Bridging Compute- and Data-Optimal Pretraining

Tian Qin[*] Harvard University

Kimia Hamidieh[*] MIT CSAIL

David Alvarez-Melis Harvard University

Abstract

Classical compute-optimal scaling laws assume an unbounded supply of fresh pretraining data, yet pretraining is increasingly entering the regime where compute is growing faster than high-quality data. We propose *Compute-Data (CD) scaling laws*, a unified framework that bridges *compute-optimal* scaling, in which data scales freely with compute, and *data-optimal* scaling, in which the corpus is fixed and compute can grow unbounded. CD scaling extends classic scaling by introducing a *token effectiveness function* η that quantifies how much a *derived* token, produced for instance by multi-epoch repetition or paraphrasing, is worth relative to a fresh one, ranging from a perfect substitute to no value at all. Fitting η for two data expansion strategies (multi-epoch repetition, paraphrasing) across model sizes ranging from 14M to 600M parameters on the Dolma-3 corpus, we find that it is far from constant: it depends jointly on model size, the tokens-per-parameter ratio, and the amount of derived data, and saturates as the corpus is expanded. The functional form of the token effectivness function η implies that substituting compute for data *diminishes* with both model size and data availability, and partitions training into three operational regimes: compute-bound, data-bound, and model-bound, showing that classic compute-optimal allocation is suboptimal across most of the practically relevant regime.

1 Introduction

The classical scaling laws (Hoffmann et al., 2022; Kaplan et al., 2020) characterize how pretraining loss decreases with model size N and dataset size D, prescribing a compute-optimal allocation between the two. Implicit in this formulation is a *data-abundance assumption*: fresh tokens are freely available, and the pretraining corpus D scales naturally with compute. But this assumption is becoming inaccurate. Compute continues to grow at exponential rates (Sevilla et al., 2022), while high-quality pretraining data remains finite and costly to curate (Villalobos et al., 2024). Pretraining is therefore entering a regime that classical scaling laws were not designed for: one in which D is bounded and additional compute must be spent on *derived tokens* D' produced from D via multi-epoch repetition, paraphrasing, or distillation. The central question becomes:

For a model of size N trained on a fixed corpus D, how much can additional compute reduce loss?

We approach this question through the lens of two limiting regimes. **Compute-optimal scaling** $\mathcal{L}^{\text{Chin}}$ describes the setting where data is unbounded and compute is the bottleneck, and it is the regime characterized by classical scaling laws. **Data-optimal scaling** $\mathcal{L}^{D}$ describes the opposite limit, where D is fixed and compute is unbounded and it is the asymptote of what additional training, in any form, can achieve on a given corpus D. Modern pretraining sits between these two limits (Figure 1*a*).

We propose **Compute-Data (CD) scaling laws**, a unified framework that bridges the compute- and data-optimal scaling. At its center is an *effectiveness function* $\eta^{\text{strat}} \in [0, 1]$ that quantifies, for a given data expansion strategy such as paraphrasing or repetition, how much a derived token D' is

[*]Equal contributions. Correspondence to `tqin@g.harvard.edu`, `hamidieh@mit.edu`

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

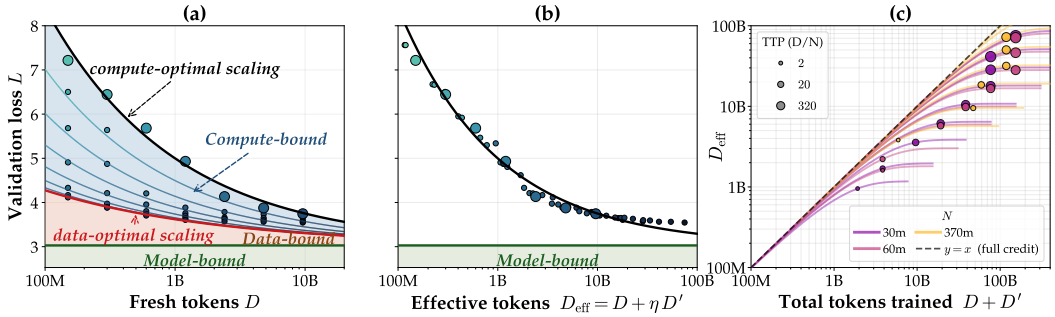


Figure 1: **CD-scaling law overview.** **(a)** For a model of size N, classical scaling laws assume 1-epoch training on fresh data, which coincides with the compute-optimal Pareto frontier. Spending additional compute on *derived* tokens D' produced from D, via data expansion strategies such as multi-epoch repetition, paraphrasing, or distillation, departs from this frontier and reaches lower loss without curating more fresh data. **(b)** To describe training behavior beyond the compute-optimal frontier, CD-scaling laws introduce a token effectiveness function η that maps each derived token to a fresh-data equivalent. Under this mapping, runs beyond the compute-optimal frontier collapse onto the classical scaling line. **(c)** Both the saturation level of effective tokens and the rate of approach depend on N and TPP: as the model grows and data becomes more abundant, additional compute brings diminishing returns.

worth relative to a fresh one (Figure 1*b*). The resulting scaling law is

$$\mathcal{L}(N, D, D') = E + \frac{A}{N^{\alpha}} + \frac{B}{(D + \eta \cdot D')^{\beta}},$$

with $\eta = 1$ meaning derived tokens are as informative as fresh data and $\eta = 0$ meaning they provide no benefit. The law continuously interpolates between the two regimes: when $D' = 0$ it recovers compute-optimal scaling, and as $D' \to \infty$ it approaches the data-optimal limit, predicting the best achievable loss for a given (N, D).

