BWB-KD: DOMAIN KNOWLEDGE DISTILLATION VIA KNOWLEDGE INJECTION, REASONING ORGANIZATION, AND ACTIVATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Transferring domain expertise from black-box teachers while preserving general capabilities remains a challenge in domain adaptation. Direct supervised adaptation can improve domain performance at the cost of broader capabilities, while subsequent reinforcement learning does not reliably resolve this trade-off. We propose BWB-KD, a stage-coordinated distillation framework that connects knowledge injection, reasoning organization, and outcome activation through complementary reasoning modes. Non-thinking supervised fine-tuning first builds a domain expert from black-box demonstrations. Thinking-mode on-policy distillation then uses the frozen expert to guide a base-initialized student along student-generated reasoning trajectories. Finally, non-thinking reinforcement learning optimizes the transferred student using outcome feedback. This OFF–ON–OFF design places expert-guided reasoning organization between knowledge acquisition and reinforcement learning. Experiments on Qwen3.6-27B across three biomedical tasks show improved domain performance over supervised fine-tuning followed by reinforcement learning, with larger gains in non-thinking inference. General-capability evaluations further show better non-thinking retention. Comparisons with offline distillation, alternative reinforcement learning methods, and different stage orders and reasoning modes support the value of coordinating how domain knowledge is acquired, transferred, and reinforced.

1 INTRODUCTION

Knowledge distillation transfers expertise from a teacher model to a student. White-box distillation exploits the teacher's token-level distributions for fine-grained supervision (Agarwal et al., 2024). For proprietary teachers, these distributions are often inaccessible, motivating black-box distillation through generated responses or evaluative feedback (Fang et al., 2026). Supervised fine-tuning (SFT) acquires domain knowledge from black-box teacher demonstrations (Hsieh et al., 2023). Reinforcement learning (RL) can further optimize the student through outcome-level feedback from task verifiers or teacher evaluations (Trung et al., 2024; Fang et al., 2026). However, transferring domain expertise from black-box teachers while preserving the student's general capabilities remains challenging.

Our experiments reveal complementary limitations of direct SFT and RL in balancing domain gains with general capability retention. In our MedXpert adaptation of Qwen3.6-27B, SFT lowers the non-thinking general-benchmark mean from 63.36 to 48.99. Subsequent GRPO improves domain performance but leaves that mean at 49.20. Direct GRPO from Base largely preserves general performance, yet achieves a lower domain mean than BWB-KD (Figure 1a). These two direct routes leave different parts of the domain–general trade-off unresolved. This motivates coordinating knowledge acquisition, student transfer, and RL.

We propose *Black–White–Black-Box Knowledge Distillation* (BWB-KD), which assigns complementary reasoning modes to *knowledge injection*, *reasoning organization*, and *outcome activation*. First, knowledge injection uses non-thinking SFT to build a domain expert from black-box teacher demonstrations, including visible reasoning and answers. Second, reasoning organization transfers this expert's guidance to a separate, base-initialized student through white-box on-policy distillation

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.

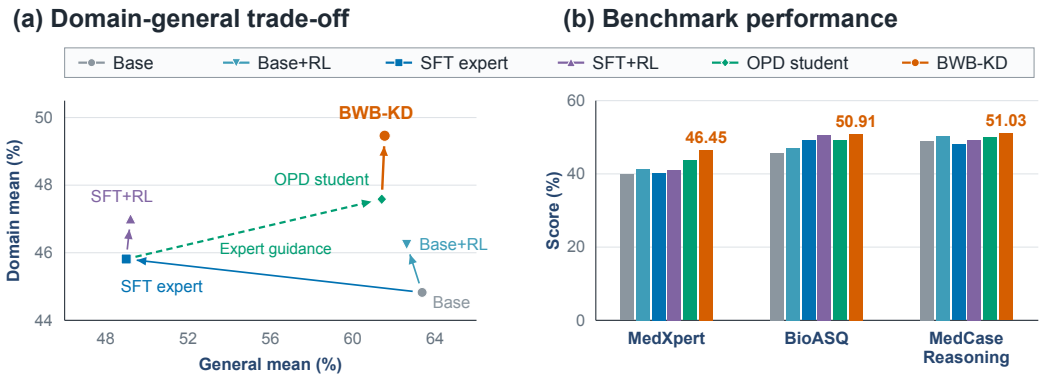

Figure 1: **Stronger domain expertise with better general-capability retention.** All results use Qwen3.6-27B with thinking OFF. (a) Method-level summaries: the vertical axis averages independently trained MedXpert, BioASQ, and MedCaseReasoning adaptations; the horizontal axis averages AIME25, ArenaHard, BBEH, and IFBench on the MedXpert branch. Base+RL retains general performance; BWB-KD achieves a higher domain mean. Solid arrows indicate continued training. The dashed arrow indicates guidance from the frozen SFT expert to an OPD student initialized from Base; BWB-KD continues that student with RL. (b) Scores on the three domain benchmarks. Base+RL applies GRPO directly to Base; RL denotes GRPO throughout. Values are point estimates.

(OPD), matching teacher distributions along the student's own thinking-mode trajectories (Agarwal et al., 2024). Finally, outcome activation applies non-thinking GRPO to the distilled student, using black-box outcome feedback to improve task performance.

Black-box and white-box describe the supervision interfaces: textual demonstrations or outcome rewards in the black-box stages, and expert token distributions in the white-box stage. Separately, ON and OFF specify whether the dedicated thinking channel is enabled; visible reasoning can appear in either mode.

Across three biomedical tasks, BWB-KD achieves stronger domain performance than direct RL or SFT followed by RL, with particularly clear gains in non-thinking inference. It also largely preserves general capabilities, addressing the pronounced degradation observed with direct SFT and SFT followed by RL (Figure 1).

Our contributions are threefold:

- **A stage-coordinated framework for black-box domain distillation.** We propose BWB-KD, which connects knowledge injection, reasoning organization, and outcome activation through complementary supervision interfaces. It converts black-box teacher demonstrations into an accessible domain expert, transfers its guidance to a separate student through white-box distillation, and refines the student through RL with outcome feedback.

- **Stage-specific reasoning-mode coordination.** We design an OFF–ON–OFF schedule that matches reasoning modes to the functional roles of the three stages: non-thinking demonstration learning for knowledge injection, thinking-mode student trajectories for reasoning organization, and non-thinking RL for outcome activation.

- **Empirical evidence for coordinated domain transfer.** Experiments across three biomedical tasks and four general benchmarks demonstrate stronger domain performance with substantial general-capability retention. Ablations validate the roles of stage order and reasoning-mode assignment, while comparisons with offline distillation and alternative RL methods demonstrate the benefits of the complete BWB-KD framework. Question-level analyses further reveal how these gains emerge across inference modes.

2 Related Work

Black-box and on-policy distillation. A common approach to black-box distillation trains on teacher-generated responses (Kim & Rush, 2016), which can include explanations alongside answers (Hsieh et al., 2023). Training on fixed teacher responses leaves a mismatch with the prefixes students generate at inference. GKD and MiniLLM address this mismatch by supervising student-generated sequences with teacher distributions (Agarwal et al., 2024; Gu et al., 2024). Without access to these distributions, black-box on-policy methods instead derive response-level feedback from discriminators, verbal scores, or rubrics (Ye et al., 2025; Xiong et al., 2026; Fang et al., 2026). BWB-KD instead uses black-box demonstrations to train a domain expert with accessible token probabilities, turning text-based supervision into distributional guidance for the student's own trajectories.

Coordinated post-training. InstructGPT combines supervised learning from demonstrations with subsequent RL from human feedback (Ouyang et al., 2022). ReFT uses the same broad ordering for reasoning, following SFT with answer-reward RL (Trung et al., 2024). Beyond this sequential recipe, ReLIFT interleaves supervised learning and RL (Ma et al., 2026), while SRFT optimizes their objectives jointly (Fu et al., 2026). PRISM addresses post-SFT distributional drift by inserting black-box adversarial OPD before multimodal RL, continuing to optimize the SFT policy (Wang et al., 2026). BWB-KD instead retains the SFT model as a frozen domain expert and distills its guidance into a Base-initialized student before outcome RL. This division separates expert acquisition from final-policy training. We further coordinate the stages' reasoning modes with their roles: non-thinking SFT for knowledge injection, thinking OPD for reasoning organization, and non-thinking RL for outcome activation.

