Chasing the Small-Frontier on a Fixed Rig

Metric Artifacts, a Learning-Rate Crack, and an Honestly Measured Gap

Jovonni L. Pharr
Georgia Cyber Warfare Range
info@gacwr.org

August 9, 2026

v0.0.1

Abstract

This work is an honest, receipt-driven account of an attempt to train a small language model to the quality of the small-frontier on a fixed pair of graphics processors, with no recourse to rented compute. The instructive result is not a state of the art but a sequence of *corrections*, each overturning a confident prior belief, and each surfaced not by additional training but by adversarial self-audit. We find, first, that our own evaluation was misleading us in *both* directions: a composite that averages unnormalized per-task margins is dominated by a single high-variance task, and one benchmark in particular is a class-prior coin flip whose apparent victory is an artifact. Correcting these and, crucially, scoring the reference through our *own* harness exposed a deeper fault—our scorer computed a different, unfair quantity for the two models and normalized by token count across incomparable vocabularies—so we rebuilt it around a single byte-normalized, joint-tokenization scorer. On that corrected ruler the picture is honest and split: the reference leads argmax accuracy by a modest three-and-a-half points, yet on the continuous log-likelihood margin our larger model leads—a gap of calibration and coverage, not of gross capability, and a gap to close rather than a ceiling. We find, second, that the learning rate used by every one of our runs, including the historical champion, was well below its optimum, a silent multiplier that had been suppressing every data intervention we had declared dead. We find, third, that we had been preparing to *manufacture*, at a cost of years of compute, a corpus that was freely available to *download* at the cost of a day. We document the diagnostic arc, the method of adversarial multi-agent auditing that produced these corrections, and the concrete and feasible plan that survives them.

Contents

I Preliminaries **2**

1 The State of Today **2**

2 The Rig **2**

II The Corrections **2**

3 The Measure Was Misleading Us **2**

4 The Learning Rate Was Too Cold **5**

5 One Should Download, Not Manufacture **6**

6 On Why More of Our Own Data Did Not Help **7**

III Method and Plan **7**

7 Adversarial Multi-Agent Auditing **7**

8 The Surviving Plan **8**

9 Related Work **8**

10 Conclusion **9**

Part I
Preliminaries

1 The State of Today

There exists a class of small language models—on the order of one hundred to five hundred million parameters—that attain a surprising fraction of the competence of their larger relatives on standard benchmarks. The existence of these models is a standing proof that a great deal of capability is reachable at small scale, given the right data and the right recipe. It is natural, for a laboratory constrained to modest hardware, to take one such model as a reference and to ask what separates one's own model of comparable size from it. The reference we adopt is a widely reported one-hundred-and-thirty-five-million-parameter model; our own model is slightly larger, at one hundred and fifty-eight million parameters, and yet it scores lower on our own evaluation harness. That a *smaller* model should outperform a larger one is, on its face, evidence that the difference is one of skill and recipe rather than of scale—and it is this premise that organizes the entire investigation.

2 The Rig

The constraint is absolute and it is worth stating precisely, for it shapes every decision that follows. The laboratory possesses two graphics processors and no others; there is no cloud, and there will be no additional hardware. Local computation is, in the sense of marginal cost, unlimited—there are no rental bills—but it is not unlimited in *throughput*: a single processor sustains on the order of seventy-four million training tokens per hour, and the trainer is not data-parallel. Models are capped at one gigabyte. Within these constraints the question is not how to buy one's way to the frontier but how to reach it by skill.

Part II
The Corrections

3 The Measure Was Misleading Us

The first and most consequential correction concerns not the model but the ruler. Our headline metric is a composite that averages, across a set of tasks $\mathcal{T}$, a smooth per-example margin between the log-likelihood assigned to the correct completion $y^\star$ and the mean log-likelihood

assigned to its distractors $y \in D(x)$:

$$\text{Composite} \;=\; \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} \underbrace{\mathbb{E}_{x\sim t}\left[\log p_\theta(y^\star \mid x) - \frac{1}{|D(x)|}\sum_{y\in D(x)} \log p_\theta(y \mid x)\right]}_{\text{margin}_t}. \tag{1}$$

This composite has a structural defect: the per-task margins margin_t are not normalized to a common scale, and a task whose margins are intrinsically large dominates the unweighted average. Figure 1 decomposes the composite of our champion by task and shows the pathology directly—the margin of a single science task, SCIQ, is of the same order as the sum of every other task's contribution, so that the composite is very nearly a one-task measurement wearing the costume of an aggregate.