We focus on two data-expansion strategies, multi-epoch repetition and paraphrasing. To fit CD-scaling laws, we sweep model sizes N from 14M to 600M parameters, fresh-data sizes D from 30M to 30B tokens, and derived-token budgets D' from 30M to 120B tokens, with hyperparameter search over learning rate and weight decay. We find that the effective-token count $D_{\text{eff}} = D + \eta D'$ saturates as $D' \to \infty$, with both the *saturation level* and *the rate of saturation* depending on N and the tokens-per-parameter ratio $\text{TPP} = D/N$ (Figure 1*c*). This observation allows us to determine the functional form of the effectiveness coefficient η, which has a power-law decay over N, D and D'. The data-constrained scaling law of Muennighoff et al. (2025) is recovered as the special case in which the saturation level is constant in (N, D).

The fitted η^{strat} yields three practical implications. First, between the compute-optimal scaling $\mathcal{L}^{\text{Chin}}$ and data-optimal scaling $\mathcal{L}^{D}$, the proposed CD scaling law identifies three regimes: **compute-bound**, **data-bound**, and **model-bound**. This informs a practitioner which resource is currently limiting their run. Second, for a given (N, D), it prescribes the optimal compute to spend on D', and generalizes Chinchilla's compute-only optimization into a *joint* compute–data Pareto frontier. Third, CD-scaling prescribes *which* expansion strategy to use and *how much* additional data to train on. Concretely, extrapolated CD-scaling law shows that the 4-epoch rule holds only for medium-scale models ($\sim$ 3B) near $1\times$ Chinchilla, with recommended epochs decreasing as N and TPP grow. Paraphrasing is preferred for small models ($\lesssim$ 600M) and sub-Chinchilla data budgets, but is ineffective for $N \geq$ 7B or large budgets ($\geq 4\times$ Chinchilla).

In summary, our contributions are:

- **A unified CD-scaling framework.** We propose CD-scaling laws, which bridge the compute-optimal and data-optimal regimes through an effectiveness function η^{strat}.
- **Effectiveness of derived tokens diminishes with scale.** Fitting η on Dolma-3 across model sizes from 14M to 600M, we find that the effectiveness of spending additional compute beyond 1-epoch training decays with both N and TPP.

- **Three practical implications.** (a) The CD law identifies three regimes—compute-, data-, and model-bound—that inform which resource is currently limiting the pretraining run. (b) For a given (N, D), it prescribes the optimal compute–data allocation, generalizing Chinchilla's compute-only optimization into a joint compute–data Pareto frontier. (c) It prescribes when to switch from repetition to paraphrasing, and provides recommendation on optimal epochs or paraphrasing based on training configurations.

2 Related Work

Scaling laws. The classical scaling laws of Kaplan et al. (2020) and Hoffmann et al. (2022) characterize loss as a function of model size N and *fresh* pretraining tokens D, prescribing a compute-optimal allocation under the assumption that D scales freely with compute. Subsequent work has investigated each side of this assumption. Gadre et al. (2024) show that the Chinchilla recipe extrapolates reliably into the substantially over-trained regime (TPP $\gg$ 20). Muennighoff et al. (2025) extend Chinchilla to the data-constrained regime, fitting a single saturation budget for repeated tokens. Yan et al. (2025) provide a complementary theoretical perspective, proving in linear regression that the effective reuse rate of repeated data grows with model size N, with $\Theta(\log N)$ scaling under strong convexity and power-law scaling under Zipf-distributed features. Kim et al. (2025) take the opposite limit, asking what loss is achievable under fixed D and unbounded compute. Concurrent to our work, Lovelace et al. (2026) also extend Chinchilla to repeated data, but model repetition through an additive, model-size-dependent overfitting penalty that captures the regime where loss *rises* with further epochs, rather than through an effectiveness function on derived tokens. Our work unifies these threads by recovering Hoffmann et al. (2022) as the $D' = 0$ limit, the law of Muennighoff et al. (2025) as the constant-saturation special case, and the data-optimal limit of Kim et al. (2025) as $D' \to \infty$.

Synthetic data for pretraining. A growing body of work treats synthetic data as a lever for trading compute against fresh-token scarcity. Maini et al. (2024) show that rephrasing web documents into structured styles accelerates pretraining by $\sim 3\times$, and DatologyAI et al. (2025) extend the recipe to trillion-token regimes. Kang et al. (2025) conduct a controlled study across synthetic-data types, finding that mixing $\sim 30\%$ rephrased data yields 5–10$\times$ training speedups. Askari-Hemmat et al. (2025) dynamically target synthetic data at the model's current knowledge gaps; Yang et al. (2025) learn inter-document relations to generate diverse synthetic continuations. Each of these works characterizes a single strategy in isolation. We instead fit a common functional form for the token-effectiveness function η across multi-epoch repetition and paraphrasing, and treat each strategy as a parameterization of the same underlying law.