Domain adaptation and capability retention. Training choices affect generalization as well as target-task performance (Chu et al., 2025). In domain adaptation, smaller SFT learning rates can reduce general-capability loss (Lin et al., 2026), and subsequent RL can recover out-of-distribution performance lost during SFT (Jin et al., 2025). Distillation provides another way to preserve existing capabilities while learning new ones: SDFT uses a demonstration-conditioned self-teacher (Shenfeld et al., 2026), while an OPD case study uses the model before adaptation to restore instruction following after domain training (Lu & Thinking Machines Lab, 2025). These studies motivate treating capability retention as a design goal alongside domain acquisition. Our focus is how expert-guided distillation and subsequent RL jointly shape this trade-off across the complete BWB-KD pipeline.

3 Method

3.1 Overview and Design Rationale

BWB-KD organizes black-box domain distillation into knowledge injection, reasoning organization, and outcome activation (Figure 2).

We consider a domain dataset $\mathcal{D} = \{(x, a)\}$, where x is a prompt and a is its reference answer. We have query access to a black-box teacher T and access to the weights of a hybrid-reasoning base model π_{θ_0}. Our goal is to transfer the teacher's domain expertise into a student while retaining the base model's general capabilities.

BWB-KD separates expert acquisition from student training. We first train a domain expert π_ϕ from the base model using black-box teacher demonstrations. We initialize a separate student π_θ from the same base parameters θ_0. The expert is then frozen, and its token distributions supervise the student on the student's own generated trajectories. This transfers domain knowledge through expert supervision rather than direct inheritance of the expert's weights. Finally, RL uses outcome feedback to refine the distilled student toward correct task responses.

We write π_θ^m for the policy under mode $m \in \{\text{OFF}, \text{ON}\}$. The chat template enables or disables the dedicated thinking channel, as in hybrid-reasoning models (Yang et al., 2025). Visible explanations can occur in either mode. We assign reasoning modes according to each stage's supervision and learning objective, yielding an OFF–ON–OFF schedule for knowledge injection, reasoning organization, and outcome activation.

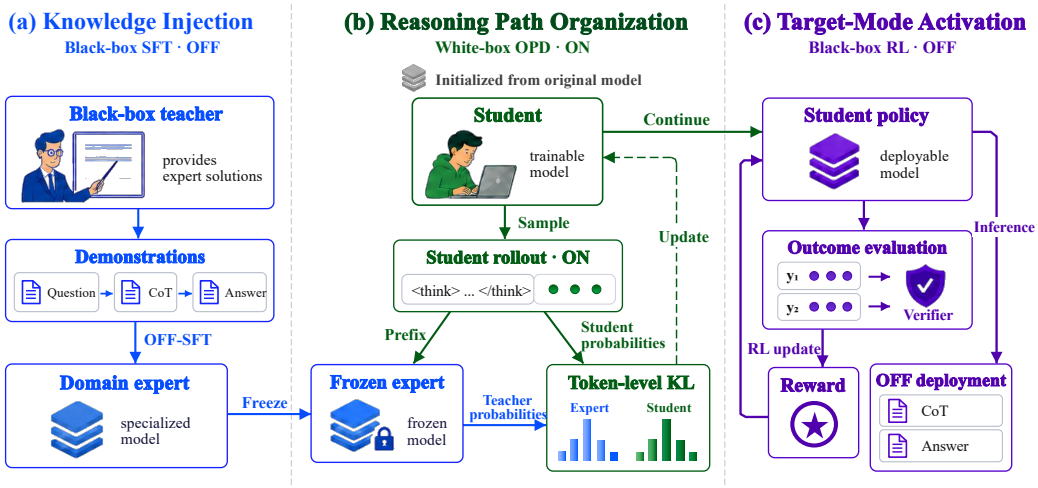

Figure 2: **Overview of BWB-KD.** (a) OFF SFT acquires a domain expert from black-box demonstrations. (b) The frozen expert guides a separate, Base-initialized student's ON trajectories through OPD. (c) The student continues with OFF outcome RL. The schedule coordinates knowledge injection, reasoning organization, and activation. OFF allows visible reasoning without the dedicated thinking channel.

3.2 Knowledge Injection through Black-Box SFT

The first stage trains a domain expert from black-box demonstrations. For each training prompt, we query T for a visible explanation and final answer. We retain responses that pass task-specific correctness and quality checks to form $\mathcal{D}_{\text{SFT}} = \{(x, y^T)\}$. Starting from θ_0, we train the expert using response-token cross-entropy, following sequence-level distillation (Kim & Rush, 2016).

In OFF mode, the expert learns both the teacher's explanation and answer as visible response tokens. This separates learning the teacher's supplied domain solutions from training the final student's own thinking process. The expert is frozen after SFT at ϕ^*, providing token distributions for the subsequent distillation stage.

3.3 Reasoning Organization through White-Box OPD

The second stage transfers domain knowledge by applying expert supervision to student-generated trajectories rather than fixed teacher explanations. We enable ON mode so that this supervision also covers the student's thinking tokens and their prefixes. The goal is to shape how the student uses domain knowledge while constructing a solution, by supervising next-token predictions along its own reasoning trajectories.

For each prompt x selected from the domain training set, the Base-initialized student samples a response $y \sim \pi_\theta^{\text{ON}}(\cdot \mid x)$. The frozen expert computes a next-token distribution at each student-generated prefix $h_t = (x, y_{<t})$. Although trained with OFF SFT, the expert uses the same ON context as the student when scoring these prefixes.

Following on-policy distillation (Agarwal et al., 2024), we generate new responses with the current student as training proceeds. For each batch $\mathcal{B} = \{(x, y)\}$, we hold the sampled prefixes fixed during the update and minimize a reverse-KL loss (Gu et al., 2024):

$$\mathcal{L}_{\text{OPD}}(\theta; \mathcal{B}) = \frac{1}{N_{\mathcal{B}}} \sum_{(x,y)\in\mathcal{B}} \sum_{t=1}^{|y|} D_{\text{KL}}(\widetilde{p}_\theta(\cdot \mid h_t) \,\|\, \widetilde{q}_{\phi^*}(\cdot \mid h_t)), \tag{1}$$

where $N_{\mathcal{B}}$ counts valid response tokens and $\widetilde{p}_\theta, \widetilde{q}_{\phi^*}$ are the ON student and frozen ON expert distributions, respectively, renormalized on the expert's top-K support. Only the student is updated. We denote the resulting student weights by θ_{OPD}. Appendix A gives the implementation details.

3.4 Outcome Activation through Black-Box RL

Matching expert distributions and producing correct answers are distinct objectives. The final stage shifts to outcome feedback: we continue from θ_{OPD}, switch thinking off, and apply GRPO (Shao et al., 2024). For each prompt, the student samples a group of responses. An outcome evaluator scores their extracted final answers against the reference. We normalize the rewards within each group to obtain advantages for the policy update. This interface is black-box: optimization uses scalar rewards rather than evaluator gradients or logits. The task-specific rewards and standard GRPO objective are specified in Appendix A.

Training in OFF applies outcome feedback directly to the distilled student's non-thinking generation. After organizing reasoning through ON distillation, we use OFF RL to turn that learning into successful answers without requiring the dedicated thinking channel at inference. The final checkpoint supports both inference modes. Sections 4.3 and 4.4 evaluate the stage order and mode assignments.

4 Experiments

4.1 Experimental Setup

Tasks and data. We study three biomedical adaptation tasks with Qwen3.6-27B: medical multiple-choice QA on MedXpertQA (Zuo et al., 2025), biomedical QA on BioASQ (Nentidis et al., 2026), and free-form diagnosis on MedCaseReasoning (Wu et al., 2025). The task branches are trained independently on 1,197, 1,812, and 7,671 queries, respectively. All stages draw from the same underlying query pool for each task. GPT-5.4 generates teacher demonstrations at temperature 1.0 and top-p 1.0. Appendices B and F give settings and prompts.

