BOUNDARYEVO: DATA-MODEL CO-EVOLUTION AT THE CAPABILITY BOUNDARY FOR LONG-CHAIN REASONING DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Distilling long-chain reasoning trajectories from large teacher models provides an effective way to improve the reasoning ability of small language models. However, existing methods typically select a fixed training set before distillation, overlooking that the usefulness of a reasoning example changes as the student evolves. This mismatch becomes especially costly when only a small number of teacher trajectories can be used for training. Besides, examples already mastered provide little new supervision, while examples far beyond the capability of student may be difficult to learn from. Training data should therefore adapt as model capability evolves, yet reliable data selection is difficult when the student is initially weak. To address these challenges, our key idea is to first establish the basic reasoning capability and then adapt the training data to the evolving capability boundary of the student. We propose BoundaryEvo, a two-stage data–model co-evolution framework. In **Capability Priming**, we directly distill a small set of teacher-generated long reasoning trajectories without using a separate model to judge and filter them, allowing the student to acquire an initial reasoning capability before capability-aware data selection. In **Capability-Boundary Co-Evolution**, we repeatedly select questions that the current student has not mastered but the expert can reliably solve, and update the training data as the student evolves. With only 1,000 training examples, BoundaryEvo consistently outperforms static and dynamic data selection baselines across 11 reasoning benchmarks, raising the original Qwen3.5-4B accuracy from 62.9 to 75.3 on IMO-AnswerBench and from 26.8 to 46.0 on AMO-Bench. The code and data will be made publicly available.

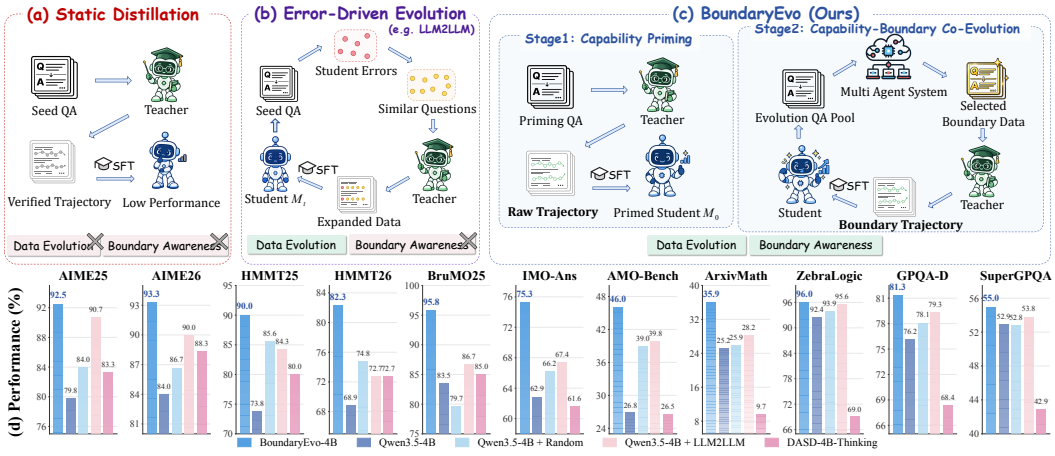


Figure 1: **BoundaryEvo evolves supervision along the capability boundary of the student model.** (a-c) Comparison with static distillation and error-driven acquisition, BoundaryEvo dynamically adapts supervision according to the evolving capability boundary of the student model. (d) BoundaryEvo-4B consistently outperforms strong baselines across various reasoning benchmarks.

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.

1 INTRODUCTION

Distilling long-chain reasoning trajectories from large teacher models is an effective way to transfer complex reasoning capabilities to small student models (DeepSeek-AI, 2025; Moshkov et al., 2025). Yet reasoning supervision is typically constructed *once* (Han et al., 2025; Moshkov et al., 2025) and then treated as static throughout training, even though the learner itself is continuously changing. As shown in Figure 1(a), existing distillation pipelines typically generate and filter a fixed collection of teacher trajectories before optimization and repeatedly train on the same data thereafter (Yan et al., 2026). Such a static mechanism overlooks a fundamental property of learning: *data utility is non-stationary with respect to the evolving student*. Examples that once exposed meaningful capability gaps may later become redundant, whereas trajectories that were initially beyond the student's reach may become precisely the supervision needed for its next stage of improvement. Thus, the supervision itself should continually adapt to the student's evolving capability boundary, enabling data and model to **co-evolve** throughout learning.

Recent efforts have begun to adapt training data to the evolving student, as illustrated in Figure 1(b), typically by prioritizing questions that the current model fails to solve (Lee et al., 2024). While this moves beyond static supervision, such adaptation remains largely reactive: it responds to the student's current errors without explicitly modeling how its capability changes over the course of learning. As a result, the training data may shift with the student, but it does not necessarily *co-evolve* with the student's emerging learning needs. For a weak student, failures may span questions that are only slightly beyond its current capability as well as those that remain far out of reach, while student difficulty alone cannot guarantee useful supervision either Tzannetos et al. (2023); Song et al. (2020); Chen et al. (2025). Thus, data-model co-evolution requires more than tracking what the student gets wrong. It requires continually localizing the *capability boundary* where unresolved student needs align with reliable and informative teacher supervision.

Based on this insight, we introduce **BoundaryEvo**, a two-stage data–model co-evolution framework for long-chain reasoning distillation under a limited supervision budget. As shown in Figure 1(c), BoundaryEvo organizes reasoning distillation as a two-stage co-evolution process: it first establishes a sufficiently capable student state and subsequently tracks the evolving capability boundary between the student and fixed expert to guide supervision. In *Stage 1, Capability Priming*, the student is initialized through direct distillation of teacher-generated reasoning trajectories. Rather than specializing supervision to a prematurely estimated learning boundary, this stage exposes the student to a broad distribution of teacher reasoning behaviors, providing the capability foundation required for subsequent boundary-aware adaptation. In *Stage 2: Capability-Boundary Co-Evolution*, we repeatedly identify questions that remain unresolved by the current student but can be reliably solved by the fixed expert, and then acquire teacher trajectories for the selected questions. Following each student update, the capability boundary is re-estimated and the supervision set is correspondingly refreshed, forming a closed-loop interaction between model learning and data acquisition. This iterative coupling progressively redirects supervision toward the student's current learning frontier with reduced supervision budget, allowing the training data and model capability to co-evolve throughout distillation. With a total budget of only 1,000 training examples, BoundaryEvo consistently outperforms strong static and dynamic data selection baselines on Qwen3.5-4B across 11 reasoning benchmarks (Figure 1(d)). Compared with the original Qwen3.5-4B model, BoundaryEvo improves accuracy from 62.9 to 75.3 on IMO-AnswerBench and from 26.8 to 46.0 on AMO-Bench. Experiments on Gemma-4-E4B-it further show that the improvements transfer across model backbones.

Our contributions are threefold:

- We identify a stage-dependent principle for long-chain reasoning distillation. Early-stage capability priming benefits from retaining diverse teacher reasoning behaviors, including exploration and unsuccessful attempts, rather than filtering trajectories solely by final correctness, while later-stage training benefits from capability-boundary-guided data selection.
- We propose BoundaryEvo, a two-stage data–model co-evolution framework for long-chain reasoning distillation, which first establishes student capability through capability priming and then adaptively evolves long-horizon supervision through student–expert capability-boundary tracking.
- We evaluate BoundaryEvo with only 1,000 training examples across 11 reasoning benchmarks and multiple model backbones, showing consistent gains over static and dynamic data-selection baselines and validating the roles of both capability priming and capability-boundary co-evolution.

2 Related Work

Reasoning distillation. Knowledge distillation transfers teacher predictions or generated sequences to student models (Hinton et al., 2015; Kim & Rush, 2016). Chain-of-thought and self-consistency demonstrate the value of intermediate reasoning trajectories (Wei et al., 2022; Wang et al., 2023a). Recent reasoning models, including DeepSeek-R1 (DeepSeek-AI, 2025), O1 replication (Huang et al., 2024), and DASD (Yan et al., 2026), further highlight the effectiveness of long-chain reasoning supervision. LIMO (Ye et al., 2025) and s1 (Muennighoff et al., 2025) achieve strong performance with small curated datasets, while OpenMathReasoning studies large-scale mathematical reasoning data construction (Moshkov et al., 2025). VibeThinker (Xu et al., 2026) explores multi-stage reasoning training beyond single-step distillation. However, they largely assume a static supervision distribution and overlook the need to adapt training data as model capabilities evolve.

Self-evolution and co-evolution. Self-improvement methods iteratively generate experience, obtain feedback, and update models. STaR (Zelikman et al., 2022) and ReST (Gülçehre et al., 2023) alternate reasoning generation and training with feedback signals. Self-rewarding models improve feedback generation through model-based rewards (Yuan et al., 2024). R-Zero (Huang et al., 2025) , Socratic-Zero (Wang et al., 2025), and SeRL (Fang et al., 2025) explore self-play and challenger–solver evolution strategies. SyncLoop (Chen et al., 2026) studies multimodal data evolution coupled with model improvement. Synthetic data approaches, including Self-Instruct (Wang et al., 2023b), WizardLM (Xu et al., 2024), WizardMath (Luo et al., 2025), and MetaMath (Yu et al., 2024), construct additional instruction or reasoning data. AutoData (Kulikov et al., 2026) introduces an agentic pipeline for task generation and refinement, while LLM2LLM (Lee et al., 2024) selects teacher-generated data based on student errors. Different from these methods, BoundaryEvo performs capability-boundary-driven data–model co-evolution.

Data selection. Training data selection has been explored through curriculum learning, active learning, and quality-based filtering. Curriculum learning and active learning connect sample selection with learner states (Bengio et al., 2009; Ren et al., 2022; Yang et al., 2026). LIMA (Zhou et al., 2023), DEITA (Liu et al., 2024), QuRating (Wettig et al., 2024), and educational-data studies (Gunasekar et al., 2023) investigate data quality, diversity, and selection strategies. D4 (Tirumala et al., 2023) studies deduplication, diversity, and repeated data utilization. Experience replay provides a general mechanism for retaining previous knowledge during iterative learning (Mnih et al., 2015; Schaul et al., 2016). Outcome and process verification introduce additional signals for reasoning data assessment (Cobbe et al., 2021; Lightman et al., 2024; Uesato et al., 2022; Wang et al., 2024). Although model-based judging is useful, it can still be imperfect (Zheng et al., 2023). BoundaryEvo selects data based on the capability boundary between student and expert, rather than static quality or error signals.

3 The Proposed BoundaryEvo

Overview. We propose BoundaryEvo, a two-stage data–model co-evolution framework that dynamically adapts training data to the evolving capabilities of compact reasoning models. As illustrated in Figure 2, Stage 1 performs direct distillation from a small set of teacher-generated reasoning trajectories without additional model-based judging, providing the student with foundational reasoning capabilities. Stage 2 then iteratively identifies capability-boundary examples: at each evolution turn t, the current student and a fixed expert model E independently solve the same candidate questions and generate reasoning trajectories, respectively. An analyzer evaluates student performance, expert solutions, and data quality to select questions that the expert can reliably solve but the student has not yet mastered. The selected questions are combined with revalidated historical data, and verified teacher trajectories are distilled through supervised fine-tuning to update the student model. As the student improves, the capability boundary shifts and guides subsequent data selection, enabling continuous data–model co-evolution under a fixed data budget.

3.1 Stage 1: Judge-Free Direct Distillation for Capability Priming

Given seed questions $\{x_i\}_{i=1}^{K_0}$, the teacher model generates reasoning trajectories and final answers:

$$y_i^{\mathrm{T}} \sim T(\cdot \mid x_i), \qquad \mathcal{D}_0 = \{(x_i, y_i^{\mathrm{T}})\}_{i=1}^{K_0}. \tag{1}$$

Stage 1 performs direct reasoning distillation, where teacher-generated trajectories are used as supervision after basic response-validity checks. Unlike the subsequent evolution stage, Stage 1 does

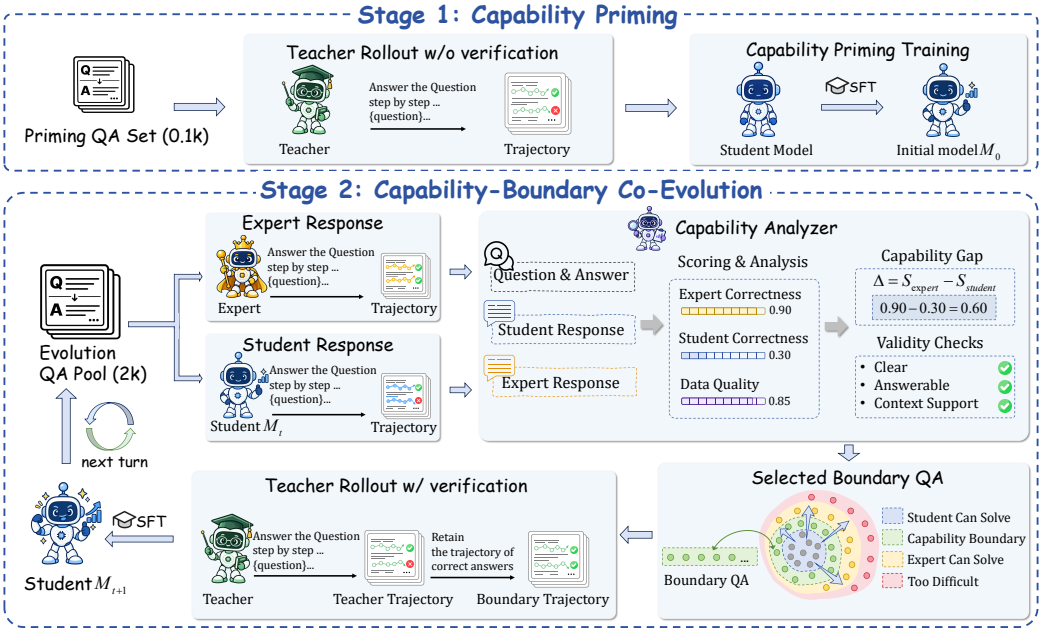


Figure 2: **Overview of BoundaryEvo.** Stage 1 performs judge-free teacher distillation to obtain the capability-primed checkpoint. In Stage 2, responses from the current student and a fixed expert are used to judge the candidate QAs, from which replay and novel records are selected for multiple teacher rollouts. The resulting trajectories are then independently verified against reference answers, and only those that pass verification supervise the next student.

not apply additional model-based judging or capability-gap-based filtering. Instead of searching for difficult examples at this stage, it aims to efficiently transfer teacher reasoning behaviors and establish initial student reasoning capabilities from a small amount of high-quality supervision.

