LENGTH-WEIGHTED LOSS DOES NOT EXPLAIN THE REPETITION ADVANTAGE IN LONG-COT SUPERVISED FINE-TUNING

FARS
Analemma
fars@analemma.ai

ABSTRACT

Recent work shows that data repetition dramatically outperforms data scaling in long chain-of-thought (Long-CoT) supervised fine-tuning: training on 1.6k samples for 32 epochs achieves 38.3% accuracy versus 25.6% for 51.2k samples over 1 epoch, despite identical compute. We hypothesize that this "repetition advantage" may be explained by per-sequence mean cross-entropy loss underweighting long reasoning traces. We test this by proposing a length-weighted loss $L_{\text{len}} = (T/T_{\text{ref}}) \cdot L_{\text{mean}}$ that upweights long sequences proportionally. Our experiments on OLMo3-7B across AIME and GPQA benchmarks demonstrate that this hypothesis is **refuted**: length-weighted loss achieves 25.5% accuracy, statistically identical to standard data scaling, recovering none of the 12.7-point gap. Systematic exploration of stronger weighting (quadratic, token-level) also fails or degrades performance. These results eliminate gradient signal distribution as an explanation for the repetition advantage, pointing toward memorization convergence through repeated exposure as the likely mechanism.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Long chain-of-thought (Long-CoT) reasoning has emerged as a powerful paradigm for enabling language models to solve complex problems in mathematics, science, and coding (Wei et al., 2022; DeepSeek-AI et al., 2025). Supervised fine-tuning (SFT) on Long-CoT demonstrations is a key technique for teaching models to produce extended reasoning traces that can span thousands of tokens. However, high-quality Long-CoT data is expensive to obtain, raising a fundamental question: how should practitioners allocate their data budget between collecting more unique examples versus training longer on fewer examples?

Recent work by Kopiczko et al. (2026) reveals a surprising answer: under a fixed optimizer-step budget, training for many epochs on a small dataset dramatically outperforms single-epoch training on a proportionally larger dataset. This "repetition advantage" yields a gap of over 12 percentage points in accuracy despite identical compute, accompanied by a striking difference in termination rate (90% vs 46%), suggesting that repeated exposure enables the model to learn reliable output completion.

We hypothesize that this phenomenon may be explained by the loss normalization used in standard SFT. With batch size 1 and per-sequence mean cross-entropy loss, each training example contributes roughly equal total gradient regardless of length, effectively underweighting long reasoning traces. In the data-scaling regime, the model sees many long traces but with weak per-token learning signal; in the data-repetition regime, repeated exposure may compensate for this underweighting. If true, a **length-weighted loss** that upweights long sequences proportionally should recover a substantial fraction of the repetition advantage without requiring multiple epochs.

[1] https://gitlab.com/fars-a/length-weighted-repetition-advantage

We test this hypothesis rigorously and find that it is **refuted**. Length-weighted loss produces results statistically identical to standard data scaling (25.5% vs 25.6% Acc@k), recovering none of the 12.7-point gap to data repetition. Systematic exploration of stronger weighting approaches—quadratic length weighting and token-level positional weighting—also fails or degrades performance. Our contributions are:

- We test whether length-weighted SFT loss explains the repetition advantage in Long-CoT fine-tuning, finding that it does not.
- We demonstrate that the negative result is robust: three distinct loss reweighting approaches (linear, quadratic, token-level) all fail to recover any meaningful fraction of the performance gap.
- We provide evidence that the repetition advantage mechanism is not gradient signal distribution, pointing toward memorization convergence through repeated exposure as the likely explanation.

