

Capacity-Dependent Effects of Data Selection for Reasoning

When should LLMs learn from easy versus hard supervision?

Cuong Dang • Hoang Anh Just • Ruoxi Jia
Virginia Tech

Conference on Language Modeling (COLM) 2026
arxiv.org/abs/2608.13721

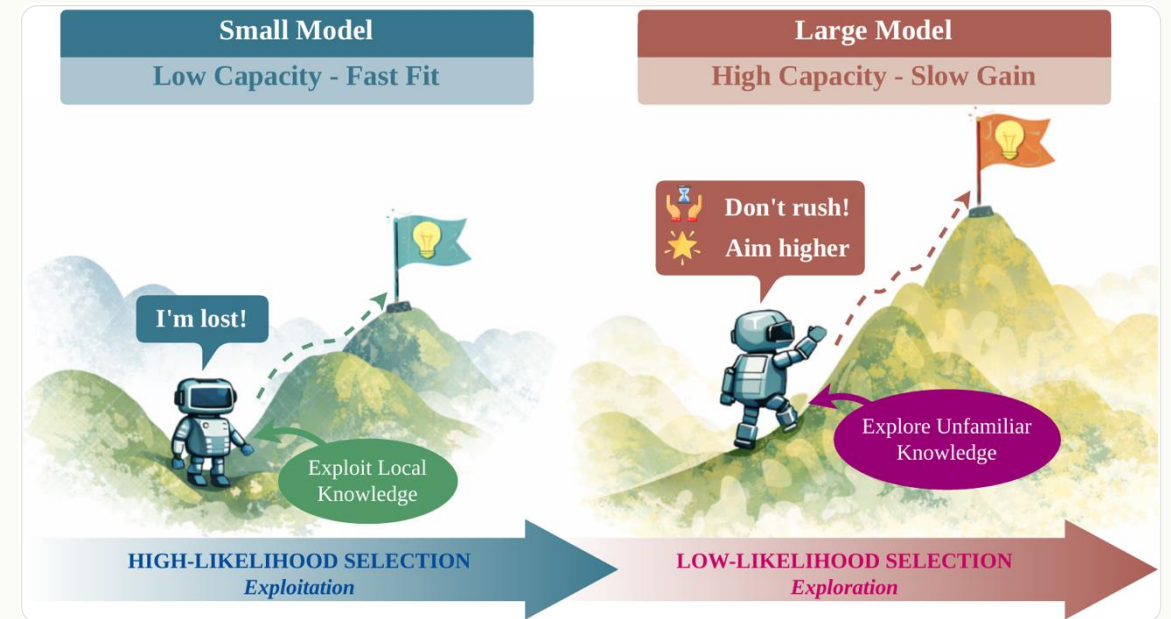


Figure 1: Capacity-dependent learning under likelihood-based data selection.

Math-specific pretraining laid the foundation for reasoning

Jun 2022 · Google

Minerva

 38.5B tokens

Continued pretraining of PaLM on arXiv papers and math web pages; no tools.

540B: MATH **50.3%** (maj@64) · 33.6% single sample

Oct 2023 · EleutherAI

Llemma

 55B tokens

Code Llama continued on Proof-Pile-2: arXiv 29B · web incl. OpenWebMath 15B · math code 11B.

Open data and weights (7B, 34B); Llemma 7B beats Minerva 8B on MATH

Feb 2024 · DeepSeek

DeepSeekMath

 120B

120B math tokens mined from Common Crawl, mixed with natural language and code; 7B model.

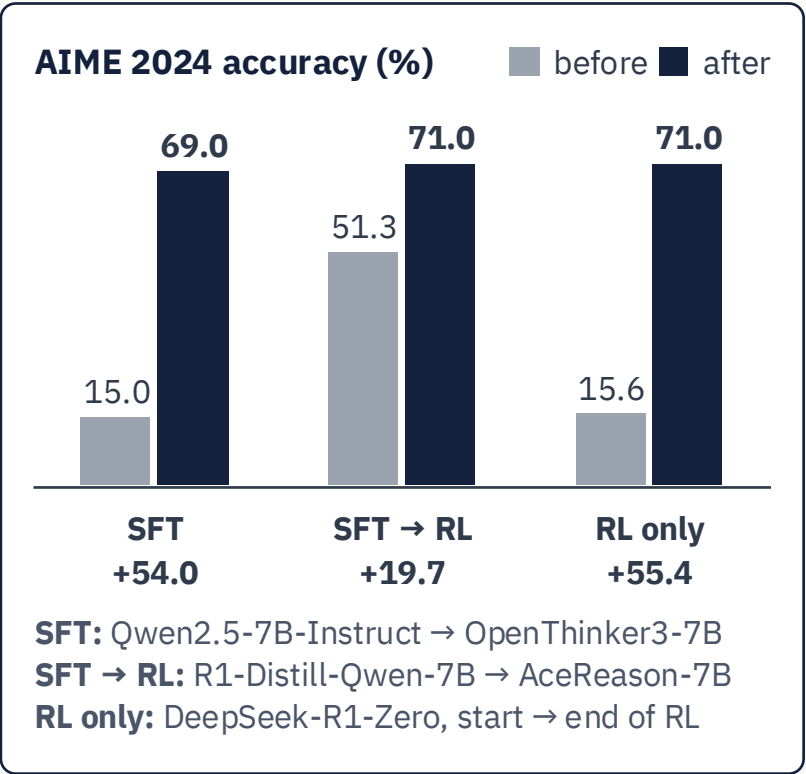
7B: MATH **51.7%** (no tools, no voting) · 60.9% self-consistency@64

Trend: larger, more targeted, increasingly open math corpora. A 7B model trained on curated math data matches a 540B model from two years earlier.

Sources: Lewkowycz et al. (2022). Solving quantitative reasoning problems with language models. NeurIPS. · Azerbayev et al. (2024). Llemma: An open language model for mathematics. ICLR. · Shao et al. (2024). DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. arXiv:2402.03300.