A more serious defect afflicts one particular benchmark, BOOLQ, a yes-or-no reading task scored by the likelihood of the single tokens "yes" and "no." Every model answers largely according to its *class prior*. Writing a_+ and a_- for accuracy on the true and false instances, the reported accuracy on a majority-+ set of positive fraction π is $\pi a_+ + (1-\pi)a_-$, which a prior-only classifier maximizes by always emitting the majority label; the honest measure is the balanced accuracy

$$\text{BA} \;=\; \tfrac{1}{2}\big(a_+ + a_-\big), \tag{2}$$

which for every model we tested, including the reference, sits at essentially $\frac{1}{2}$. The instability is visible without any per-class breakdown at all: across seventy logged runs, reported BOOLQ accuracy swings with standard deviation 0.091 about the 0.50 chance line, while a genuine four-way task, ARC_EASY, is three times steadier at 0.031 (Figure 2). The apparent victory of our champion on BOOLQ is therefore an artifact of a prior; it simultaneously inflated our champion's standing and concealed the true shape of the gap to the reference.

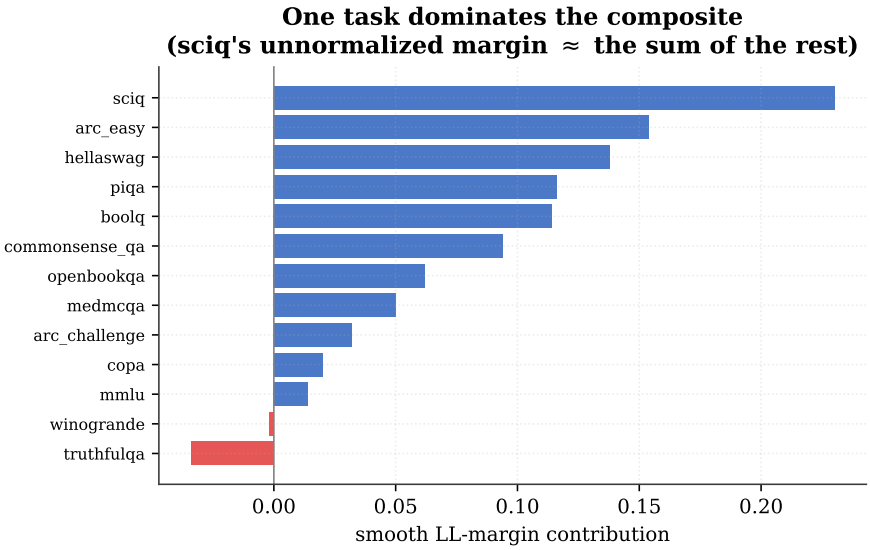


Figure 1: Task decomposition of the champion's composite (Eq. 1). A single task, SCIQ, contributes a margin comparable to the sum of all others: the aggregate is scale-dominated.