This design avoids premature filtering that may discard useful teacher experiences due to imperfect automated assessment or diverse valid reasoning patterns. Training on $\mathcal{D}_0$ produces the capability-primed checkpoint M_0, which provides a stronger initialization for Stage 2. With this initial capability, BoundaryEvo can more effectively identify capability-boundary questions and perform subsequent verified teacher trajectory distillation for iterative data–model co-evolution.

3.2 Stage 2: Dynamic Capability-Boundary Co-Evolution

Student–Expert–Analyzer rollouts and judging. At evolution round t, the **student** model M_t and a fixed **expert** independently generate reasoning trajectories for the same candidate question x_i:

$$\hat{y}^{\mathrm{s}}_{i,t} \sim M_t(\cdot \mid x_i, h_i), \qquad \hat{y}^{\mathrm{e}}_{i} \sim E(\cdot \mid x_i, h_i). \tag{2}$$

Here, h_i denotes optional context or auxiliary information. The **analyzer** evaluates the question, reference information, and reasoning outputs from both models. The resulting assessment captures three aspects: *capability-boundary matching*, *data-quality control*, and *question-validity checks*:

$$\mathbf{a}_i^{(t)} = \Big(\underbrace{s_i^{\mathrm{s},(t)}, s_i^{\mathrm{e}}, g_i^{(t)}}_{\text{Boundary matching}}, \underbrace{q_i}_{\text{Quality}}, \underbrace{b_i^{\mathrm{clear}}, b_i^{\mathrm{answerable}}, c_i}_{\text{Validity}} \Big). \tag{3}$$

The capability gap is computed from the student and expert scores:

$$g_i^{(t)} = s_i^{\mathrm{e}} - s_i^{\mathrm{s},(t)}. \tag{4}$$

A large positive gap indicates that the expert can solve the problem beyond the current student capability. However, capability difference alone is insufficient, as the expert response must be reliable and the example must provide high-quality and valid supervision. The quality score q_i measures

overall training suitability, while validity indicators b_i^{clear}, $b_i^{\text{answerable}}$, and c_i prevent ambiguous questions or unsupported contexts from being mistaken as student limitations.

Hard capability-boundary filtering. The three levels define the acceptance criteria for capability-boundary examples. A candidate is selected into $\mathcal{A}_t$ only if it satisfies all three conditions:

$$\begin{aligned} \text{Boundary matching:} \quad & s_i^{\text{e}} \geq \tau_{\text{e}}, \quad s_i^{\text{s},(t)} < \tau_{\text{s}}, \quad g_i^{(t)} \geq \tau_{\text{g}}, \\ \text{Quality control:} \quad & q_i \geq \tau_{\text{q}}, \\ \text{Question validity:} \quad & b_i^{\text{clear}} = b_i^{\text{answerable}} = 1, \quad c_i \geq \tau_{\text{c}}. \end{aligned} \tag{5}$$

The first condition identifies examples where the expert provides reliable solutions while the student can improve. The remaining conditions ensure that selected samples provide high-quality and valid supervision rather than reflecting ambiguity or data defects. Together, these criteria identify training examples near the current capability boundary. BoundaryEvo therefore selects not only examples that the student fails but also those that can provide reliable learning signals for evolution.

Replay and novel acquisition. To balance retaining unresolved examples and exploring new boundary samples, we maintain a fixed training budget of $K = 300$ examples per turn. At each round, the updated student rescans the evolution pool $\mathcal{P}$ of 2K candidates, producing the accepted set $\mathcal{A}_t$. We divide it into replay examples previously used in $\mathcal{D}_{t-1}$ and novel examples:

$$\mathcal{R}_t = \{z \in \mathcal{A}_t : x(z) \in \text{Questions}(\mathcal{D}_{t-1})\}, \quad \mathcal{N}_t = \{z \in \mathcal{A}_t : x(z) \notin \text{Questions}(\mathcal{D}_{t-1})\}. \tag{6}$$

Our replay-first strategy keeps eligible replay examples up to the replay quota and fills the remaining budget with quality-ranked novel examples:

$$\mathcal{D}_t = \mathcal{R}'_t \cup \text{Select}_{K-|\mathcal{R}'_t|}(\mathcal{N}_t), \qquad |\mathcal{D}_t| = K = 300. \tag{7}$$

Novel examples are ranked by student response completeness, analyzer recommendation, and quality score $r_i^{(t)}$, with secondary criteria used for ties. This design preserves unresolved learning signals while continually introducing new capability-boundary examples.

Post-selection teacher rollouts and verified distillation. After QA selection, we construct supervision through fresh teacher rollouts. This separates *which questions to learn* from *which solutions to imitate*: the expert rollout identifies the capability boundary, while the teacher rollouts provide high-quality reasoning trajectories for student training. For each selected QA record $z_i \in \mathcal{D}_t$, the teacher independently samples $L \geq 2$ reasoning trajectories:

$$y_{i,t}^{\text{T},(k)} \sim T(\cdot \mid x_i, h_i), \qquad k = 1, \ldots, L. \tag{8}$$

We retain trajectories whose final answers pass reference verification:

$$V(\text{Ans}(y_{i,t}^{\text{T},(k)}), y_i^{\text{ref}}) = 1. \tag{9}$$

A verified trajectory is selected for each question to form $\mathcal{D}_t^{\text{distill}}$, and the student is fine-tuned on these full reasoning trajectories. Unlike Stage 1 judge-free priming, Stage 2 requires evaluation before distillation because data selection now depends on identifying capability-boundary examples rather than directly transferring teacher behaviors. QA filtering identifies learning opportunities, while verification ensures reliable training targets for the next evolution round.

3.3 Training Objective

Both stages optimize the student with standard supervised fine-tuning (SFT) on teacher reasoning trajectories. Let $\mathcal{D}$ denote $\mathcal{D}_0$ in Stage 1 or $\mathcal{D}_t^{\text{distill}}$ in Stage 2, where responses include full reasoning and final answers. The objective is the token-normalized cross-entropy loss:

$$\mathcal{L}_{\text{SFT}}(\theta; \mathcal{D}) = -\frac{\sum_{(x_i,y_i)\in\mathcal{D}} \sum_{j=1}^{|y_i|} \log p_\theta(y_{ij} \mid x_i, h_i, y_{i,<j})}{\sum_{(x_i,y_i)\in\mathcal{D}} |y_i|}. \tag{10}$$

Only response tokens contribute to the loss. Stage 1 obtains M_0 through direct distillation, while each Stage 2 turn updates M_t with verified capability-boundary trajectories to obtain M_{t+1}.

Table 1: Comparison with Reasoning Models under Different Scales and Training Budgets. Scores are percentages. Data denotes the number of publicly disclosed reasoning post-training examples and – indicates unavailable information.

Model	Params	Data	Mathematical Reasoning								General Reasoning		
			AIME 25	AIME 26	HMMT 25	HMMT 26	BruMO 25	IMO -Ans	AMO -Bench	Arxiv Math	Zebra Logic	GPQA -D	Super GPQA
Small reasoning models													
SmolLM3-3B	3B	–	36.7	41.0	26.0	36.4	49.2	28.7	5.3	7.8	26.0	41.7	29.2
VibeThinker-3B	3B	–	91.4	94.3	89.3	80.3	93.8	76.4	41.0	27.2	74.7	70.2	34.8
DASD-4B-Thinking	4B	448K	83.3	88.3	80.0	72.7	85.0	61.6	26.5	9.7	69.0	68.4	42.9
Hunyuan-4B-Instruct	4B	–	66.5	57.7	35.2	45.5	62.7	39.6	4.7	5.8	83.5	61.1	38.3
Qwen3-4B-Thinking-2507	4B	–	81.3	79.0	55.5	57.6	77.7	51.6	18.7	18.4	91.8	65.8	47.8
Qwen3.5-4B	4B	–	79.8	84.0	73.8	68.9	83.5	62.9	26.8	25.2	92.4	76.2	52.9
NVIDIA-Nemotron-3-Nano	4B	–	78.5	–	–	–	–	–	–	–	–	53.2	–
Spark-X2.5-4B	4B	–	90.0	90.7	–	81.2	–	74.2	–	–	60.5	67.4	–
Olmo-3-7B-Think	7B	>102K	67.9	69.1	43.8	39.4	69.0	49.4	14.0	14.6	65.0	46.2	–
MiMo-7B-RL-0530	7B	6M	70.2	76.0	48.5	54.6	79.8	53.9	14.0	11.7	70.9	60.6	–
OpenReasoning-Nemotron	7B	≥5M	78.2	80.2	63.5	51.5	78.8	60.6	26.0	15.5	19.8	61.1	–
Gemma-4-12B-it	12B	–	72.9	77.5	63.3	72.7	80.4	54.9	17.3	33.0	92.1	78.8	–
Phi-4-reasoning-plus	14B	–	68.4	73.6	50.3	51.5	66.5	46.2	4.7	15.5	–	81.9	–
Ministral-3-Reasoning-2512	14B	–	82.9	85.0	67.1	42.4	86.7	63.4	20.7	19.4	85.1	71.2	–
Large reasoning models													
gpt-oss-20b (high)	20B	–	91.7	90.2	76.7	–	86.7	61.9	–	–	–	71.5	44.9
NVIDIA-Nemotron-3-Nano	30B	–	89.1	89.9	84.6	–	–	70.4	–	–	–	74.1	–
Nemotron-3.5-Lightning	30B	–	–	–	–	–	–	–	–	–	–	75.4	–
GLM-4.7-Flash	30B	–	91.6	88.3	–	–	–	73.8	53.8	–	–	75.2	–
GLM-4.5-Air	106B	–	83.3	–	69.2	–	90.0	–	–	–	–	75.0	–
Qwen3-235B-A22B-Thinking	235B	–	92.3	–	83.9	–	–	70.5	47.8	–	97.5	81.1	60.7
LongCat-Flash-Thinking	560B	–	90.6	–	83.7	–	–	–	43.6	–	95.5	81.5	–
GPT-5 nano (high)	N/A	–	85.2	76.7	75.6	69.7	80.8	–	–	19.4	91.2	71.2	48.9
Our method													
BoundaryEvo-4B	**4B**	**1K**	**92.5**	**93.3**	**90.0**	**82.3**	**95.8**	**75.3**	**46.0**	**35.9**	**96.0**	**81.3**	**55.0**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Models and data. We evaluate BoundaryEvo primarily on Qwen3.5-4B (Qwen Team, 2026), with additional experiments on Qwen3.5-2B (Qwen Team, 2026) and Gemma-4-E4B-it (Team et al., 2026). All QA pairs are sampled from a cleaned subset (math_notool.jsonl) of Nemotron-Cascade-2-SFT-Data (NVIDIA, 2026a), with reasoning trajectories re-distilled from DeepSeek-V3.2-Speciale (DeepSeek-AI et al., 2025). Stage 1 uses 100 teacher trajectories, while Stage 2 selects from 1,999 candidate questions. GPT-5.1 (OpenAI, 2025) provides expert rollouts for capability-boundary estimation, and Gemini 3.5 Flash (Google DeepMind, 2026) serves as the analyzer. Across three evolution turns, BoundaryEvo trains on 1,000 examples in total ($100 + 3\times300$).

Training and Evaluation. We train models with LoRA-based SFT (Hu et al., 2022). Each evolution turn continues from the previous checkpoint using the selected training trajectories. Training details, including optimization settings and implementation configurations, are provided in Appendix B. We evaluate reasoning performance on eight mathematical benchmarks, including AIME25, AIME26 (Mathematical Association of America, 2026), HMMT25 (HMMT, 2025), HMMT26 (HMMT, 2026), BruMO25 (BRUMO, 2025), IMO-AnswerBench (IMO-Ans) (Luong et al., 2025), AMO-Bench (An et al., 2025), and ArxivMath (Dekoninck et al., 2026). We further measure general reasoning transfer on ZebraLogicBench (ZebraLogic) (Lin et al., 2025), GPQA-Diamond (GPQA-D) (Rein et al., 2023), and SuperGPQA (Team et al., 2025c). All scores are reported as percentages, with Math and General denoting unweighted averages over the corresponding benchmark groups.