Post-training became a powerful, data-efficient route to math reasoning

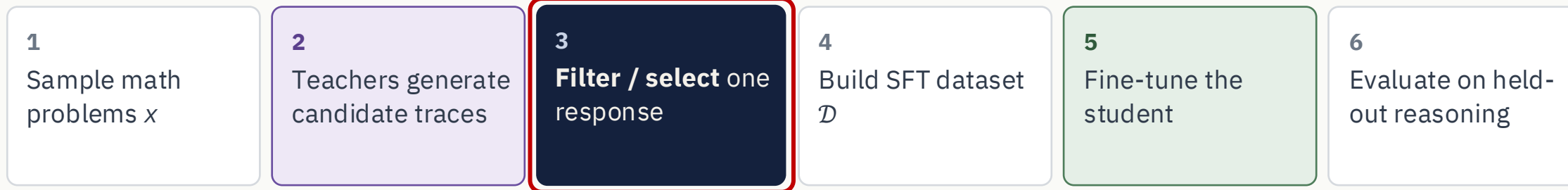
MetaMath Sep 2023 · SFT	395K bootstrapped GSM8K/MATH questions → 7B: GSM8K 66.5% , +11.5 over same-size SOTA
OpenAI o1 Sep 2024 · RL	Large-scale RL over long chains of thought → AIME 2024: 74% single sample vs GPT-4o 12%
DeepSeek-R1 Jan 2025 · RL + distill	R1-Zero: pure RL on DeepSeek-V3-Base, no SFT; reasoning then distilled into small dense students
OpenThoughts Jun 2025 · open SFT, small models	1.2M QwQ-32B traces → OpenThinker3-7B: AIME25 53.3% , +15.3 over R1-Distill-Qwen-7B. A stronger model is not always a better teacher.



Sources: Yu et al. (2024). MetaMath: Bootstrap your own mathematical questions for large language models. ICLR. · OpenAI (2024). Learning to reason with LLMs. · Guo et al. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv:2501.12948. · Guha et al. (2026). OpenThoughts: Data recipes for reasoning models. ICLR. · Liu et al. (2026). AceReason-Nemotron 1.1: Advancing math and code reasoning through SFT and RL synergy. ICLR.

1 · INTRODUCTION & BACKGROUND

Strong teachers generate reasoning Smaller students learn to imitate



Supervised fine-tuning objective

$$\min_{\theta} - \sum_{(x, y) \in \mathcal{D}} \log \pi_{\theta}(y | x)$$

x — math question · y — teacher reasoning response

π — student policy · $\mathcal{D}$ — the selected training data

The objective is fixed. Step 3 alone decides what the student should imitate.

1 · INTRODUCTION & BACKGROUND

The final answer is not the only criterion — the distribution matters too

Question:

Solve $x^2 - 5x + 6 = 0$

Answer A · concise, familiar

$(x - 2)(x - 3) = 0$, so $x = 2$ or 3

Answer B · longer, less familiar

Move the constant: $x^2 - 5x = -6$. Complete the square by adding $(5/2)^2 = 25/4$ to both sides: $(x - 5/2)^2 = 1/4$. Take square roots: $x - 5/2 = \pm 1/2$, so $x = 3$ or $x = 2$.

Both correct. Which is better?

2022 · generation

SLiC

Sequence **likelihood affects task performance** in text generation.

Zhao et al.

2024 · preferences

On-policy data

Training on data **likely under the model** improves fine-tuning results.

Tajwar et al.

2025 · SFT

GRAPE

Pick the response the **target model finds most likely**: up to +13.8% avg over distilling from the strongest model.

Zhang, Dai & Peng

2025 · math CoT

Shape of Thought

Model CoTs, **even with wrong answers**, beat human-written ones; paraphrasing human traces toward the model helps.

Chandra et al.

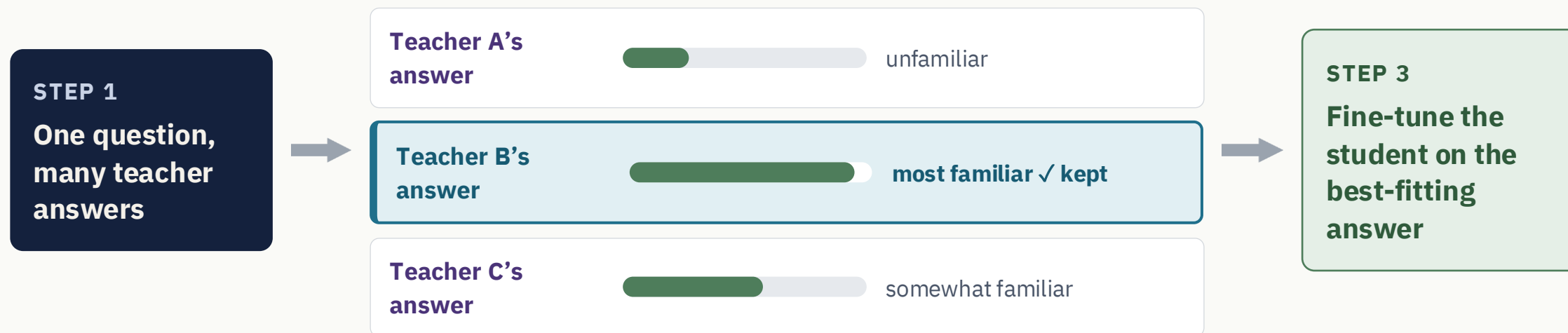
It depends on the student: good supervision is correct **and** learnable. **Our question:** is closer always better?

Sources: Zhao et al. (2022). Calibrating sequence likelihood improves conditional language generation. arXiv:2210.00045. · Tajwar et al. (2024). Preference fine-tuning of LLMs should leverage suboptimal, on-policy data. ICML. · Zhang, Dai & Peng (2025). The best instruction-tuning data are those that fit. NeurIPS. · Chandra et al. (2025). Shape of Thought: When distribution matters more than correctness in reasoning tasks. arXiv:2512.22255.

1 · INTRODUCTION & BACKGROUND

GRAPE selects the responses that best fit the student

THE STUDENT ASKS: “HOW NATURAL IS THIS FOR ME?”



Why: teachers differ in style and complexity; answers close to the student should be easier to absorb.

Cost: one scoring pass by the student per candidate; no extra training signal needed.

Prior research offers two opposing selection principles

PERSPECTIVE A

High-likelihood data is better

Motivated by GRAPE, on-policy distillation, student-distribution alignment, learnability and stable optimization.

Hypothesis: supervision closer to the student policy produces more effective learning.

Learnability
vs
Novelty

PERSPECTIVE B

Hard examples are better

Motivated by hard-example mining, active learning, curriculum research, informative or surprising examples.

Hypothesis: examples far from the student policy can teach capabilities it does not yet have.

Caveat. Low likelihood $\neq$ useful mathematical difficulty. It can also reflect response style, length or other mismatch.