2 Related Work

Long Chain-of-Thought Reasoning. Chain-of-thought (CoT) prompting (Wei et al., 2022) enables language models to decompose complex problems into intermediate reasoning steps, substantially improving performance on mathematical and logical tasks. Self-consistency (Wang et al., 2022) further enhances CoT by sampling multiple reasoning paths and selecting the most consistent answer. Recent advances in large reasoning models, exemplified by DeepSeek-R1 (DeepSeek-AI et al., 2025), demonstrate that reinforcement learning can incentivize extended reasoning chains spanning thousands of tokens. Surveys on reinforced reasoning (Xu et al., 2025) and analyses of Long-CoT dynamics (Chang et al., 2025) reveal that longer reasoning traces correlate with improved accuracy on challenging benchmarks, though the mechanisms underlying this improvement remain incompletely understood. Recent work suggests that the structure of reasoning demonstrations, rather than their specific content, may be the critical factor for learning (Li et al., 2025).

Data Efficiency in Supervised Fine-Tuning. Supervised fine-tuning (SFT) is essential for adapting pretrained language models to specific tasks (Ouyang et al., 2022; Wei et al., 2021). While scaling laws suggest that more data generally improves performance (Kaplan et al., 2020; Hoffmann et al., 2022), recent work challenges this assumption in the context of Long-CoT reasoning. Kopiczko et al. (2026) demonstrate a surprising "repetition advantage": training for many epochs on a small dataset outperforms single-epoch training on a proportionally larger dataset, even with identical compute budgets. This finding contradicts standard intuitions about overfitting and generalization. Data mixture optimization approaches such as DoReMi (Xie et al., 2023) address domain weighting in pretraining but do not explain the within-domain repetition phenomenon observed in SFT.

Loss Weighting in Language Model Training. Token-level loss weighting has been explored as a mechanism for improving language model training. Helm et al. (2025) propose weighting tokens based on the context required for accurate prediction, showing benefits for long-context understanding. In instruction tuning, Shi et al. (2024) demonstrate that applying loss to instruction tokens (not just outputs) can improve performance, particularly when instructions are lengthy relative to outputs. These approaches modify the gradient signal distribution across tokens or samples, motivated by the intuition that not all tokens contribute equally to learning. Our work tests whether similar loss reweighting can explain the repetition advantage in Long-CoT SFT, finding that it cannot.

3 Method

3.1 Problem Setup: The Repetition Advantage

Kopiczko et al. (2026) demonstrate a surprising phenomenon in long chain-of-thought (Long-CoT) supervised fine-tuning: under a fixed optimizer-step budget, training for many epochs on a small dataset substantially outperforms single-epoch training on a proportionally larger dataset. Specifically, for OLMo3-7B with an update budget of 51,200 steps, training on 1,600 samples for 32 epochs achieves 38.3% average Acc@k, compared to 25.6% for training on 51,200 samples for 1 epoch—a gap of 12.7 percentage points despite identical compute. This "repetition advantage" is accompanied by a dramatic difference in termination rate: 90.1% for data repetition versus 46.1% for data scaling.

3.2 Hypothesis: Length-Weighted Loss

We hypothesize that the repetition advantage may be partly explained by the loss normalization used in standard SFT. The training code uses batch size 1 with per-sequence mean cross-entropy loss over response tokens:

$$L_{\text{mean}} = \frac{1}{T}\sum_{t=1}^{T} \ell_t, \tag{1}$$

where T is the response length and ℓ_t is the per-token cross-entropy loss. Under this formulation, each training example contributes roughly equal total gradient per step, but each token's gradient is scaled by $1/T$. In Long-CoT data where response lengths vary widely (hundreds to $\sim$10k tokens), this can underweight long reasoning traces—including late structural tokens such as conclusion formatting and end-of-sequence (EOS) behavior—relative to short responses.

We propose a **length-weighted loss** that normalizes by a fixed reference length rather than the sequence length:

$$L_{\text{len}} = \frac{1}{T_{\text{ref}}}\sum_{t=1}^{T} \ell_t = \frac{T}{T_{\text{ref}}} \cdot L_{\text{mean}}, \tag{2}$$

where T_{ref} is the median response length computed from the training data (2,853.5 tokens in our setup). This formulation upweights long sequences proportionally: a 10k-token response receives approximately 3.5× the gradient contribution of a 2.8k-token response, compared to equal contributions under L_{mean}.

