Self-Distillation for Data-Scarce Language Model Pretraining

author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2026

Abstract

Language model training is increasingly bottlenecked by data. Naive multi-epoch training quickly overfits, which motivates the search for effective regularization under data scarcity. In this work, we study the effectiveness of self-distillation as a regularization method. Considering eight scarcity levels and two model scales, we find that self-distillation serves as effective regularization when data are scarcer than what Chinchilla prescribes, outperforming both direct training and common regularization methods such as weight decay and exponential moving average. For practical recommendations on self-distillation, we suggest using larger models which are more data efficient despite the greater potential to overfit, and using plain teacher logits without temperature scaling.

1. Introduction

High-quality data are a limiting resource in language model pretraining. The supply is not growing at the rate of compute [19]. Collecting additional data is expensive and slow, and the standard alternative, multi-epoch training on a fixed corpus, overfits to the training set quickly with standard regularizers offering only partial relief [10, 24]. This calls for effective regularization for language model pretraining when data, rather than compute, are the primary bottleneck.

This paper investigates whether self-distillation serves as such an effective regularizer. Self-distillation is the special case of knowledge distillation [8] in which teacher and student have the same architecture, and the student is trained on a weighted combination of the teacher's output and the ground-truth labels. Its regularization effect has been studied theoretically [2, 14, 16, 26] and recently in language models [3, 10, 18]. In particular, [10] shows that when the amount of pretraining data is 1/32 that of Chinchilla, self-distillation leads to lower perplexity compared to multi-epoch training with standard regularization such as weight decay. However, it's unclear how the comparison changes as the scarcity level varies.

In this work, we study self-distillation and characterize the data regimes under which self-distillation improves pretraining under data scarcity. We test self-distillation across eight data-scarcity levels (from Chinchilla-optimal to $1/128$ Chinchilla), two model scales (OLMo-2-13M and OLMo-2-186M), and epoch counts from 1 to 128, a wider sweep over both data scarcity and repeated training than prior compute-controlled work [15]. Our work contrasts prior results on distillation scaling laws [4] which has largely focused on data-abundant regimes where the main question is how to allocate compute between teacher and student. Specifically, we find:

- **Self-distillation outperforms direct training under data scarcity** (Section 3). While direct training leads to better performance when data are abundant, self-distillation wins when the scarcity level goes beyond a model-scale-dependent crossover threshold, and the gap grows with scarcity. Smaller model sees the self-distillation benefit sooner: the crossover is at $1/4$ Chinchilla on OLMo-2-13M and $1/8$ Chinchilla on OLMo-2-186M.
- **Self-distillation outperforms standard regularization baselines** (Section 3.1), specifically weight decay and exponential moving average (EMA) which prior work find effective [2, 10, 23, 24].
- **Self-distillation requires little hyperparameter tuning** (Section 3.2). Defaults ($\tau = 1$, $\alpha = 0.5$) work across the scarcity levels and scales we study, and direct training's tuned learning rate and weight decay transfer to self-distillation without modification. At matched data, self-distillation of OLMo-2-186M outperforms direct training of two smaller OLMo-2 variants.

2. Regularization methods under data scarcity

Let f_θ be a causal, decoder-only language model with N parameters, trained on the next-token prediction objective $\mathcal{L}_{\text{CE}}(\theta)$ on a corpus $\mathcal{D}$ of D tokens. We consider training under *data scarcity*, where scarcity is defined relative to the Chinchilla-optimal token count $D^* = 20N$ [9].

We consider the following four training strategies. When $D \ll D^*$, the most naive approach is multi-epoch (1) **direct training (DT)** which minimizes $\mathcal{L}_{\text{CE}}$ on $\mathcal{D}$ directly. Prior work has found (2) **weight decay (WD)** to be helpful [24], and [10] finds the optimal decay value (denoted λ) to be much larger than what's commonly assumed. We hence consider weight decay as a baseline regularization method and tune λ per scarcity level (Appendix C). Given prior work's positive results on ensemble [2, 10, 23], we consider a temporal ensemble implemented as the (3) **exponential moving average (EMA)** of training checkpoints, $\theta_{\text{ema}} \leftarrow \gamma\theta_{\text{ema}} + (1-\gamma)\theta$, is an implicit regularizer evaluated on the averaged weights; we tune the decay γ per epoch count (Appendix C, Table 6). Finally, we consider (4) **self-distillation (SD)**, which first trains a teacher model by DT on $\mathcal{D}$ and then trains a randomly-initialized same-size student using both the teacher output and the data:

$$\mathcal{L}_{\text{SD}} = \alpha\,\tau^2\,\text{KL}(p_T^\tau \| p_S^\tau) + (1-\alpha)\,\text{CE}(p_S, y), \tag{1}$$

where $p_T^\tau = \text{softmax}(z_T/\tau)$ and $p_S^\tau = \text{softmax}(z_S/\tau)$ apply temperature τ to the teacher and student logits z_T, z_S; y is the ground-truth next-token label; and α is the distillation weight [8]. We default to $\alpha = 0.5$ and $\tau = 1.0$; we ablate τ in Section 3.2 and α in Appendix C.