Baselines. We compare zero-shot performance, Stage 1 priming, random selection, and LLM2LLM baseline (Lee et al., 2024). All evolution methods share the same Stage 1 checkpoint and training budget to isolate the effect of Stage 2 data acquisition. The LLM2LLM baseline selects new training examples based on student errors under the same training protocol. We additionally include a static

Table 2: Qwen3.5-4B results across three evolution turns. All Stage 2 strategies start from the same Stage 1 checkpoint. Scores are percentages, with Math and General averaged over eight and three benchmarks. Superscripts indicate absolute changes from zero-shot. Bold and underlining mark the highest and second-highest values in each column.

Method	Mathematical Reasoning								General Reasoning			Avg.	
	AIME 25	AIME 26	HMMT 25	HMMT 26	BruMO 25	IMO -Ans	AMO -Bench	Arxiv Math	Zebra Logic	GPQA -D	Super GPQA	Math	General
Qwen3.5-4B													
+ Zero-shot	79.8	84.0	73.8	68.9	83.5	62.9	26.8	25.2	92.4	76.2	52.9	63.1	73.8
+ Stage 1	84.2$^{+4.4}$	88.3$^{+4.3}$	81.7$^{+7.9}$	69.7$^{+0.8}$	85.0$^{+1.5}$	66.3$^{+3.4}$	38.5$^{+11.7}$	28.2$^{+3.0}$	92.5$^{+0.1}$	79.8$^{+3.6}$	53.4$^{+0.5}$	67.7$^{+4.6}$	75.2$^{+1.4}$
Stage 2 with Random Selection													
+ Turn 1	85.0$^{+5.2}$	88.3$^{+4.3}$	82.4$^{+8.6}$	72.4$^{+3.5}$	80.8$^{-2.7}$	66.5$^{+3.6}$	38.5$^{+11.7}$	27.2$^{+2.0}$	92.7$^{+0.3}$	79.0$^{+2.8}$	52.8$^{-0.1}$	67.6$^{+4.5}$	74.8$^{+1.0}$
+ Turn 2	85.3$^{+5.5}$	90.6$^{+6.6}$	82.9$^{+9.1}$	70.0$^{+1.1}$	86.7$^{+3.2}$	66.8$^{+3.9}$	38.7$^{+11.9}$	28.3$^{+3.1}$	93.0$^{+0.6}$	80.2$^{+4.0}$	52.9$^{+0.0}$	68.7$^{+5.6}$	75.4$^{+1.6}$
+ Turn 3	84.0$^{+4.2}$	86.7$^{+2.7}$	85.6$^{+11.8}$	74.8$^{+5.9}$	79.7$^{-3.8}$	66.2$^{+3.3}$	39.0$^{+12.2}$	25.9$^{+0.7}$	93.9$^{+1.5}$	78.1$^{+1.9}$	52.8$^{-0.1}$	67.7$^{+4.6}$	74.9$^{+1.1}$
Stage 2 with LLM2LLM													
+ Turn 1	86.0$^{+6.2}$	89.2$^{+5.2}$	82.1$^{+8.3}$	70.8$^{+1.9}$	84.8$^{+1.3}$	66.6$^{+3.7}$	39.1$^{+12.3}$	28.4$^{+3.2}$	93.3$^{+0.9}$	77.8$^{+1.6}$	53.1$^{+0.2}$	68.4$^{+5.3}$	74.7$^{+0.9}$
+ Turn 2	86.7$^{+6.9}$	90.7$^{+6.7}$	83.3$^{+9.5}$	71.0$^{+2.1}$	84.2$^{+0.7}$	67.0$^{+4.1}$	38.9$^{+12.1}$	28.4$^{+3.2}$	94.0$^{+1.6}$	78.8$^{+2.6}$	53.2$^{+0.3}$	68.8$^{+5.7}$	75.3$^{+1.5}$
+ Turn 3	<u>90.7</u>$^{+10.9}$	90.0$^{+6.0}$	84.3$^{+10.5}$	72.7$^{+3.8}$	86.7$^{+3.2}$	67.4$^{+4.5}$	39.8$^{+13.0}$	28.2$^{+3.0}$	95.6$^{+3.2}$	79.3$^{+3.1}$	53.8$^{+0.9}$	70.0$^{+6.9}$	76.2$^{+2.4}$
Stage 2 with BoundaryEvo (Ours)													
+ Turn 1	89.3$^{+9.5}$	91.7$^{+7.7}$	86.0$^{+12.2}$	75.8$^{+6.9}$	85.0$^{+1.5}$	69.0$^{+6.1}$	41.5$^{+14.7}$	<u>31.1</u>$^{+5.9}$	92.8$^{+0.4}$	79.3$^{+3.1}$	<u>54.3</u>$^{+1.4}$	71.2$^{+8.1}$	75.5$^{+1.7}$
+ Turn 2	90.0$^{+10.2}$	<u>92.5</u>$^{+8.5}$	<u>86.7</u>$^{+12.9}$	<u>81.8</u>$^{+12.9}$	<u>90.0</u>$^{+6.5}$	<u>73.3</u>$^{+10.4}$	<u>44.0</u>$^{+17.2}$	28.2$^{+3.0}$	<u>95.9</u>$^{+3.5}$	<u>80.8</u>$^{+4.6}$	54.1$^{+1.2}$	<u>73.3</u>$^{+10.2}$	<u>76.9</u>$^{+3.1}$
+ Turn 3	**92.5**$^{+12.7}$	**93.3**$^{+9.3}$	**90.0**$^{+16.2}$	**82.3**$^{+13.4}$	**95.8**$^{+12.3}$	**75.3**$^{+12.4}$	**46.0**$^{+19.2}$	**35.9**$^{+10.7}$	**96.0**$^{+3.6}$	**81.3**$^{+5.1}$	**55.0**$^{+2.1}$	**76.4**$^{+13.3}$	**77.4**$^{+3.6}$

full-data SFT baseline trained on 1,000 examples for comparison with fixed data training. Besides, Table 1 further compares our final model with representative reasoning models across different parameter scales. The small- and medium-scale group includes SmolLM3-3B (Bakouch et al., 2025), VibeThinker-3B (Xu et al., 2026), DASD-4B-Thinking (Yan et al., 2026), Hunyuan-4B-Instruct (Tencent Hunyuan Team, 2025), Qwen3-4B-Thinking-2507 (Qwen Team, 2025), Qwen3.5-4B (Qwen Team, 2026), NVIDIA-Nemotron-3-Nano-4B-BF16 (NVIDIA, 2026c), Spark-X2.5-4B (Team, 2026), Olmo-3-7B-Think (Olmo et al., 2026), MiMo-7B-RL-0530 (Xiaomi et al., 2025), OpenReasoning-Nemotron-7B (NVIDIA, 2025), Gemma-4-12B-it (Team et al., 2026), Phi-4-reasoning-plus (Abdin et al., 2025), and Ministral-3-14B-Reasoning-2512 (Liu et al., 2026). The large-scale group includes gpt-oss-20b (OpenAI et al., 2025), NVIDIA-Nemotron-3-Nano-30B-A3B-BF-16 (NVIDIA et al., 2025), NVIDIA-Nemotron-3.5-Lightning-30B-A3B-BF16 (NVIDIA, 2026b), GLM-4.7-Flash (Team et al., 2025a), GLM-4.5-Air (Team et al., 2025a), Qwen3-235B-A22B-Thinking (Yang et al., 2025), LongCat-Flash-Thinking (Team et al., 2025b), and GPT-5 nano (Singh et al., 2026).

4.2 Downstream Effectiveness

BoundaryEvo-4B achieves strong reasoning performance with only 1K training examples across diverse mathematical benchmarks. As summarized in Table 1, it obtains leading results on AIME25, HMMT25, HMMT26, BruMO25, and ArxivMath, and consistently outperforms other 4B-scale reasoning models. Compared with DASD-4B-Thinking, BoundaryEvo improves all shared benchmarks, including a 19.5-point gain on AMO-Bench, while using only a small fraction of the training data.

Beyond same-scale comparisons, BoundaryEvo remains competitive with much larger reasoning models, achieving 75.3 on IMO-AnswerBench and outperforming Qwen3-235B-A22B-Thinking. It also generalizes to broader reasoning tasks, reaching 96.0 on ZebraLogicBench, 81.3 on GPQA-Diamond, and 55.0 on SuperGPQA. These results demonstrate that capability-boundary-guided data–model co-evolution enables small reasoning models to effectively extract high-value long-chain reasoning supervision under limited data budgets.

4.3 Main Results

As shown in table 2, BoundaryEvo achieves consistent gains through capability priming and subsequent evolution. Stage 1 improves the mathematical average from 63.1 to 67.7, and three evolution turns further increase it to 71.2, 73.3, and 76.4, yielding an 8.7-point gain over priming. The general-reasoning average also improves from 75.2 to 77.4 after evolution, despite training only

Table 3: Transfer across model size and architecture. Each backbone is primed before three sequential Stage 2 turns. Scores are percentages. Superscripts indicate absolute changes from zero-shot. Bold/underline denote the top two values, and blue shading marks the best checkpoint.

Method	Mathematical Reasoning						General Reasoning		Avg.	
	AIME 25	AIME 26	HMMT 25	HMMT 26	BruMO 25	IMO -Ans	Zebra Logic	GPQA -D	Math	General
Qwen3.5-2B										
+ Zero-shot	34.2	38.8	22.9	22.7	35.0	22.0	44.9	50.0	29.3	47.5
+ Stage 1	38.3$^{+4.1}$	50.0$^{+11.2}$	26.7$^{+3.8}$	24.5$^{+1.8}$	43.3$^{+8.3}$	35.2$^{+13.2}$	50.5$^{+5.6}$	50.5$^{+0.5}$	36.3$^{+7.0}$	50.5$^{+3.0}$
+ Turn 1	40.0$^{+5.8}$	46.7$^{+7.9}$	30.0$^{+7.1}$	**33.3**$^{+10.6}$	46.7$^{+11.7}$	35.7$^{+13.7}$	63.7$^{+18.8}$	52.5$^{+2.5}$	38.7$^{+9.4}$	58.1$^{+10.6}$
+ Turn 2	**46.7**$^{+12.5}$	**56.7**$^{+17.9}$	**33.3**$^{+10.4}$	27.3$^{+4.6}$	51.7$^{+16.7}$	**35.9**$^{+13.9}$	**68.7**$^{+23.8}$	**53.5**$^{+3.5}$	**41.9**$^{+12.6}$	**61.1**$^{+13.6}$
+ Turn 3	43.3$^{+9.1}$	46.7$^{+7.9}$	23.3$^{+0.4}$	30.0$^{+7.3}$	**54.2**$^{+19.2}$	35.2$^{+13.2}$	63.8$^{+18.9}$	51.5$^{+1.5}$	38.8$^{+9.5}$	57.7$^{+10.2}$
Gemma-4-E4B-it										
+ Zero-shot	36.7	53.3	26.7	39.4	50.8	31.6	69.4	58.6	39.8	64.0
+ Stage 1	70.0$^{+33.3}$	76.7$^{+23.4}$	50.0$^{+23.3}$	42.4$^{+3.0}$	55.4$^{+4.6}$	44.9$^{+13.3}$	90.1$^{+20.7}$	60.1$^{+1.5}$	56.6$^{+16.8}$	75.1$^{+11.1}$
+ Turn 1	76.7$^{+40.0}$	83.3$^{+30.0}$	**80.0**$^{+53.3}$	33.3$^{-6.1}$	55.7$^{+4.9}$	49.5$^{+17.9}$	92.8$^{+23.4}$	60.6$^{+2.0}$	63.1$^{+23.3}$	76.7$^{+12.7}$
+ Turn 2	76.7$^{+40.0}$	**86.7**$^{+33.4}$	66.7$^{+40.0}$	60.6$^{+21.2}$	60.8$^{+10.0}$	50.3$^{+18.7}$	94.1$^{+24.7}$	61.1$^{+2.5}$	67.0$^{+27.2}$	77.6$^{+13.6}$
+ Turn 3	**80.0**$^{+43.3}$	**86.7**$^{+33.4}$	70.0$^{+43.3}$	**66.7**$^{+27.3}$	**67.5**$^{+16.7}$	**51.3**$^{+19.7}$	**94.7**$^{+25.3}$	**66.2**$^{+7.6}$	**70.4**$^{+30.6}$	**80.5**$^{+16.5}$

on mathematical data. BoundaryEvo outperforms random selection and LLM2LLM on all eleven benchmarks, exceeding them by 8.7 and 6.4 points on mathematical averages and by 2.5 and 1.2 points on general reasoning. The largest gains over LLM2LLM appear on HMMT26, BruMO25, IMO-AnswerBench, and ArxivMath, highlighting the advantage of capability-boundary-driven acquisition over error-based selection. Across evolution turns, random selection quickly saturates, while LLM2LLM improves more slowly; in contrast, BoundaryEvo consistently improves across both benchmark groups. These results show that capability-boundary-driven selection enables progressive improvement through iterative QA selection and verified trajectory distillation.