Distillation and other compute-for-data exchanges. Distillation (Hinton et al., 2015) has seen renewed interest in LLM pretraining as a way to extract additional signal from a fixed corpus. Kim et al. (2025) show that single-student distillation from a multi-epoch teacher recovers most of the loss improvement of the ensemble. Busbridge et al. (2025) establish a teacher–student scaling law that allocates compute optimally between training the teacher and the student. These works model distillation as an internal allocation of compute under a fixed corpus; in our framework, self-distilled tokens are a natural instance of D' that the same η function describes, although our empirical fits in this paper focus on multi-epoch repetition and paraphrasing.

Data allocation and mixing. A complementary line of work allocates compute across the *composition* of the fresh corpus rather than between fresh and derived tokens: DoReMi (Xie et al., 2023), RegMix (Liu et al., 2025), and the data-mixing scaling laws of Ye et al. (2024) optimize mixture weights across pretraining domains, and mixture dependent scaling laws of Hamidieh et al. (2026) predict model performance by accounting for data synergy. CD-scaling is orthogonal: we hold the fresh-corpus composition fixed and study how compute should be split between fresh D and derived D'.

3 Methodology and Experiment Setup

3.1 Compute-Data Scaling Law

The classical Chinchilla scaling law is

$$\mathcal{L}^{\text{Chin}}(N, D) \;=\; E + \frac{A}{N^{\alpha}} + \frac{B}{D^{\beta}}, \tag{1}$$

where N is the number of model parameters and D is the number of fresh pretraining tokens, each seen once. To accommodate synthetic and augmented data, we extend Eq. (1) into a *compute–data* (CD) scaling law that shares the constants (E, A, B, α, β):

$$\mathcal{L}^{\text{CD}}(N, D, D') = E + \frac{A}{N^{\alpha}} + \frac{B}{\left(D + \eta^{\text{strat}}(N, \text{TPP}, D'/D) \cdot D'\right)^{\beta}}, \tag{2}$$

where D' denotes additional tokens produced from a data-expansion strategy (multi-epoch repetition, paraphrasing, or other forms of synthetic data generation and augmentation methods). The *effectiveness coefficient* $\eta \in [0, 1]$ predicts how much one derived token in D' is worth relative to a fresh token from D. In Section 4.1, we motivate parameterizing η in terms of the three more interpretable quantities $(N, \text{TPP}, D'/D)$ rather than (N, D, D'), and in Section 4.2, we derive a parametric form for η^{strat}.

Eq. (2) has three natural limits. Setting $D' = 0$ recovers the classic scaling law $\mathcal{L}^{\text{Chin}}$. Taking $D \to \infty$ sends the third term to zero, leaving the model-bound floor $\mathcal{L}^{N} = E + A/N^{\alpha}$, the lowest loss a model of size N can reach under unlimited fresh data. Finally, and most importantly, taking $D' \to \infty$ at fixed D defines *data-optimal scaling* $\mathcal{L}^{D}$: the lowest loss attainable on a fixed corpus D when unlimited compute is spent on derived tokens. The exact form of $\mathcal{L}^{D}$ depends on η and is derived in Section 4.2.

3.2 Experiment Setup

Data, model, and training. We use the OLMo3 (Olmo et al., 2025) architecture and training infrastructure for all pretraining runs. We train models ranging from $N = 14$M to $N = 600$M parameters, with fresh data D ranging from 30M to 30B tokens and derived tokens D' from 30M to 120B. Fresh tokens are sampled from the Dolma-3 150B corpus (Olmo et al., 2025)[2], with smaller D always a strict subset of larger D. All models are trained at sequence length 4096, batch size 512, with AdamW (Loshchilov and Hutter, 2019) and cosine learning-rate decay. We sweep learning rate $\in \{1\text{e-}4, 3\text{e-}4, 1\text{e-}3, 3\text{e-}3\}$ and weight decay $\in \{0.1, 0.2, 0.4, 0.8, 1.6\}$ at $N \in \{30\text{M}, 370\text{M}\}$, selecting the configuration that minimizes validation loss; for other model sizes we perform a local grid search around the selected configuration. Our primary metric is validation loss on a held-out set of 5M Dolma-3 documents.

Data expansion strategies. We extend pretraining compute beyond 1-epoch training on D via two data-expansion strategies, each producing additional tokens D':

1. **Multi-epoch** (Muennighoff et al., 2025): repeat the original corpus for additional epochs.
2. **Paraphrasing (i.e., augmentation)** (Maini et al., 2024; Kang et al., 2025): rephrase each document in D using SmolLM2-1.7B-Instruct as the paraphrasing model (Allal et al., 2025), sampling up to 16 paraphrases per document with the style for each seed drawn uniformly from {question, math (or wiki when not applicable), FAQ, table}, a four-style mixture identified as most effective in prior analysis (Niklaus et al., 2026). Models are then trained on a mixture of fresh D and paraphrased D' for one epoch. Paraphrased documents are always derived from documents in D, and smaller paraphrased corpora are nested in larger ones.

We do not include the FLOPs spent on paraphrase generation in training compute, as paraphrasing can be performed asynchronously and is not part of the training loop.

