_{\mathrm{OOD}} g}{g^\top F_{\mathrm{in}} g} + \mathcal{O}(\epsilon). \tag{23}$$

Update magnitude cancels in the ratio of quadratic terms, leaving the relative influence of the direction on the two domains. This motivates calibrating in-domain change to compare the OOD influence of directions induced by different feedback sources.

Comparing calibrated directions. The local comparison reuses the same prefixes generated by Base and computes A/B/C update directions separately. For each direction, we adjust only its overall magnitude to reach a common in-domain JS target on calibration prefixes. We then measure in-domain JS, OOD JS, and ρ on independent evaluation prefixes, so the reported in-domain JS can differ. Reference RLVR directions use parameter differences along the student's original RLVR training, with the same calibration procedure.

Update direction	In-domain JS $\times 10^5$	OOD JS $\times 10^5$	ρ
Qwen3-1.7B			
A: Base teacher	1.079	0.659	0.611
B: RLVR teacher	1.072	0.079	0.074
C: RLVR increment only	0.938	0.053	0.056
RLVR parameter change (step 60)	1.228	0.071	0.058
RLVR parameter change (step 780)	1.098	0.069	0.063
Qwen3-4B			
A: Base teacher	0.835	0.070	0.084
B: RLVR teacher	1.538	0.025	0.016
C: RLVR increment only	1.573	0.024	0.015

Table 15: Independent-prefix measurements after calibrating in-domain JS to 10^{-5}. Incorporating the teacher's RLVR changes reduces relative OOD influence and brings it closer to the paired direct RLVR parameter directions.

E.3 Teacher Information Relative to Base

Fix a student prefix s and let s^{T} contain the same problem and continuation under the teacher's input format, including privileged context when present. On a common finite vocabulary with positive probabilities, the Base-relative teacher preference is

$$\begin{aligned} b_q(v, s) &= \log \frac{q(v \mid s^{\mathrm{T}})}{p_0(v \mid s)} \\ &= \log \frac{q(v \mid s^{\mathrm{T}})}{p_0(v \mid s^{\mathrm{T}})} + \log \frac{p_0(v \mid s^{\mathrm{T}})}{p_0(v \mid s)}. \end{aligned} \tag{24}$$

The first term compares teacher and Base under identical inputs; the second isolates the effect of the teacher input on the Base model. For Self-OPD (SFT) and Self-OPD (RLVR), the first term contains the changes acquired during teacher training. For a larger Base teacher, it contains differences between pretrained models; a post-trained larger teacher additionally includes preferences acquired during post-training. For OPSD, the teacher is a frozen copy of Base, so the first term is zero and the signal comes from the augmented input.

Information in the A/B/C teachers. For the larger Base and RLVR teachers, the shared-input decomposition in Equation (6) gives

$$b_{q_B} = \delta_{\mathrm{pre}}, \qquad b_{q_R} = \delta_{\mathrm{pre}} + \delta_{\mathrm{RL}}, \qquad b_{q_C} = \delta_{\mathrm{RL}} - \log Z(s), \tag{25}$$

where $Z(s) = \sum_v p_0(v \mid s) \exp[\delta_{\mathrm{RL}}(v \mid s)]$. The last equality follows directly from normalizing $p_0 \exp(\delta_{\mathrm{RL}})$. It identifies the information retained by the constructed teacher while accounting for its normalization.

Adding and subtracting $\log p_0(v \mid s)$ gives

$$\log \frac{q(v \mid s^{\mathrm{T}})}{p_\theta(v \mid s)} = b_q(v, s) - \log \frac{p_\theta(v \mid s)}{p_0(v \mid s)}. \tag{26}$$

At the initial student, the entire coefficient is $b_q(v, s)$. As the student changes, its Base-relative log probability is subtracted from this fixed teacher preference. The per-token identity also holds on the student's Top-16 support, before applying the detached weights used in training. Student-generated prefixes and candidate weights determine which teacher preferences contribute to each update.