Sources: Zhang, Dai & Peng (2025). The best instruction-tuning data are those that fit. NeurIPS. · Agarwal et al. (2024). On-policy distillation of language models: Learning from self-generated mistakes. ICLR. · Lewis (1995). A sequential algorithm for training text classifiers. SIGIR Forum. · Shrivastava, Gupta & Girshick (2016). Training region-based object detectors with online hard example mining. CVPR. · Robinson et al. (2021). Contrastive learning with hard negative samples. ICLR. · Paul, Ganguli & Dziugaite (2021). Deep learning on a data diet: Finding important examples early in training. NeurIPS. · Lin et al. (2024). Not all tokens are what you need for pretraining. NeurIPS.

2 · RESEARCH MOTIVATION

The best training data may depend on the learning context



PRIOR AXES

Learning stage: curriculum & self-paced learning (Bengio 2009; Kumar 2010; Weinshall 2018).

Data size: keep easy examples when data is scarce, hard when abundant (Sorscher 2022).

THIS WORK

Adds **student capacity** × **training duration** for student-likelihood response selection.

Sources: Bengio et al. (2009). Curriculum learning. ICML. · Kumar, Packer & Koller (2010). Self-paced learning for latent variable models. NeurIPS. · Weinshall, Cohen & Amir (2018). Curriculum learning by transfer learning: Theory and experiments with deep networks. ICML. · Sorscher et al. (2022). Beyond neural scaling laws: Beating power law scaling via data pruning. NeurIPS.

Is there a universal winner between high and low likelihood?

RQ1

Does high-likelihood supervision always win?

Compare five student sizes over short and long training.

RQ2

Why do students respond differently?

Track generated outputs under the teacher and the original student.

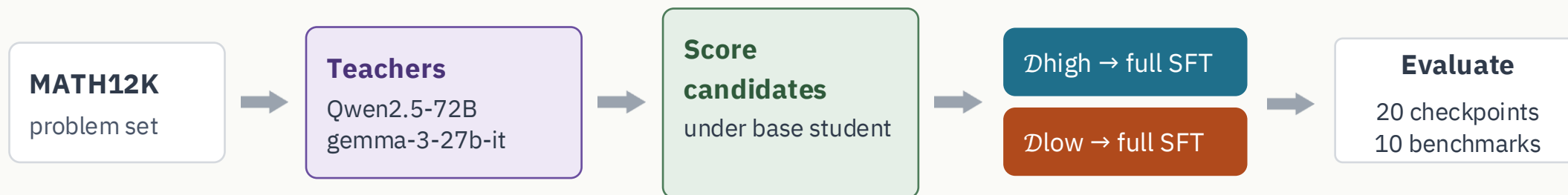
RQ3

What mechanism governs transfer?

Data span × knowledge gap × feasible student capacity.

3 · EXPERIMENTAL SETUP

Controlled response selection isolates the effect of student likelihood



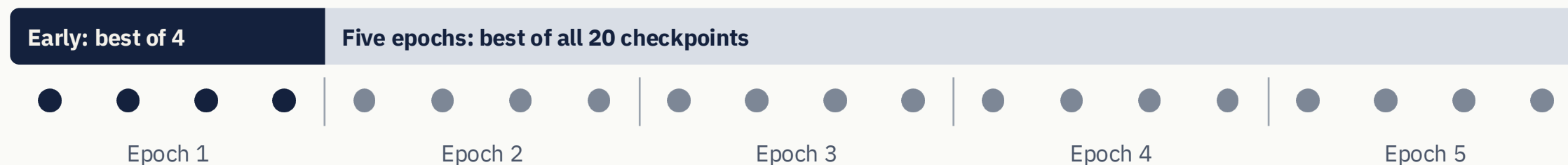
STUDENTS

Size	Model
1.5B	Qwen/Qwen2.5-1.5B
3B	Qwen/Qwen2.5-3B-Instruct
4B	Qwen/Qwen3-4B
7B	Qwen/Qwen2.5-Math-7B
8B	Qwen/Qwen3-8B

Sources: Hendrycks et al. (2021). Measuring mathematical problem solving with the MATH dataset. NeurIPS. · Qwen (2025). Qwen2.5 technical report. · Yang et al. (2024a). Qwen2.5-Math technical report: Toward mathematical expert model via self-improvement. · Yang et al. (2025). Qwen3 technical report. · Gemma Team (2025). Gemma 3 technical report.

3 · EXPERIMENTAL SETUP

Evaluating both early learning and long-run performance



PROTOCOL

- Pass@1, temperature 0.6
- Scores are **best-checkpoint envelopes** per benchmark, not final checkpoints

TEN BENCHMARKS

AIME24 AMC CHMATH Gaokao GPQA GradeSchool
KAOYAN MATH500 Minerva OlympiadBench

Sources: AIMO (2024). AIMO validation sets: AIME 2024 and AMC. · Wei et al. (2023). CMATH: Can your language model pass Chinese elementary school math test? · Zhang et al. (2023). Evaluating the performance of large language models on GAOKAO benchmark. · Rein et al. (2024). GPQA: A graduate-level Google-proof Q&A benchmark. COLM. · Hendrycks et al. (2021). Measuring mathematical problem solving with the MATH dataset. NeurIPS. · Lewkowycz et al. (2022). Solving quantitative reasoning problems with language models. NeurIPS. · He et al. (2024). OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. ACL. · Ye et al. (2025). LIMO: Less is more for reasoning. COLM.