Setup. We train OLMo-2 [20] at two scales, i.e., OLMo-2-13M and OLMo-2-186M (Table 2), on DCLM [11]. The Chinchilla-optimal count is $D^* \approx 252$M for OLMo-2-13M, $D^* \approx 3.7$B for OLMo-2-186M. We test eight data-scarcity levels from 1 to $1/128$ Chinchilla at both scales, evaluate on a held-out DCLM split, and report bits per byte (BPB). All methods train on the same fixed token pool at each scarcity level. We optimize with AdamW [13] ($\beta_1 = 0.9$, $\beta_2 = 0.95$) using a cosine learning rate schedule with linear warmup over the first 5% of steps and gradient clipping at norm 1.0. The batch size is 64 sequences of 1,024 tokens; all training uses bf16 mixed precision.

3. Self-distillation in the data-scarce regime

We show that self-distillation serves as an effective regularizer, especially under severe data scarcity, outperforming direct training, weight decay, and EMA.

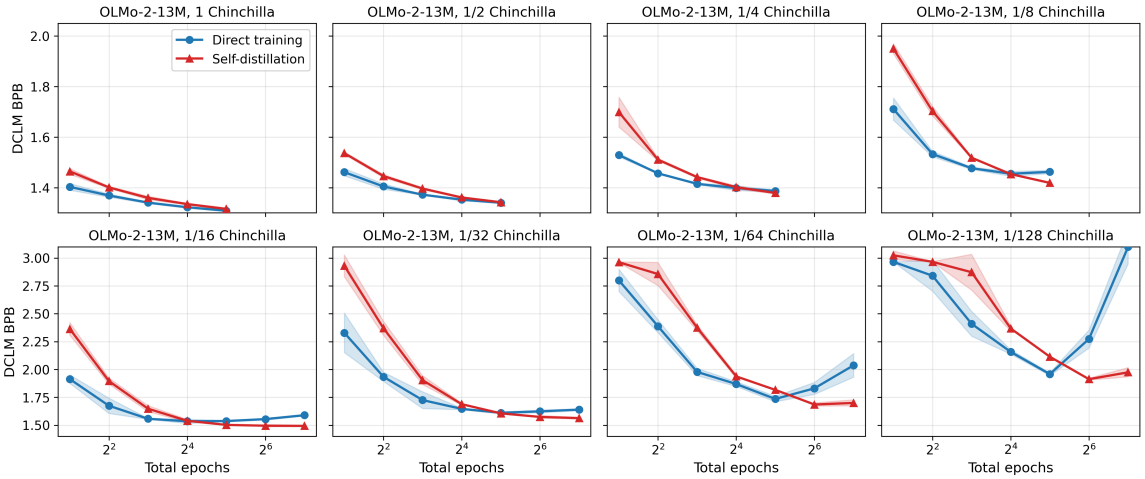

Figure 1: **At data-scarce levels, self-distillation continues to improve at high epoch counts where direct training overfits.** BPB as a function of total training epochs, on OLMo-2-13M across eight scarcity levels from 1 to $1/128$ Chinchilla. For self-distillation, total epochs is the sum of the teacher and student training. Shaded bands: 95% CI across 3 seeds.

Direct training wins with abundant data; self-distillation wins under scarcity. We vary the amount of data-scarcity levels to characterize when self-distillation helps relative to direct training. We sweep eight scarcity levels (1 to $1/128$ Chinchilla) and epoch counts $\{1, 2, 4, 8, 16, 32, 64, 128\}$ on OLMo-2-13M, comparing DT and SD. For self-distillation, the epoch count includes the training of both the teacher and the student. [1]

Self-distillation tolerates more epochs before overfitting (Figure 1): at data-scarce levels, direct training peaks at around 32 total epochs and degrades thereafter; self-distillation continues to improve through 128 epochs. Self-distillation hurts at 1 Chinchilla (consistent with Busbridge et al. [4]) and ties direct training at $1/2$ Chinchilla, but reaches the lowest BPB at every level from $1/4$ Chinchilla downward (Figure 2). Adding EMA narrows but does not close the gap at OLMo-2-13M, $1/32$ Chinchilla (Figure 4); at OLMo-2-186M, EMA tracks direct training to within 0.01 BPB and provides no meaningful benefit (Appendix C.3). Sodhani et al. [18] observed a similar growing benefit of codistillation as the dataset shrinks. These patterns persist when scaling up to OLMo-2-186M (Figure 3): self-distillation reaches the lowest BPB at every level from $1/8$ Chinchilla downward.