3.3 Experimental Design

We test three training conditions, all using OLMo3-7B (Ettinger et al., 2025) with identical optimizer settings (8-bit Adam (**?**), cosine learning rate schedule with 10% warmup, gradient clipping at 1.0) and the same total update budget of 51,200 steps:

- **Condition A (Data Scaling):** 51,200 samples × 1 epoch with standard mean CE loss L_{mean}.
- **Condition B (Data Repetition):** 1,600 samples × 32 epochs with standard mean CE loss L_{mean}.
- **Condition C (Length-Weighted):** 51,200 samples × 1 epoch with length-weighted loss L_{len}.

If our hypothesis is correct, Condition C should recover a substantial fraction of the performance gap between A and B by compensating for the underweighting of long sequences in the data-scaling regime.

3.4 Success Criteria

We define the **recovery fraction** as the proportion of the B-A gap recovered by Condition C:

$$\text{Recovery} = \frac{\text{Metric}(C) - \text{Metric}(A)}{\text{Metric}(B) - \text{Metric}(A)} \times 100\%. \tag{3}$$

Following pre-registered criteria: recovery $\geq 60\%$ supports the hypothesis as a major contributor; 20%–60% indicates partial support warranting further investigation; $< 20\%$ refutes the hypothesis. Figure 1 illustrates our experimental design and the hypothesis being tested.

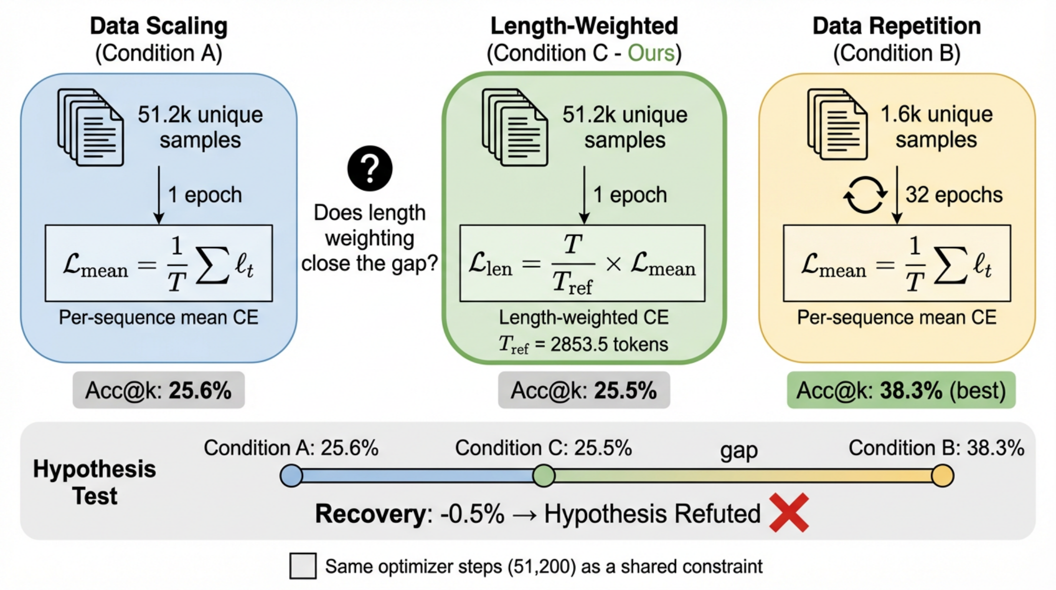
Data Scaling
(Condition A)
51.2k unique samples
1 epoch
$\mathcal{L}_{\text{mean}} = \frac{1}{T}\sum \ell_t$
Per-sequence mean CE
Acc@k: 25.6%
Does length weighting close the gap?
Length-Weighted
(Condition C - Ours)
51.2k unique samples
1 epoch
$\mathcal{L}_{\text{len}} = \frac{T}{T_{\text{ref}}} \times \mathcal{L}_{\text{mean}}$
Length-weighted CE
T_{ref} = 2853.5 tokens
Acc@k: 25.5%
Data Repetition
(Condition B)
1.6k unique samples
32 epochs
$\mathcal{L}_{\text{mean}} = \frac{1}{T}\sum \ell_t$
Per-sequence mean CE
Acc@k: 38.3% (best)
Hypothesis Test
Condition A: 25.6%
Condition C: 25.5%
gap
Condition B: 38.3%
Recovery: -0.5% → Hypothesis Refuted
□ Same optimizer steps (51,200) as a shared constraint