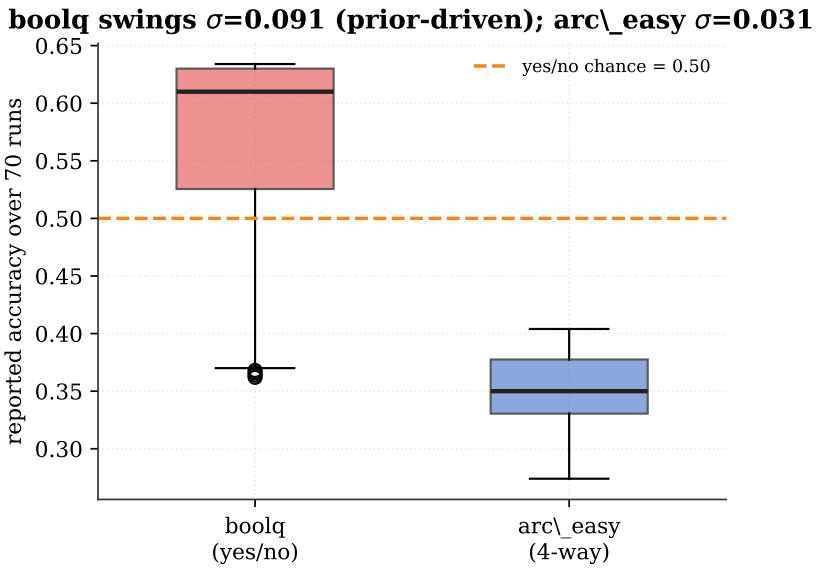


Figure 2: BOOLQ accuracy over 70 runs swings three times as widely as ARC_EASY and straddles the yes/no chance line—the signature of a prior-driven coin flip, not a skill.

Removing the coin-flip benchmark and giving the remaining tasks equal weight is necessary but not sufficient, for it corrects only our own scores. Scoring the reference through our own harness is the next step—but it exposed a deeper fault, and it is the fourth correction: *the ruler itself was miscalibrated.* Our two scoring paths did not compute the same quantity. Our own checkpoints were scored by encoding context and answer separately and concatenating the token identifiers, which retokenizes the answer as though it began a document; the reference was scored by a joint encoding whose boundary index silently dropped the first answer token on multi-token answers; and both normalized by *token count*, a quantity that is not comparable across a sixteen-thousand and a forty-nine-thousand vocabulary. We replaced both with a single shared scorer: one joint tokenization, an answer boundary recovered by longest common prefix, and normalization by the answer's *byte* length, which is tokenizer-invariant. Only on this corrected ruler is a comparison meaningful.

The corrected result is more honest and, in one respect, less sobering. On argmax accuracy the reference still leads, but the two metrics now *disagree* (Table 1, Figures 3 and 4): the reference leads mean accuracy 0.413 to 0.377, a gap of 0.036, and leads six of seven tasks; yet on the byte-normalized smooth log-likelihood margin—the continuous, capability-sensitive signal—*our* model leads, +0.197 to +0.163. A model that trails on the discrete decision while leading on the underlying margin is a model whose knowledge is close but whose calibration at the decision boundary lags. That a smaller reference of one hundred and thirty-five million parameters still edges our one hundred and fifty-eight million on accuracy is not a ceiling but a skill-and-data gap; the corrected measurement tells us it is a gap of confidence and coverage, not of gross capability.

Table 1: Head-to-head with *both* models scored on our single fixed, byte-normalized scorer ($n = 300$ per task, re-measured after the harness correction). The reference leads argmax accuracy by 0.036; on the smooth byte-normalized LL-margin our model leads (+0.197 vs +0.163).

	MMLU	ARC_E	ARC_C	HELLA	WINO	OBQA	CSQA	**mean**
ours (v412, 158M)	0.290	0.483	**0.283**	0.367	0.517	0.333	0.367	0.377
reference (135M, ours)	**0.373**	**0.553**	0.273	**0.423**	**0.527**	**0.360**	**0.383**	**0.413**
gap (ref − ours)	+.083	+.070	−.010	+.056	+.010	+.027	+.016	+.036

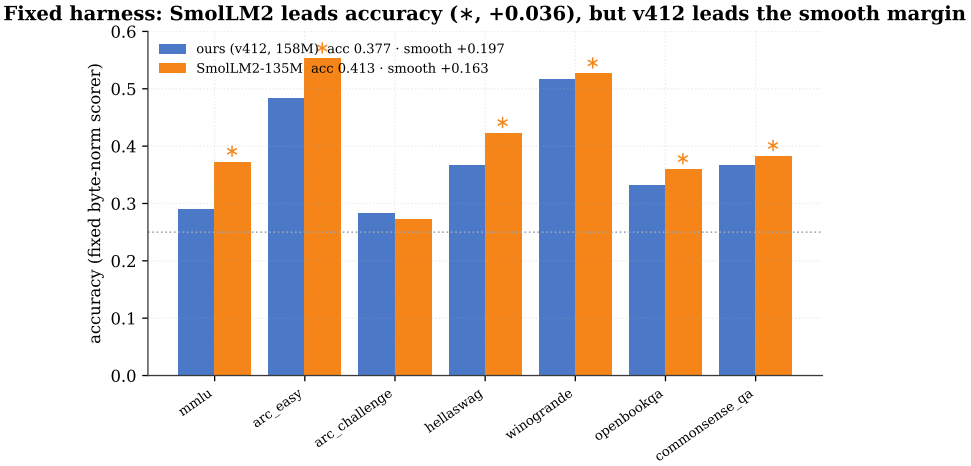


Figure 3: Head-to-head on our fixed byte-normalized harness (both models, one shared scorer). The reference leads argmax accuracy (∗) on six of seven tasks; on the smooth margin our model leads.