3.1. Weight decay and EMA narrow but does not close the gap

Motivated by the view that self-distillation may act as a regularizer, we sweep weight decay to test whether it can explain the observed gains [24]. We sweep λ on OLMo-2-13M at $1/32$ Chinchilla, 128 total epochs (Figure 4). Direct training and self-distillation have different optima – direct training prefers $\lambda = 10$, self-distillation prefers $\lambda = 5$ – and self-distillation has a flatter response across λ.

1. Due to teacher forward pass, self-distillation costs roughly 18% more FLOPs than direct training at matched epochs. The estimation is based on Hoffmann et al. [9]: a training step costs $\sim 6N$ FLOPs per token and an inference step costs $\sim 2N$; at equal teacher and student token counts self-distillation is $\sim 7ND$ vs direct training's $6ND$.

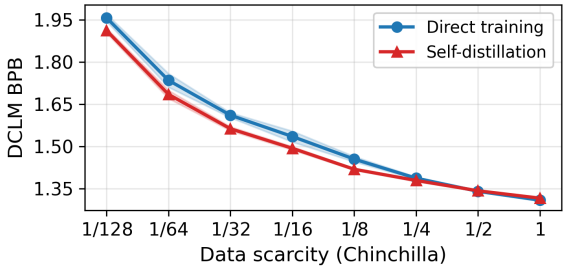

Figure 2: **Self-distillation reaches the lowest BPB at every level $1/4$ Chinchilla and below on OLMo-2-13M.** For each method, we report the best BPB over total epochs $\{2, \ldots, 128\}$, learning rate, and weight decay. Learning rate and weight decay are tuned by coordinate descent per data scarcity level (Appendix C). Shaded bands: 95% CI across 3 seeds.

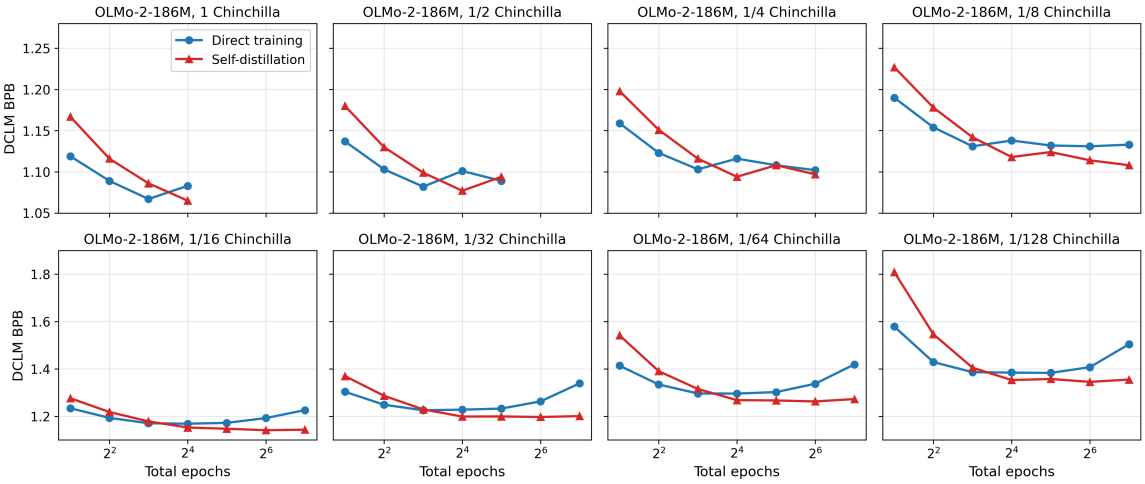

Figure 3: **Self-distillation reaches the lowest BPB from $1/8$ Chinchilla downward on OLMo-2-186M.** The gap grows monotonically with data scarcity.

At their respective optima, self-distillation outperforms direct training by 0.077 BPB. EMA, plotted as a reference line in Figure 4, narrows the gap further but, like weight decay, does not close it.