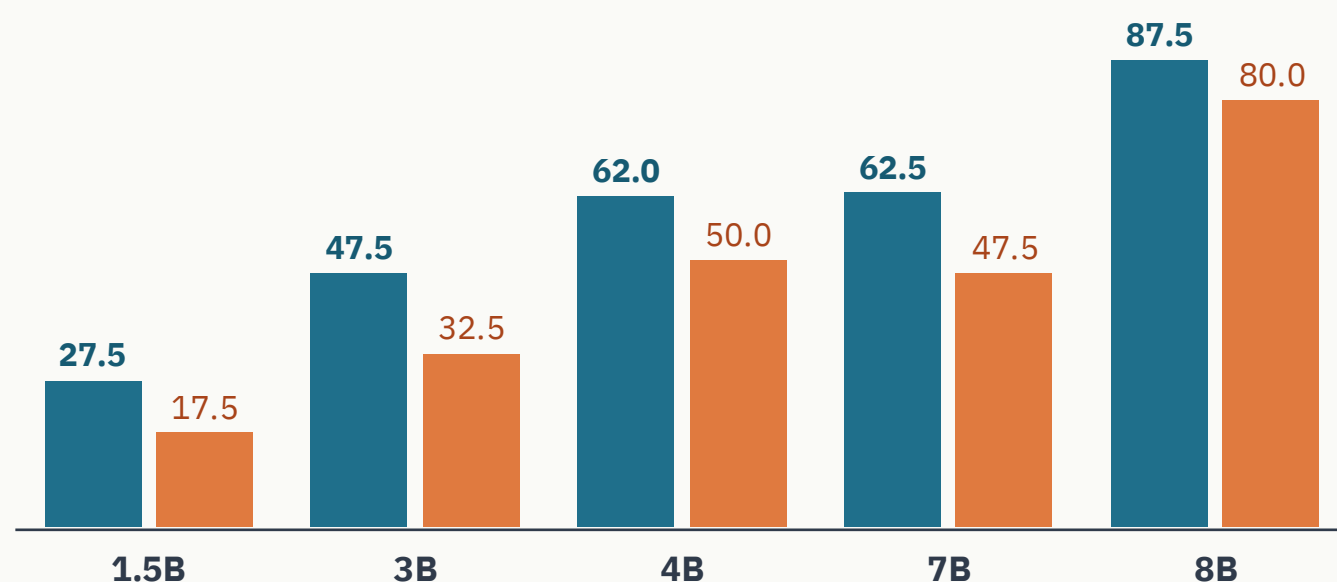
4 · KEY EMPIRICAL RESULTS

High-likelihood data consistently wins during early training

AMC pass@1 (%), best of 4 first-epoch checkpoints

■ High ■ Low

ALL 10 BENCHMARKS, EPOCH 1

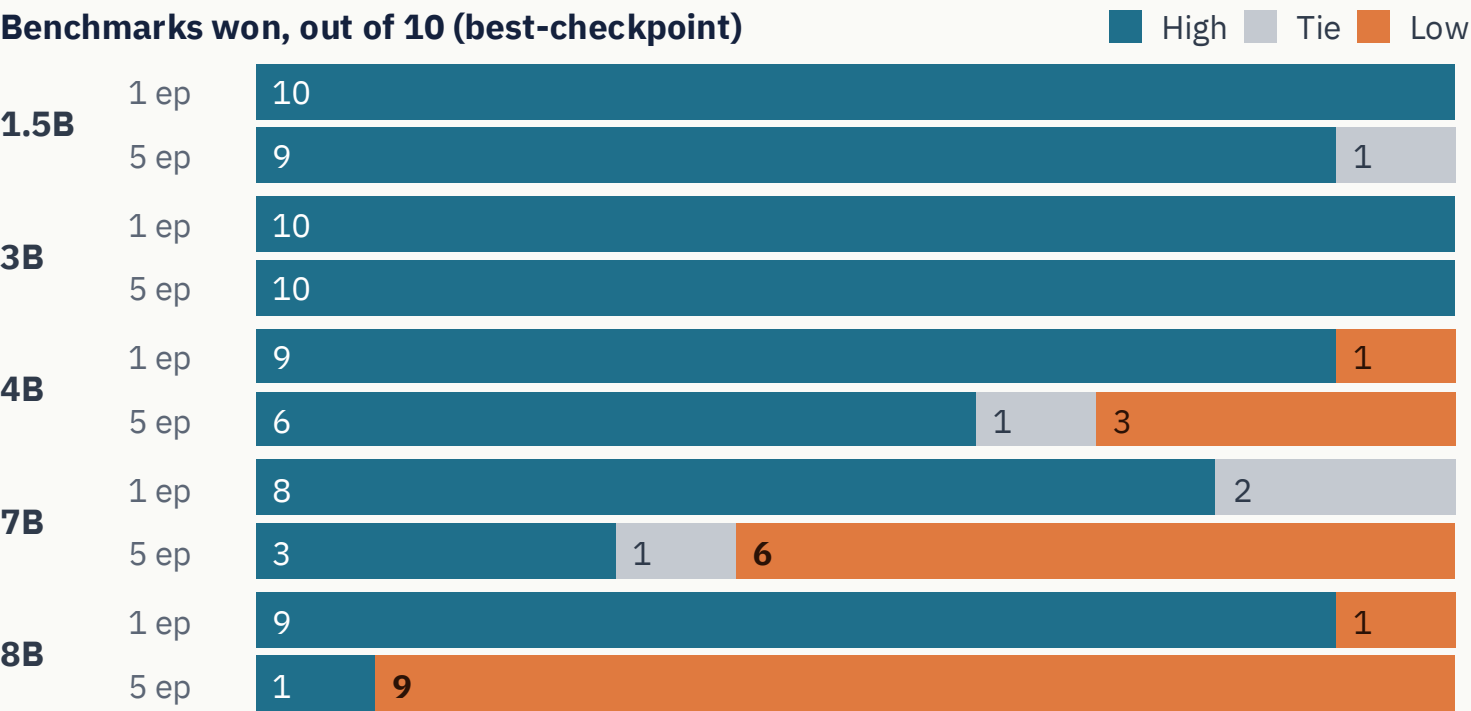


Student	High wins
1.5B	10 / 10
3B	10 / 10
4B	9 / 10
7B	8 / 10 (2 ties)
8B	9 / 10

Observation 1. High-likelihood data yields faster early gains, especially for smaller students.

4 · KEY EMPIRICAL RESULTS

After five epochs, low likelihood wins for the larger students



EXAMPLES, TABLE 2

Case	High	Low
1.5B AMC	35.0	22.5
3B AMC	55.0	47.5
7B GPQA	39.39	44.95
7B MATH	79.00	80.06
8B AMC	90.0	92.5
8B MATH	92.0	93.0

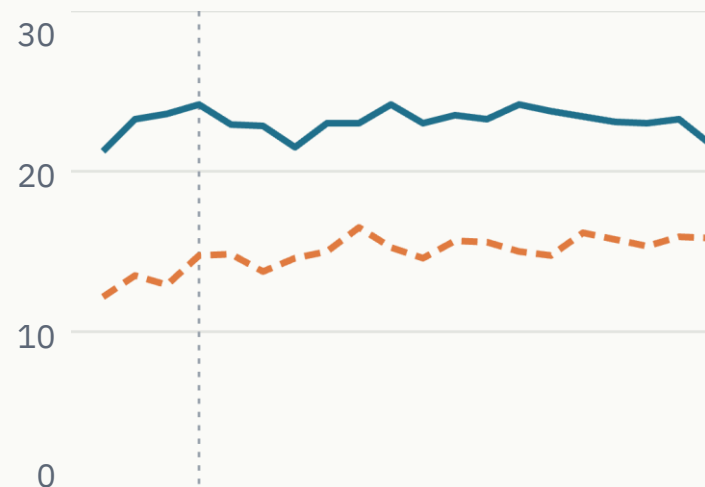
Not uniform: 8B still prefers High on Gaokao (74.68 vs 67.09).