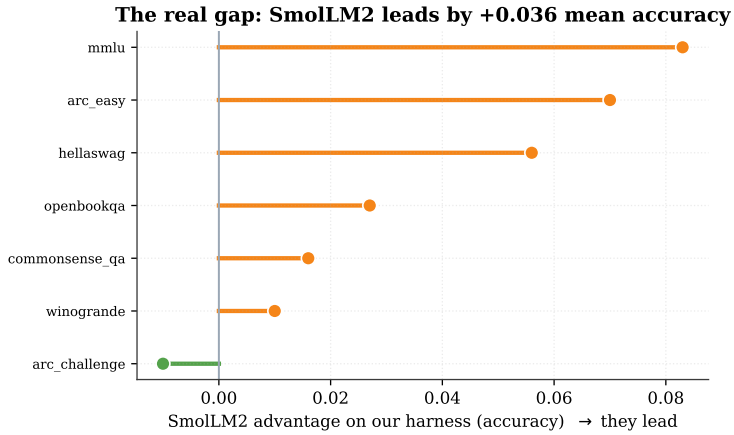


Figure 4: The real per-task gap. Positive means the reference leads; the advantage is spread across knowledge and reasoning, not a single column.

4 The Learning Rate Was Too Cold

The second correction concerns the most elementary of hyperparameters. A controlled sweep of the optimizer's peak learning rate η, at matched token budget, describes a clean inverted-U in the smooth metric $M(\eta)$ (Table 2, Figure 5): the value that every one of our prior runs

had used—including the historical champion—sits at $\eta = 6\times10^{-4}$, well to the cold side of the optimum $\eta^\star \approx 1.5\times10^{-3}$, so that

$$\frac{M(\eta^\star)}{M(6\times10^{-4})} \;=\; \frac{0.204}{0.069} \;\approx\; 3.0. \tag{3}$$

The optimum yields roughly three times the smooth metric at the *same* compute, and a small-scale proxy confirms that $\eta^\star$ transfers across scale. This diagnosis is corroborated from an independent direction: the reference model's own technical report discloses that its hundred-and-thirty-five-million-parameter model was trained at a peak learning rate of 3×10^{-3}—five times hotter than our historical default and of the same order as the optimum our sweep independently located, and far above the 5×10^{-4} used for the same family's much larger model. The elementary knob we had mis-set is one the reference had set hot on purpose. The implication is uncomfortable and clarifying at once: every previous verdict of the form "this data intervention does not help" was rendered at the wrong learning rate, and is therefore void until repeated at the correct one. Indeed, an intervention we had recorded as harmful reverses to helpful once the learning rate is corrected. A single mis-set knob had been silently taxing every experiment we had run.

Table 2: Peak-learning-rate sweep at matched budget. The champion's setting is 2.5× colder than the optimum.

Muon peak η	6×10^{-4} (prior default)	1.5×10^{-3} (optimum)	2.5×10^{-3}	4×10^{-3}
smooth metric $M(\eta)$	0.069	**0.204**	0.075	0.026