3.2. Practical recommendations: self-distill rather than shrink, use the natural temperature

Self-distill rather than shrink model size. We ask whether training a smaller model can serve as an alternative to self-distillation for mitigating overfitting. We compare against two smaller models derived from OLMo-2-186M by halving all transformer dimensions once (OLMo-2-43M) and twice (OLMo-2-15M); both configurations are in Appendix B. Each model is trained past its overfitting point. At every matched data budget, direct training of either smaller model is worse than self-distillation of OLMo-2-186M (Table 1); the smaller OLMo-2-15M never overtakes OLMo-2-43M.

Use the natural temperature $\tau = 1$. We consider another natural design choice in self-distillation: adjusting the teacher's temperature τ to control the softness of its targets. Lower τ sharpens the teacher's distribution toward hard targets, whereas higher τ produces a flatter distribution. We sweep $\tau \in \{0.01, 0.1, 0.5, 1.0, 2.0, 5.0\}$ at OLMo-2-13M, $1/32$ Chinchilla, 128 total epochs, $\alpha = 0.5$, $\lambda = 1$ (Figure 5). The teacher's natural temperature ($\tau = 1$) is optimal at 1.599 BPB, and performance

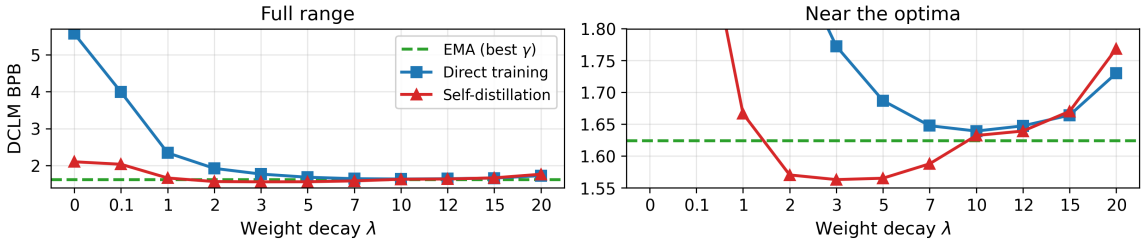

Figure 4: **Weight decay and EMA narrow but do not close the self-distillation gap.** OLMo-2-13M at $1/32$ Chinchilla, 128 total epochs. Direct training and self-distillation have different λ optima (10 vs 5); SD has a flatter response. The EMA reference line is the best BPB across γ at 128 epochs (Table 6). The right panel zooms near the optima.

degrades on both sides. Collapsing the teacher's distribution toward hard targets is dramatically worse: $\tau = 0.1$ reaches 1.910 and $\tau = 0.01$ reaches 1.934, more than 0.3 BPB worse than $\tau = 1$.

Table 1: **Direct training of either smaller model is worse than self-distillation of OLMo-2-186M at every matched data budget.** Best BPB across pass counts swept past overfitting (15M to 512 epochs, 43M to 256, 186M to 128).

	Chinchilla		
Method	1/32	1/64	1/128
OLMo-2-15M DT	1.395	1.425	1.494
OLMo-2-43M DT	1.251	1.319	1.388
OLMo-2-186M DT	1.225	1.296	1.383
OLMo-2-186M SD	**1.197**	**1.263**	**1.345**

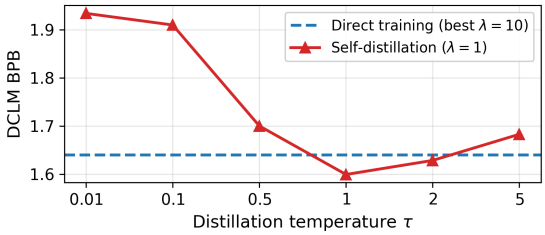

Figure 5: **The teacher's natural temperature ($\tau = 1$) is optimal; collapsing toward hard targets is dramatically worse.** OLMo-2-13M at $1/32$ Chinchilla, 128 total epochs, $\alpha = 0.5$; self-distillation at $\lambda = 1$, direct training at its optimum $\lambda = 10$ (Figure 4).

4. Discussion

In this work, we find that self-distillation is an effective pretraining method under data scarcity. Self-distillation is a strong regularizer and postpones overfitting compared to direct training. Although weight decay and EMA can narrow the gap between self-distillation and direct training, they do not close it. In addition to its regularization benefits, the soft teacher distribution in self-distillation provides a richer source of supervision that may be diminished when training with hard targets. These findings suggest a practical guideline for self-distillation: use self-distillation when operating under a data scarce setting, typically when $D < D^*/4$, with a soft teacher by setting $\tau = 1$ and $\alpha = 0.5$. Overall, these findings motivate further investigation into self-distillation and the mechanisms underlying its success.

References

[1] Rishabh Agarwal, Nino Vieillard, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. *arXiv preprint arXiv:2306.13649*, 2024.