For the full-vocabulary local objective, the same decomposition gives

$$\begin{aligned} &\mathrm{KL}\big(p_\theta(\cdot \mid s) \,\|\, q(\cdot \mid s^{\mathrm{T}})\big) \\ &\quad = \mathrm{KL}\big(p_\theta(\cdot \mid s) \,\|\, p_0(\cdot \mid s)\big) - \mathbb{E}_{v \sim p_\theta(\cdot \mid s)}[b_q(v, s)]. \end{aligned} \tag{27}$$

This follows by expanding the log ratio inside the expectation. All teacher configurations share this form; their information sources determine the additional preferences b_q. For Group C,

$$\mathrm{KL}(p_\theta \| q_C) = \mathrm{KL}(p_\theta \| p_0) - \mathbb{E}_{p_\theta}[\delta_{\mathrm{RL}}] + \log Z(s). \tag{28}$$

At a fixed prefix, $Z(s)$ is independent of student parameters, so its gradient is zero. Group C therefore combines fitting the RLVR increment with the current student's departure from Base. On any candidate set, centering with common normalized weights also cancels $\log Z(s)$.

Teacher distance and student change. For fixed evaluation prefixes, define the teacher's relative distance from student Base by

$$\rho_{\mathrm{T}}(q; p_0) = \frac{D_{\mathrm{OOD}}(q, p_0)}{D_{\mathrm{in}}(q, p_0)}, \qquad D_{\mathrm{in}}(q, p_0) > 0. \tag{29}$$

The ratio compares the frozen target with student Base. The student's ρ instead measures changes reached along its optimization path, whose directions also depend on student prefix probabilities and logit Jacobians (Appendix E.1).

A fixed-Base signal compares teacher and Base on the same input: $A_\Delta(v, s) = \log[q(v \mid s^{\mathrm{T}})/p_0(v \mid s^{\mathrm{T}})]$. Using $p_0(\cdot \mid s^{\mathrm{T}})$ as the reference gives

$$\begin{aligned} &\mathrm{KL}\big(p_\theta(\cdot \mid s) \,\|\, q(\cdot \mid s^{\mathrm{T}})\big) \\ &\quad = \mathrm{KL}\big(p_\theta(\cdot \mid s) \,\|\, p_0(\cdot \mid s^{\mathrm{T}})\big) - \mathbb{E}_{v \sim p_\theta(\cdot \mid s)}[A_\Delta(v, s)]. \end{aligned} \tag{30}$$

Thus the current-student reference balances the teacher–Base preference against departure from the frozen reference distribution.

E.4 Fixed-Base Feedback and Candidate Centering

A fixed teacher-to-Base log ratio provides a reference for the current-student feedback used by ordinary OPD and Group C. Following the delta-feedback construction of Heo et al. (2026), let q be a frozen teacher and p_0 the frozen student Base reference, with a student p_θ initialized from Base. At a student-generated response prefix, let s denote the Raw student input and s^{T} the same problem and continuation under the teacher's input format. Both q and the frozen Base are scored on s^{T}, so their difference compares identical inputs. Define the ordinary and delta signals by

$$\begin{aligned} r_a(s) &= \log q(a \mid s^{\mathrm{T}}) - \log p_\theta(a \mid s), \\ d_a(s) &= \log q(a \mid s^{\mathrm{T}}) - \log p_0(a \mid s^{\mathrm{T}}). \end{aligned} \tag{31}$$

All log probabilities use the full vocabulary and the same scoring temperature. Let $\mathcal{K}(s)$ be a candidate set selected by the student. We normalize the student probabilities on this set and center each signal using those weights:

$$\begin{aligned} w_a(s) &= \frac{p_\theta(a \mid s)}{\sum_{v \in \mathcal{K}(s)} p_\theta(v \mid s)}, \qquad a \in \mathcal{K}(s), \\ \bar{r}_a &= r_a - \sum_{v \in \mathcal{K}(s)} w_v r_v, \qquad \bar{d}_a = d_a - \sum_{v \in \mathcal{K}(s)} w_v d_v. \end{aligned} \tag{32}$$

The weighted coefficient passed to the actor update is

$$A_a^{\mathrm{delta}}(s) = w_a \bar{d}_a \, \mathbf{1}\{\bar{r}_a \bar{d}_a > 0\}. \tag{33}$$

Zero or disagreeing signs give zero coefficients. These coefficients are computed without gradients, masked to valid response positions, and held fixed during the same probability-ratio update used by ordinary Self-OPD. No separate KL penalty is added.