Training paths. We compare Base, SFT expert, and OPD student, each before and after GRPO. GRPO on the OPD student yields BWB-KD. The principal schedule uses OFF SFT, ON OPD from Base, and OFF GRPO. Additional comparisons vary distillation trajectories, stage order, reasoning modes, and the RL method using ROPD (Fang et al., 2026) and DAPO (Yu et al., 2025). SFT and OPD use learning rates of 2×10^{-6} and 2×10^{-7}, respectively, with an effective batch size of 32. GRPO uses eight responses per prompt.

Evaluation and selection. We evaluate both inference modes. MedXpert reports mean correctness over four responses to each of 577 questions. BioASQ uses a closed-book semantic macro average over four question types on 340 questions, and MedCaseReasoning uses single-response diagnostic accuracy on 897 test cases. The MedXpert-trained models are also evaluated on fixed sets from AIME25, ArenaHard (Li et al., 2025), BBEH (Kazemi et al., 2025), and IFBench (Pyatkin et al., 2025). We report their individual scores and arithmetic mean. For each method, we select its best checkpoint according to validation-set performance in OFF mode, then evaluate that same checkpoint in both OFF and ON modes (Appendix B.5). Bold marks BWB-KD throughout the result tables.

4.2 Domain Transfer and General-Capability Retention

BWB-KD achieves the highest score on each domain task in both inference modes (Table 1). Averaged across the three tasks, it scores 49.46 in OFF and 50.92 in ON, improving over SFT+GRPO by 2.46 and 0.63 points, respectively. The length analysis further shows that BWB-KD's OFF policy approaches SFT+GRPO's ON accuracy with substantially shorter outputs (Section 5.1).

These domain gains accompany substantial general-capability retention (Table 2). In OFF, SFT reduces the general mean from 63.36 to 48.99, and subsequent GRPO leaves it at 49.20. OPD and BWB-KD instead score 61.40 and 61.54, close to Base. In ON, the six training paths have similar general means, ranging from 71.12 to 73.01. BWB-KD thus improves domain performance while avoiding the large OFF general-capability deficit of SFT-initialized routes.

4.3 Coordinating Knowledge Transfer and Reinforcement Learning

On-policy versus offline distillation. OPD and offline forward-KL (FKL) and reverse-KL (RKL) distillation share the SFT expert and Base initialization (Table 3). Unlike OPD, offline distillation

Table 1: **Domain results with Qwen3.6-27B.** Each task uses a separately trained model. MedCase denotes MedCaseReasoning; Mean averages the three tasks. The OPD student is Base-initialized and supervised by the SFT expert.

	Thinking OFF				Thinking ON			
Method	MedXpert	BioASQ	MedCase	Mean	MedXpert	BioASQ	MedCase	Mean
Base	39.82	45.72	48.94	44.83	43.24	45.78	50.72	46.58
SFT expert	40.29	49.10	48.05	45.81	47.92	48.36	52.51	49.60
OPD student	43.67	49.14	49.94	47.58	46.10	48.92	51.51	48.84
Base+GRPO	41.38	47.06	50.31	46.25	43.15	47.37	51.76	47.43
SFT+GRPO	41.07	50.67	49.25	47.00	47.44	51.11	52.31	50.29
BWB-KD	**46.45**	**50.91**	**51.03**	**49.46**	**48.35**	**51.14**	**53.26**	**50.92**

Table 2: **General capability of the MedXpert-trained models.** The same six training paths are evaluated in both modes on the frozen general benchmarks. Mean is their arithmetic average.

	Thinking OFF					Thinking ON				
Method	AIME 25	Arena Hard	BBEH	IF Bench	Mean	AIME 25	Arena Hard	BBEH	IF Bench	Mean
Base	79.31	53.94	77.00	43.20	63.36	94.83	51.83	83.25	57.82	71.93
SFT expert	51.72	40.91	63.75	39.59	48.99	94.83	56.24	82.00	58.16	72.81
OPD student	75.86	54.60	71.25	43.88	61.40	94.83	51.73	83.50	54.42	71.12
Base+GRPO	79.31	54.69	73.25	43.20	62.61	96.55	52.39	85.25	57.82	73.00
SFT+GRPO	50.00	42.09	65.25	39.46	49.20	94.83	55.15	84.25	57.82	73.01
BWB-KD	**72.41**	**57.31**	**74.25**	**42.18**	**61.54**	**93.10**	**53.64**	**84.25**	**57.14**	**72.03**

uses fixed teacher trajectories. All three have OFF general means of 61.21–64.22, versus SFT's 48.99. Under reverse KL, OPD achieves higher domain accuracy in both modes, with a larger advantage after GRPO. BWB-KD also exceeds FKL+GRPO, the stronger offline baseline, by 1.04 MedXpert points in OFF and 1.38 in ON. These results favor OPD for the intermediate transfer stage: beyond preserving general capabilities, it transfers domain expertise more effectively, with gains that persist after RL.

Initialization for RL. Averaged across the three tasks, OPD students gain more from subsequent GRPO than SFT experts (Table 11). In ON inference, the SFT expert initially has a higher domain mean than the OPD student (49.60 versus 48.84), but the ordering reverses after GRPO (50.29 versus 50.92). Intermediate accuracy alone therefore does not determine the quality of an RL starting point: reasoning organization prepares the student to benefit more from subsequent RL.

Stage order. On MedXpert, *RL-expert OPD* applies GRPO to the SFT expert before distilling it into a fresh Base-initialized student. BWB-KD instead applies GRPO to the student after distillation, achieving 1.48 points higher accuracy in OFF and 0.99 in ON, with similar output lengths and general-capability retention (Table 3). This supports placing outcome activation after reasoning organization, so that RL builds on the student's acquired domain capabilities.

4.4 Stage-Specific Reasoning Modes

Table 4 compares stage-specific training modes under both inference modes. For knowledge injection, BWB-KD trains the expert with OFF SFT on teacher-provided explanations and answers. Changing SFT to ON lowers MedXpert accuracy from 40.29/47.92 to 39.17/39.60 in OFF/ON inference and reduces ON general performance from 72.81 to 61.09. These results favor learning teacher solutions as visible responses to construct the domain expert for OPD.

For reasoning organization, BWB-KD uses ON OPD to supervise the student's thinking trajectories. Disabling thinking during OPD lowers OFF accuracy from 43.67 to 39.30, with similar general-

Table 3: **Knowledge transfer and stage order on MedXpert.** Base is the unadapted reference. FKL/RKL use fixed teacher trajectories. RL-expert OPD transfers an SFT+GRPO expert into a fresh Base student; BWB-KD applies GRPO after OPD. General is the four-benchmark mean.

	Thinking OFF		Thinking ON	
Method	MedXpert	General	MedXpert	General
Base	39.82	63.36	43.24	71.93
Offline FKL	41.51	61.21	45.62	69.74
Offline RKL	41.20	64.22	45.54	70.34
OPD student	43.67	61.40	46.10	71.12
FKL+GRPO	45.41	63.13	46.97	69.15
RKL+GRPO	42.37	62.42	45.28	67.84
RL-expert OPD	44.97	62.58	47.36	69.53
BWB-KD	**46.45**	**61.54**	**48.35**	**72.03**

Table 4: **Stage-specific reasoning modes on MedXpert.** Left: training modes. Right: accuracy, general mean, and mean generated tokens. The all-dash row is Base; other dashes indicate omitted stages. Tokens include thinking tokens when enabled. Bold identifies BWB-KD.