4 Parametric Form of CD-Scaling Law

We first characterize η empirically (Section 4.1), then we use those observations to motivate a functional form (Section 4.2), and fit the resulting closed-form (Section 4.3). We validate that

[2] https://huggingface.co/datasets/allenai/dolma3_mix-150B-1025

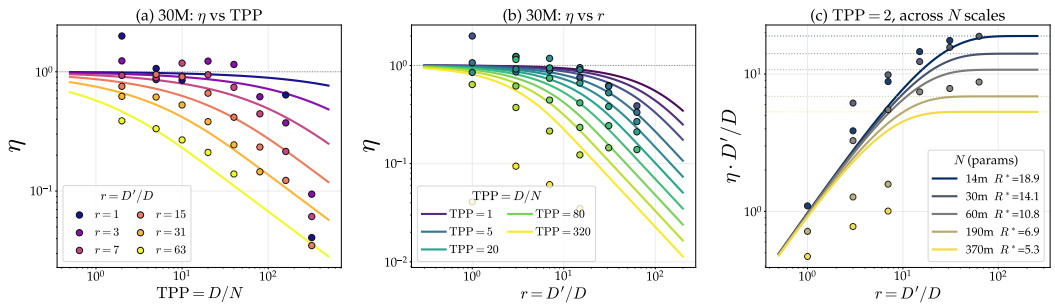


Figure 2: **Empirical characterization of** η (y-axis is η_{emp} throughout). *(a)* η vs. TPP at fixed D'/D: η decreases monotonically with TPP, so the portion of fresh data in the pretraining set matters. *(b)* η vs. D'/D at fixed (N, D): monotone decay; together with (c), this is decay *toward a finite saturation ceiling*, not decay to zero. *(c)* $\eta \cdot D'/D$ vs. r at fixed (N, D): extra effective tokens *plateau* as D' grows. The plateau is a fundamental property of η (and does not arise from under-fitting): no matter how many derived tokens we add, the loss curve flattens, and any functional form for which $\eta \cdot D'$ diverges with D' is ruled out. We also observe that larger models saturate faster, consistent with findings in Section 5.1.

the fit extrapolates to held-out model sizes and show ablations on the parametric form choice (Section 4.4). Finally, we confirm that validation loss improvements transfer to downstream benchmark performance (Section B.3) across training settings.

4.1 Empirical Characterization of η

Before specifying a parametric form for η^{strat}, we measure it empirically and examine how it varies with (N, D, D'). To obtain η_{emp}, we first fit the classic scaling law (Eq. (1)) to obtain constants (E, A, B, α, β). For each repetition and paraphrasing run, we then define the *empirical* coefficient η_{emp} as the unique value that, substituted into Eq. (2) alongside these constants, reproduces the observed loss $\mathcal{L}$. We expect $\eta_{\text{emp}} \in [0, 1]$, though it can exceed 1 when the 1-epoch fit underestimates a run's loss, as any such residual is absorbed into η_{emp}. We discuss this limitation in Section 6.

Change of variables. Although η^{strat} is naturally a function of (N, D, D'), we recast it in terms of three more interpretable variables: the tokens-per-parameter ratio $\text{TPP} = D/N$, the expansion ratio $r = D'/D$, and the model size N. TPP measures fresh-data availability relative to model capacity, and is the conventional unit in which Chinchilla scaling is described. The expansion ratio r measures how aggressively D' expands beyond D. In the multi-epoch case, r is simply the number of extra epochs. We retain N as a separate variable so that TPP captures data availability and N captures model size alone.

Empirical observations. In Figure 2, we examine how η_{emp} depends on (r, TPP, N). Panel *(a)* shows that η_{emp} decays with TPP, and the decay steepens as r grows. Panel *(b)* shows that η_{emp} decays with r, and the decay rate depends on TPP: slowly when TPP is small, sharply when TPP is large. The quantity that matters for total loss, however, is not η but the effective derived data $\eta \cdot D'$, or $\eta \cdot r$ once normalized by D. This quantity represents the fresh-equivalent tokens the derived corpus D' contributes. Rather than growing without bound, $\lim_{r\to\infty} \eta \cdot r$ should saturate. We visualize the saturation behavior in panel *(c)*. Both the saturation limit and the rate of saturation depend on N, with larger models saturating at a lower value and at a faster rate.

4.2 Functional Form of η

The empirical observations impose three requirements: (i) $\eta_{\text{emp}} \to 1$ as $r \to 0$, (ii) $\eta \cdot r$ saturates to a finite limit as $r \to \infty$, and (iii) the saturation behavior depends on TPP and N. Among the functional families satisfying the first two conditions, we adopt the exponential form below, and Section 4.4 shows it provides the best fit when compared to all other forms considered:

$$\eta = \frac{R^*}{r}\left(1 - e^{-r/R^*}\right) \tag{3}$$

Table 1: **Fitted CD-Scaling parameters with 95% CIs.** *Top:* 1-epoch Chinchilla parameters, *Bottom:* per-strategy CD-Scaling parameters. RMSE on $\log \mathcal{L}$, reported separately for 1-epoch, repetition and paraphrasing training runs.