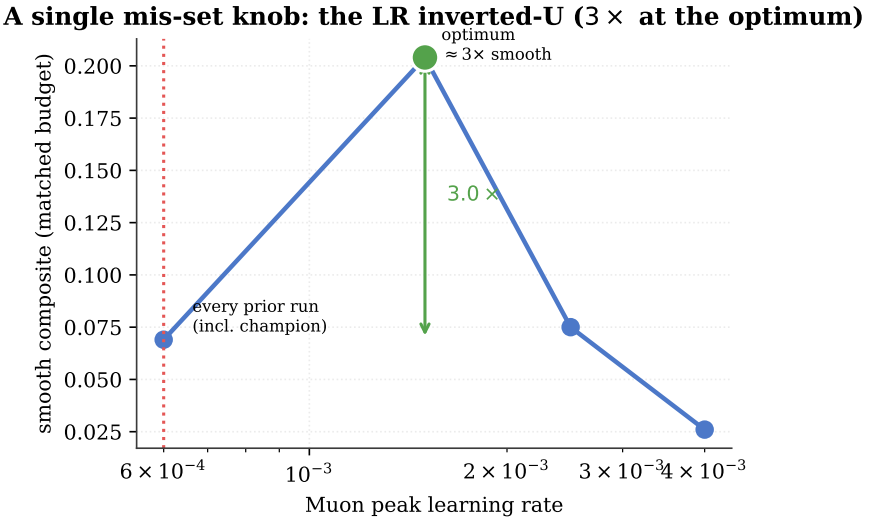


Figure 5: The learning-rate inverted-U (Eq. 3). Every historical run, including the champion, sat on the cold shoulder; the optimum recovers a factor of three at fixed compute.

5 One Should Download, Not Manufacture

The third correction concerns the acquisition of data, and it was the most humbling. The measured deficit is at bottom a deficit of knowledge and reasoning, and the natural response is to supply that as data. We had been preparing to *generate* it, by distillation from a local teacher model. Two independent adversarial audits arrived at the same objection: the generation is infeasible and the manufactured data would be poor. A teacher of moderate size emits on the

order of one million tokens per hour; the tens of billions of tokens required would consume years of compute, and the output of a weak generator is itself weak. The corpus we needed was, in fact, the reference model's own training data—curated synthetic textbooks, mathematics, and filtered web text—and it is openly published and re-tokenizable to our vocabulary at the cost of a day of ordinary input and output. We had been on the verge of spending years to manufacture, badly, what could be downloaded in a day.

6 On Why More of Our Own Data Did Not Help

A pattern underlies all three corrections. We ran a controlled token-scaling series over our single narrow corpus—six models identical but for the number of passes over the same data—and measured, for each, the held-out bits-per-byte and the fair equal-weight accuracy of Table 1. The result is the clean signature of memorization (Figure 6): bits-per-byte falls monotonically, from 1.161 to 1.126, as the model compresses the corpus ever more tightly, while the fair accuracy does not move at all, holding at 0.389 ± 0.003—a spread indistinguishable from evaluation noise. Compression improves; competence does not. The literature on data-constrained scaling anticipates exactly this: past a few epochs, the marginal capability of an additional pass over a fixed corpus decays to zero. The lever is not more passes over the data one has, but more unique, diverse, and high-quality data—which, per the third correction, is to be acquired rather than fabricated.

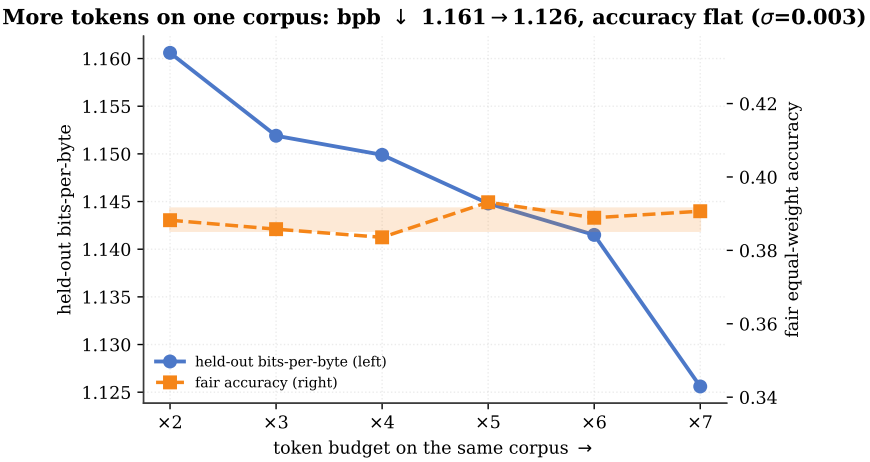


Figure 6: More tokens on the *same* corpus. Bits-per-byte (left axis) falls monotonically while fair accuracy (right axis, shaded $\pm 1\sigma$ band) is flat within noise. The gain is compression, not competence.