Training mode			Thinking OFF			Thinking ON		
SFT	OPD	RL	MedXpert	General	Tokens	MedXpert	General	Tokens
–	–	–	39.82	63.36	1613.6	43.24	71.93	3193.0
OFF	–	–	40.29	48.99	285.1	47.92	72.81	2827.8
ON	–	–	39.17	56.78	394.8	39.60	61.09	544.7
OFF	ON	–	43.67	61.40	1408.0	46.10	71.12	2936.6
OFF	OFF	–	39.30	61.67	295.0	44.76	70.96	2753.6
ON	ON	–	42.59	60.30	1003.8	46.53	71.29	1571.1
OFF	**ON**	**OFF**	**46.45**	**61.54**	**1541.8**	**48.35**	**72.03**	**3073.5**
OFF	OFF	OFF	40.03	62.97	315.9	46.71	71.83	2799.1
OFF	ON	ON	45.58	62.66	2377.6	49.05	72.98	5518.8
ON	ON	ON	44.18	63.24	1438.2	48.27	72.89	3604.1

capability retention. After both students undergo OFF RL, their OFF accuracies reach 46.45 and 40.03, respectively, further widening the gap. Thinking-mode transfer thus improves domain performance even when inference disables the thinking channel.

For outcome activation, BWB-KD uses OFF RL to reinforce non-thinking generation. Changing only the final RL stage to ON lowers OFF accuracy from 46.45 to 45.58, while raising ON accuracy from 48.35 to 49.05 and increasing output length (Section 5.1). This trade-off motivates retaining OFF RL in the final stage. An all-ON schedule achieves 44.18/48.27 in OFF/ON inference, below BWB-KD in both modes. Together, these results support assigning reasoning modes to individual stages rather than enabling thinking throughout training.

4.5 Can Alternative RL Methods Replace the Pipeline?

We compare GRPO, ROPD, and DAPO from Base and SFT initializations to test whether changing the RL method can replace BWB-KD's staged knowledge transfer (Table 5).

Starting from Base, DAPO achieves strong domain performance while largely retaining general capabilities, as does BWB-KD. BWB-KD nevertheless achieves higher MedXpert accuracy by 0.91/4.63 points in OFF/ON inference, with shorter mean outputs: 1,541.8/3,073.5 versus 2,705.9/4,277.9 tokens. These results support transferring domain expertise before RL rather than relying on outcome feedback alone.

Table 5: **Replacing RL versus using the BWB pipeline.** GRPO, ROPD, and DAPO start from Base or the SFT expert. General is the four-benchmark mean. Tokens are mean generated tokens on MedXpert.

	Thinking OFF			Thinking ON		
Method	MedXpert	General	Tokens	MedXpert	General	Tokens
Base	39.82	63.36	1613.6	43.24	71.93	3193.0
Base+GRPO	41.38	62.61	1881.4	43.15	73.00	3211.2
Base+ROPD	41.46	59.61	1503.8	43.02	67.17	3313.3
Base+DAPO	45.54	65.12	2705.9	43.72	66.46	4277.9
SFT+GRPO	41.07	49.20	373.8	47.44	73.01	2818.2
SFT+ROPD	42.59	51.34	669.4	48.31	70.18	3441.5
SFT+DAPO	39.17	48.87	157.2	47.83	70.13	2644.1
BWB-KD	**46.45**	**61.54**	**1541.8**	**48.35**	**72.03**	**3073.5**

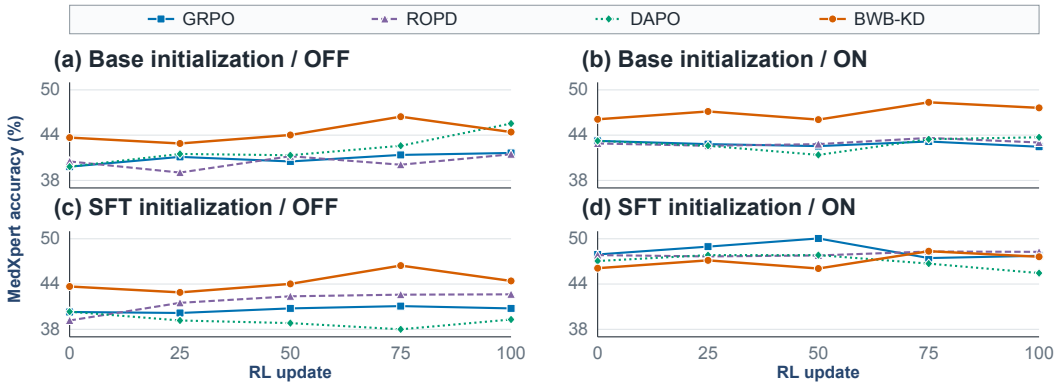

Figure 3: **RL trajectories on MedXpert.** GRPO, ROPD, and DAPO are grouped by initialization and inference mode. BWB-KD starts from its OPD student and is repeated as a reference in each panel. Points are measured every 25 RL updates.

Applying RL after SFT leaves the large OFF general-capability deficit unresolved. GRPO, ROPD, and DAPO obtain general means of 49.20, 51.34, and 48.87, versus 61.54 for BWB-KD. SFT+ROPD approaches BWB-KD in ON domain accuracy (48.31 versus 48.35), but remains behind in OFF (42.59 versus 46.45). The unresolved deficit motivates transferring the expert's knowledge to a separate student before RL, rather than seeking recovery solely by changing how the expert is reinforced.

Figure 3 highlights the role of initialization in RL training. With the same DAPO method, Base initialization yields gains in OFF accuracy, whereas SFT initialization ends below its starting accuracy. Changing the RL method alone therefore does not ensure effective adaptation; the policy entering RL also matters. This finding reinforces BWB-KD's design of transferring domain expertise to the student before outcome-based RL.

5 Analysis

5.1 Accuracy and Generation Length

Training-mode coordination. Compared with the OFF–ON–ON schedule, BWB-KD generates 35.2% fewer tokens and scores 0.87 points higher in OFF (Table 4, Figure 4). In ON, it uses 44.3% fewer tokens with 0.70-point lower accuracy. Longer outputs after ON RL thus do not improve OFF accuracy.

Inference-mode trade-off. Within BWB-KD, OFF uses 49.8% fewer tokens than ON, scoring 46.45 versus 48.35. SFT+GRPO's OFF outputs are shorter (373.8 versus 1,541.8 tokens), but its accuracy is

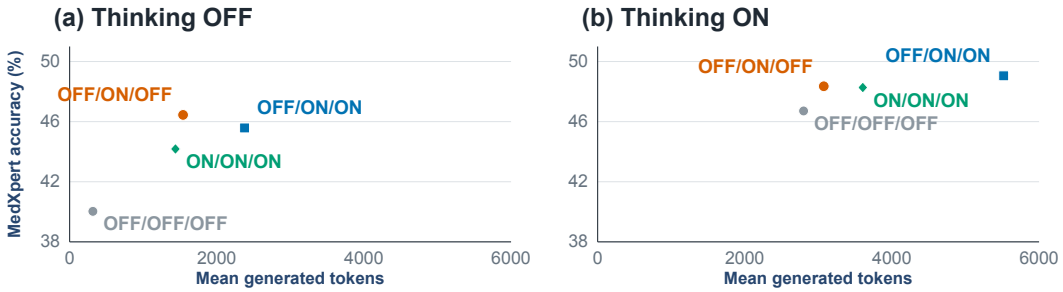

Figure 4: **Reasoning-mode schedules and output length.** Point labels list SFT/OPD/RL modes. Both panels use the same axes; length includes thinking tokens when enabled.

Table 6: **Question-level recovery and paired uncertainty on MedXpert.** Gains and 95% CIs are in pp. Recovery counts target OFF successes within the source ON-strong/OFF-weak cohort. ON retention and joint outcomes appear in Appendix D.

Comparison	OFF gain [95% CI]	ON gain [95% CI]	OFF recovery
SFT → OPD	+3.38 [+0.65, +6.11]	-1.82 [-4.03, +0.43]	20/41 (48.78%)
SFT+RL → BWB	+5.37 [+2.82, +8.02]	+0.91 [-1.30, +3.12]	29/48 (60.42%)
Base → OPD	+3.86 [+1.65, +6.02]	+2.86 [+0.87, +4.94]	19/28 (67.86%)
Base+RL → BWB	+5.07 [+2.99, +7.19]	+5.20 [+2.95, +7.54]	15/21 (71.43%)

lower (41.07). BWB-KD's OFF accuracy instead approaches SFT+GRPO's ON score of 47.44, with 45.3% fewer tokens (1,541.8 versus 2,818.2), offering an effective non-thinking alternative.