Observation 2. For high-capacity models, high likelihood peaks early, while low likelihood improves later and can surpass it.

4 · KEY EMPIRICAL RESULTS

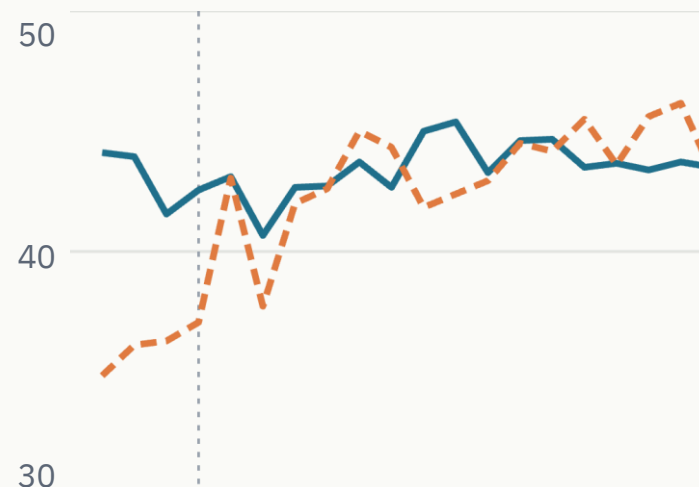
High likelihood fits quickly; low likelihood can gain more eventually

1.5B · Fast-Fit, no catch-up



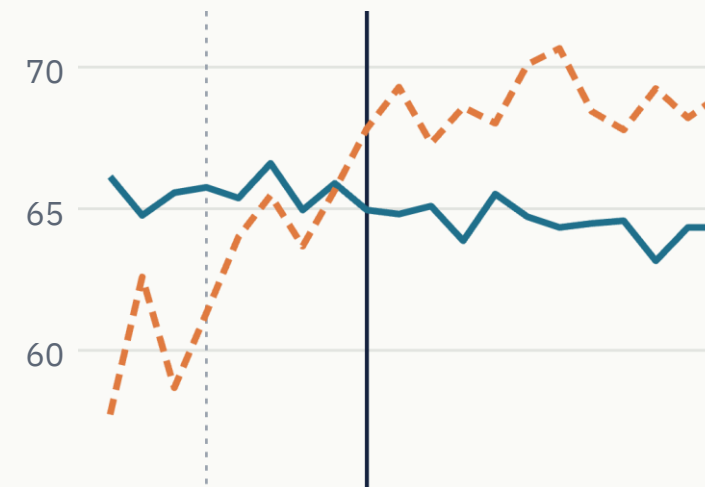
High 24.16 by epoch 1; Low's best is only 16.50.

7B · Unstable Slow-Gain



Low peaks at 46.19 > High's 45.41, but leads at only 5 of 20 checkpoints.

8B · Sustained Slow-Gain



Early → best: High +0.48, Low +8.07 (to 70.69).

— High (solid) — Low (dashed) · Simple average of 10 benchmarks, pass@1, every checkpoint

4 · KEY EMPIRICAL RESULTS

Small models can fail to absorb unfamiliar reasoning

Question: There is a sequence $a_1, a_2, \dots, a_{2023}$ such that $\tan(2023x) = \dots$. What is a_{2023} ?		
Qwen2.5-1.5B	Qwen2.5-3B-Instruct	Qwen2.5-Math-7B
There is a sequence $a_1, a_2, \dots, a_{2023}$ such that $\tan(2023x) = \dots$. What is a_{2023} ?	Let $P(x) = \tan 2023x$. Then $\tan(2023x) = \frac{\tan(2022x) + \tan x}{1 - \tan(2022x) \tan x}$	Let $t = \tan x$. Then we have... We know that $\tan(A + B) = \frac{\tan A + \tan B}{1 - \tan A \tan B}$
2023 2023 2023 2023 ...	$\tan(2023x) = \frac{\tan(2022x) + \tan x}{1 - \tan(2022x) \tan x}$...	... Final answer: 1 .

1.5B: repeats the question, then loops on an incorrect answer — shallow imitation.

3B: begins valid steps, then gets lost repeating them.

7B: completes the reasoning and reaches the correct answer.

Observation 3. Small model: repetition, shallow copying, incomplete reasoning. Larger model: delayed but meaningful reasoning improvement.

4 · KEY EMPIRICAL RESULTS

Selection effectiveness depends on capacity × training time

	Short training	Long training	Regime
Small 1.5B · 3B · 4B	High strongly favored	High remains favored	Fast-Fit, no catch-up
Intermediate 7B	High favored	Low reaches higher peaks; lead is checkpoint-sensitive	Fast-Fit, unstable Slow-Gain
Large Qwen3-8B · Llama-3.1-8B	High often favored	Low can outperform, persistently	Fast-Fit, sustained Slow-Gain

Patterns in the evaluated models — not a universal scaling threshold. Model families and specializations also differ across sizes.

Takeaway: Choose supervision difficulty to match the student’s capacity and compute budget.

Learning dynamics reveal what the student actually learns

Accuracy alone can't tell **meaningful movement toward the teacher** from **mere drift away from the starting point**.