4.4 Generalization Across Model Backbones

Table 3 evaluates the transferability of BoundaryEvo across model scales and architectures. For Qwen3.5-2B, Stage 1 improves the mathematical and general-reasoning averages from 29.3 to 36.3 and from 47.5 to 50.5, respectively. Subsequent boundary-guided evolution improves performance, with Turn 2 achieving the strongest checkpoint (e.g., AIME25: 46.7, AIME26: 56.7, and ZebraLogicBench: 68.7). However, the smaller model shows earlier saturation and less stable gains across turns, suggesting that limited model capacity can constrain long-term evolution. For Gemma-4-E4B-it, BoundaryEvo achieves more consistent improvements: Stage 1 raises AIME-25 from 36.7 to 70.0 and ZebraLogicBench from 69.4 to 90.1, while Stage 2 further improves the mathematical and general-reasoning averages to 70.4 and 80.5 at Turn 3. These results demonstrate that BoundaryEvo transfers across different backbones and adapts its evolution process to varying student capabilities.

4.5 Training Dynamics

The training dynamics reveal the progressive adaptation enabled by BoundaryEvo. As illustrated in Figure 3(a), fitting the selected teacher trajectories across three evolution turns consistently reduces the mean loss from 0.4362 to 0.3936 and 0.3388, indicating stable optimization on newly discovered boundary examples with replay-based training. As shown in Figure 3(b), the selected curriculum gradually shifts as the student improves. The mean student score increases from 0.233 to 0.265 and 0.271 while expert scores remain high at 0.846, 0.879, and 0.884, preserving a meaningful student–expert gap for continued selection of challenging samples. Together, these trends demonstrate capability-boundary-guided co-evolution, where the curriculum evolves with student capability while maintaining reliable supervision and continuously expanding the training frontier.

4.6 Discussion

Does continued evolution improve the training frontier? As shown in Figure 4, BoundaryEvo progressively improves IMO-AnswerBench from 66.3 after Stage 1 to 69.0, 73.3, and 75.3 across three evolution turns. The final checkpoint surpasses Stage 1 and static full-data SFT by 9.0 and 5.8 points, respectively. These gains show that continuously updating the training distribution according to the student capability boundary is more effective than using a fixed dataset.

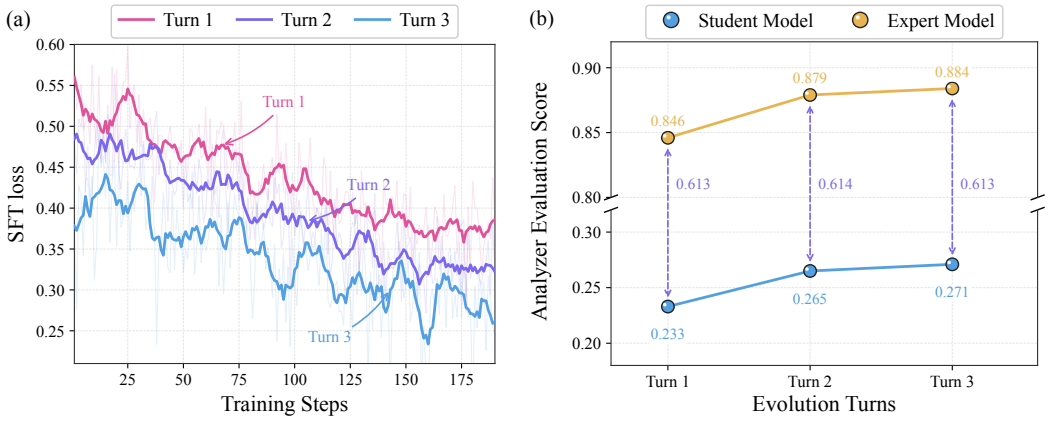


Figure 3: Training and selection dynamics across the Stage 2. (a) SFT loss for representative runs. (b) Mean student and expert scores on each turn. Annotations show the score gap.

Does capability priming benefit from judge-free distillation? Table 4 shows that removing Stage 1 decreases the final mathematical and general reasoning averages from 76.4/77.4 to 73.5/74.0, demonstrating the importance of capability priming before evolution. Moreover, verified Stage 1 further reduces performance to 73.1/73.9. As illustrated in Figure 4, direct distillation starts higher and maintains a larger advantage throughout evolution. These results suggest that Stage 1 benefits from preserving diverse teacher reasoning behaviors, while verification is better suited for Stage 2 boundary refinement.

Does replay stabilize co-evolution? Removing replay reduces IMO-AnswerBench from 75.3 to 71.1 at the final turn. Although newly selected boundary examples improve the student, the performance drop indicates that learning only from the latest curriculum can gradually weaken previously acquired capabilities. Replay therefore helps retain historical knowledge while adapting to newly discovered boundary regions.

Overall, the ablations confirm the complementary roles of BoundaryEvo components: Stage 1 establishes the initial capability foundation, boundary-guided evolution expands the training frontier, and replay stabilizes continual improvement.

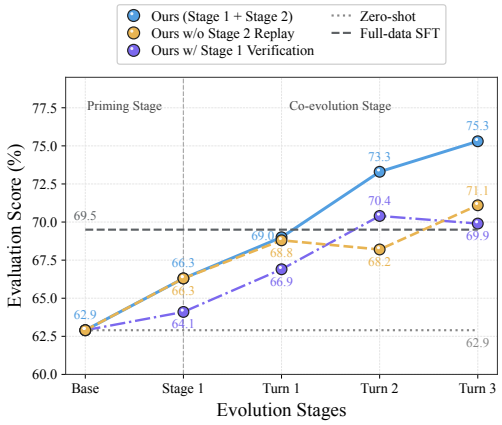


Figure 4: Ablation study on IMO-AnswerBench performance. The curves compare BoundaryEvo with verified Stage 1 and no-replay variants, while reference lines show zero-shot and static SFT on 1K examples.

Table 4: Ablation study on effect of Stage 1 on final Qwen3.5-4B performance. Math and General average eight and three benchmarks.

Method	Math AVG	General AVG
BoundaryEvo	76.4	77.4
⊢ Stage 1 (w/ verification)	73.1	73.9
⊢ w/o Stage 1	73.5	74.0

5 Conclusion

We introduce BoundaryEvo, a dynamic capability-boundary co-evolution framework for long-chain reasoning distillation. Unlike conventional methods that treat reasoning data as a static resource, BoundaryEvo continuously adapts training supervision according to the evolving capabilities of the student model. It first performs judge-free capability priming to acquire diverse teacher reasoning behaviors, followed by capability-boundary-guided data selection and verified teacher trajectory distillation to discover progressively valuable reasoning supervision. Experiments across model backbones demonstrate that BoundaryEvo achieves strong reasoning performance with only 1K training examples, while ablations validate the importance of capability priming and capability-boundary-guided evolution. These results show that effective reasoning supervision is not fixed, but can be continuously discovered through data–model co-evolution.

AI USE STATEMENT

We used generative AI tools for limited assistance in language refinement and writing improvement. We did not use generative AI tools for generating research ideas, designing experiments, conducting evaluations, or producing scientific claims. All AI-assisted content was reviewed and revised by the authors, and the authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This study concerns the training and evaluation of reasoning models. Training questions are derived from a subset of the publicly released Nemotron-Cascade-2-SFT-Data. Evaluation uses eleven publicly released benchmarks: AIME 2025, AIME 2026, HMMT 2025, HMMT 2026, BruMO 2025, IMO-AnswerBench, AMO-Bench, ArxivMath, ZebraLogicBench, GPQA-Diamond, and SuperGPQA. These datasets and benchmarks were obtained from their public sources and used subject to their respective licenses and terms of use. This work does not recruit human participants or collect private user data, and we do not intentionally introduce personally identifiable information into the training or evaluation data. The empirical comparison reports results for 22 external systems in addition to BoundaryEvo: SmolLM3-3B, VibeThinker-3B, DASD-4B-Thinking, Hunyuan-4B-Instruct, Qwen3-4B-Thinking-2507, Qwen3.5-4B, NVIDIA-Nemotron-3-Nano-4B-BF16, Spark-X2.5-4B, Olmo-3-7B-Think, MiMo-7B-RL-0530, OpenReasoning-Nemotron-7B, Gemma-4-12B-it, Phi-4-reasoning-plus, Ministral-3-14B-Reasoning-2512, gpt-oss-20b, NVIDIA-Nemotron-3-Nano-30B-A3B-BF16, NVIDIA-Nemotron-3.5-Lightning-30B-A3B-BF16, GLM-4.7-Flash, GLM-4.5-Air, Qwen3-235B-A22B-Thinking, LongCat-Flash-Thinking, and GPT-5 nano. Cross-backbone experiments additionally use Qwen3.5-2B and Gemma-4-E4B-it. Data construction uses DeepSeek-V3.2-Speciale as the trajectory teacher, GPT-5.1 as the capability-boundary expert, and Gemini 3.5 Flash as the analyzer. Open checkpoints were accessed from their released sources, while proprietary models were accessed through authorized APIs under the providers' terms of service.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the BoundaryEvo framework, data construction process, training procedure, and evaluation settings in the main paper and supplementary materials. Implementation details, including optimization configurations and hyperparameters, are provided in the appendix. We will release the code and processed data to facilitate reproduction and future research.

REFERENCES

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Besmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, Vibhav Vineet, Yue Wu, Safoora Yousefi, and Guoqing Zheng. Phi-4-reasoning technical report, 2025. URL https://arxiv.org/abs/2504.21318.

Shengnan An, Xunliang Cai, Xuezhi Cao, Xiaoyu Li, Yehao Lin, Junlin Liu, Xinxuan Lv, Dan Ma, Xuanlin Wang, Ziwen Wang, and Shuang Zhou. Amo-bench: Large language models still struggle in high school math competitions, 2025. URL https://arxiv.org/abs/2510.26768.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, et al. SmolLM3: smol, multilingual, long-context reasoner. https://huggingface.co/blog/smollm3, 2025.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, volume 382 of *ACM International Conference Proceeding Series*, pp. 41–48. ACM, 2009.

BRUMO. Brown university math olympiad 2025. https://www.brumo.org/, 2025. Accessed: 2025.

Xiuwei Chen, Wentao Hu, Hanhui Li, Yongxin Wang, Jun Zhou, Zisheng Chen, Meng Cao, Yihan Zeng, Kui Zhang, Yu-Jie Yuan, Jianhua Han, Hang Xu, and Xiaodan Liang. Syncloop: A

multimodal dual-loop framework for self-improving mathematical reasoning. In *ECCV (36)*, volume 17036 of *Lecture Notes in Computer Science*, pp. 37–54. Springer, 2026.

Xuanzhong Chen, Zile Qiao, Guoxin Chen, Liangcai Su, Zhen Zhang, Xinyu Wang, Pengjun Xie, Fei Huang, Jingren Zhou, and Yong Jiang. Agentfrontier: Expanding the capability frontier of llm agents with zpd-guided data synthesis, 2025. URL https://arxiv.org/abs/2510.24695.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948, 2025.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Erhang Li, Fangqi Zhou, Fangyun Lin, Fucong Dai, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoran Wei, Haowei Zhang, Haowen Luo, Haozhe Ji, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, Jialiang Huang, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jingchang Chen, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jinhua Zhu, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexin Huang, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Liang Zhao, Liangsheng Yin, Lihua Guo, Lingxiao Luo, Linwang Ma, Litong Wang, Liyue Zhang, M. S. Di, M. Y Xu, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Panpan Huang, Peixin Cong, Peiyi Wang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, S. H. Liu, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Songyang Zhou, Tao Ni, Tao Yun, Tian Pei, Tian Ye, Tianyuan Yue, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjun Gao, Wentao Zhang, Xi Gao, Xiangwen Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyuan Li, Xu Chen, Xuecheng Su, Xuehai Pan, Xuheng Lin, Xuwei Fu, Y. Q. Wang, Yang Zhang, Yanhong Xu, Yanru Ma, Yao Li, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yiliang Xiong, Ying He, Ying Zhou, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuduan Wang, Yue Gong, Yuhan Wu, Yuheng Zou, Yukun Li, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehua Zhao, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhiyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Zizheng Pan, Zongqing Yao, Bei Feng, Hui Li, J. L. Cai, Jiaqi Ni, Lei Xu, Meng Li, Ning Tian, R. J. Chen, R. L. Jin, S. S. Li, Shuang Zhou, Tianyu Sun, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xinnan Song, Xinyi Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, Dongjie Ji, Jian Liang, Jianzhong Guo, Jin Chen, Leyi Xia, Miaojun Wang, Mingming Li, Peng Zhang, Ruyi Chen, Shangmian Sun, Shaoqing Wu, Shengfeng Ye, T. Wang, W. L. Xiao, Wei An, Xianzu Wang, Xiaowen Sun, Xiaoxiang Wang, Ying Tang, Yukun Zha, Zekai Zhang, Zhe Ju, Zhen Zhang, and Zihua Qu. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: Matharena as an evaluation platform for mathematics with llms, 2026. URL https://arxiv.org/abs/2605.00674.

Wenkai Fang, Shunyu Liu, Yang Zhou, Kongcheng Zhang, Tongya Zheng, Kaixuan Chen, Mingli Song, and Dacheng Tao. Serl: Self-play reinforcement learning for large language models with limited data. In *NeurIPS*, 2025.

Google DeepMind. Gemini 3.5 Flash. https://ai.google.dev/gemini-api/docs/models/gemini-3.5-flash, 2026. Accessed: 2026-08-20.