5.2 Question-Level Recovery across Inference Modes

Compared with SFT+GRPO, BWB-KD reduces the difference between ON and OFF domain accuracy from 3.29 to 1.46 percentage points. General performance shows a similar narrowing of the mode gap. We therefore examine whether the improvement in OFF performance includes questions that SFT+GRPO could previously answer reliably only in ON mode, using question-level results on MedXpert.

We first consider 48 questions that SFT+GRPO answers correctly at least three times out of four in ON mode, but at most once in OFF mode. BWB-KD answers 26 of these questions correctly at least twice in each mode, yielding a joint recovery rate of 54.17%. Across the full test set, the number of questions answered correctly at least three times in both modes also increases, from 160 to 196 out of 577. These results show that BWB-KD improves OFF performance on previously ON-dependent questions while expanding the set of questions answered successfully in both modes.

Paired bootstrap analysis further supports the OFF accuracy gain over SFT+GRPO, which is 5.37 percentage points with a 95% confidence interval of $[2.82, 8.02]$. The ON gain is smaller, and its confidence interval includes zero (Table 6). Compared with Base+GRPO, BWB-KD improves accuracy in both modes, with both confidence intervals above zero. Appendix D provides the resampling procedure and detailed recovery counts.

6 Conclusion

We presented BWB-KD for black-box domain knowledge distillation. It coordinates knowledge injection through OFF SFT, reasoning organization through ON OPD, and outcome activation through OFF RL. Across three biomedical tasks, BWB-KD improves domain performance in both inference modes, especially non-thinking inference, while largely preserving general capabilities. Ablations and baseline comparisons support transferring expert knowledge to the student before RL and assigning reasoning modes according to each stage's role.

AI USE STATEMENT

AI tools generated black-box teacher demonstrations, evaluated responses as LLM judges, and assisted with literature retrieval and discovery, writing, language editing, code debugging, and organizing and packaging the supplementary code. The authors are responsible for the final content.

REPRODUCIBILITY STATEMENT

Section 3 and Appendices A–B detail training and evaluation. Appendices C–D provide supplementary results; Appendix F provides generation and verification prompts. Code for the main experiments is included in the supplementary materials.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 10818–10838, 2025. URL https://proceedings.mlr.press/v267/chu25c.html.

Junfeng Fang, Zhepei Hong, Mao Zheng, Mingyang Song, Gengsheng Li, Houcheng Jiang, Dan Zhang, Haiyun Guo, Xiang Wang, and Tat-Seng Chua. Rubric-based On-policy Distillation. *arXiv preprint arXiv:2605.07396*, 2026. URL https://arxiv.org/abs/2605.07396.

Yuqian Fu, Tinghong Chen, Jiajun Chai, Xihuai Wang, Songjun Tu, Guojun Yin, Wei Lin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. SRFT: A Single-Stage Method with Supervised and Reinforcement Fine-Tuning for Reasoning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a79699db176ed0efc04a9da171e52112-Abstract-Conference.html.

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://proceedings.iclr.cc/paper_files/paper/2024/hash/8ac015d409635f196f9e3e9dcfb9a94e-Abstract-Conference.html.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8003–8017, 2023. URL https://aclanthology.org/2023.findings-acl.507/.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What Disease Does This Patient Have? A Large-Scale Open Domain Question Answering Dataset from Medical Exams. *Applied Sciences*, 11(14):6421, 2021. doi: 10.3390/app11146421. URL https://doi.org/10.3390/app11146421.

Hangzhan Jin, Sitao Luan, Tianwei Ni, Sicheng Lyu, Guillaume Rabusseau, Reihaneh Rabbany, Doina Precup, and Mohammad Hamdaqa. RL Fine-Tuning Heals OOD Forgetting in SFT. *arXiv preprint arXiv:2509.12235*, 2025. URL https://arxiv.org/abs/2509.12235.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. PubMedQA: A Dataset for Biomedical Research Question Answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2567–2577. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1259. URL https://aclanthology.org/D19-1259/.

Mehran Kazemi, Bahare Fatemi, Hritik Bansal, John Palowitch, Chrysovalantis Anastasiou, Sanket Vaibhav Mehta, Lalit K. Jain, Virginia Aglietti, Disha Jindal, Peter Chen, Nishanth Dikkala, Gladys Tyen, Xin Liu, Uri Shalit, Silvia Chiappa, Kate Olszewska, Yi Tay, Vinh Q. Tran, Quoc V. Le, and Orhan Firat. BIG-Bench Extra Hard. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26473–26501, 2025. URL https://aclanthology.org/2025.acl-long.1285/.

Yoon Kim and Alexander M. Rush. Sequence-Level Knowledge Distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, 2016. URL https://aclanthology.org/D16-1139/.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From Crowdsourced Data to High-quality Benchmarks: Arena-Hard and Benchbuilder Pipeline. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34209–34231. PMLR, 2025. URL https://proceedings.mlr.press/v267/li25h.html.

Jiacheng Lin, Zhongruo Wang, Kun Qian, Tian Wang, Arvind Srinivasan, Hansi Zeng, Ruochen Jiao, Xie Zhou, Jiri Gesi, Dakuo Wang, Yufan Guo, Kai Zhong, Weiqi Zhang, Sujay Sanghavi, Changyou Chen, Hyokun Yun, and Lihong Li. SFT Doesn't Always Hurt General Capabilities: Revisiting Domain-Specific Fine-Tuning in LLMs. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/4e447acb68f57e29234bc0eb19896f11-Abstract-Conference.html.

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

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, Bin Cui, and Wentao Zhang. Learning What Reinforcement Learning Can't: Interleaved Online Fine-Tuning for Hardest Questions. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/82c4ad0c3d3d4898a77be5df7b2ede3a-Abstract-Conference.html.

Anastasios Nentidis, Georgios Katsimpras, Anastasia Krithara, Martin Krallinger, Miguel Rodríguez-Ortega, Eduard Rodriguez-López, Natalia Loukachevitch, Andrey Sakhovskiy, Elena Tutubalina, Dimitris Dimitriadis, Grigorios Tsoumakas, George Giannakoulas, Alexandra Bekiaridou, Athanasios Samaras, Giorgio Maria Di Nunzio, Nicola Ferro, Stefano Marchesin, Marco Martinelli, Gianmaria Silvello, and Georgios Paliouras. Overview of BioASQ 2025: The Thirteenth BioASQ Challenge on Large-Scale Biomedical Semantic Indexing and Question Answering. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction (CLEF 2025)*, volume 16089 of *Lecture Notes in Computer Science*, pp. 173–198. Springer Nature Switzerland, 2026. doi: 10.1007/978-3-032-04354-2_12. URL https://link.springer.com/chapter/10.1007/978-3-032-04354-2_12.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A Large-scale Multi-Subject Multi-Choice Dataset for Medical domain Question Answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 248–260. PMLR, 2022. URL https://proceedings.mlr.press/v174/pal22a.html.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. Generalizing Verifiable Instruction Following. In *Advances in Neural Information Processing Systems*, volume 38, pp. 54870–54896, 2025. doi: 10.52202/085713-1645. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/46499a0622ecf568b72d17b61e45dbd5-Abstract-Datasets_and_Benchmarks_Track.html.

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, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-Distillation Enables Continual Learning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/61434.

Luong Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. ReFT: Reasoning with Reinforced Fine-Tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024. URL https://aclanthology.org/2024.acl-long.410/.

Sudong Wang, Weiquan Huang, Xiaomin Yu, Zuhao Yang, Hehai Lin, Keming Wu, Chaojun Xiao, Chen Chen, Wenxuan Wang, Beier Zhu, Yunjian Zhang, and Chengwei Qin. Beyond SFT-to-RL: Pre-alignment via Black-Box On-Policy Distillation for Multimodal RL. *arXiv preprint arXiv:2604.28123*, 2026. URL https://arxiv.org/abs/2604.28123.