Fine-tuned student π_t

generates answers y at each checkpoint t



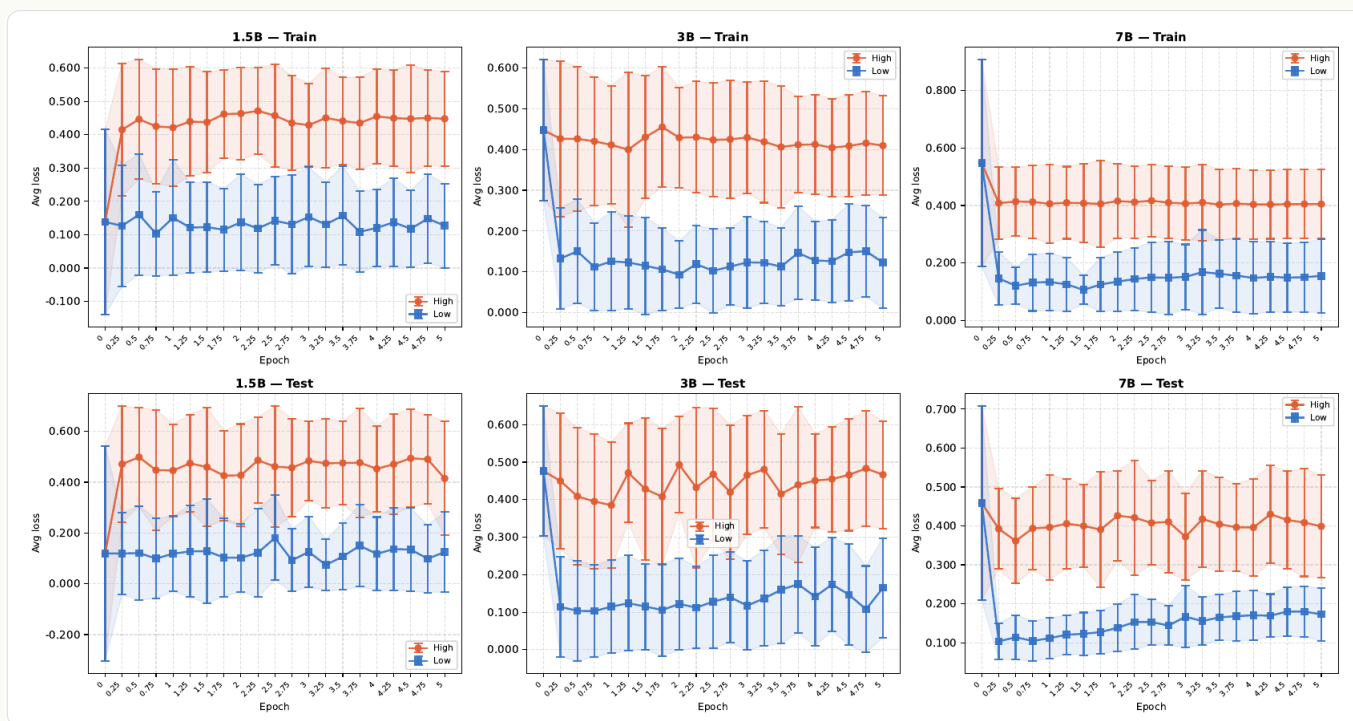
Teacher view: is the student moving *toward* the teacher?

$$\mathcal{L}_T(t) = - \mathbb{E}_x \mathbb{E}_{y \sim \pi_{\theta_t}(\cdot | x)} [\log \pi_T(y | x)]$$

Base-student view (π_0): how far has it moved *away* from where it started?

$$\mathcal{L}_0(t) = - \mathbb{E}_x \mathbb{E}_{y \sim \pi_{\theta_t}(\cdot | x)} [\log \pi_{\theta_0}(y | x)]$$

Low-likelihood data pulls capable students toward the teacher



1.5B: High training raises teacher loss (drifts away); Low leaves it roughly unchanged.

3B: High stays flat; Low **decreases** teacher loss.

7B: both decrease; Low reaches the lower loss.

Observation 4. Low-likelihood data moves larger models closer to the teacher but fails to do so for small models.

Figure: Loss under teacher model. Lower = closer to teacher.

Low-likelihood training pushes students beyond their initial distribution

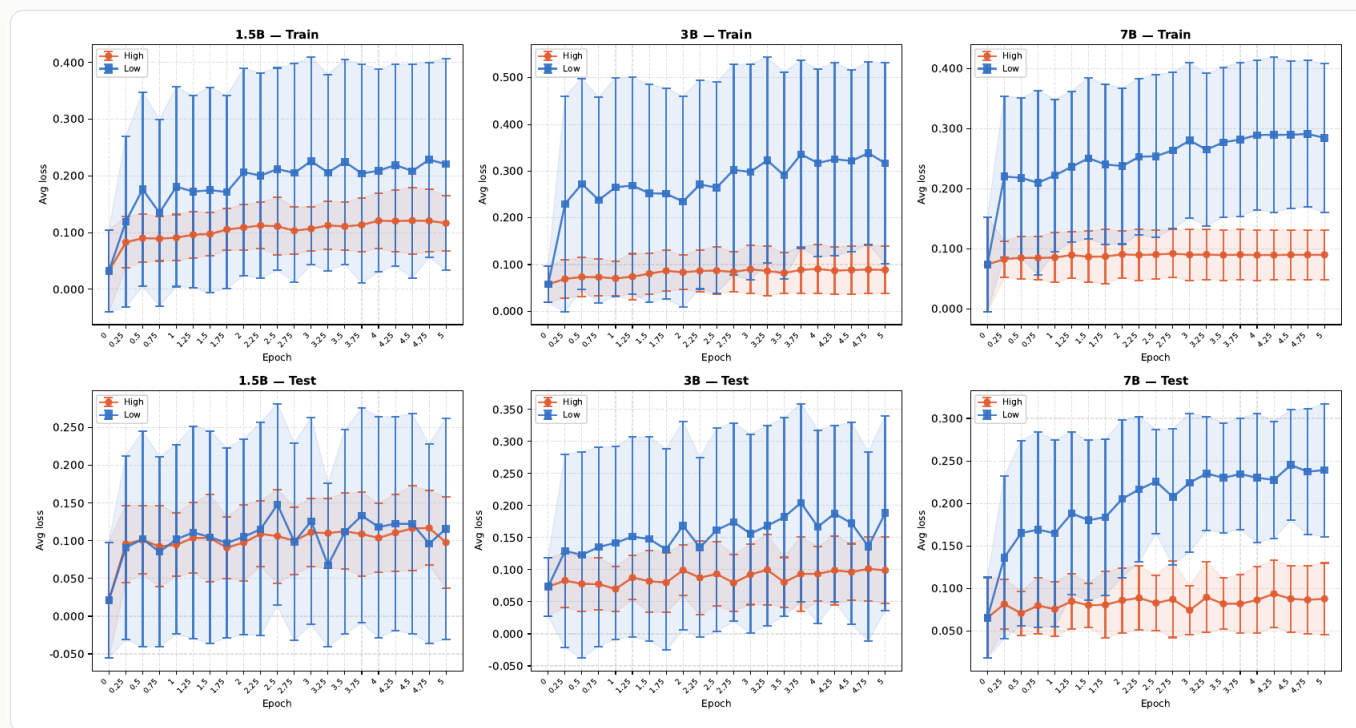


Figure: Loss under original student model. Higher = farther from the base student.