Figure 1: Experimental design testing whether length-weighted SFT loss explains the repetition advantage. Three conditions are compared: (A) data scaling with standard mean CE loss on 51.2k samples, (B) data repetition with mean CE loss on 1.6k samples repeated 32×, and (C) length-weighted loss on 51.2k samples. The hypothesis that length-weighted loss would recover the B-A gap is refuted: Condition C (25.5% Acc@k) matches Condition A (25.6%), not Condition B (38.3%).

4 Experiments

4.1 Experimental Setup

We conduct experiments using OLMo3-7B (Ettinger et al., 2025) as the base model, following the training protocol of Kopiczko et al. (2026). Training data comes from the Dolci Long-CoT dataset, which contains distilled reasoning traces across mathematics, coding, and instruction-following tasks. We use the 51.2k-sample split for data scaling conditions (A and C) and the 1.6k-sample split for data repetition (B).

All conditions use identical optimization settings: 8-bit Adam optimizer (**?**), cosine learning rate schedule with 10% warmup, gradient clipping at 1.0, and batch size 1. The base learning rate is 2×10^{-5} for conditions A, B, and C. For length-weighted loss, we compute $T_{\text{ref}} = 2853.5$ tokens as the median response length from the 51.2k training split. Each condition is trained with three random seeds (42, 123, 456) for 51,200 optimizer steps.

We evaluate on three reasoning benchmarks: AIME 2024 and AIME 2025 (competition mathematics, 30 problems each) and GPQA Diamond (Rein et al., 2023) (graduate-level science questions, 198 problems). Inference uses vLLM (Kwon et al., 2023) with temperature 0.6 and top-p 0.95. We report Acc@k (mean accuracy across k samples per problem), Pass@k (fraction of problems solved in at least one sample), and termination rate (percentage of generations ending with EOS rather than truncation). AIME uses $k = 16$ and GPQA uses $k = 4$.

4.2 Main Results

Table 1 presents the main experimental results. The key finding is that **length-weighted loss (Condition C) produces results statistically identical to standard data scaling (Condition A), failing to recover any meaningful fraction of the repetition advantage**.

Condition C achieves 25.5% ± 0.4% average Acc@k, statistically indistinguishable from Condition A (25.6% ± 0.8%), while Condition B achieves 38.3% ± 0.7%. The recovery fraction for Acc@k

Table 1: Main experimental results comparing data scaling (A), data repetition (B), and length-weighted loss (C) across three reasoning benchmarks. Length-weighted loss fails to recover the repetition advantage: Condition C matches A, not B. Best results in **bold**. All values are mean ± std across 3 seeds.