Kevin Wu, Eric Wu, Rahul Thapa, Kevin Wei, Angela Zhang, Arvind Suresh, Jacqueline J. Tao, Min Woo Sun, Alejandro Lozano, and James Zou. MedCaseReasoning: Evaluating and Learning Diagnostic Reasoning from Clinical Case Reports. *arXiv preprint arXiv:2505.11733*, 2025. URL https://arxiv.org/abs/2505.11733.

Jing Xiong, Hui Shen, Shansan Gong, Yuxin Cheng, Jianghan Shen, Chaofan Tao, Haochen Tan, Haoli Bai, Lifeng Shang, and Ngai Wong. OVD: On-policy Verbal Distillation. *arXiv preprint arXiv:2601.21968*, 2026. doi: 10.48550/arXiv.2601.21968. URL https://arxiv.org/abs/2601.21968.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-Box On-Policy Distillation of Large Language Models. *arXiv preprint arXiv:2511.10643*, 2025. doi: 10.48550/arXiv.2511.10643. URL https://arxiv.org/abs/2511.10643.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An Open-Source LLM Reinforcement Learning System at Scale. In *Advances in Neural Information Processing Systems*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Yuxin Zuo, Shang Qu, Yifei Li, Zhang-Ren Chen, Xuekai Zhu, Ermo Hua, Kaiyan Zhang, Ning Ding, and Bowen Zhou. MedXpertQA: Benchmarking Expert-Level Medical Reasoning and Understanding. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. URL https://proceedings.mlr.press/v267/zuo25a.html.

A Training Objectives and Implementation

A.1 Supervised Expert Acquisition

For a batch $\mathcal{B}$ of selected teacher demonstrations, the SFT objective is

$$\mathcal{L}_{\mathrm{SFT}}(\phi;\mathcal{B}) = -\frac{1}{N_{\mathcal{B}}} \sum_{(x,y^T)\in\mathcal{B}} \sum_{t=1}^{|y^T|} \log \pi_\phi^{\mathrm{OFF}}(y_t^T \mid x, y_{<t}^T), \tag{2}$$

where $N_{\mathcal{B}}$ counts supervised response tokens. Prompt and padding tokens are masked. The target contains the teacher's visible explanation and final answer. The expert is initialized from θ_0 and frozen at ϕ^* after training.

A.2 On-Policy Distribution Supervision

The OPD student is independently initialized from θ_0. Training uses student rollouts throughout, rather than replaying the SFT target as the continuation. At prefix h_t, let $\mathcal{V}_t$ be the expert's top-K tokens, with $K = 32$. We apply softmax to the temperature-scaled student and expert logits restricted to $\mathcal{V}_t$, yielding $\widetilde{p}_t$ and $\widetilde{q}_t$. The reverse-KL term in Equation 1 is

$$D_{\mathrm{KL}}(\widetilde{p}_t \| \widetilde{q}_t) = \sum_{v\in\mathcal{V}_t} \widetilde{p}_t(v) \log \frac{\widetilde{p}_t(v)}{\widetilde{q}_t(v)}. \tag{3}$$

Both distributions are normalized on this restricted support. The loss averages valid generated tokens, including thinking tokens. Gradients flow only through the student distribution, with no backpropagation through sampling or updates to the expert. The resulting student weights initialize RL.

A.3 Outcome Rewards and GRPO

For each prompt, the rollout policy $\pi_{\mathrm{old}}^{\mathrm{OFF}}$ samples G responses. An evaluator assigns $r_i = R(x, \mathrm{Extract}(y_i), a)$. MedXpert uses option correctness, MedCaseReasoning uses a semantic diagnosis judge, and BioASQ requires two final-answer judges to agree that the answer is correct. All three provide binary rewards without scoring intermediate reasoning.

Let $\bar{r}_x$ and s_x denote the mean and standard deviation of the rewards within a prompt's group. With $\epsilon > 0$ stabilizing normalization, the response-level advantage and token probability ratio are

$$\widehat{A}_i = \frac{r_i - \bar{r}_x}{s_x + \epsilon}, \qquad \rho_{i,t}(\theta) = \frac{\pi_\theta^{\mathrm{OFF}}(y_{i,t} \mid x, y_{i,<t})}{\pi_{\mathrm{old}}^{\mathrm{OFF}}(y_{i,t} \mid x, y_{i,<t})}. \tag{4}$$

Using $\langle\cdot\rangle_{\mathcal{B}}$ to average valid response tokens in the rollout batch, GRPO maximizes

$$\mathcal{J}_{\mathrm{RL}}(\theta) = \mathbb{E}_{\mathcal{B}}\Big[\Big\langle \min\Big\{\rho_{i,t}\widehat{A}_i, \mathrm{clip}(\rho_{i,t}, 1-\delta, 1+\delta)\widehat{A}_i\Big\} - \lambda_{\mathrm{KL}} k_{i,t}\Big\rangle_{\mathcal{B}}\Big], \tag{5}$$

where δ is the clipping radius. The reference policy is a frozen copy of the RL initialization, $\pi_{\mathrm{ref}} = \pi_{\theta_{\mathrm{OPD}}}$. We use the sampled KL penalty $k_{i,t} = \exp(d_{i,t}) - d_{i,t} - 1$, with $d_{i,t} = \log \pi_{\mathrm{ref}}^{\mathrm{OFF}}(y_{i,t} \mid x, y_{i,<t}) - \log \pi_\theta^{\mathrm{OFF}}(y_{i,t} \mid x, y_{i,<t})$. The penalty is applied in the policy objective, separately from the outcome reward. The main configuration uses $G = 8$ and $\lambda_{\mathrm{KL}} = 0.001$.

B Data, Training, and Evaluation

B.1 Task-Specific Adaptation Data

The final 1,197-query MedXpert training set contains 500 MedMCQA (Pal et al., 2022), 191 MedQA (Jin et al., 2021), 58 PubMedQA (Jin et al., 2019), and 448 MedXpert-source questions. A separate 63-question development set is reserved for validation. Data construction combines questions the Base model finds difficult with easier anchor questions, retains teacher-correct demonstrations, and deduplicates by sample identifier before a deterministic training/development split. Evaluation uses

Table 7: **Adaptation and evaluation sizes.** Each task uses one underlying query pool for SFT, OPD, and RL. OFF validation performance selects one checkpoint for both test modes. The task branches are trained independently.

Task	Train	Validation	Test	Domain metric
MedXpert	1,197	63	577	Mean correctness, four responses
BioASQ	1,812	200	340	Four-type semantic macro average
MedCaseReasoning	7,671	500	897	Single-response diagnosis accuracy

Table 8: **Core training settings.** The expert and OPD student are independently Base-initialized. RL continues the indicated student or baseline initialization.

Setting	SFT	OPD	Main GRPO
Training mode	OFF	ON	OFF
Effective prompt batch	32	32	32
Learning rate	2×10^{-6}	2×10^{-7}	5×10^{-7}
Supervision	Response CE	Teacher top-32 RKL	Outcome reward
Rollout group size	–	1	8
Initialization	Base	Base	Branch checkpoint
Frozen model	–	SFT expert	Initial reference policy
KL regularization coefficient	–	–	0.001
Policy rollouts per update	–	32	32×8

a fixed 577-question MedXpert held-out subset. BioASQ uses task-13b training questions and a 340-question golden evaluation set, with a separate 200-question validation set. MedCaseReasoning uses clinical diagnostic cases from the official training split, a 500-case validation set, and the official 897-case test split. For each task, all three stages share the underlying training query pool in Table 7.

Teacher demonstrations. We use GPT-5.4 at temperature 1.0 and top-p 1.0 to generate visible explanations and final answers. Task-specific correctness checks select the demonstrations: option matching for medical multiple-choice QA and semantic answer verification for BioASQ and MedCaseReasoning. The retained response uses `<reasoning>` and `<answer>` blocks. SFT supervises both blocks with hidden thinking disabled; prompt and padding tokens are masked. Appendix F gives the task templates and diagnostic and biomedical answer-verification prompts.