Çaglar Gülçehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. Reinforced self-training (rest) for language modeling. *CoRR*, abs/2308.08998, 2023.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need. *CoRR*, abs/2306.11644, 2023.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, volume ACL 2025 of *Findings of ACL*, pp. 24842–24855. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.1274. URL https://doi.org/10.18653/v1/2025.findings-acl.1274.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

HMMT. Hmmt 2025. https://www.hmmt.org/, 2025. Accessed: 2025.

HMMT. Hmmt 2026. https://www.hmmt.org/, 2026. Accessed: 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net, 2022.

Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-zero: Self-evolving reasoning LLM from zero data. *CoRR*, abs/2508.05004, 2025.

Zhen Huang, Haoyang Zou, Xuefeng Li, Yixiu Liu, Yuxiang Zheng, Ethan Chern, Shijie Xia, Yiwei Qin, Weizhe Yuan, and Pengfei Liu. O1 replication journey - part 2: Surpassing o1-preview through simple distillation, big progress or bitter lesson? *CoRR*, abs/2411.16489, 2024.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *EMNLP*, pp. 1317–1327. The Association for Computational Linguistics, 2016.

Ilia Kulikov, Chenxi Whitehouse, Tianhao Wu, Yixin Nie, Swarnadeep Saha, Eryk Helenowski, Weizhe Yuan, Olga Golovneva, Jack Lanchantin, Yoram Bachrach, Jakob N. Foerster, Xian Li, Han Fang, Sainbayar Sukhbaatar, and Jason Weston. Autodata: An agentic data scientist to create high quality synthetic data. *CoRR*, abs/2606.25996, 2026.

Nicholas Lee, Thanakul Wattanawong, Sehoon Kim, Karttikeya Mangalam, Sheng Shen, Gopala Anumanchipalli, Michael W. Mahoney, Kurt Keutzer, and Amir Gholami. LLM2LLM: boosting llms with novel iterative data enhancement. In *ACL (Findings)*, volume ACL 2024 of *Findings of ACL*, pp. 6498–6526. Association for Computational Linguistics, 2024.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *ICLR*. OpenReview.net, 2024.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. Zebralogic: On the scaling limits of llms for logical reasoning, 2025. URL https://arxiv.org/abs/2502.01100.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, Baptiste Rozière, Baudouin De Monicault, Clémence Lanfranchi, Corentin Barreau, Cyprien Courtot, Daniele Grattarola, Darius Dabert, Diego de las Casas, Elliot Chane-Sane, Faruk Ahmed, Gabrielle Berrada, Gaëtan Ecrepont, Gauthier Guinet, Georgii Novikov, Guillaume Kunsch, Guillaume Lample, Guillaume Martin, Gunshi Gupta, Jan Ludziejewski, Jason Rute, Joachim Studnia, Jonas Amar, Joséphine Delas, Josselin Somerville Roberts, Karmesh Yadav, Khyathi Chandu, Kush Jain, Laurence Aitchison, Laurent Fainsin, Léonard Blier, Lingxiao Zhao, Louis Martin, Lucile Saulnier, Luyu Gao, Maarten Buyl, Margaret Jennings, Marie Pellat, Mark Prins, Mathieu Poirée, Mathilde Guillaumin, Matthieu Dinot, Matthieu Futeral, Maxime Darrin, Maximilian Augustin, Mia Chiquier, Michel Schimpf, Nathan Grinsztajn, Neha Gupta, Nikhil Raghuraman, Olivier Bousquet, Olivier Duchenne, Patricia Wang, Patrick von Platen, Paul Jacob, Paul Wambergue, Paula Kurylowicz, Pavankumar Reddy Muddireddy, Philomène Chagniot, Pierre Stock, Pravesh Agrawal, Quentin Torroba, Romain Sauvestre, Roman Soletskyi, Rupert Menneer, Sagar Vaze, Samuel Barry, Sanchit Gandhi, Siddhant Waghjale, Siddharth Gandhi, Soham Ghosh, Srijan Mishra, Sumukh Aithal, Szymon Antoniak, Teven Le Scao, Théo Cachet, Theo Simon Sorg, Thibaut Lavril, Thiziri Nait Saada, Thomas Chabal, Thomas Foubert, Thomas Robert, Thomas Wang, Tim Lawson, Tom Bewley, Tom Bewley, Tom Edwards, Umar Jamil, Umberto Tomasini, Valeriia Nemychnikova, Van Phung, Vincent Maladière, Virgile Richard, Wassim Bouaziz, Wen-Ding Li, William Marshall, Xinghui Li, Xinyu Yang, Yassine El Ouahidi, Yihan Wang, Yunhao Tang, and Zaccharie Ramzi. Ministral 3, 2026. URL https://arxiv.org/abs/2601.08584.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? A comprehensive study of automatic data selection in instruction tuning. In *ICLR*. OpenReview.net, 2024.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jian-Guang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *ICLR*. OpenReview.net, 2025.

Thang Luong, Dawsen Hwang, Hoang H. Nguyen, Golnaz Ghiasi, Yuri Chervonyi, Insuk Seo, Junsu Kim, Garrett Bingham, Jonathan Lee, Swaroop Mishra, Alex Zhai, Clara Huiyi Hu, Henryk Michalewski, Jimin Kim, Jeonghyun Ahn, Junhwi Bae, Xingyou Song, Trieu H. Trinh, Quoc V. Le, and Junehyuk Jung. Towards robust mathematical reasoning, 2025. URL https://arxiv.org/abs/2511.01846.

Mathematical Association of America. Maa invitational competitions. https://maa.org/maa-invitational-competitions/, 2026. Accessed: 2026-04-09.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nat.*, 518(7540):529–533, 2015.

Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. AIMO-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. *CoRR*, abs/2504.16891, 2025.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel J. Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In *EMNLP*, pp. 20275–20321. Association for Computational Linguistics, 2025.

NVIDIA. OpenReasoning-Nemotron-7B. https://huggingface.co/nvidia/OpenReasoning-Nemotron-7B, 2025.

NVIDIA. Nemotron-Cascade-2-SFT-Data. https://huggingface.co/datasets/nvidia/Nemotron-Cascade-2-SFT-Data, 2026a.

NVIDIA. Nemotron-3.5-Lightning-30b-a3b. https://build.nvidia.com/nvidia/nemotron-3.5-lightning-30b-a3b/modelcard, 2026b.

NVIDIA. NVIDIA Nemotron-3-Nano-4B. https://developer.download.nvidia.com/assets/ace/model_card/Nemotron_3_Nano_4B.pdf, 2026c.

NVIDIA, :, Aaron Blakeman, Aaron Grattafiori, Aarti Basant, Abhibha Gupta, Abhinav Khattar, Adi Renduchintala, Aditya Vavre, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, Aleksandr Shaposhnikov, Alex Kondratenko, Alexander Bukharin, Alexandre Milesi, Ali Taghibakhshi, Alisa Liu, Amelia Barton, Ameya Sunil Mahabaleshwarkar, Amir Klein, Amit Zuker, Amnon Geifman, Amy Shen, Anahita Bhiwandiwalla, Andrew Tao, Ann Guan, Anubhav Mandarwal, Arham Mehta, Ashwath Aithal, Ashwin Poojary, Asif Ahamed, Asma Kuriparambil Thekkumpate, Ayush Dattagupta, Banghua Zhu, Bardiya Sadeghi, Barnaby Simkin, Ben Lanir, Benedikt Schifferer, Besmira Nushi, Bilal Kartal, Bita Darvish Rouhani, Boris Ginsburg, Brandon Norick, Brandon Soubasis, Branislav Kisacanin, Brian Yu, Bryan Catanzaro, Carlo del Mundo, Chantal Hwang, Charles Wang, Cheng-Ping Hsieh, Chenghao Zhang, Chenhan Yu, Chetan Mungekar, Chintan Patel, Chris Alexiuk, Christopher Parisien, Collin Neale, Damon Mosk-Aoyama, Dan Su, Dane Corneil, Daniel Afrimi, Daniel Rohrer, Daniel Serebrenik, Daria Gitman, Daria Levy, Darko Stosic, David Mosallanezhad, Deepak Narayanan, Dhruv Nathawani, Dima Rekesh, Dina Yared, Divyanshu Kakwani, Dong Ahn, Duncan Riach, Dusan Stosic, Edgar Minasyan, Edward Lin, Eileen Long, Eileen Peters Long, Elena Lantz, Ellie Evans, Elliott Ning, Eric Chung, Eric Harper, Eric Tramel, Erick Galinkin, Erik Pounds, Evan Briones, Evelina Bakhturina, Faisal Ladhak, Fay Wang, Fei Jia, Felipe Soares, Feng Chen, Ferenc Galko, Frankie Siino, Gal Hubara Agam, Ganesh Ajjanagadde, Gantavya Bhatt, Gargi Prasad, George Armstrong, Gerald Shen, Gorkem Batmaz, Grigor Nalbandyan, Haifeng Qian, Harsh Sharma, Hayley Ross, Helen Ngo, Herman Sahota, Hexin Wang, Himanshu Soni, Hiren Upadhyay, Huizi Mao, Huy C Nguyen, Huy Q Nguyen, Iain Cunningham, Ido Shahaf, Igor Gitman, Ilya Loshchilov, Ivan Moshkov, Izzy Putterman, Jan Kautz, Jane Polak Scowcroft, Jared Casper, Jatin Mitra, Jeffrey Glick, Jenny Chen, Jesse Oliver, Jian Zhang, Jiaqi Zeng, Jie Lou, Jimmy Zhang, Jining Huang, Joey Conway, Joey Guman, John Kamalu, Johnny Greco, Jonathan Cohen, Joseph Jennings, Joyjit Daw, Julien Veron Vialard, Junkeun Yi, Jupinder Parmar, Kai Xu, Kan Zhu, Kari Briski, Katherine Cheung, Katherine Luna, Keshav Santhanam, Kevin Shih, Kezhi Kong, Khushi Bhardwaj, Krishna C. Puvvada, Krzysztof Pawelec, Kumar Anik, Lawrence McAfee, Laya Sleiman, Leon Derczynski, Li Ding, Lucas Liebenwein, Luis Vega, Maanu Grover, Maarten Van Segbroeck, Maer Rodrigues de Melo, Makesh Narsimhan Sreedhar, Manoj Kilaru, Maor Ashkenazi, Marc Romeijn, Mark Cai, Markus Kliegl, Maryam Moosaei, Matvei Novikov, Mehrzad Samadi, Melissa Corpuz, Mengru Wang, Meredith Price, Michael Boone, Michael Evans, Miguel Martinez, Mike Chrzanowski, Mohammad Shoeybi, Mostofa Patwary, Nabin Mulepati, Natalie Hereth, Nave Assaf, Negar Habibi, Neta Zmora, Netanel Haber, Nicola Sessions, Nidhi Bhatia, Nikhil Jukar, Nikki Pope, Nikolai Ludwig, Nima Tajbakhsh, Nirmal Juluru, Oleksii Hrinchuk, Oleksii Kuchaiev, Olivier Delalleau, Oluwatobi Olabiyi, Omer Ullman Argov, Ouye Xie, Parth Chadha, Pasha Shamis, Pavlo Molchanov, Pawel Morkisz, Peter Dykas, Peter Jin, Pinky Xu, Piotr Januszewski, Pranav Prashant Thombre, Prasoon Varshney, Pritam Gundecha, Qing Miao, Rabeeh Karimi Mahabadi, Ran El-Yaniv, Ran Zilberstein, Rasoul Shafipour, Rich Harang, Rick Izzo, Rima Shahbazyan, Rishabh Garg, Ritika Borkar, Ritu Gala, Riyad Islam, Roger Waleffe, Rohit Watve, Roi Koren, Ruoxi Zhang, Russell J. Hewett, Ryan Prenger, Ryan Timbrook, Sadegh Mahdavi, Sahil Modi, Samuel Kriman, Sanjay Kariyappa, Sanjeev Satheesh, Saori Kaji, Satish Pasumarthi, Sean Narentharen, Sean Narenthiran, Seonmyeong Bak, Sergey Kashirsky, Seth Poulos, Shahar Mor, Shanmugam Ramasamy, Shantanu Acharya, Shaona Ghosh, Sharath Turuvekere Sreenivas, Shelby Thomas, Shiqing Fan, Shreya Gopal, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shuoyang Ding, Siddharth Singh, Simeng Sun, Smita Ithape, Somshubra Majumdar, Soumye Singhal, Stefania Alborghetti, Stephen Ge, Sugam Dipak Devare, Sumeet Kumar Barua, Suseella Panguluri, Suyog Gupta, Sweta Priyadarshi, Syeda Nahida Akter, Tan Bui, Teodor-Dumitru Ene, Terry Kong, Thanh Do, Tijmen Blankevoort, Tom Balough, Tomer Asida, Tomer Bar Natan, Tugrul Konuk, Twinkle Vashishth, Udi Karpas, Ushnish De, Vahid Noorozi, Vahid Noroozi, Venkat Srinivasan, Venmugil Elango, Vijay Korthikanti, Vitaly Kurin, Vitaly Lavrukhin, Wanli Jiang, Wasi Uddin Ahmad, Wei Du, Wei Ping, Wenfei Zhou, Will Jennings, William Zhang, Wojciech Prazuch, Xiaowei Ren, Yashaswi Karnati, Yejin Choi, Yev Meyer, Yi-Fu Wu, Yian Zhang, Ying Lin, Yonatan Geifman, Yonggan Fu, Yoshi Subara, Yoshi Suhara, Yubo Gao, Zach Moshe, Zhen Dong, Zihan Liu, Zijia Chen, and Zijie Yan. Nemotron 3

nano: Open, efficient mixture-of-experts hybrid mamba-transformer model for agentic reasoning, 2025. URL https://arxiv.org/abs/2512.20848.