E	A	B	α	β	RMSE_{1ep}
1.35 [1.0, 1.6]	205 [90, 676]	16,597 [7,677, 34,475]	0.283 [0.22, 0.37]	0.435 [0.40, 0.47]	0.043

	$\log K$	ρ	σ	RMSE
Repetition	10.93 [9.7, 13.5]	−0.42 [−0.71, −0.15]	−0.41 [−0.55, −0.35]	0.035
Paraphrase	30.50 [30.4, 30.8]	−1.52 [−1.71, −1.22]	−1.30 [−1.35, −1.26]	0.024

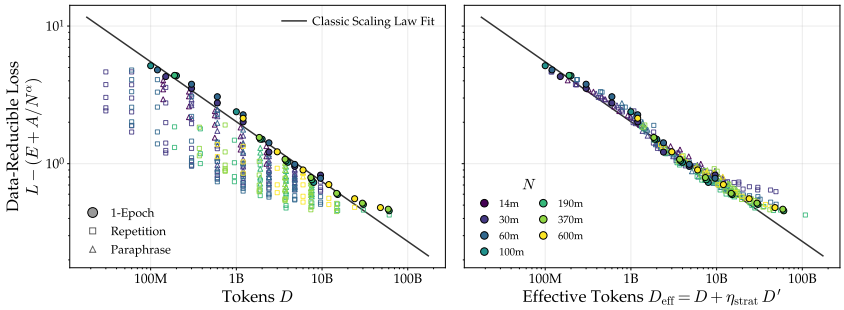


Figure 3: **Fitted CD-scaling law across model sizes and strategies.** Data-reducible loss $\mathcal{L} - (E + A/N^\alpha)$ across model sizes and training settings (1-epoch, repetition, and paraphrase). The solid line is the classic scaling-law fit on 1-epoch runs. *Left:* Data-reducible loss against *fresh* tokens D. Repetition and paraphrase runs train on extra D' tokens, pushing their loss below what fresh D alone achieves, so these runs fall beneath the classic scaling law. *Right:* The CD-scaling law converts each D' into its fresh-token equivalent $\eta_{\text{strat}} D'$. Re-plotted against the effective token count $D_{\text{eff}} = D + \eta_{\text{strat}} D'$, the paraphrase and repetition runs reduce back to the classic scaling law.

Interpretation of R^*. The exponential form gives the identity $\lim_{r\to\infty} \eta \cdot r = R^*$, so R^* can be interpreted as the *saturation ceiling*: the maximum number of fresh-equivalent tokens a corpus of size D can yield through data expansion, expressed as a multiple of D. This ceiling implies that total effective data saturates at $\lim_{D'\to\infty} D_{\text{eff}} = D(1 + R^*)$. Applying this identity to Eq. (2) yields a closed form for data-optimal scaling $\mathcal{L}^D$, the lowest loss a given data-expansion strategy can reach on a fixed corpus of size D:

$$\mathcal{L}^D(N, D) \;=\; E + \frac{A}{N^\alpha} + \frac{B}{\left[D\,\left(1 + R^*\,(D, N)\right)\right]^\beta}. \tag{4}$$

Functional form of R^*. Based on observations in Section 4.1, we propose a power law for R^*:

$$R^*(D, N) \;=\; K \cdot (D/N)^\rho \cdot N^\sigma, \tag{5}$$

where ρ governs how R^* tightens with TPP and σ how R^* tightens with N. We overlay the proposed form in Figure 2, and confirm that the power law fits the observed qualitative behavior.

Together, Eqs. (3) and (5) determine the token effectiveness function η. Substituting η into Eq. (2) gives the complete CD-scaling law, containing eight constants: the five 1-epoch Chinchilla parameters (E, A, B, α, β) and three strategy-specific parameters (K, ρ, σ). The repeated-data law of Muennighoff et al. (2025) is the special case in which R^* is constant and independent of (D, N). Ablations in Section 4.4 (Table 3) confirm that this constant-R^* assumption fits worse, and that the (D, N)-dependence in Eq. (5) is what makes the law predictive across model sizes and data-availability regimes.

4.3 Fitting CD-Scaling Law

We follow the fitting procedure introduced by Besiroglu et al. (2024). Namely, we use a Huber loss on the residuals of $\log \mathcal{L}$. We jointly optimize all parameters: the 5 parameters of the 1-epoch scaling

law (E, A, B, α, β), together with 3 per-strategy saturation parameters (K, ρ, σ) for repetition and paraphrasing. We provide the detailed fitting procedure in Appendix A and report the fitted parameters in Table 1.

To assess fit quality, we separately report the RMSE of $\log \mathcal{L}$ for 1-epoch, repetition, and paraphrasing runs. We also use 95% confidence intervals (CIs) to quantify uncertainty in the fitted parameters. Specifically, we resample training runs with replacement and refit the parameters on the resampled data. We repeat this resampling and refitting procedure 200 times and take the 95% confidence interval. We report RMSE and per-parameter confidence intervals in Table 1 as well.

4.4 Cross-Scale Validation and Ablations

Cross-scale validation. A practical motivation for fitting Eq. (3) is to predict loss for the expensive large-model runs from cheaper small-model runs. We now examine whether the CD-scaling law can extrapolate beyond the model sizes used in fitting. To test this, we refit Eq. (3) on runs at $N \in \{14, 30\}$M only, then predict held-out losses at $N \in \{60, 100, 190, 370, 600\}$M. The held-out set spans model sizes $2\times$ to $20\times$ larger than any used in fitting, TPP up to $\sim$160 ($8\times$ Chinchilla), and r up to 63 (64 epochs), which covers token-count regimes well outside the fitting range. Additionally, we sweep on model sizes used for fitting in Appendix B.1.