1.5B: Low gives a broader, more divergent pattern that still overlaps the initial range.

3B: clearer shift; High–Low separation grows.

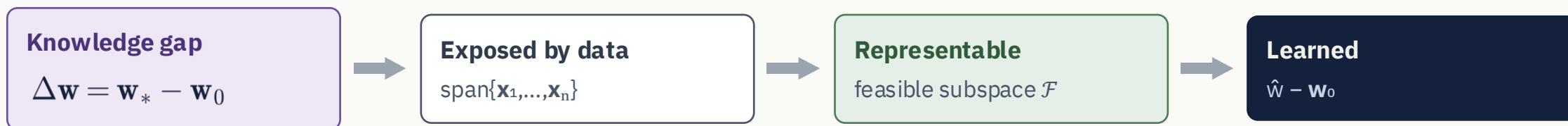
7B: smaller variance than 3B — a more stable, focused shift.

Shift alone is not proof of better reasoning.

Observation 5. As size grows, behavior is more stable, generalization improves, and the High/Low distinction becomes more pronounced.

6 • THEORETICAL EXPLANATION

Teacher knowledge is filtered by data coverage and student capacity



SETTING

Linear teacher and student classifiers:

$$h^*(\mathbf{x}) = \mathbf{1}\{\mathbf{w}_*^\top \mathbf{x} \geq 0\}, \quad h(\mathbf{x}) = \mathbf{1}\{\mathbf{w}^\top \mathbf{x} \geq 0\}$$

Soft teacher labels $y_i = \sigma(\mathbf{w}_*^\top \mathbf{x}_i)$

Gradient flow on the cross-entropy loss:

$$L(\mathbf{w}) = -\frac{1}{n} \sum_{i=1}^n [y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1 - y_i) \log (1 - \sigma(\mathbf{w}^\top \mathbf{x}_i))]$$

Scope

A simplified binary-classification model of distillation — not a nonlinear LLM. Low capacity = student confined to a subspace $\mathcal{F} \subseteq \mathbb{R}^d$.

6 · THEORETICAL EXPLANATION

Training data reveal only part of the teacher's knowledge

High-capacity student. Let P_X project onto $\text{span}\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. As $\tau \rightarrow \infty$, $\mathbf{w}(\tau) \rightarrow \hat{\mathbf{w}}$ with

$$\hat{\mathbf{w}} = \mathbf{w}_0 + P_X(\mathbf{w}_* - \mathbf{w}_0)$$

- The learned change is the **projection of the knowledge gap** onto the data span
- Directions orthogonal to the data are **never updated**; the student stays anchored to $\mathbf{w}_0$ there

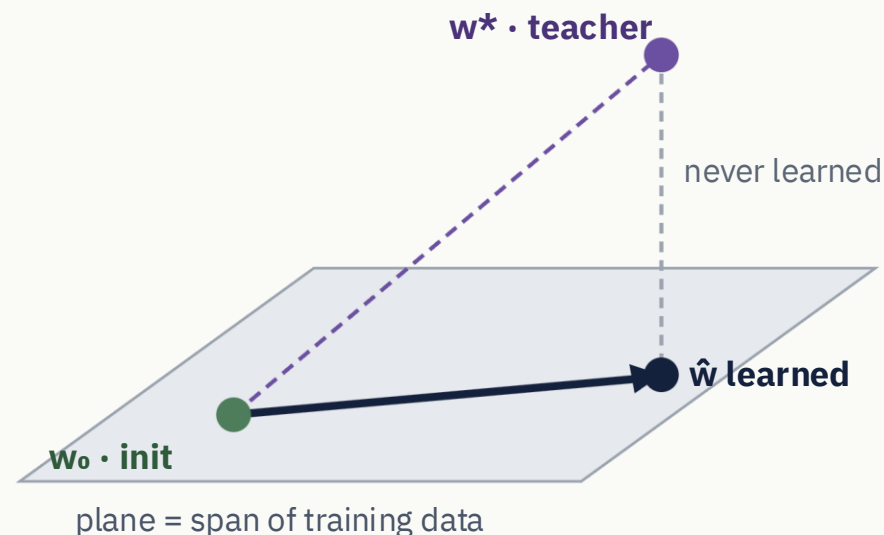


Figure: **Distillation as projection.** The student moves from $\mathbf{w}_0$ only within the data span, reaching $\hat{\mathbf{w}}$; the teacher's off-plane component is never learned.

6 • THEORETICAL EXPLANATION

Low-capacity students learn only representable teacher knowledge

Let $\mathbf{X}\mathcal{F}$ be the data span projected onto $\mathcal{F}$, and $P_{\mathbf{X}\mathcal{F}}$ its projector.

As $\tau \rightarrow \infty$:

$$\hat{\mathbf{w}} = \mathbf{w}_0 + P_{\mathbf{X}\mathcal{F}}(\mathbf{w}_* - \mathbf{w}_0)$$

Takeaway 2. Low-capacity distillation needs data that expose teacher knowledge *and* place it within the student's representable subspace.

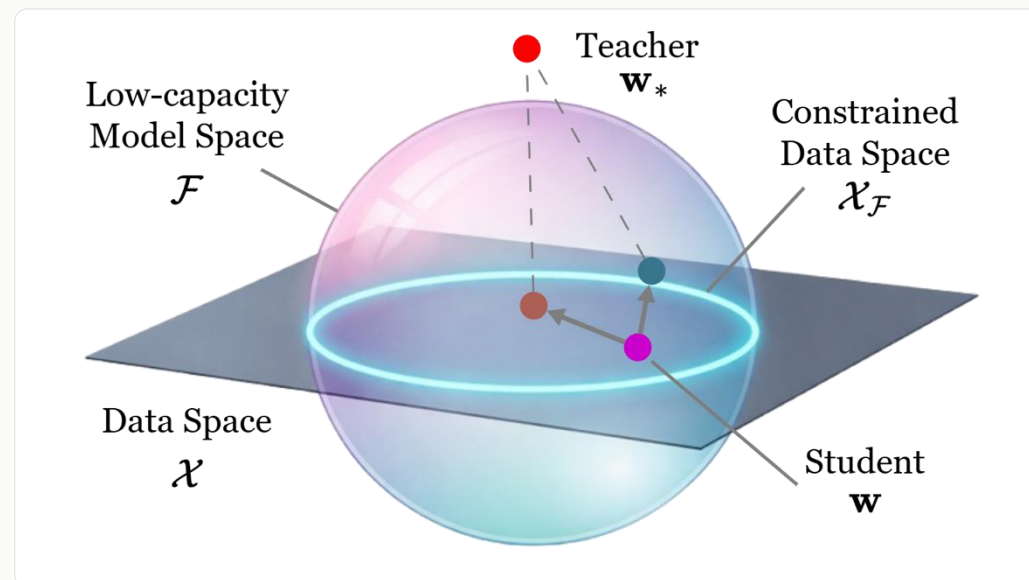


Figure: High-capacity students move toward the **brown** point (optimal solution); low-capacity students only toward the **teal** point (representable part – suboptimal solution). The remainder is unrepresentable.