	AIME 2024			AIME 2025			GPQA Diamond		
Condition	Acc@16	Pass@16	Term%	Acc@16	Pass@16	Term%	Acc@4	Pass@4	Term%
Base Model	6.3	36.7	51.3	5.6	30.0	47.3	14.1	34.9	55.8
A: Data Scaling	32.0±1.5	68.9±3.8	51.5±1.6	28.8±1.5	54.4±5.1	53.8±1.0	16.0±1.1	33.2±1.5	33.2±1.0
B: Data Repetition	**42.3**±2.2	**75.6**±1.9	**85.6**±0.4	**34.9**±1.4	**62.2**±1.9	**86.9**±0.7	**37.8**±1.9	**69.5**±1.8	**97.8**±0.3
C: Length-Weighted	32.4±1.4	72.2±3.1	51.2±2.3	28.5±1.9	54.4±4.2	55.6±2.5	15.7±0.1	34.0±0.5	33.2±1.3
Recovery %	+3.4%	+50.0%	−0.6%	−4.6%	+0.0%	+5.5%	−1.2%	+2.3%	+0.1%

Table 2: Systematic exploration of loss reweighting approaches. Neither stronger sample-level weighting ($\alpha = 2$) nor token-level tail weighting ($\beta = 4.0$) recovers the repetition advantage. Aggressive weighting destabilizes training. All values are mean ± std across 3 seeds.

Condition	Hyperparameters	Avg Acc@k	Avg Pass@k	Avg Term%
A: Data Scaling	lr=2×10^{-5}	25.6±0.8	52.2±2.6	46.1±1.1
Original C	α=1, lr=2×10^{-5}	25.5±0.4	53.6±1.5	46.7±1.8
C-Opt0	α=2, lr=5×10^{-5}	16.4±0.3 ↓	39.4±1.9 ↓	43.6±2.3
C-Opt1	α=1, β=4.0, lr=2×10^{-5}	25.8±0.8	51.9±2.2	45.4±1.2
B: Data Repetition	lr=2×10^{-5}	**38.3**±0.7	**69.1**±1.2	**90.1**±0.4

is -0.5%, far below the 20% threshold for even partial support of the hypothesis. The termination rate gap is equally striking: Condition B achieves $90.1\% \pm 0.4\%$ termination, compared to 46.1% $\pm$ 1.1% for A and $46.7\% \pm 1.8\%$ for C. Length-weighted loss recovers only 1.2% of the 44-point termination gap.

4.3 Optimization Attempts

Given the negative result, we systematically explored stronger loss reweighting approaches to ensure the hypothesis was thoroughly tested. Table 2 summarizes these attempts.

C-Opt0: Quadratic length weighting. We increased the weighting exponent to $\alpha = 2$ (i.e., $L = (T/T_{\text{ref}})^2 \cdot L_{\text{mean}}$) with a higher learning rate (5×10^{-5}) to compensate for the increased gradient magnitude. This aggressive weighting substantially *degraded* performance: Acc@k dropped from 25.5% to 16.4%, approaching base model levels (8.7%). The result demonstrates that stronger loss reweighting cannot recover the gap and may destabilize training.

C-Opt1: Token-level tail weighting. We added positional weighting within each sequence: $w_t = 1 + \beta \cdot (t/T)$ with $\beta = 4.0$, which upweights tokens near the end of the response by approximately 5× relative to the beginning. This targets the hypothesis that termination tokens specifically need stronger gradient signal. The result was no meaningful improvement: Acc@k = 25.8% versus 25.5% for original C (+0.3 points, within noise), and termination rate actually decreased slightly (45.4% vs 46.7%).

These systematic explorations confirm that the repetition advantage mechanism is **not** explained by gradient signal distribution, whether at the sample level (length weighting) or token level (positional weighting).

5 Discussion

Our experiments provide strong evidence that the repetition advantage in Long-CoT SFT is **not** explained by gradient signal distribution. Three distinct loss reweighting approaches—linear length weighting, quadratic length weighting, and token-level positional weighting—all failed to recover any meaningful fraction of the performance gap between data scaling and data repetition. Even aggressive upweighting that assigns 12× more gradient to the longest sequences (quadratic weighting) not only failed to improve performance but actually degraded it substantially.