The small-N fit achieves RMSE 0.079 on $\log \mathcal{L}$ on the held-out points, against an in-sample RMSE of 0.048. Our scaling law therefore transfers cleanly across an order of magnitude in model size, and a practitioner can fit Eq. (3) on small models with modest budget and read off $R^*(D, N)$ at production scale to anticipate how much additional data will reduce loss.

Ablations. We validate the exponential functional form Eq. (3) and the R^* form Eq. (5) using multi-epoch training runs, reporting leave-one-out (LOO) RMSE on $\log \mathcal{L}$ under the same fitting procedure.

For the functional form of η, we compare nine candidates (Appendix A.3, Table 3). The forms that let R^* depend on (D, N) via Eq. (5) dominate the rest at every parameter count; in particular, the constant-R^* data-repetition baseline (Muennighoff et al., 2025) is rejected by $\sim$40% in relative LOO RMSE. Within the top group, our chosen form is statistically indistinguishable from the best, and we prefer it because it generalizes the constant-R^* form and satisfies $\eta(0) = 1$ exactly.

For R^*, we remove terms from Eq. (5) and refit (Appendix A.4, Table 4). Dropping either exponent degrades the fit, and dropping both recovers the constant-R^* baseline: each exponent is individually necessary, and using both is strictly better than either alone.

Loss-to-Downstream Transfer So far we have compared validation loss across training settings. In Appendix Section B.3 we show that now validation loss remains a valid predictor of downstream capability across data-expansion strategies and model scales.

5 CD-Scaling Laws Bridge Compute- and Data-Optimal Training

In this section, we discuss three implications of CD-scaling laws. First, for a fixed N, the CD law predicts the compute-data Pareto allocation between D and D' (Section 5.1). Second, it partitions training into three regimes, identifying which resource is the binding constraint for a given (N, D, D') (Section 5.2). Third, it tells a practitioner which data-expansion strategy to adopt and how much compute to spend before returns saturate (Section 5.3).

5.1 Compute-Data Pareto-Optimal Allocation

A practitioner training a model of size N on a fixed corpus D must decide how much compute to spend on further loss reduction. Under Chinchilla scaling, compute is fixed by (N, D) via $C \approx 6ND$, leaving no such freedom. Decoupling C from D through multi-epoch training or paraphrasing adds a degree of freedom, giving a two-dimensional loss surface over (C, D) with $C \approx 6N(D + D')$. Figure 4 *(left)* shows this empirical surface for $N = 30$M, with compute C on the x-axis and fresh data D on the y-axis. The black dashed line traces single-epoch runs, where the surface collapses onto a single ray. The contours around single-epoch training are far steeper in C

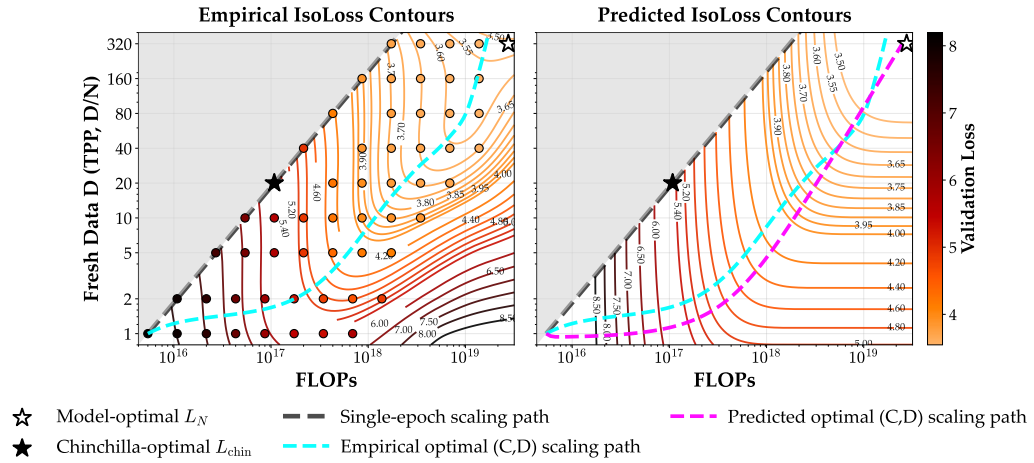


Figure 4: **Optimal compute-data allocation on (C, D) loss surfaces.** *Left:* 1-epoch compute-optimal training scales compute C proportionally with fresh data D, reducing the (C, D) loss surface into a single ray (*black*). Loss contour lines around it indicates that small increase in C would significantly lower the loss. Tracing the sharpest direction of descent, the empirical optimal scaling path (*teal*) prefers scaling compute at data-scarce regime, then scaling both data and compute. *Right:* Our proposed CD-scaling law predicts the optimal path that jointly optimizes (C, D).

than in D, indicating that modest compute beyond the single-epoch buys substantial loss reductions that single-epoch scaling leaves on the table.