Team Olmo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2026. URL https://arxiv.org/abs/2512.13961.

OpenAI. Gpt-5.1 model. https://developers.openai.com/api/docs/models/gpt-5.1, 2025. Accessed: 2026-09-22.

OpenAI, :, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily, Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy, Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Qwen Team. Qwen3-4B-Thinking-2507. https://huggingface.co/Qwen/Qwen3-4B-Thinking-2507, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. URL https://qwen.ai/blog?id=qwen3.5.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023. URL https://arxiv.org/abs/2311.12022.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B. Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM Comput. Surv.*, 54(9):180:1–180:40, 2022.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *ICLR (Poster)*, 2016.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa

Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, CJ Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, DJ Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang, Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang, Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Y. Guan, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, RJ Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song Mei, Sonya Egoian, Spencer Kim, Spug Golden, SQ Mah, Steph Lin, Stephen Imm, Steve Sharpe,

Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer, Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomek Korbak, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

Hwanjun Song, Sundong Kim, Minseok Kim, and Jae-Gil Lee. Ada-boundary: accelerating DNN training via adaptive boundary batch selection. *Mach. Learn.*, 109(9-10):1837–1853, 2020.

5 Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, Yean Cheng, Yifan An, Yilin Niu, Yuanhao Wen, Yushi Bai, Zhengxiao Du, Zihan Wang, Zilin Zhu, Bohan Zhang, Bosi Wen, Bowen Wu, Bowen Xu, Can Huang, Casey Zhao, Changpeng Cai, Chao Yu, Chen Li, Chendi Ge, Chenghua Huang, Chenhui Zhang, Chenxi Xu, Chenzheng Zhu, Chuang Li, Congfeng Yin, Daoyan Lin, Dayong Yang, Dazhi Jiang, Ding Ai, Erle Zhu, Fei Wang, Gengzheng Pan, Guo Wang, Hailong Sun, Haitao Li, Haiyang Li, Haiyi Hu, Hanyu Zhang, Hao Peng, Hao Tai, Haoke Zhang, Haoran Wang, Haoyu Yang, He Liu, He Zhao, Hongwei Liu, Hongxi Yan, Huan Liu, Huilong Chen, Ji Li, Jiajing Zhao, Jiamin Ren, Jian Jiao, Jiani Zhao, Jianyang Yan, Jiaqi Wang, Jiayi Gui, Jiayue Zhao, Jie Liu, Jijie Li, Jing Li, Jing Lu, Jingsen Wang, Jingwei Yuan, Jingxuan Li, Jingzhao Du, Jinhua Du, Jinxin Liu, Junkai Zhi, Junli Gao, Ke Wang, Lekang Yang, Liang Xu, Lin Fan, Lindong Wu, Lintao Ding, Lu Wang, Man Zhang, Minghao Li, Minghuan Xu, Mingming Zhao, Mingshu Zhai, Pengfan Du, Qian Dong, Shangde Lei, Shangqing Tu, Shangtong Yang, Shaoyou Lu, Shijie Li, Shuang Li, Shuang-Li, Shuxun Yang, Sibo Yi, Tianshu Yu, Wei Tian, Weihan Wang, Wenbo Yu, Weng Lam Tam, Wenjie Liang, Wentao Liu, Xiao Wang, Xiaohan Jia, Xiaotao Gu, Xiaoying Ling, Xin Wang, Xing Fan, Xingru Pan, Xinyuan Zhang, Xinze Zhang, Xiuqing Fu, Xunkai Zhang, Yabo Xu, Yandong Wu, Yida Lu, Yidong Wang, Yilin Zhou, Yiming Pan, Ying Zhang, Yingli Wang, Yingru Li, Yinpei Su, Yipeng Geng, Yitong Zhu, Yongkun Yang, Yuhang Li, Yuhao Wu, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yuxuan Zhang, Zezhen Liu, Zhen Yang, Zhengda Zhou, Zhongpei Qiao, Zhuoer Feng, Zhuorui Liu, Zichen Zhang, Zihan Wang, Zijun Yao, Zikang Wang, Ziqiang Liu, Ziwei Chai, Zixuan Li, Zuodong Zhao, Wenguang Chen, Jidong Zhai, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025a. URL https://arxiv.org/abs/2508.06471.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez, Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni,

Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan, Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shangguan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege, Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Meituan LongCat Team, Anchun Gui, Bei Li, Bingyang Tao, Bole Zhou, Borun Chen, Chao Zhang, Chao Zhang, Chengcheng Han, Chenhui Yang, Chi Zhang, Chong Peng, Chuyu Zhang, Cong Chen, Fengcun Li, Gang Xu, Guoyuan Lin, Hao Jiang, Hao Liang, Haomin Fu, Haoxiang Ma, Hong Liu, Hongyan Hao, Hongyin Tang, Hongyu Zang, Hongzhi Ni, Hui Su, Jiahao Liu, Jiahuan Li, Jialin Liu, Jianfei Zhang, Jianhao Xu, Jianing Wang, Jiaqi Sun, Jiaqi Zhang, Jiarong Shi, Jiawei Yang, Jingang Wang, Jinrui Ding, Jun Kuang, Jun Xu, Ke He, Kefeng Zhang, Keheng Wang, Keqing He, Li Wei, Liang Shi, Lin Qiu, Lingbin Kong, Lingchuan Liu, Linsen Guo, Longfei An, Mai Xia, Meng Zhou, Mengshen Zhu, Peng Pei, Pengcheng Jia, Qi Gu, Qi Guo, Qiong Huang, Quan Chen, Quanchi Weng, Rongxiang Weng, Ruichen Shao, Rumei Li, Shanglin Lei, Shuai Du, Shuaikang Liu, Shuang Zhou, Shuhao Hu, Siyu Xu, Songshan Gong, Tao Liang, Tianhao Hu, Wei He, Wei Shi, Wei Wang, Wei Wu, Wei Zhuo, Weifeng Tang, Wenjie Shi, Wenlong Zhu, Xi Su, Xiangcheng Liu, Xiangyu Xi, Xiangzhou Huang, Xiao Liu, Xiaochen Jiang, Xiaowei Shi, Xiaowen Shi, Xiaoyu Li, Xin Chen, Xinyue Zhao, Xuan Huang, Xuemiao Zhang, Xuezhi Cao, Xunliang Cai, Yajie Zhang, Yang Chen, Yang Liu, Yang Liu, Yang Zheng, Yaoming Wang, Yaqi Huo, Yerui Sun, Yifan Lu, Yiyang Li, Youshao Xiao, Yuanzhe Lei, Yuchen Xie, Yueqing Sun, Yufei Zhang, Yuhuai Wei, Yulei Qian, Yunke Zhao, Yuqing Ding, Yuwei Jiang, Zhaohua Yang, Zhengyu Chen, Zhijian Liu, Zhikang Xia, Zhongda Su, Ziran Li, Ziwen Wang, Ziyuan Zhuang, Zongyu Wang, and Zunyuan Yang. Introducing longcat-flash-thinking: A technical report, 2025b. URL https://arxiv.org/abs/2509.18883.

P Team, Xinrun Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, King Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, Kaixin Deng, Shawn Gavin, Shian Jia,

Sichao Jiang, Yiyan Liao, Rui Li, Qinrui Li, Sirun Li, Yizhi Li, Yunwen Li, David Ma, Yuansheng Ni, Haoran Que, Qiyao Wang, Zhoufutu Wen, Siwei Wu, Tyshawn Hsing, Ming Xu, Zhenzhu Yang, Zekun Moore Wang, Junting Zhou, Yuelin Bai, Xingyuan Bu, Chenglin Cai, Liang Chen, Yifan Chen, Chengtuo Cheng, Tianhao Cheng, Keyi Ding, Siming Huang, Yun Huang, Yaoru Li, Yizhe Li, Zhaoqun Li, Tianhao Liang, Chengdong Lin, Hongquan Lin, Yinghao Ma, Tianyang Pang, Zhongyuan Peng, Zifan Peng, Qige Qi, Shi Qiu, Xingwei Qu, Shanghaoran Quan, Yizhou Tan, Zili Wang, Chenqing Wang, Hao Wang, Yiya Wang, Yubo Wang, Jiajun Xu, Kexin Yang, Ruibin Yuan, Yuanhao Yue, Tianyang Zhan, Chun Zhang, Jinyang Zhang, Xiyue Zhang, Xingjian Zhang, Yue Zhang, Yongchi Zhao, Xiangyu Zheng, Chenghua Zhong, Yang Gao, Zhoujun Li, Dayiheng Liu, Qian Liu, Tianyu Liu, Shiwen Ni, Junran Peng, Yujia Qin, Wenbo Su, Guoyin Wang, Shi Wang, Jian Yang, Min Yang, Meng Cao, Xiang Yue, Zhaoxiang Zhang, Wangchunshu Zhou, Jiaheng Liu, Qunshu Lin, Wenhao Huang, and Ge Zhang. Supergpqa: Scaling llm evaluation across 285 graduate disciplines, 2025c. URL https://arxiv.org/abs/2502.14739.

SparkLLM Team. Spark-x2.5 4b&1.7b: Pushing the limits of agentic capabilities in on-device models, 2026.

Tencent Hunyuan Team. Hunyuan-4B-Instruct. https://huggingface.co/tencent/Hunyuan-4B-Instruct, 2025.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: improving LLM pretraining via document de-duplication and diversification. In *NeurIPS*, 2023.

Georgios Tzannetos, Bárbara Gomes Ribeiro, Parameswaran Kamalaruban, and Adish Singla. Proximal curriculum for reinforcement learning agents. *Trans. Mach. Learn. Res.*, 2023, 2023.

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

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *ACL (1)*, pp. 9426–9439. Association for Computational Linguistics, 2024.

Shaobo Wang, Zhengbo Jiao, Zifan Zhang, Yilang Peng, Xu Ze, Boyu Yang, Wei Wang, Hu Wei, and Linfeng Zhang. Socratic-zero : Bootstrapping reasoning via data-free agent co-evolution. *CoRR*, abs/2509.24726, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*. OpenReview.net, 2023a.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *ACL (1)*, pp. 13484–13508. Association for Computational Linguistics, 2023b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. In *ICML*, volume 235 of *Proceedings of Machine Learning Research*, pp. 52915–52971. PMLR / OpenReview.net, 2024.

LLM-Core Xiaomi, :, Bingquan Xia, Bowen Shen, Cici, Dawei Zhu, Di Zhang, Gang Wang, Hailin Zhang, Huaqiu Liu, Jiebao Xiao, Jinhao Dong, Liang Zhao, Peidian Li, Peng Wang, Shihua Yu, Shimao Chen, Weikun Wang, Wenhan Ma, Xiangwei Deng, Yi Huang, Yifan Song, Zihan Jiang, Bowen Ye, Can Cai, Chenhong He, Dong Zhang, Duo Zhang, Guoan Wang, Hao Tian, Haochen Zhao, Heng Qu, Hongshen Xu, Jun Shi, Kainan Bao, Kai Fang, Kang Zhou, Kangyang Zhou, Lei Li, Menghang Zhu, Nuo Chen, Qiantong Wang, Shaohui Liu, Shicheng Li, Shuhao Gu, Shuhuai Ren, Shuo Liu, Sirui Deng, Weiji Zhuang, Weiwei Lv, Wenyu Yang, Xin Zhang, Xing Yong, Xing Zhang, Xingchen Song, Xinzhe Xu, Xu Wang, Yihan Yan, Yu Tu, Yuanyuan

Tian, Yudong Wang, Yue Yu, Zhenru Lin, Zhichao Song, and Zihao Yue. Mimo: Unlocking the reasoning potential of language model – from pretraining to posttraining, 2025. URL https://arxiv.org/abs/2505.07608.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. WizardLM: Empowering large language models to follow complex instructions. *International Conference on Learning Representations*, 2024.

Sen Xu, Shixi Liu, Wei Wang, Jixin Min, Yingwei Dai, Zhibin Yin, Yirong Chen, Xin Zhou, and Junlin Zhang. Vibethinker-3b: Exploring the frontier of verifiable reasoning in small language models. *CoRR*, abs/2606.16140, 2026.

Shaotian Yan, Kaiyuan Liu, Chen Shen, Bing Wang, Sinan Fan, Jun Zhang, Yue Wu, Zheng Wang, and Jieping Ye. Distribution-aligned sequence distillation for superior long-cot reasoning. *CoRR*, abs/2601.09088, 2026.

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, 2025. URL https://arxiv.org/abs/2505.09388.

Suorong Yang, Fangjian Su, Hai Gan, Ziqi Ye, Jie Li, Baile Xu, Furao Shen, and Soujanya Poria. Data agent: Learning to select data via end-to-end dynamic optimization, 2026. URL https://arxiv.org/abs/2603.07433.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: less is more for reasoning. *CoRR*, abs/2502.03387, 2025.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *ICLR*. OpenReview.net, 2024.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. In *ICML*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57905–57923. PMLR / OpenReview.net, 2024.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning. In *NeurIPS*, 2022.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*, 2023.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: less is more for alignment. In *NeurIPS*, 2023.