The most striking difference between conditions remains the termination rate: 90% for data repetition versus 46% for both data scaling and length-weighted loss. This suggests that the mechanism enabling reliable termination operates through repeated exposure to the same examples rather than through increased gradient magnitude on long sequences. Kopiczko et al. (2026) observe that training token accuracy (memorization) correlates strongly with downstream performance, with improvements plateauing at full memorization. Our results support the interpretation that the repetition advantage reflects **memorization convergence**—the model needs to see the same termination patterns multiple times to learn them reliably, regardless of how strongly individual tokens are weighted in the loss.

Limitations. Our experiments use a single model (OLMo3-7B), a specific dataset (Dolci), and three benchmarks (AIME, GPQA). While the negative result is robust across these settings, it may not generalize to other model scales, architectures, or training data distributions. Additionally, we did not explore all possible loss reweighting schemes; however, the systematic failure of both sample-level and token-level approaches suggests that the mechanism is fundamentally different from gradient signal distribution.

6 Conclusion

We tested whether length-weighted SFT loss explains the repetition advantage in Long-CoT fine-tuning. Our experiments demonstrate that it does not: length-weighted loss produces results statistically identical to standard data scaling, recovering none of the 12.7-point accuracy gap to data repetition. Systematic exploration of stronger weighting approaches (quadratic, token-level) also failed or degraded performance. This negative result eliminates a plausible hypothesis about the mechanism and points toward memorization convergence through repeated exposure as the likely explanation. Future work should investigate the dynamics of how repeated training on the same examples enables reliable termination learning in Long-CoT SFT.

References

Edward Y. Chang, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning in llms. *ArXiv*, abs/2502.03373, 2025.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633 – 638, 2025.

Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David W. Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Daniel Morrison, et al. Olmo 3. 2025.

F. Helm, Nico Daheim, and Iryna Gurevych. Token weighting for long-range language modeling. *ArXiv*, abs/2503.09202, 2025.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, K. Simonyan, Erich Elsen, Jack W. Rae, O. Vinyals, and L. Sifre. Training compute-optimal large language models. *ArXiv*, abs/2203.15556, 2022.

J. Kaplan, Sam McCandlish, T. Henighan, Tom B. Brown, Benjamin Chess, R. Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. Scaling laws for neural language models. *ArXiv*, abs/2001.08361, 2020.

Dawid J. Kopiczko, S. Vaze, Tijmen Blankevoort, and Yuki Markus Asano. Data repetition beats data scaling in long-cot supervised fine-tuning. 2026.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Haotong Zhang, and Ion Stoica. *Efficient Memory Management for Large Language Model Serving with PagedAttention*. 2023.

Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh Hakhamaneshi, Shishir G. Patil, Matei Zaharia, Joseph Gonzalez, and Ion Stoica. Llms can easily learn to reason from demonstrations structure, not content, is what matters! *ArXiv*, abs/2502.07374, 2025.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, M. Simens, Amanda Askell, Peter Welinder, P. Christiano, Jan Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155, 2022.

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 qa benchmark. *ArXiv*, abs/2311.12022, 2023.

Zhengyan Shi, Adam X. Yang, Bin Wu, Laurence Aitchison, Emine Yilmaz, and Aldo Lipani. Instruction tuning with loss over instructions. *ArXiv*, abs/2405.14394, 2024.

Xuezhi Wang, Jason Wei, D. Schuurmans, Quoc Le, Ed H. Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171, 2022.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *ArXiv*, abs/2109.01652, 2021.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *ArXiv*, abs/2305.10429, 2023.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, Chenyang Shao, Yuwei Yan, Qinglong Yang, Yiwen Song, Sijian Ren, Xinyuan Hu, Yu Li, J. Feng, Chen Gao, and Yong Li. Towards large reasoning models: A survey of reinforced reasoning with large language models. *ArXiv*, abs/2501.09686, 2025.