The empirical Pareto frontier is non-trivial. The teal curve traces the loss-minimizing path through (C, D) space via steepest descent on the empirical surface. Along it, the contours bend sharply in both directions, meaning increasing C or D *alone* would leave loss roughly unchanged. Since neither resource can reduce loss in isolation, this line traces the compute-data Pareto frontier. This frontier gives practitioners a general guidance on resource allocation. In the data-scarce, low-compute regime (bottom-left), increasing compute is far more effective than adding fresh data, even when data is scarce. As C grows, the frontier rotates toward scaling C and D in equal proportion, then enters a basin where fresh data dominates and further compute yields diminishing returns.

CD-scaling predicts this frontier. Tracing the steepest-descent path on the CD-scaling loss surface (Figure 4 *right*) yields a trajectory that closely tracks the empirical Pareto curve, generalizing Chinchilla's compute-only optimization into a joint compute-data optimization.

5.2 Three Training Regimes

As fresh data become the binding constraint, instead of plotting validation loss against training compute, we plot against fresh data, with color indicating the compute budget $C \approx 6N(D + D')$, shown in Figure 5. The three limits of CD-scaling: $\mathcal{L}^{\text{Chin}}$ ($D' = 0$), $\mathcal{L}^{D}$ ($D' \to \infty$), and $\mathcal{L}^{N}$ ($D \to \infty$) appear as three boundary curves that carve the ($\mathcal{L}$, D) plane into three distinct regions:

- **Compute-bound** (between $\mathcal{L}^{\text{Chin}}$ and $\mathcal{L}^{D}$): for a given D, additional D' still reduces loss. Compute is the binding resource.
- **Data-bound** (between $\mathcal{L}^{D}$ and $\mathcal{L}^{N}$): D' has saturated to R^*D, only additional data can further reduce loss.
- **Model-bound** (at $\mathcal{L}^{N}$): both compute and data have saturated. N is the binding resource.

We then overlay the three region boundaries predicted by CD-law against empirical training losses. Single-epoch points lie on $\mathcal{L}^{\text{Chin}}$. Intermediate-$D'$ traces countour lines in the compute-bound region. Importantly, the predicted data-optimal frontier $\mathcal{L}^{D}$ coincides with the observed $D' \to \infty$ asymptote of the training runs, validating that the CD-scaling's saturation limit correctly models the empirical limit.

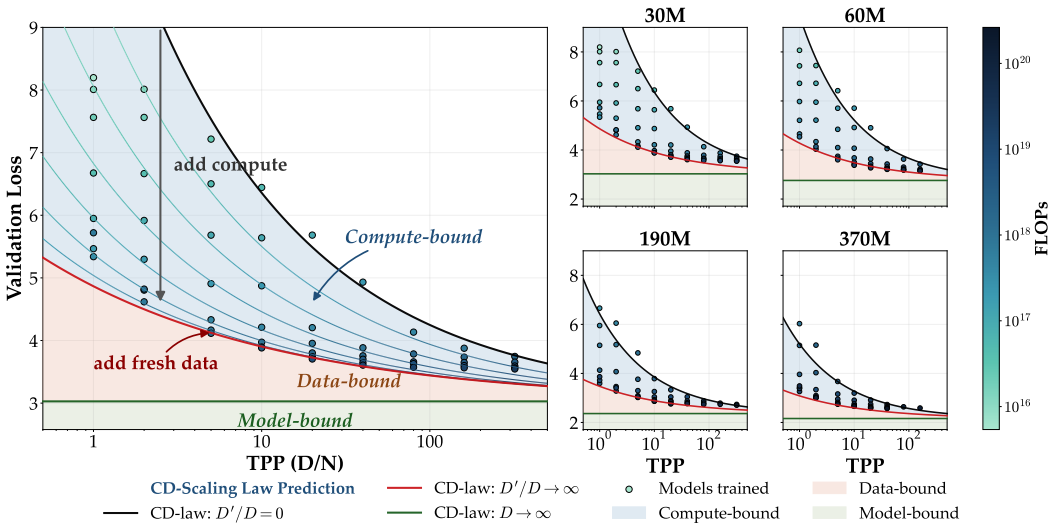


Figure 5: **The CD-scaling law captures three training regimes.** *Left*: $N = 30$M empirical losses overlaid with CD-scaling law predictions. By sweeping D' from 0 ($\mathcal{L}^{\text{Chin}}$, compute-optimal limit) to ∞ ($\mathcal{L}^{D}$, data-optimal limit), CD-scaling law traces out all achievable losses from $\mathcal{L}^{\text{Chin}}$ to $\mathcal{L}^{D}$. *Right*: The same decomposition across model sizes N . As N grows, the compute-bound region bounded by $\mathcal{L}^{\text{Chin}}$ and $\mathcal{L}^{D}$ tightens, indicating that for larger models, increasing compute C yields diminishing returns in further loss reduction.

The compute-bound region shrinks with both TPP **and** N**.** The vertical gap between $\mathcal{L}^{\text{Chin}}$ and $\mathcal{L}^{D}$ is governed by the saturation ceiling R^*. When R^* is large, the data-optimal frontier sits well below the compute-optimal one. In this training regime, additional training compute spent on D' delivers substantial loss reductions. In contrast, when R^* is small, the two frontiers nearly coincide and additional compute spent on D' is largely ineffective. In Section 4.1, we empirically observed that R^* shrinks with both N and TPP. In Figure 5, we confirm that the fitted CD-scaling law correctly models the shrinking rate over TPP *(left)*, and N *(right)*.