B.2 Optimization Settings

SFT uses a 4,096-token sequence limit. OPD uses a 32,768-token context limit, a 1,500-token completion limit, and a warmup ratio of 0.05. Main GRPO uses sampling temperature 0.9, top-p 1.0, and top-k 32, with a constant learning-rate schedule and weight decay 0.01. The PPO mini-batch contains 32 prompt groups. MedXpert uses prompt and response limits of 2,048 and 8,192 tokens, respectively.

B.3 Baseline Configurations

Offline FKL and RKL share learning rate 2×10^{-7}, effective batch 32, completion limit 1,500, and teacher top-32 logits. They use fixed teacher-generated trajectories rather than current student rollouts. The reversed-order student uses the SFT+GRPO expert, the same underlying MedXpert query pool, batch 32, and ON-mode OPD.

ROPD follows the released rubric construction and verification procedure (Fang et al., 2026). GPT-5.4 supplies four offline references per training query and serves as the online rubricator and verifier. Reference generation uses temperature 1.0, top-p 1.0, and an 8,192-token output limit. The rubricator and verifier use high reasoning effort and the same output limit. Both Base- and SFT-initialized ROPD use OFF training, prompt batch 32, eight policy rollouts per query, and learning rate 10^{-6}.

DAPO (Yu et al., 2025) uses asymmetric ratio clipping with lower and upper radii 0.2 and 0.28, token-mean loss aggregation, and dynamic sampling. It generates batches of 96 prompts and removes groups with zero reward variance, accumulating 32 retained prompt groups per update with at most ten generation batches. Each prompt has eight responses. The learning rate is 5×10^{-7}, with zero warmup and weight decay 0.01. Rollout temperature, top-p, and top-k are 0.9, 1.0, and 32. The response limit is 8,192 tokens, with a 2,048-token overlong buffer and penalty factor 1.0. Both reward-side and actor-side KL penalties are disabled. Base and SFT initializations use the same DAPO configuration.

B.4 Domain and General Evaluation

MedXpert correctness is determined by matching the extracted option against the reference. The extraction procedure handles answer tags, option identifiers, and common formatting variants. Each question has four sampled responses per inference mode. We average correctness across responses and then across questions, rather than counting a question as correct whenever any sample succeeds.

BioASQ evaluation uses the 340-question golden set without reference snippets. Final answers are normalized by question type: polarity for yes/no, entity or numerical answers for factoid, entity sets for list, and complete answer text for summary. Two independent GPT-5.4 judgments must both accept the answer as correct or an acceptable semantic variant. The reported score equally weights yes/no, factoid, list, and summary accuracy. This is a closed-book semantic adaptation evaluation, distinct from the challenge's retrieval-assisted exact-answer and summary metrics.

MedCaseReasoning is scored using final-diagnosis semantic correctness with GPT-5.4. The judge receives the prediction and reference diagnosis and returns a binary yes/no decision. The reported metric is single-response diagnostic accuracy.

The general suite uses frozen evaluation sets with 58 AIME25 items, 750 ArenaHard items, 400 BBEH items, and 294 IFBench prompts. AIME25 uses numerical-answer extraction and rule scoring; BBEH uses task-specific rule scoring. ArenaHard uses GPT-5.4 pairwise judgments in both answer orders against fixed reference answers stored with the evaluation records. The same reference texts and frozen win-rate aggregation are used for every model. IFBench uses prompt-level strict verification with its deterministic constraint evaluator. The general mean is an arithmetic summary of these four scores.

Thinking-enabled generation uses the Qwen reasoning parser, with final answers passed to the task evaluator. Output-token statistics include generated thinking tokens when enabled and visible response tokens. IFBench uses temperature 0 and a 32,768-token output limit. The frozen Hard3 generation configuration uses temperature 1.0, top-p 0.95, and top-k 20, with a 65,536-token context limit and no separate output cap. The ArenaHard judge uses temperature 1.0 and an 8,192-token output limit. MedXpert uses temperature 0.6, top-p 0.95, top-k 20, and an 8,192-token output limit. ArenaHard scoring aggregates valid judgments; the reversed-order model has 746/750 valid OFF and 741/750 valid ON items.

For option-scored tasks, an incorrect or unparseable final answer is scored as incorrect. Failed API requests are retried and logged separately from answer correctness. BioASQ judgments evaluate final answers against gold aliases, ideal answers, and evidence; the evaluated model receives no gold evidence.

B.5 Model Selection

For each method, we select its best checkpoint according to validation-set performance in OFF mode, using the task-specific domain metric. SFT and OPD have a budget of up to three epochs, and GRPO has a budget of up to 100 updates. We then evaluate the same selected checkpoint on the test set in both OFF and ON modes.

C Additional Benchmark Results

Table 9 expands the distillation and stage-order comparisons in Table 3; Table 10 expands the alternative-RL comparisons in Table 5. Both report individual general benchmarks alongside MedX-

Table 9: **Transfer and stage-order alternatives: individual benchmark scores.** Base and BWB-KD provide common reference points in both inference modes. Mean averages the four general benchmarks, excluding MedXpert.

Method	Mode	MedXpert	AIME25	ArenaHard	BBEH	IFBench	Mean
Base	OFF	39.82	79.31	53.94	77.00	43.20	63.36
Offline FKL	OFF	41.51	72.41	51.71	77.50	43.20	61.21
Offline RKL	OFF	41.20	84.48	54.18	75.00	43.20	64.22
FKL+GRPO	OFF	45.41	77.59	53.96	76.75	44.22	63.13
RKL+GRPO	OFF	42.37	81.03	54.29	72.50	41.84	62.42
RL-expert OPD	OFF	44.97	81.03	51.60	74.50	43.20	62.58
BWB-KD	**OFF**	**46.45**	**72.41**	**57.31**	**74.25**	**42.18**	**61.54**
Base	ON	43.24	94.83	51.83	83.25	57.82	71.93
Offline FKL	ON	45.62	96.55	52.73	70.50	59.18	69.74
Offline RKL	ON	45.54	96.55	52.24	73.75	58.84	70.34
FKL+GRPO	ON	46.97	93.10	53.04	70.25	60.20	69.15
RKL+GRPO	ON	45.28	94.83	52.81	67.25	56.46	67.84
RL-expert OPD	ON	47.36	94.83	50.92	75.25	57.14	69.53
BWB-KD	**ON**	**48.35**	**93.10**	**53.64**	**84.25**	**57.14**	**72.03**

Table 10: **Alternative RL: individual benchmark scores.** Base and BWB-KD provide common reference points in both inference modes. Mean averages the four general benchmarks, excluding MedXpert.

Method	Mode	MedXpert	AIME25	ArenaHard	BBEH	IFBench	Mean
Base	OFF	39.82	79.31	53.94	77.00	43.20	63.36
Base+ROPD	OFF	41.46	72.41	53.73	72.50	39.80	59.61
Base+DAPO	OFF	45.54	84.48	57.67	76.50	41.84	65.12
SFT+ROPD	OFF	42.59	55.17	42.38	70.75	37.07	51.34
SFT+DAPO	OFF	39.17	51.72	40.33	62.25	41.16	48.87
BWB-KD	**OFF**	**46.45**	**72.41**	**57.31**	**74.25**	**42.18**	**61.54**
Base	ON	43.24	94.83	51.83	83.25	57.82	71.93
Base+ROPD	ON	43.02	94.83	51.61	64.75	57.48	67.17
Base+DAPO	ON	43.72	93.10	53.04	63.25	56.46	66.46
SFT+ROPD	ON	48.31	94.83	54.81	73.25	57.82	70.18
SFT+DAPO	ON	47.83	94.83	53.46	74.75	57.48	70.13
BWB-KD	**ON**	**48.35**	**93.10**	**53.64**	**84.25**	**57.14**	**72.03**

pert accuracy. The OFF general-capability deficit is concentrated in SFT-initialized paths, while offline and on-policy distillation both retain much of the Base model's general performance. Table 11 supports the RL-initialization analysis in Section 4.3 by tabulating pre-RL to post-RL differences from Tables 1 and 2. The BioASQ difference in Table 11 compares separately selected pre-RL and post-RL models.