[2] Zeyuan Allen-Zhu and Yuanzhi Li. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. *arXiv preprint arXiv:2012.09816*, 2023.

[3] Rohan Anil, Gabriel Pereyra, Alexandre Passos, Robert Ormandi, George E. Dahl, and Geoffrey E. Hinton. Large scale distributed neural network training through online distillation. *arXiv preprint arXiv: 1804.03235*, 2018.

[4] Dan Busbridge et al. Distillation scaling laws. *arXiv preprint arXiv:2502.08606*, 2025.

[5] Tommaso Furlanello, Zachary C. Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. *arXiv preprint arXiv:1805.04770*, 2018.

[6] Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2024.

[7] Danny Hernandez, Tom Brown, Tom Conerly, Nova DasSarma, Dawn Drain, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Tom Henighan, Tristan Hume, et al. Scaling laws and interpretability of learning from repeated data. *arXiv preprint arXiv:2205.10487*, 2022.

[8] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[9] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *Advances in Neural Information Processing Systems*, 35, 2022.

[10] Tatsunori Kim et al. Pre-training under infinite compute. *arXiv preprint arXiv:2509.14786*, 2025.

[11] Jeffrey Li et al. DataComp-LM: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*, 2024.

[12] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2016.

[13] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2019.

[14] Hossein Mobahi, Mehrdad Farajtabar, and Peter Bartlett. Self-distillation amplifies regularization in hilbert space. *Advances in Neural Information Processing Systems*, 33, 2020.

[15] Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36, 2023.

[16] Minh Pham, Minsu Cho, Ameya Joshi, and Chinmay Hegde. Revisiting self-distillation. *arXiv preprint arXiv:2206.08491*, 2022.

[17] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[18] Shagun Sodhani, Olivier Delalleau, Mahmoud Assran, Koustuv Sinha, Nicolas Ballas, and Michael Rabbat. A closer look at codistillation for distributed training. *arXiv preprint arXiv:2010.02838*, 2020.

[19] Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Will we run out of data? limits of LLM scaling based on human-generated data. *arXiv preprint arXiv:2211.04325*, 2024.

[20] Pete Walsh et al. OLMo 2. *arXiv preprint arXiv:2501.00656*, 2025.

[21] 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. *arXiv preprint arXiv:2212.10560*, 2023.

[22] Kaiyue Wen, David Hall, Tengyu Ma, and Percy Liang. Fantastic pretraining optimizers and where to find them. *arXiv preprint arXiv:2509.02046*, 2025.

[23] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 23965–23998. PMLR, 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

[24] Fuzhao Xue, Yao Fu, Wangchunshu Zhou, Zangwei Zheng, and Yang You. To repeat or not to repeat: Insights from scaling LLM under token-crisis. *arXiv preprint arXiv:2305.13230*, 2023.

[25] Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

[26] Zhilu Zhang and Mert R. Sabuncu. Self-distillation as instance-specific label smoothing. *Advances in Neural Information Processing Systems*, 33, 2020.

Appendix A. Related work

Knowledge distillation. Hinton et al. [8] introduced knowledge distillation, training a student network to match the softened output distribution of a teacher. The soft targets encode information beyond the ground-truth label – relative probabilities across incorrect classes that Hinton et al. [8] term "dark knowledge." Subsequent work has applied distillation to language model compression [6, 17] and addressed the mismatch between teacher and student distributions through on-policy training [1]. These methods distill a large teacher into a smaller student using the teacher's logits. We study same-size self-distillation, also using logits, in a setting where data scarcity rather than model compression is the motivation.

Self-distillation. Furlanello et al. [5] showed that same-size students consistently outperform their teachers on image classification, even when the dark knowledge signal is ablated – suggesting the benefit does not require correctly attributed inter-class similarities. This is distinct from our $\tau \to 0$ result (Section 3.2): collapsing the teacher's distribution toward hard targets removes softness entirely and destroys the benefit. Their language modeling experiments are less conclusive: on Penn Treebank, pure distillation fails, and a cross-entropy term is required.

Three theoretical accounts predict the regime we observe. The *multi-view* account of Allen-Zhu and Li [2] proves that when the data admits multiple equally predictive feature sets, ensembles capture features individual models miss, and the gain transfers through distillation – a structure plausibly present in multi-epoch language model training, where each epoch exposes the model to overlapping but not identical statistical patterns. The *regularization* view [14, 16, 26] casts the teacher's softened distribution as an instance-specific smoothing of the targets that suppresses high-frequency components of the learned function and steers the student toward flatter minima. The *soft-target information* view [8] holds that the teacher's relative probabilities across non-target classes carry similarity information that one-hot labels discard. Our results are consistent with all three; the dramatic collapse at $\tau = 0.01$ (Section 3.2) specifically supports the soft-target information view. All prior analyses use image classification at a small scale.