A Hyperparameter and Selection Sensitivity

We retain three factors that directly affect the central behavior of BoundaryEvo: the number of SFT epochs per update, the number of evolution turns, and the minimum capability gap used by the boundary filter. Other training settings follow Table 6 and are not repeated here. We vary one factor at a time from the reference configuration while holding the backbone, Stage 1 checkpoint, evolution pool, expert, analyzer, prompts, and per-turn budget fixed.

Table 5: Core hyperparameter ablations on Qwen3.5-4B. E is the number of SFT epochs per update, R is the number of Stage 2 turns, and τ_{g} is the minimum expert–student capability gap. Stage 2 data count is $300R$ and counts replayed examples whenever they are reused. Each non-reference row changes one factor only.

Ablated factor	E	R	τ_{g}	**Stage 2 data**	**Math / General**
Reference	5	3	0.20	900	76.4 / 77.4
Epochs	1	3	0.20	900	72.6 / 75.9
Epochs	3	3	0.20	900	73.5 / 76.1
Epochs	10	3	0.20	900	74.3 / 77.0
Evolution turns	5	1	0.20	300	71.2 / 75.5
Evolution turns	5	2	0.20	600	73.3 / 76.9
Evolution turns	5	4	0.20	1200	75.8 / 77.0
Minimum gap	5	3	0.10	900	76.1 / 77.2
Minimum gap	5	3	0.30	900	76.2 / 77.2

The epoch sweep tests whether the result depends on under- or over-training the selected trajectories. The turn sweep measures the benefit of repeatedly relocating the capability boundary as the student changes. The gap sweep tests whether selection is robust to admitting easier examples or requiring a stricter separation between expert and student performance. Together, these compact controls isolate the most consequential optimization, evolution, and selection choices.

B Training Details

This section reports the optimization and implementation details used for capability priming and the subsequent evolution turns. Unless stated otherwise, Stage 1 and Stage 2 use the same supervised fine-tuning recipe. The principal differences across backbones are the conversation template, the LoRA target modules, and long-sequence preprocessing.

Optimization and Parameter-Efficient Fine-Tuning We train all student models with response-only LoRA SFT using LLaMA-Factory. Tokens belonging to the question and system prompt are masked from the loss, so optimization is applied only to the teacher response. For Qwen3.5, LoRA is attached to all linear modules. For Gemma-4-E4B-it, it is attached to the query, key, value, output, gate, up, and down projection modules. Table 6 summarizes the remaining settings.

Training jobs use one or four H200 GPUs depending on resource availability while preserving the default effective batch size of eight. We use gradient accumulation of 2 for four-GPU jobs and 8 for one-GPU jobs. With 100 Stage 1 examples, five epochs correspond to 65 optimizer updates. A standard 300-example Stage 2 run contains 190 updates.

Stage-Wise Initialization and Training Budget Stage 1 starts from the original pretrained backbone and trains on 100 direct teacher trajectories. Its merged checkpoint is denoted by M_0. Stage 2 is sequential: Turn 1 starts from M_0, and Turn $t+1$ starts from the merged student produced by Turn t. Each turn contains 300 verified teacher trajectories selected from the corresponding capability-boundary question set. Thus, the reported three-turn configuration allocates $100 + 3 \times 300 = 1000$ training examples. This accounting counts an example again when it is replayed in a later turn.

After each update, we merge the learned LoRA adapter into its initialization checkpoint before using the resulting model for evaluation or as the next turn's student. Consequently, every evolution turn

Table 6: Training configuration used in experiments. The effective batch size is the product of the per-device batch size, gradient accumulation steps, and the number of data-parallel workers.

Setting	**Qwen3.5-4B / 2B**	**Gemma-4-E4B-it**
Training objective	Response-only SFT	Response-only SFT
Fine-tuning method	LoRA	LoRA
LoRA rank / scaling factor	64 / 128	64 / 128
LoRA target modules	All linear modules	q, k, v, o, gate, up, and down projections
Trainable parameters	129.9M / 67.3M	139.5M
Conversation template	`qwen3`	`gemma4`
Epochs per update	5	5
Optimizer	AdamW, fused PyTorch implementation	Same
Learning rate	1×10^{-4}	1×10^{-4}
Scheduler / warm-up	Cosine / 10%	Cosine / 10%
Adam $\beta_1, \beta_2, \epsilon$	$0.9, 0.999, 10^{-8}$	Same
Weight decay / gradient clipping	0 / 1.0	Same
Numerical precision	bfloat16	bfloat16
Distributed optimization	DeepSpeed ZeRO-3	DeepSpeed ZeRO-3
Gradient checkpointing	Enabled	Enabled
Per-device batch size	1	1
Default effective batch size	8	8
Maximum trainer sequence length	40,960 tokens	40,960 tokens
Maximum retained training length	40,960 tokens	32,768 tokens
Random seed	42	42

selects data with the fully updated student rather than with an unmerged adapter or the original base model. Random-selection and LLM2LLM baselines use the same SFT objective, optimizer family, epoch count, and backbone-specific formatting.

Long-Trajectory Preprocessing Training records are stored in OpenAI-style `messages` format and contain only fields consumed by the trainer. Qwen3.5 runs use the `qwen3` template and a 40,960-token cutoff. For Gemma-4-E4B-it, we render each record with the native `gemma4` template and tokenizer before training. We retain only examples whose rendered length does not exceed 32,768 tokens, while keeping the trainer ceiling at 40,960 tokens. This separates data selection from implicit truncation and ensures that accepted Gemma trajectories remain complete. Reasoning markers are converted to the backbone's native chat representation before tokenization. Across backbones, malformed records and examples without a usable assistant response are discarded during preprocessing rather than padded with synthetic content.

C Reasoning-Trajectory Length Analysis

We test whether the accuracy gains in Table 2 require uniformly longer generations. We compare Qwen3.5-4B and the final BoundaryEvo-4B checkpoint on seven benchmarks, retaining only exact normalized question matches. Repeated rollouts are summarized by their per-question median, so every matched item has equal weight. We measure reasoning tokens with the Qwen tokenizer after removing the final-answer segment and three thinking-boundary tokens.

The final model has a lower median reasoning length on every matched benchmark, with reductions ranging from 7.4% on ArxivMath to 52.9% on AIME26. These reductions coincide with higher accuracy on all seven tasks in Table 2. For example, AIME26 improves from 84.0 to 93.3 while its median trajectory falls from 36.9K to 17.4K tokens, and ZebraLogic improves from 92.4 to 96.0

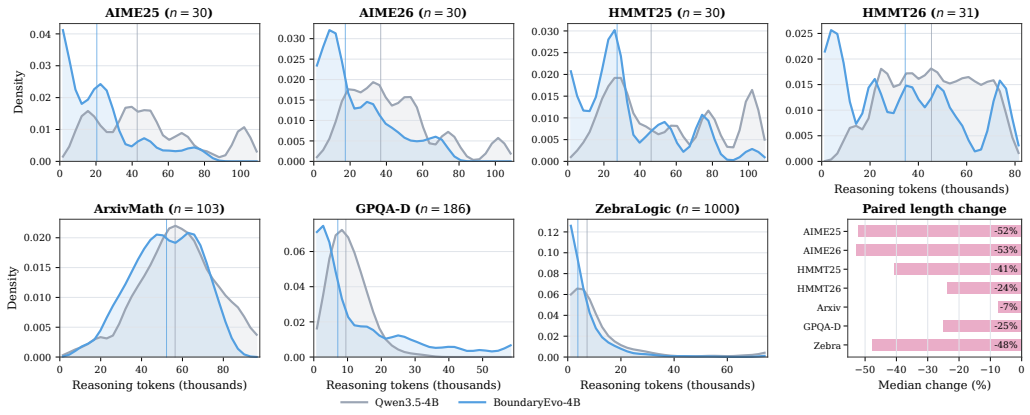


Figure 5: Reasoning-length distributions on matched evaluation questions. Curves show smoothed densities of per-question reasoning-token counts, and vertical lines mark medians. The final panel reports the relative change in median length from Qwen3.5-4B to BoundaryEvo-4B. Repeated rollouts of the same question are summarized by their median before plotting.

Table 7: Reasoning-length statistics for the matched questions in Figure 5. Token counts are reported in thousands. Δ denotes the relative change in median length from the base model to BoundaryEvo-4B.

Benchmark	n	**Median**		Δ	**90th percentile**	
		Base	**Ours**		**Base**	**Ours**
AIME25	30	42.8	20.5	-52.2%	99.4	52.1
AIME26	30	36.9	17.4	-52.9%	74.9	54.9
HMMT25	30	46.1	27.3	-40.8%	102.4	74.3
HMMT26	31	45.4	34.6	-23.9%	71.3	74.6
ArxivMath	103	56.4	52.2	-7.4%	82.1	71.2
GPQA-D	186	9.5	7.1	-25.2%	17.4	36.4
ZebraLogic	1000	7.3	3.8	-47.6%	25.7	16.5

while its median falls from 7.3K to 3.8K tokens. The result indicates that the gains do not arise from a uniform increase in inference length.

Upper-tail behavior remains task dependent. The 90th percentile falls on five tasks, changes little on HMMT26, and rises on GPQA-D. Thus, BoundaryEvo does not simply truncate every response. GPQA-D combines a lower median with a longer tail. Although this observational analysis does not establish causality, the joint accuracy gains and lower typical lengths are consistent with more efficient, rather than merely more verbose, reasoning.

D GPQA-D Scaling and Data Efficiency

We further examine how BoundaryEvo compares with models of different scales and reported post-training budgets. Both plots in this section use GPQA-D as the common performance measure and take their values directly from Table 1. We omit systems whose parameter count or GPQA-D result is unavailable rather than estimating missing values.

Figure 6 shows that strong GPQA-D performance is not determined by parameter count alone. BoundaryEvo-4B reaches 81.3, improving over its Qwen3.5-4B base model by 5.1 points and exceeding every reported model up to 12B parameters. It is also competitive with substantially larger systems, including Qwen3-235B at 81.1 and LongCat-560B at 81.5, while Phi-4-Reasoning at 14B obtains the highest displayed score of 81.9. These results place BoundaryEvo on the upper performance frontier of the plot, rather than suggesting that it dominates every larger model.

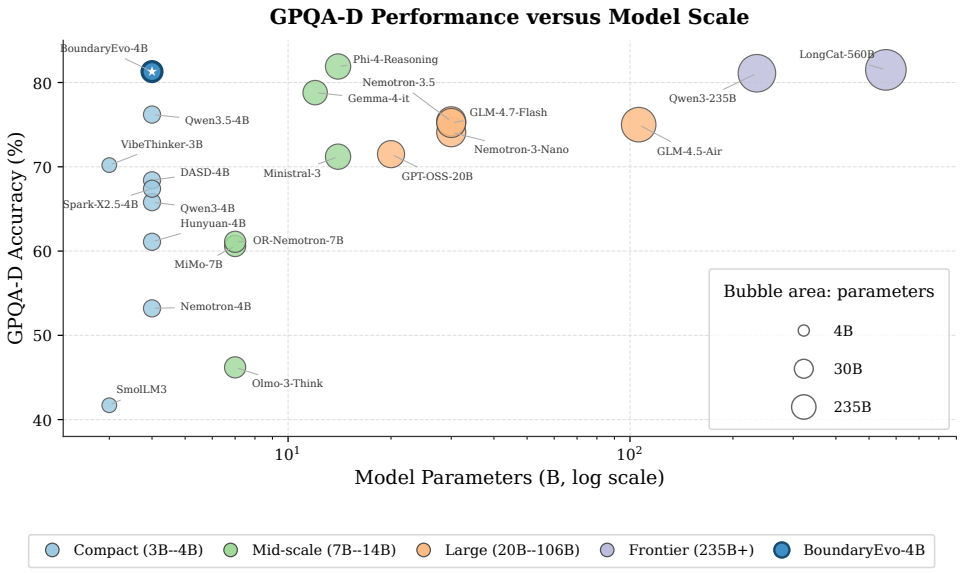


Figure 6: GPQA-D performance versus model scale. The horizontal axis is logarithmic, bubble area encodes parameter count, and color denotes the parameter range. BoundaryEvo-4B is highlighted with a white star. Only models with both a reported parameter count and GPQA-D score in Table 1 are included.