D Question-Level Statistics

For each source–target comparison, let $c_m^S(x)$ and $c_m^T(x)$ count correct responses out of four for the source and target models in mode m. The recovery cohort contains questions with $c_{\text{ON}}^S(x) \geq 3$ and $c_{\text{OFF}}^S(x) \leq 1$. Within this cohort, OFF recovery requires $c_{\text{OFF}}^T(x) \geq 2$, and ON retention requires $c_{\text{ON}}^T(x) \geq 2$. Joint recovery requires both conditions; its rate uses the cohort size as the denominator. Across all 577 questions, dual-mode success instead requires at least three correct responses in each mode, counted separately for source and target models.

Confidence intervals use 20,000 question-paired bootstrap resamples, with the full four-response result for each selected question retained. For each resample, we average the target-minus-source

Table 11: **Differences between pre-RL and post-RL reported models (pp).** General scores refer to the MedXpert branch. BioASQ compares separately selected pre-RL and post-RL models.

Initialization	Mode	ΔMedXpert	ΔBioASQ	ΔMedCase	ΔDomain	ΔGeneral
Base	OFF	+1.56	+1.34	+1.37	+1.42	-0.75
SFT expert	OFF	+0.78	+1.57	+1.20	+1.18	+0.21
OPD student	OFF	+2.77	+1.77	+1.09	+1.88	+0.14
Base	ON	-0.09	+1.59	+1.04	+0.85	+1.07
SFT expert	ON	-0.48	+2.75	-0.20	+0.69	+0.20
OPD student	ON	+2.25	+2.22	+1.75	+2.07	+0.91

Table 12: **ON retention, joint recovery, and dual-mode success.** Retention and joint recovery use a target threshold of two correct responses out of four. Dual-mode success requires three in each mode over all 577 questions.

Source / target	Cohort	ON retained	Joint	Joint (%)	Dual-mode
SFT / OPD student	41	30	18	43.90	$154 \rightarrow 177$
SFT+GRPO / BWB-KD	48	40	26	54.17	$160 \rightarrow 196$
Base / OPD student	28	25	18	64.29	$170 \rightarrow 177$
Base+GRPO / BWB-KD	21	18	14	66.67	$183 \rightarrow 196$

correctness difference and use the 2.5th and 97.5th percentiles. These intervals quantify question-sampling uncertainty for the evaluated checkpoints. The main MedXpert results and this analysis share the same underlying response batches. Table 12 complements the separate recovery and retention columns with their intersection.

E Code and Evaluation Scope

The supplementary materials include code for the main experiments; the public code release will include full configurations and data splits.

Domain averages summarize independently trained task branches; general benchmark and token analyses use the MedXpert branch.

F Generation and Evaluation Prompts

We provide templates for generating teacher demonstrations and verifying final answers, followed by the prompt sources for baseline evaluation. Braced fields are replaced with the corresponding task data.

F.1 TEACHER-DEMONSTRATION GENERATION

GPT-5.4 generates explanations and final answers for SFT using the following task-specific templates. Correctness checks select the retained demonstrations, as described in Appendix B.

MedXpert. The teacher receives a medical question and its options and returns an explanation and an option identifier. The {options} field lists one option per line as `- opt_A: ...`.

MedXpert Teacher-Demonstration Template

```
Below is a medical question and a set of options. Analyze the
question and select the best answer.

Question:
{question}

Options:
{options}

You should follow the output format below:
<reasoning>
Provide Your reasoning process of the Question here.
</reasoning>
<answer>
Provide the option identifier as your final answer, e.g., opt_A.
</answer>
```

MedCaseReasoning. The teacher receives the case presentation and returns a diagnostic explanation followed by the final diagnosis.

MedCaseReasoning Teacher-Demonstration Template

```
Read the following case presentation and give the most likely
diagnosis.
First, provide your diagnostic reasoning within the tags
<reasoning> ... </reasoning>.
Then, output the final diagnosis (just the name of the
disease/entity) within the tags <answer> ... </answer>.

----------------------------------------
CASE PRESENTATION
----------------------------------------
{case_presentation}

----------------------------------------
OUTPUT TEMPLATE
----------------------------------------
<reasoning>
...your diagnostic reasoning...
</reasoning>
<answer>
...the name of the disease/entity...
</answer>
```

BioASQ. The teacher receives the question and its type, without gold evidence. The {answer_contract} field is filled with the single instruction matching that type from the second box below. The teacher returns an explanation and a type-specific final answer.

BioASQ Teacher-Demonstration Template

```
Answer the biomedical question using reliable biomedical
knowledge. Reason explicitly and concisely, and do not invent
unsupported facts.
Return exactly two XML blocks in this order:
<reasoning>your step-by-step biomedical reasoning</reasoning>
<answer>your final answer</answer>
{answer_contract}

Question type: {question_type}
Question: {question}
```

BioASQ Answer Requirements by Question Type

```
YESNO:
In <answer>, begin with an explicit Yes or No, followed by a
concise biomedical justification.

FACTOID:
In <answer>, state the most specific requested entity or value
first, followed by a concise biomedical answer.

LIST:
In <answer>, list the distinct requested items, followed by a
concise biomedical explanation.

SUMMARY:
In <answer>, provide one concise, self-contained paragraph that
covers the decisive biomedical facts.
```

F.2 Answer Verification

MedCaseReasoning. GPT-5.4 receives the extracted prediction and reference diagnosis and returns a binary y/n judgment, using the diagnosis-correctness prompt of MedCaseReasoning (Wu et al., 2025).

MedCaseReasoning Answer-Verification Template

```
Is our predicted diagnosis correct (y/n)?
Predicted diagnosis: {predicted_diagnosis},
True diagnosis: {actual_diagnosis}
Answer [y/n].
```

BioASQ. The box below gives the GPT-5.4 verifier's system prompt. Each call also supplies the question, official exact-answer aliases, ideal answers, gold evidence snippets, and anonymized candidate final answers. The verifier returns a JSON object with a correctness judgment for each candidate. Two independent judgments must both accept an answer, as described in Appendix B.

BioASQ Answer-Verifier System Prompt

```
You are a strict biomedical answer verifier.

For each anonymized candidate, decide only whether its final
```

```
answer is fully correct for the BioASQ question. Return a binary
true/false decision. Do not score reasoning style, verbosity,
confidence, XML formatting, or hidden chain of thought.

Use the official exact-answer aliases, ideal answers, and gold
evidence as the reference. Apply these rules strictly:

- YESNO: the polarity must match exactly; hedging or
  contradiction is false.
- FACTOID: every requested entity, value, rank, direction,
  subtype, time relation, and material qualifier must match.
  Nearby values, a broader range, or a related entity are false.
- LIST: the required semantic entity set must be complete.
  Missing required members or adding unsupported alternative
  members is false.
- SUMMARY: all central biomedical claims requested by the
  question must be present and supported. Missing a central
  mechanism, relation, indication, limitation, or comparison
  is false. Fabricated or contradictory material is false.

Harmless spelling variants, exact aliases, unambiguous
abbreviation expansion, and exact unit conversions may be true.
When uncertain, return false. If the official reference is
insufficient or internally inconsistent, mark
reference_quality accordingly instead of guessing.

Return exactly one JSON object:
{
  "reference_quality": "valid|questionable|insufficient",
  "reference_quality_reason": "...",
  "judgments": [
    {
      "label": "C01",
      "is_correct": true,
      "reason": "one concise reference-grounded reason"
    }
  ]
}
```

F.3 Baseline Evaluation Prompts

ROPD uses its released rubricator and verifier templates (Fang et al., 2026); their instructions and structured output format are unchanged. ArenaHard uses the fixed pairwise evaluator template and reference answers described in Appendix B.