Self-distillation for language model pretraining. Kim et al. [10] show that self-distillation improves validation loss by sampling synthetic tokens from a trained teacher and continuing training on a mixture of real and generated data, but do not account for the compute spent training the teacher or generating tokens. Busbridge et al. [4] derive scaling laws for compression-style distillation on abundant data and find that the compute spent on the teacher is better allocated to direct training – the regime opposite to ours. Beyond pretraining, privileged-information distillation [12] treats the teacher as carrying signals the student cannot observe at deployment, and instruction-tuning self-improvement methods [21, 25] pair an already-capable instruction-tuned model with rejection-sampling and reward heuristics; neither setup applies to pretraining a language model from scratch. We extend self-distillation to the data-scarce pretraining regime, distilling directly from the teacher's logits rather than generated tokens, and find that self-distillation outperforms direct training in the data-scarce regime, even after accounting for the additional compute.

Data repetition. When training data are limited, models must train for multiple epochs over the same tokens, and performance eventually degrades [15]. Muennighoff et al. [15] show that pretraining language models on the same data for more than four epochs, in their setting, can result

in diminishing returns, and even increase the loss. Furthermore, Hernandez et al. [7] shows that repeating even a small fraction of data can severely damage induction heads and copying mechanisms in large language models. Finally, Xue et al. [24] systematically studies multi-epoch degradation and finds that standard regularization (e.g., weight decay, label smoothing) provides little relief. While they show that only dropout helps, it requires careful tuning as model size increases. In this work, we test whether self-distillation can serve as an alternative regularizer, which does not require careful tuning in a data-constrained setting.

Appendix B. Model and dataset details

Models. All models use the OLMo2 architecture [20]: post-norm RMSNorm after attention and MLP blocks, QK-norm, SwiGLU activations, RoPE positional embeddings, and untied input and output embeddings. We train at two scales (Table 2); the smaller-models comparison in Section 3.2 adds two additional configurations (Table 3). With the full OLMo2 vocabulary (100,278 tokens), embedding parameters would dominate the total at these model sizes; at all scales we use a subset of the OLMo2 tokenizer's BPE merges so the embedding fraction roughly matches OLMo-2-1B ($\sim$28%).

Table 2: **OLMo-2-13M is the development model; OLMo-2-186M is the primary scaling model.**

	OLMo-2-13M	OLMo-2-186M
Hidden dimension	256	1,024
Layers	8	8
Attention heads	8	8
Head dimension	32	128
Intermediate dimension	1,024	4,096
Vocabulary size	8,192	25,088
Context length	1,024	1,024
Non-embedding parameters	8.4M	134M
Total parameters	12.6M	185.6M

Table 3: **Smaller-model configurations used as direct-training baselines against self-distillation of OLMo-2-186M.** Both share the OLMo-2-186M tokenizer (25k vocabulary) and context length (1,024).

	OLMo-2-15M	OLMo-2-43M
Hidden dimension	256	512
Layers	2	4
Attention heads	2	4
Head dimension	128	128
Non-embedding parameters	2.1M	16.8M
Total parameters	14.9M	42.5M

Appendix C. Hyperparameter sweeps

We tune hyperparameters by coordinate descent, following Wen et al. [22]. Starting from default values, we sweep one hyperparameter at a time while holding all others fixed, accepting any value that strictly improves final validation BPB. We cycle through all hyperparameters until no change is made in a full pass. Hyperparameters are optimized for direct training and applied unchanged to self-distillation and EMA.

C.1. OLMo-2-13M

For direct training, we sweep learning rate and weight decay at all eight scarcity levels, and β_2 at one level. We run the coordinate-descent procedure at epoch counts $\{8, 32\}$, skipping high epoch counts at scarcity levels where overfitting does not occur. We additionally sweep at $\{128, 256\}$ epoch counts at the four scarcity levels ($1/16$ to $1/128$) to choose hyperparameters for the high-epoch-count production runs (Table 5).

We find that $\beta_2 = 0.95$ is confirmed at its default. Table 4 reports the chosen direct-training hyperparameters at $\{8, 32\}$ epochs. At 8 epochs, the optimal learning rate increases with data scarcity while weight decay stays low (0.1–0.5). At 32 epochs, the pattern is unchanged at data-rich levels, but for severe scarcity ($1/32$ and below) the learning rate drops to 5×10^{-3} and weight decay rises to 5.0 to prevent overfitting. Beyond 32 epochs the optimal learning rate is stable at 5×10^{-3} and weight decay continues to rise: doubling between 32 and 128 epochs at $1/32$–$1/64$ ($5 \to 10$) and $4\times$ at $1/128$ ($5 \to 20$), then plateauing at $1/128$ between 128 and 256 epochs (Table 5).