Part III
Method and Plan

7 Adversarial Multi-Agent Auditing

It is worth remarking upon the method by which these corrections were obtained, for it was not the method by which they might have been expected. Each correction was produced not by a training run but by convening a small panel of independent analytical agents, charged with proposing a diagnosis and then adversarially auditing one another's diagnoses against the saved record of results. This process is what caught the coin-flip artifact, the scale-domination

of the composite, and the futility of manufacturing data. The lesson generalizes: in a regime where each experiment is expensive, the cheapest and most reliable source of progress is the disciplined, adversarial re-examination of what one already believes.

8 The Surviving Plan

The plan that survives the corrections is no longer a guess, because the reference's technical report supplies the recipe we must match. The reference model was trained on the order of two trillion tokens; we do not have that budget, and we do not pretend to. But its report is explicit that, at our scale, the lever is not tokens but the *mixture*: its small models use a single-stage schedule of consistently high-quality, freely downloadable corpora—a roughly sixty-forty blend of educational and filtered web text, an educational-code slice, and, above all, a step-by-step mathematics corpus that the report credits with doubling grade-school-math accuracy. Every one of these is downloadable, which is the third correction made concrete. Our plan is therefore to reproduce the reference architecture to the parameter, train on a feasible budget of some one hundred billion unique downloaded tokens—on the order of a thousand tokens per parameter, a run of some two processor-weeks—at the corrected hot learning rate, over this same mixture, and to grade the result against the reference *on our own harness*, the only comparison this work has shown to be trustworthy. The objective is no longer to defend a tie but to erase a measured, honest gap of three points of mean accuracy; the levers to do so—a hotter learning rate we had left on the table, and a mathematics-and-education data mixture we can simply download—are now identified and in hand.

9 Related Work

This study sits at the intersection of several mature literatures; our contribution is not to found any of them but to expose, on a fixed rig, how their known results interact—and how neglecting them silently taxed every experiment we ran. We organize the prior art into the threads that each of our corrections touches.

Small models and the data-centric turn. The premise that a 100–500M model can capture a large fraction of frontier competence is by now well established. Our reference, SmolLM2 [1], is the direct exemplar, and its lineage runs through the "textbooks" program—phi-1 [6] and phi-1.5 [7]—and through TinyStories [8], which showed that curated, synthetic data lets sub-10M models produce coherent text. Cosmopedia [4] is the open reproduction of that synthetic-textbook recipe. Architectural and training-suite work on the same scale band—MobileLLM [10], the controlled Pythia suite [9], and the Chinchilla-scaled Cerebras-GPT family [11]—frames our own 158M model as a point in a well-charted design space. The common thread, which our third correction (§6) rediscovers the hard way, is that data quality and mixture, not parameter count, separate these models.

Compute-optimal scaling and over-training. The scaling laws of Kaplan et al. [12] and the compute-optimal correction of Chinchilla [13] fix the tokens-per-parameter ratio for a *single* training pass on *fresh* data. Two refinements bear on our memorization result (Figure 6). First, small deployable models are deliberately trained far past the Chinchilla point, a regime whose predictability Gadre et al. [14] characterize as reliable over-training. Second, Muennighoff et al. [3] show that when data is *repeated* rather than fresh, the marginal value of additional epochs decays toward zero after a handful of passes—exactly the flat-accuracy, falling-bits-per-byte signature we observe. Our finding is thus a single-rig instance of a documented law, not a novel phenomenon.

Learning-rate and schedule science. Our second correction—an inverted-U in peak learning rate with the historical default sitting 2.5× too cold (Eq. 3)—is elementary in isolation but connects to an active thread. The warmup-stable-decay (WSD) schedule introduced with MiniCPM [15], and the constant-plus-cooldown analysis of Hägele et al. [16], decouple the decay phase from a fixed horizon and thereby rationalize the warm-start continuation we pursue. The claim that our optimal rate transfers from a small-scale proxy is precisely the maximal-update-parametrization (μTransfer) result of Yang et al. [17]. That our runs use the Muon optimizer [5]—whose scalability at LLM scale was subsequently established by Liu et al. [18]—matters because the optimal peak rate is optimizer- and batch-size-dependent; the gradient-noise-scale model of McCandlish et al. [19] and the learning-rate/batch-size interchange of Smith et al. [20] are why a single mis-set rate cannot be read off another lab's report without re-tuning.