The raw delta d_a measures predictive changes accumulated during teacher training, while the visited prefixes, weights, and gate depend on the current student. Group C uses the normalized constructed teacher q_C in ordinary OPD, without this sign-agreement gate. Its feedback also subtracts the current student's log probability, as shown in Appendix C.10.

E.5 The Current Policy as a Feedback Reference

Sampling and feedback both depend on the student. Let μ_θ be the distribution of prefixes generated by the current student. Compare the two raw feedback choices on the same prefixes drawn from

μ_θ. Ordinary OPD additionally uses p_θ in its log-probability ratio (Equation (31)). For full-vocabulary score updates with positive probabilities, define $g_A = \mathbb{E}_{s\sim\mu_\theta,\, v\sim p_\theta(\cdot|s)}[A(v,s)\nabla_\theta \log p_\theta(v \mid s)]$. Before candidate truncation and the sign gate, holding sampled prefixes fixed when differentiating gives

$$g_\Delta - g_{\text{self}} = \mathbb{E}_{s\sim\mu_\theta}\left[\nabla_\theta \mathrm{KL}\big(p_\theta(\cdot \mid s)\|p_0(\cdot \mid s^{\mathrm{T}})\big)\right]. \tag{34}$$

This follows by substituting the advantage difference and using $\mathbb{E}_{p_\theta}\nabla_\theta \log p_\theta = 0$. The ordinary reference therefore corrects for the evolving student's departure from the frozen Base distribution on the prefixes that this same student visits.

Feedback decreases as preferences are acquired. Fix a prefix, use identical input formats, and suppress the prefix notation. The teacher q may also be the constructed q_C. Let p_0 and q have positive probabilities on a finite vocabulary, and consider the family

$$b_v = \log \frac{q_v}{p_{0,v}}, \qquad p_{\alpha,v} = \frac{p_{0,v}\exp(\alpha b_v)}{Z_\alpha}, \qquad Z_\alpha = \sum_v p_{0,v}\exp(\alpha b_v), \quad 0 \le \alpha \le 1. \tag{35}$$

The endpoints are p_0 and q. Substitution yields

$$A_{\text{self},v} = (1-\alpha)b_v + \log Z_\alpha, \qquad \overline{A}_{\text{self},v} = (1-\alpha)\overline{A}_{\Delta,v}, \tag{36}$$

where bars subtract the expectation under p_α. The centered identity also holds on a candidate set when both signals use the same normalized centering weights. The common constant has zero full-vocabulary score gradient. Thus the current-student reference reduces feedback in proportion to the remaining preference difference along this family. For $\alpha < 1$, the nonzero centered signals agree in sign, so the delta gate in Equation (33) keeps them. At $\alpha = 1$, the teacher–student signal and the gated update vanish.

A reward-aligned model of the RLVR source. For a fixed problem and a common input format, let $r(y)$ be the Base distribution over a finite set of complete responses, with positive probabilities, and let $R(y) \in \{0,1\}$ denote correctness. Consider a teacher that reweights these responses by reward and a student along the same family:

$$q_\beta(y) = \frac{r(y)e^{\beta R(y)}}{Z_\beta}, \qquad p_\tau(y) = \frac{r(y)e^{\tau R(y)}}{Z_\tau}, \qquad 0 \le \tau < \beta. \tag{37}$$

The teacher is the optimizer of $\mathbb{E}_q R - \beta^{-1}\mathrm{KL}(q\|r)$, as follows by rewriting this objective as $\beta^{-1}\log Z_\beta - \beta^{-1}\mathrm{KL}(q\|q_\beta)$. This model represents training that changes the probability of correctness while preserving Base's relative preferences among responses with the same reward.