Why low-likelihood supervision can fail for small models

Low-capacity student

1. Low-likelihood data expose unfamiliar teacher behavior
2. Much of the gap lies outside the feasible space $\mathcal{F}$
3. The supervision cannot be fully assimilated
4. Training yields shallow copying or repetition
5. More difficulty $\neq$ better reasoning

Higher-capacity student

1. Low-likelihood data expose unfamiliar teacher behavior
2. More of the gap is representable
3. Continued optimization lets it adapt
4. It approaches the teacher distribution
5. Accuracy can surpass the high-likelihood baseline

The trajectories **motivate** this reading; the theorems establish a mechanism **in the simplified linear model** — not an exact account of transformer training, nor a proof of Fast-Fit / Slow-Gain dynamics.

CONCLUSION

There is no universally optimal likelihood-based selection rule

1

Familiarity matters

High-likelihood supervision gives faster, more stable early gains — across every student size.

2

Capacity matters

Low-likelihood data pays off only when the student can represent and absorb unfamiliar teacher knowledge.

3

Duration matters

Short budgets favor easy-to-learn data; larger models trained longer can gain from challenging data.

The best reasoning data is not always the easiest or the hardest — it is the data the student can *productively learn from*.

Effective data selection is a function of student capacity, learning dynamics and training budget — not sample difficulty alone.

Thank you for your attention!

REFERENCES (1 / 2)

References

Dang, Just & Jia (2026). Capacity-Dependent Effects of Data Selection for Reasoning. *COLM*. [arXiv:2608.13721](https://arxiv.org/abs/2608.13721)

Agarwal et al. (2024). On-policy distillation of language models. *ICLR*. openreview.net/forum?id=3zKtaqxLhW

Azerbayev et al. (2024). Llemma: An open language model for mathematics. *ICLR*. [arXiv:2310.10631](https://arxiv.org/abs/2310.10631)

Bengio et al. (2009). Curriculum learning. *ICML*.

Chandra et al. (2025). Shape of Thought: When distribution matters more than correctness in reasoning tasks. [arXiv:2512.22255](https://arxiv.org/abs/2512.22255)

Gao et al. (2025). Principled data selection for alignment. [arXiv:2502.09650](https://arxiv.org/abs/2502.09650)

Guha et al. (2026). OpenThoughts: Data recipes for reasoning models. *ICLR*. openreview.net/forum?id=7xjoTuaNmN

Guo et al. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via RL. [arXiv:2501.12948](https://arxiv.org/abs/2501.12948)

Kumar, Packer & Koller (2010). Self-paced learning for latent variable models. *NeurIPS*.

Lewis (1995). A sequential algorithm for training text classifiers. *SIGIR Forum*.

Lewkowycz et al. (2022). Solving quantitative reasoning problems with language models (Minerva). *NeurIPS*. [arXiv:2206.14858](https://arxiv.org/abs/2206.14858)

Li et al. (2025). Small models struggle to learn from strong reasoners. *Findings of ACL*.

Lin et al. (2024). Not all tokens are what you need for pretraining. *NeurIPS*.

Liu et al. (2026). AceReason-Nemotron 1.1: Math and code reasoning through SFT and RL synergy. *ICLR*. openreview.net/forum?id=IaEqjWXd1d

References, continued

OpenAI (2024). Learning to reason with LLMs. openai.com/index/learning-to-reason-with-llms

Paul, Ganguli & Dziugaite (2021). Deep learning on a data diet. *NeurIPS*.

Robinson et al. (2021). Contrastive learning with hard negative samples. *ICLR*.

Shao et al. (2024). DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. [arXiv:2402.03300](https://arxiv.org/abs/2402.03300)

Shrivastava, Gupta & Girshick (2016). Online hard example mining. *CVPR*.

Sorscher et al. (2022). Beyond neural scaling laws: beating power law scaling via data pruning. *NeurIPS*.

Tajwar et al. (2024). Preference fine-tuning of LLMs should leverage suboptimal, on-policy data. *ICML*. [arXiv:2404.14367](https://arxiv.org/abs/2404.14367)

Weinshall, Cohen & Amir (2018). Curriculum learning by transfer learning. *ICML*.

Xia et al. (2024). LESS: Selecting influential data for targeted instruction tuning. *ICML*.

Yu et al. (2024). MetaMath: Bootstrap your own mathematical questions for LLMs. *ICLR*. [arXiv:2309.12284](https://arxiv.org/abs/2309.12284)

Zhang, Dai & Peng (2025). The best instruction-tuning data are those that fit (GRAPE). *NeurIPS*. openreview.net/forum?id=4jFSekBaDT

Zhao et al. (2022). Calibrating sequence likelihood improves conditional language generation (SLiC). [arXiv:2210.00045](https://arxiv.org/abs/2210.00045)

Data & models: Hendrycks et al. 2021 (MATH); Qwen 2025 ([arXiv:2412.15115](https://arxiv.org/abs/2412.15115)); Yang et al. 2024a (Qwen2.5-Math); Yang et al. 2025 (Qwen3, [arXiv:2505.09388](https://arxiv.org/abs/2505.09388)); Gemma Team 2025 ([arXiv:2503.19786](https://arxiv.org/abs/2503.19786)).

Benchmarks: AIMO 2024 (AIME24, AMC); Wei et al. 2023 (CMATH); Zhang et al. 2023 (Gaokao); Rein et al. 2024 (GPQA); Lewkowycz et al. 2022 (Minerva); He et al. 2024 (OlympiadBench); Ye et al. 2025 (LIMO).