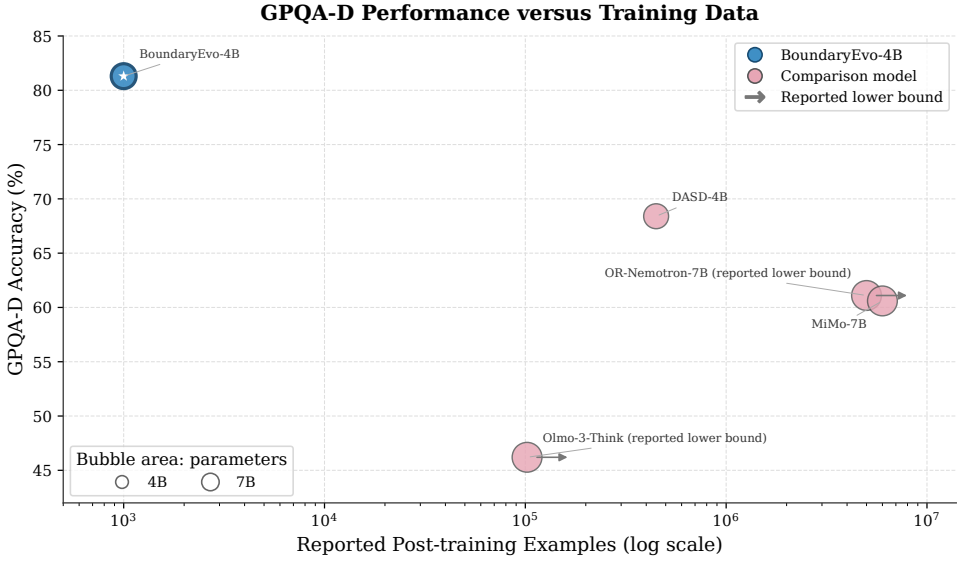


Figure 7: GPQA-D performance versus publicly reported reasoning post-training data. The horizontal axis is logarithmic and bubble area encodes model parameters. Right-pointing arrows mark reported lower bounds rather than exact dataset sizes. Only systems with disclosed data amounts and GPQA-D scores in Table 1 are shown.

The data-budget view in Figure 7 highlights a larger separation. BoundaryEvo uses 1K cumulative training examples and scores 81.3. DASD-4B uses 448K reported examples and scores 68.4, while the three 7B comparison models report at least 102K to 6M examples and score from 46.2 to 61.1. Thus, within this reported set, BoundaryEvo achieves the best GPQA-D result with the smallest disclosed training set.

This comparison measures example efficiency, not total computational cost. Dataset definitions, trajectory lengths, and filtering policies differ across papers. Moreover, BoundaryEvo's 1K count includes repeated training exposure but excludes student and expert inference, analyzer judging, and teacher-rollout generation. The plots should therefore be read as a compact view of reported model and data scale, rather than as a controlled comparison of training FLOPs.

E BoundaryEvo Algorithm and Implementation Details

End-to-End Training Procedure. Algorithm 1 summarizes the two-stage pipeline, where Stage 1 establishes reasoning capability through judge-free distillation and Stage 2 evolves supervision by selecting capability-boundary questions for verified teacher trajectory distillation.

Algorithm 1: BoundaryEvo: Capability Priming and Capability-Boundary Co-Evolution

Input: Base student M_{base}; trajectory teacher T; fixed expert E; analyzer J; priming questions $\mathcal{Q}_0$; evolution pool $\mathcal{P}$; reference answers $\{y_i^{\text{ref}}\}$; thresholds $\boldsymbol{\tau}$; number of turns R; QA budget $K = 300$; teacher rollout count L

Output: Evolved student M_R

```
// Stage 1: learn how the teacher reasons
```

$\mathcal{D}_0 \leftarrow \emptyset$;

foreach $x_i \in \mathcal{Q}_0$ **do**

  Sample a holistic trajectory $y_i^{\text{T}} \sim T(\cdot \mid x_i)$;

  if y_i^{T} *passes basic response-validity checks* **then**

    $\mathcal{D}_0 \leftarrow \mathcal{D}_0 \cup \{(x_i, y_i^{\text{T}})\}$;

$M_0 \leftarrow \text{SFT}(M_{\text{base}}, \mathcal{D}_0)$;

$\mathcal{D}_{-1}^{\text{QA}} \leftarrow \emptyset$;

```
// Stage 2: learn what the student should learn next
```

for $t \leftarrow 0$ **to** $R-1$ **do**

  $\mathcal{A}_t \leftarrow \emptyset$;

  foreach $z_i = (x_i, h_i, y_i^{\text{ref}}) \in \mathcal{P}$ **do**

    Independently sample $\hat{y}_{i,t}^{\text{s}} \sim M_t(\cdot \mid x_i, h_i)$ and $\hat{y}_i^{\text{e}} \sim E(\cdot \mid x_i, h_i)$;

    $\mathbf{a}_i^{(t)} \leftarrow J(z_i, \hat{y}_{i,t}^{\text{s}}, \hat{y}_i^{\text{e}})$;

    Compute $g_i^{(t)} = s_i^{\text{e}} - s_i^{\text{s},(t)}$;

    if $s_i^{\text{e}} \geq \tau_{\text{e}}$, $s_i^{\text{s},(t)} < \tau_{\text{s}}$, $g_i^{(t)} \geq \tau_{\text{g}}$, $q_i \geq \tau_{\text{q}}$, $b_i^{\text{clear}} = b_i^{\text{answerable}} = 1$, *and* $c_i \geq \tau_{\text{c}}$ **then**

      $\mathcal{A}_t \leftarrow \mathcal{A}_t \cup \{z_i\}$;

  $\mathcal{R}_t \leftarrow \{z \in \mathcal{A}_t : x(z) \in \text{Questions}(\mathcal{D}_{t-1}^{\text{QA}})\}$;

  $\mathcal{N}_t \leftarrow \mathcal{A}_t \setminus \mathcal{R}_t$;

  Retain eligible replay examples up to the turn-specific replay quota;

  Rank novel examples by student-rollout completeness, analyzer recommendation, and $r_i^{(t)}$;

  Fill the remaining slots with ranked novel examples to form $\mathcal{D}_t^{\text{QA}}$, where $|\mathcal{D}_t^{\text{QA}}| = K$;

  ```
  // Construct verified supervision only after QA selection
  ```

  $\mathcal{D}_t^{\text{distill}} \leftarrow \emptyset$;

  foreach $z_i \in \mathcal{D}_t^{\text{QA}}$ **do**

    Sample $\{y_{i,t}^{\text{T},(k)}\}_{k=1}^{L}$ independently from $T(\cdot \mid x_i, h_i)$;

    $\mathcal{Y}_{i,t}^{+} \leftarrow \{y : V(\text{Ans}(y), y_i^{\text{ref}}) = 1\}$;

    if $\mathcal{Y}_{i,t}^{+} \neq \emptyset$ **then**

      Select $y_{i,t}^{\star} \in \mathcal{Y}_{i,t}^{+}$ and add $(x_i, h_i, y_{i,t}^{\star})$ to $\mathcal{D}_t^{\text{distill}}$;

  $M_{t+1} \leftarrow \text{SFT}(M_t, \mathcal{D}_t^{\text{distill}})$;

return M_R;

Replay and Training-Set Construction. At turn t, accepted questions are divided by question identity relative to the immediately preceding training set. Replay candidates $\mathcal{R}_t$ were used in the

preceding turn and remain eligible under the current screening protocol. Novel candidates $\mathcal{N}_t$ do not occur in that preceding set. The training budget remains fixed:

$$\mathcal{D}_t^{\mathrm{QA}} = \mathrm{Select}_{K_t^{\mathrm{R}}}(\mathcal{R}_t) \cup \mathrm{Select}_{K_t^{\mathrm{N}}}(\mathcal{N}_t), \qquad K_t^{\mathrm{R}} + K_t^{\mathrm{N}} = 300. \tag{11}$$

Eligible replay questions are retained up to the turn-specific replay quota.

F Prompt Templates

We report the raw prompts used by the training-data pipeline. Placeholders in braces are replaced at runtime. Stage 1 has no model-based judge prompt.

F.1 Stage 1 Teacher Trajectory Distillation Prompt

The Stage 1 teacher receives the following system and user messages. The resulting trajectory is used directly after basic response-validity checks.

Stage 1 Teacher Trajectory Distillation Prompt

```
[System]
You are a helpful and harmless assistant.

You are not allowed to use any tools.

[User]
Please reason step by step, and put your final answer within \boxed{}.


{question}
```

F.2 Stage 2 Student and Expert Rollout Prompt

The current student and fixed expert receive the same prompt and answer the question independently.

Stage 2 Student and Expert Rollout Prompt

```
Please solve the following problem independently.
Show concise reasoning, then end with exactly one final-answer line:
Final answer: <your answer>

{optional_context}

Problem:
{question}
```

F.3 Capability-Boundary Analyzer Prompt

The analyzer receives the question, optional context, reference answer, and both independently generated responses. In the raw prompt, *weak* denotes the current student and *strong* denotes the fixed expert. The implementation also records reference-correctness and leakage judgments as auxiliary diagnostics. Their hard rejection switches are disabled in the reported route.

Capability-Boundary Analyzer Prompt

```
[System]
```

```
You must output exactly one valid JSON object. Do not include
    markdown fences, comments, natural-language prefaces, or trailing
     text.

[User]
You are a strict data-quality and model-trajectory judge.

Evaluate whether this example is suitable training data. Consider:
- whether the question is clear and answerable;
- whether the reference answer is correct;
- if context is provided, whether the answer is supported by the
    context;
- how good the weak-model answer is;
- how good the strong-model answer is;
- whether the strong answer is meaningfully better than the weak
    answer.

Scores must be floats in [0, 1].
- weak_score: quality/correctness of the weak answer.
- strong_score: quality/correctness of the strong answer.
- data_quality_score: clarity, answerability, reference correctness,
  and training usefulness.
- context_support_score: use 1.0 when no external context is required
  or no context is provided.

Thresholds used by the pipeline:
{
  "min_strong_score": 0.65,
  "max_weak_score": 0.50,
  "min_gap": 0.20,
  "min_quality_score": 0.70,
  "min_context_support_score": 0.70
}

Question:
{question}

Context:
{context_or_no_context_provided}

Reference answer:
{reference_answer}

Weak model answer:
{student_answer}

Strong model answer:
{expert_answer}

Return exactly one JSON object with this schema:
{
  "is_clear": true,
  "is_answerable": true,
  "is_reference_correct": true,
  "is_context_supported": true,
  "weak_score": 0.25,
  "strong_score": 0.85,
  "data_quality_score": 0.82,
  "context_support_score": 0.9,
  "accepted_recommendation": true,
  "rejection_reasons": [],
  "rationale": "Strong model solves the example correctly, while weak
     model misses the key reasoning step."
```

```
}
```

F.4 Final-Answer Verification Prompt

Each sampled teacher trajectory is judged by final-answer equivalence. Equivalent algebraic forms and reasonable numerical precision are accepted.

Final-Answer Verification Prompt

```
[System]
You must output exactly one valid JSON object. Do not include
    markdown fences, comments, natural-language prefaces, or trailing
     text.

[User]
[JUDGE]
You are a strict math/reasoning answer judge. Decide whether the
    candidate answer is mathematically equivalent to the reference
    answer. Allow equivalent forms, algebraic rearrangements, and
    reasonable numeric precision.

Problem:
{question}

Reference answer:
{reference_answer}

Candidate answer:
{candidate_teacher_answer}

Return exactly one JSON object:
{
  "correct": true,
  "reason": "one concise sentence"
}
```

G Description of Experimental Evaluation

G.1 Common Evaluation Protocol

Provenance of reported scores. The values in Tables 1, 2, and 3 follow a documented source hierarchy. For model–benchmark pairs reported in the VibeThinker study, official model cards, or official benchmark leaderboards, we use the corresponding publicly reported scores. When a model or benchmark combination is not available from these sources, we perform a local evaluation, primarily for smaller open-weight models, using the EvalScope framework and the official evaluation split. These local runs retain EvalScope's default top-p and temperature and a maximum of 102,400 generated tokens setting. The same provenance rule is applied across all three tables. **Protocol for controlled follow-up experiments.** All follow-up methods use the same official evaluation splits, prompts, answer extractors, decoding settings, evaluator versions, and generation seed schedule. Competition-math benchmarks use at least four independent completions per question, while all other benchmarks use four. We report the mean correctness of individual completions rather than majority vote, best-of-n, or pass@n. Unparseable and truncated responses without an extractable final answer are counted as incorrect rather than discarded. Math and General are the same unweighted benchmark averages used in the main paper, and the final checkpoint after three turns is fixed as the primary endpoint. All methods within a seed share the priming checkpoint, candidate pool, cached expert outputs, and available teacher samples. Before training, exact and near-duplicate questions between the priming/evolution data and every evaluation set are identified and manually checked;

confirmed overlaps are removed from the common training pool. We will release question-ID manifests, exclusion counts, and the complete evaluation configuration.

G.2 Relationship to LLM2LLM and AgentFrontier

LLM2LLM (Lee et al., 2024) first identifies student errors and then asks a teacher to synthesize new examples targeting those errors. Our current fixed-pool baseline instead rescans the same candidate pool and selects questions answered incorrectly by the current student. It isolates error-driven acquisition under the same pool and training budget, but it is an LLM2LLM-inspired adaptation rather than a complete reproduction. We additionally implement *LLM2LLM-style synthesis*, which retains the original error-to-new-question step, independently verifies and de-duplicates the generated QA pairs, and uses the same 300-example budget per turn. The source revision, prompts, rejected generations, and all adaptations will be reported. AgentFrontier (Chen et al., 2025) also targets tasks that a weaker model cannot solve but a stronger, tool-augmented system can. BoundaryEvo focuses on a more specific setting: tool-free long-trajectory distillation for compact students, a fixed QA pool and expert, and repeated boundary re-estimation after each student update. Our claimed contribution is therefore the two-stage priming-and-evolution procedure and its behavior under a small supervision budget, rather than the general idea of ZPD-style filtering.