Data curation, filtering, and mixture. The corpora our surviving plan (§8) targets are products of a large curation literature: FineWeb and its educational filter FineWeb-Edu [21], the DataComp-LM benchmark and DCLM-Baseline [22], and the classifier-filtered lineage from The Pile [23] through RefinedWeb [24] to Dolma [25]; the code slice descends from StarCoder2 and The Stack v2 [26]. Complementary techniques explain *why* these outperform raw scale: deduplication [27], importance-resampling selection (DSIR) [28], and learned mixture weights (DoReMi) [29]. Crucially, the SmolLM2 report [1] and the domain-upsampling study of Blakeney et al. [30] show that a late, high-quality *annealing* phase—the decay stage of WSD [15]—is where mathematics and educational data pay off.

Knowledge distillation as an alternative to curation. The distillation route we considered and rejected on feasibility grounds (§5) rests on classical work: logit distillation [31], sequence-level distillation [32], and on-policy LLM distillation [33]. The synthetic-textbook recipes [6, 4] are effectively offline distillation from a strong teacher; our audit's conclusion was that our *local* teacher was too weak and slow for this to beat simply downloading the curated corpora—a scope-limited claim, not a universal one.

Evaluation methodology. Our first and most consequential correction concerns the ruler, and here the literature is explicit that our failure modes are known ones. Schaeffer et al. [2] argue that the choice between a discrete and a smooth metric can manufacture or dissolve apparent capability jumps—the same split we find between argmax accuracy and the log-likelihood margin (Table 1)—qualifying the emergent-ability claims of Wei et al. [34]. The reproducibility lessons of Biderman et al. [35], from the lm-evaluation-harness, document precisely the normalization and tokenization pitfalls (per-token vs. per-byte length normalization, answer-boundary handling) that our harness correction repairs. That multiple-choice benchmarks carry large seed- and prompt-variance is quantified by Madaan et al. [36] and, for MMLU, by Gema et al. [37]; this is the general form of our BOOLQ coin-flip (Figure 2, Eq. 2). Our use of paired bootstrap confidence intervals follows standard significance-testing practice [38, 39]. Our contribution here is not that benchmarks are noisy but the specific, receipt-backed decomposition (Eq. 1) showing a single unnormalized task dominating our composite, and the rebuilt byte-normalized joint-tokenization scorer that makes a cross-vocabulary comparison legitimate.

10 Conclusion

Training a small model to the frontier upon a fixed rig is governed less by the heroism of one's compute than by three disciplines: measuring the right quantity, setting the obvious knob correctly, and acquiring rather than manufacturing one's data. Each of our findings was the correction of a confident belief, and each was surfaced by adversarial self-audit rather than by further training. That, more than any single benchmark number, is the transferable result.

References

[1] L. Ben Allal et al. *SmolLM2: When Smol Goes Big.* 2025. arXiv:2502.02737.

[2] R. Schaeffer, B. Miranda, S. Koyejo. *Are Emergent Abilities of Large Language Models a Mirage?* NeurIPS, 2023. arXiv:2304.15004.

[3] N. Muennighoff et al. *Scaling Data-Constrained Language Models.* NeurIPS, 2023. arXiv:2305.16264.

[4] L. Ben Allal et al. *Cosmopedia.* Hugging Face, 2024.

[5] K. Jordan et al. *Muon and modded-nanoGPT.* 2024.

[6] S. Gunasekar et al. *Textbooks Are All You Need.* 2023. arXiv:2306.11644.

[7] Y. Li et al. *Textbooks Are All You Need II: phi-1.5 Technical Report.* 2023. arXiv:2309.05463.

[8] R. Eldan, Y. Li. *TinyStories: How Small Can Language Models Be and Still Speak Coherent English?* 2023. arXiv:2305.07759.

[9] S. Biderman et al. *Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling.* ICML, 2023. arXiv:2304.01373.

[10] Z. Liu et al. *MobileLLM: Optimizing Sub-billion Parameter Language Models for On-Device Use Cases.* ICML, 2024. arXiv:2402.14905.

[11] N. Dey et al. *Cerebras-GPT: Open Compute-Optimal Language Models Trained on the Cerebras Wafer-Scale Cluster.* 2023. arXiv:2304.03208.