The token-level implication follows from conditioning on prefixes. Let $V(s) = \Pr_r(R = 1 \mid s)$ and $Z_b(s) = \mathbb{E}_r[e^{bR} \mid s] = 1 + (e^b - 1)V(s)$. Since $q_b(v \mid s) = r(v \mid s)Z_b(sv)/Z_b(s)$, the two raw signals are

$$\begin{aligned} A_\Delta(v,s) &= \log Z_\beta(sv) - \log Z_\beta(s), \\ A_{\text{self}}(v,s) &= \log \frac{Z_\beta(sv)}{Z_\tau(sv)} - \log \frac{Z_\beta(s)}{Z_\tau(s)}. \end{aligned} \tag{38}$$

For a fixed s, both signals are increasing functions of $V(sv)$. Writing $k_b = e^b - 1$, their derivatives with respect to $V(sv)$ are

$$\frac{k_\beta}{1 + k_\beta V(sv)} > 0, \qquad \frac{k_\beta - k_\tau}{[1 + k_\beta V(sv)][1 + k_\tau V(sv)]} > 0. \tag{39}$$

Thus the two references preserve the same ordering of token continuations by eventual success in this model, with feedback strength depending on the current student's progress. The student's success probability, $e^\tau V(sv)/Z_\tau(sv)$, is also increasing in $V(sv)$, so this ordering agrees with expected correctness under the current student. The statement holds at every visited prefix even as on-policy rollouts change.

An SFT teacher fits a demonstration distribution whose probability ratios relative to Base can also vary among equally correct responses. Its fixed source signal can consequently retain response preferences beyond correctness as the student learns. Equation (36) describes how the current-student reference reduces the remaining feedback as these preferences are acquired. The effect of these preferences on retained capabilities depends on their cross-domain influence, described in Appendix E.2.

F Related Work

Learning and forgetting in post-training. Acquiring new skills while retaining prior capabilities is a central problem in continual learning (Wang et al., 2024). Empirical studies show that language-model fine-tuning can impair capabilities beyond the target task, motivating evaluations that jointly measure adaptation and retention (Luo et al., 2025). For language-model post-training, RL's Razor connects the retention advantage of online RL to limited KL changes on the new task (Shenfeld et al., 2025). A complementary data perspective analyzes how the distributions supplying supervision affect learning and interference with prior knowledge (Zhang et al., 2026). OPD combines on-policy student responses with teacher-provided token preferences, making it a useful setting for examining how supervision changes this learning–forgetting relationship. OPDArena compares training objectives and teacher sources under a shared setup, measuring both target-domain learning and general capability retention.

On-policy distillation. Training on student-generated trajectories exposes a distillation teacher to the contexts its student actually visits, reducing the mismatch between training and deployment distributions (Agarwal et al., 2024). MiniLLM uses reverse-KL distillation to align student and teacher predictions (Gu et al., 2024). Subsequent analysis identifies compatible thinking patterns and additional teacher capabilities as important conditions for successful OPD, and traces learning to alignment on high-probability tokens at student-visited states (Li et al., 2026). Related work also finds that smaller models can struggle to learn from strong reasoning demonstrations (Li et al., 2025). These studies motivate looking beyond teacher task accuracy when choosing supervision. Our analysis examines the information teachers add relative to student Base and how fitting these preferences affects predictions across domains.

Self-distillation and teacher information. Teacher information can enter through both model parameters and conditioning context. Self-Distilled Reasoner introduces OPSD, in which access to a reference solution provides the teacher with privileged information hidden from the student (Zhao et al., 2026). On-policy context distillation similarly transfers information available in a teacher's context into student parameters (Ye et al., 2026). Teacher conditioning, update rules, and the response tokens receiving supervision influence the resulting training signal (Zhao et al., 2026; Ye et al., 2026; Wang et al., 2026). We compare these contextual sources with preferences contained in larger pretrained models and acquired through SFT or RLVR. On-policy delta distillation extracts post-training information through teacher log-probability differences (Heo et al., 2026); our controlled teacher comparisons use such differences to examine the relationship between information sources and relative OOD influence.

Distribution changes and parameter paths. Prediction and parameter measurements provide complementary views of post-training. Distribution-based analyses connect supervision to changes in model behavior (Shenfeld et al., 2025; Zhang et al., 2026), while PEFT-Arena jointly evaluates learning and retention across fine-tuning parameterizations and uses interpolation to study behavior along adaptation paths (Huang et al., 2026). We compare prediction changes on in-domain and OOD prefixes and use their ratio to measure the relative cross-domain influence of updates. For Self-OPD, weight scaling supplies a reference for behavior along the Base–teacher parameter line, while paired parameter and prediction trajectories reveal how teacher preferences are inherited during distillation. Together, these measurements relate teacher information to both the selectivity of student updates and the different rates of task learning and forgetting.