Table 4: **At data-scarce levels, the optimal learning rate at 32 epochs is $\sim 6\times$ smaller than at 8 epochs, and weight decay rises by $10\times$.** OLMo-2-13M direct training hyperparameters chosen by coordinate descent; all settings use $\beta_2 = 0.95$. Dashes indicate epoch counts not swept at levels where overfitting does not occur.

	8 epochs		32 epochs	
Scarcity	LR	WD	LR	WD
1 Chinchilla	8×10^{-3}	0.1	—	—
$1/2$	1.6×10^{-2}	0.1	8×10^{-3}	0.1
$1/4$	3.2×10^{-2}	0.1	3.2×10^{-2}	0.1
$1/8$	3.2×10^{-2}	0.1	3.2×10^{-2}	0.1
$1/16$	3.2×10^{-2}	0.5	3.2×10^{-2}	$0.5^{\dagger}$
$1/32$	3.2×10^{-2}	0.5	5×10^{-3}	5.0
$1/64$	3.2×10^{-2}	0.5	5×10^{-3}	5.0
$1/128$	3.2×10^{-2}	0.5	5×10^{-3}	5.0

†At $1/16$ Chinchilla, 32 epochs, the coordinate-descent optimum was LR $= 1.6 \times 10^{-2}$, WD $= 1.0$ (BPB 1.523), marginally better than the reported LR $= 3.2 \times 10^{-2}$, WD $= 0.5$ (BPB 1.541). We report the 8-epoch values since the difference (0.018 BPB) is within run-to-run noise and the entire top-5 of the grid lies within a similar range.

Table 5: **Beyond 32 epochs, the optimal learning rate stabilizes at 5×10^{-3} while weight decay continues to rise.** OLMo-2-13M direct training hyperparameters at high epoch counts, data-scarce levels only.

	128 epochs		256 epochs	
Scarcity	LR	WD	LR	WD
$1/32$	5×10^{-3}	10.0	5×10^{-3}	15.0
$1/64$	5×10^{-3}	10.0	5×10^{-3}	15.0
$1/128$	5×10^{-3}	20.0	5×10^{-3}	20.0

For self-distillation, we sweep the distillation weight $\alpha \in \{0.25, 0.5, 0.75, 0.9\}$ and temperature $\tau \in \{0.5, 1.0, 2.0\}$ at $1/32$, $1/64$, and $1/128$ Chinchilla, 32 total epochs (16 teacher, 16 student). The landscape is flat: the entire 4×3 grid differs by at most 0.05 BPB at $1/32$. Lower α is slightly better than higher α, and $\tau = 1.0$ is optimal or near-optimal at all levels. We use $\alpha = 0.5$ and $\tau = 1.0$ as defaults; we ablate τ at SD's operating point ($1/32$ Chinchilla, 128 total epochs) in Section 3.2.

At $1/32$ Chinchilla, 32 epochs, and the optimal weight decay ($\lambda = 5$), varying attention dropout from 0 to 0.3 changes both direct training and self-distillation by less than 0.02 BPB and leaves the gap unchanged.

For EMA, we sweep the decay $\gamma \in \{0.9, 0.95, 0.98, 0.99, 0.999\}$ at $1/32$ Chinchilla across epoch counts $\{8, 32, 128\}$. The optimal decay increases with epoch count (Table 6): more training steps support slower averaging. We set γ per epoch count accordingly.

Table 6: **The optimal EMA decay γ increases with epoch count: more updates support a slower (staler) average.** OLMo-2-13M at $1/32$ Chinchilla, using direct-training HPs for each epoch count. EMA BPB on DCLM. Bold cells mark the chosen γ per epoch count.

γ	8 epochs	32 epochs	128 epochs
0.9	1.728	1.612	1.692
0.95	**1.693**	1.608	1.698
0.98	1.705	1.608	1.686
0.99	1.710	**1.598**	1.703
0.999	2.476	1.716	**1.630**

Combining EMA with self-distillation does not help. We test whether tracking an EMA of the student's weights during self-distillation provides additional benefit beyond either method alone (Table 7). At OLMo-2-13M, $1/32$ Chinchilla, the EMA copy of the SD student matches SD alone at 32 total epochs (1.634 vs 1.628 BPB, $\gamma = 0.99$) and is 0.028 BPB worse at 128 total epochs (1.594 vs 1.566, $\gamma = 0.999$). EMA and self-distillation appear to address related failure modes; stacking them does not help.