[12] J. Kaplan et al. *Scaling Laws for Neural Language Models.* 2020. arXiv:2001.08361.

[13] J. Hoffmann et al. *Training Compute-Optimal Large Language Models.* NeurIPS, 2022. arXiv:2203.15556.

[14] S. Y. Gadre et al. *Language Models Scale Reliably with Over-training and on Downstream Tasks.* 2024. arXiv:2403.08540.

[15] S. Hu et al. *MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies.* 2024. arXiv:2404.06395.

[16] A. Hägele et al. *Scaling Laws and Compute-Optimal Training Beyond Fixed Training Durations.* NeurIPS, 2024. arXiv:2405.18392.

[17] G. Yang et al. *Tensor Programs V: Tuning Large Neural Networks via Zero-Shot Hyperparameter Transfer.* NeurIPS, 2021. arXiv:2203.03466.

[18] J. Liu et al. *Muon is Scalable for LLM Training.* 2025. arXiv:2502.16982.

[19] S. McCandlish, J. Kaplan, D. Amodei, OpenAI Dota Team. *An Empirical Model of Large-Batch Training.* 2018. arXiv:1812.06162.

[20] S. L. Smith, P.-J. Kindermans, C. Ying, Q. V. Le. *Don't Decay the Learning Rate, Increase the Batch Size.* ICLR, 2018. arXiv:1711.00489.

[21] G. Penedo et al. *The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale.* NeurIPS Datasets & Benchmarks, 2024. arXiv:2406.17557.

[22] J. Li et al. *DataComp-LM: In Search of the Next Generation of Training Sets for Language Models.* NeurIPS Datasets & Benchmarks, 2024. arXiv:2406.11794.

[23] L. Gao et al. *The Pile: An 800GB Dataset of Diverse Text for Language Modeling.* 2020. arXiv:2101.00027.

[24] G. Penedo et al. *The RefinedWeb Dataset for Falcon LLM.* NeurIPS, 2023. arXiv:2306.01116.

[25] L. Soldaini et al. *Dolma: An Open Corpus of Three Trillion Tokens for Language Model Pretraining Research.* ACL, 2024. arXiv:2402.00159.

[26] A. Lozhkov et al. *StarCoder 2 and The Stack v2: The Next Generation.* 2024. arXiv:2402.19173.

[27] K. Lee et al. *Deduplicating Training Data Makes Language Models Better.* ACL, 2022. arXiv:2107.06499.

[28] S. M. Xie et al. *Data Selection for Language Models via Importance Resampling.* NeurIPS, 2023. arXiv:2302.03169.

[29] S. M. Xie et al. *DoReMi: Optimizing Data Mixtures Speeds Up Language Model Pretraining.* NeurIPS, 2023. arXiv:2305.10429.

[30] C. Blakeney et al. *Does Your Data Spark Joy? Performance Gains from Domain Upsampling at the End of Training.* COLM, 2024. arXiv:2406.03476.

[31] G. Hinton, O. Vinyals, J. Dean. *Distilling the Knowledge in a Neural Network.* NeurIPS Deep Learning Workshop, 2015. arXiv:1503.02531.

[32] Y. Kim, A. M. Rush. *Sequence-Level Knowledge Distillation.* EMNLP, 2016. arXiv:1606.07947.

[33] Y. Gu et al. *MiniLLM: On-Policy Distillation of Large Language Models.* ICLR, 2024. arXiv:2306.08543.

[34] J. Wei et al. *Emergent Abilities of Large Language Models.* TMLR, 2022. arXiv:2206.07682.

[35] S. Biderman et al. *Lessons from the Trenches on Reproducible Evaluation of Language Models.* 2024. arXiv:2405.14782.

[36] L. Madaan et al. *Quantifying Variance in Evaluation Benchmarks.* 2024. arXiv:2406.10229.

[37] A. P. Gema et al. *Are We Done with MMLU?* NAACL, 2025. arXiv:2406.04127.

[38] P. Koehn. *Statistical Significance Tests for Machine Translation Evaluation.* EMNLP, 2004.

[39] R. Dror, G. Baumer, S. Shlomov, R. Reichart. *The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing.* ACL, 2018.