Table 7: **Combining EMA with self-distillation does not help.** OLMo-2-13M at $1/32$ Chinchilla, BPB. SD alone is the raw student; SD+EMA is the EMA copy of the SD student. EMA decay γ chosen per epoch count from Table 6.

Total epochs	SD alone	SD+EMA
32	1.628	1.634
128	1.566	1.594

C.2. OLMo-2-186M

For OLMo-2-186M we run a 3×3 grid search of learning rate against weight decay at the three scarcest levels ($1/32$, $1/64$, $1/128$) at both 8 and 32 epochs. At 8 epochs, the optimal learning rate doubles with each scarcity halving while weight decay stays at 1.0; at 32 epochs, the optimal learning rate is stable at 1.5×10^{-3} and weight decay doubles with each scarcity halving (Table 8).

Table 8: **At data-scarce levels, the optimal learning rate at 32 epochs is $\sim 5\times$ smaller than at 8 epochs, and weight decay rises by up to $10\times$.** OLMo-2-186M direct-training hyperparameters chosen by grid search.

	8 epochs		32 epochs	
Scarcity	LR	WD	LR	WD
$1/32$	2×10^{-3}	1.0	1.5×10^{-3}	2.5
$1/64$	4×10^{-3}	1.0	1.5×10^{-3}	5.0
$1/128$	8×10^{-3}	1.0	1.5×10^{-3}	10.0

C.3. Exponential moving average at OLMo-2-186M

We sweep EMA decay γ at OLMo-2-186M across four data-scarce levels and find no meaningful improvement over direct training (Table 9).

Table 9: **EMA does not help at OLMo-2-186M.** Best BPB across pass counts (1–128) and EMA decay γ; data-scarce levels only.

Scarcity	DT	EMA
$1/16$	1.169	1.168
$1/32$	1.225	1.225
$1/64$	1.296	1.294
$1/128$	1.383	1.371

Appendix D. Recursive self-distillation

If one round of self-distillation helps, do additional rounds help further? We test recursive self-distillation on OLMo-2-13M: train for one epoch, distill into a new student on the same data, and repeat for up to 28 rounds. The control is direct training over 32 epochs with a single cosine schedule. We test four teacher variants at each round: (1) the most recent model, (2) the round-0 model as a fixed anchor, (3) a uniform ensemble of all prior rounds, and (4) an EMA across rounds (decay 0.5). Each variant is tested with and without re-initializing the student at each round.

Direct training dominates all recursive variants at both scarcity levels tested ($1/32$ and $1/128$ Chinchilla, Table 10). With warm-start (student continues from the previous round's weights), the latest-model variant improves steadily over rounds but does not catch direct training: at $1/32$, the best warm-start recursive variant reaches 1.91 BPB after 28 rounds versus direct training's 1.61 at 32 epochs; at $1/128$, the best warm-start recursive variant reaches 2.58 BPB versus direct training's 1.96. The continuous cosine schedule of direct training is more effective than resetting it each round. Re-initializing the student at each round eliminates all benefit – performance stagnates (2.6–2.9 BPB at $1/32$ and 2.96–3.09 at $1/128$) and the ensemble and EMA reinit variants diverge from their minimum across later rounds.

Table 10: **Direct training dominates all recursive self-distillation variants at both scarcity levels; re-initializing the student between rounds eliminates all benefit.** OLMo-2-13M after 28 rounds of recursive self-distillation, BPB. Each round trains the student for 1 epoch using a teacher derived from prior rounds; "warm-start" continues from the previous round's student weights, "reinit" starts each round from a fresh random initialization. Direct training over 32 epochs is shown for reference.

	$1/32$ Chinchilla		$1/128$ Chinchilla	
Teacher used at each round	Warm-start	Reinit	Warm-start	Reinit
Direct training (32 epochs)	1.611		1.958	
Latest model (previous round)	1.906	2.693	2.614	2.965
Round-0 anchor (frozen teacher)	2.154	2.690	2.588	2.965
Ensemble of all prior rounds	2.137	2.848	2.582	3.090
EMA across rounds ($\gamma = 0.5$)	2.007	2.860	2.634	3.051

Recursive self-distillation's failure is consistent with theoretical predictions of diminishing returns from iterative self-distillation [14]. It also reflects a practical limitation: each round resets the learning rate schedule, thereby forfeiting the long-horizon optimization afforded by a single continuous schedule.