5.3 Practical Implications

By visualizing saturation ceiling R^* predicted by CD-scaling laws, we can directly understand at different N and TPP, which of the two data expansion strategy to adopt and the best achievable outcome, as shown in Figure 6 (*top*). Figure 6 (*bottom*) then reports the training effort needed to reach 90% of such R^*'s. Namely, we report the total epochs for repetition runs and total paraphrasing passes for paraphrase runs. We provide three takeaways:

- The recommended epoch count **decreases** as model size and data budget **grow**. Extrapolating the fitted law to large model scales indicates that 4-epoch training is recommended only for medium-scale models ($\sim$ 3B) around $1\times$ Chinchilla data scale.
- For smaller models and data budgets, paraphrasing is preferred. Concretely, paraphrasing beats repetition for models $\lesssim$ 600M and small data budgets ($< 1\times$ Chinchilla).
- Extrapolating the fitted law to large model scales indicates that paraphrasing becomes ineffective for $N \geq$ 7B and large data budgets ($\geq 4\times$ Chinchilla).

6 Conclusion & Discussion

We establish a unified framework that bridges compute- and data-optimal training. Through the effectiveness function η^{strat}, we characterize how much additional compute can reduce loss on a fixed pretraining corpus D. Building on it, we derive three practical implications: the compute-data Pareto-optimal allocation between D and D', the three-regime picture (compute-bound, data-bound, and model-bound) that identifies the binding constraint, and guidance on which expansion strategy to

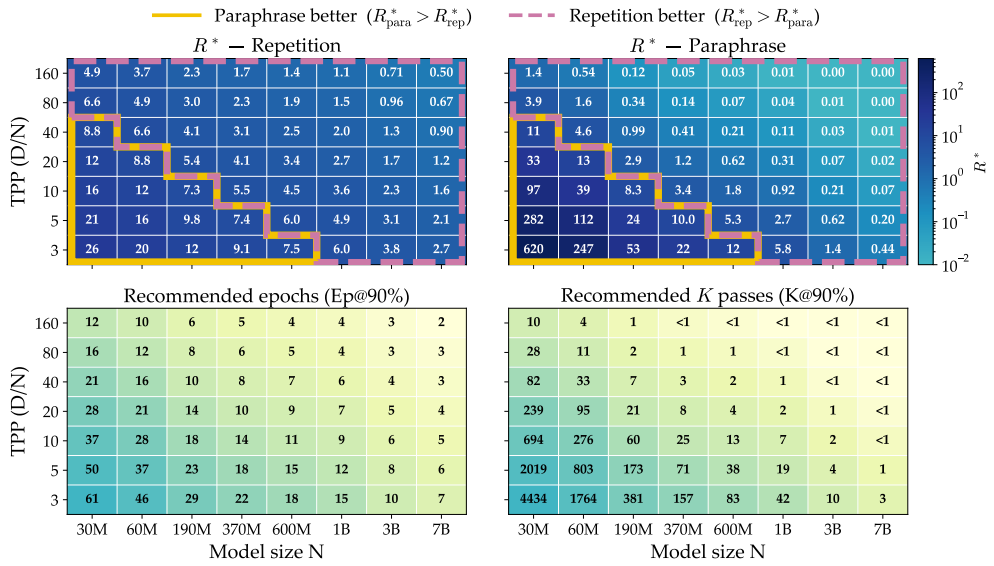


Figure 6: **When to repeat, when to paraphrase, and how much.** *Top:* Saturation ceiling R^* across N and TPP for repetition (*left*) and paraphrase (*right*). The boundary marks where the winning strategy switches. Paraphrase wins in the lower-left (small N, low data budget), repetition wins in the upper-right. *Bottom:* Training compute to reach 90% of R^*. We provide recommended epochs (*left*) for repetition and passes K (*right*) for paraphrase. Both decrease as N and TPP grow.

adopt and how much compute to spend before returns saturate. Together these give practitioners a principled basis for deciding when to invest in compute, fresh data, or model capacity.

Predicting η from corpus statistics. We currently fit η^{strat} empirically per strategy. A natural extension is to predict η from dataset statistics alone, using summary statistics such as token diversity, n-gram overlap, or perplexity distributions. Such a predictive theory would let practitioners assess an expansion strategy on a given corpus before committing compute, making the framework substantially more practical.

Limitations. The functional form we adopt has a clean interpretation, but η absorbs any residual misspecification of the 1-epoch fit (and can push η_{emp} above 1), so the overall fit is sensitive to the quality of the 1-epoch runs. Within each run, we swept only learning rate and weight decay. Other hyperparameters (batch size, sequence length, schedule) were held fixed and could in principle interact with η. Establishing these laws incurs substantiation computation cost. To complete training runs reported in this study, we have consumed over 250k H100-hours. Even with this compute budget, we could only fit scaling laws for two data expansion strategy on small-to-medium scale model sizes. Finally, we evaluate two data-expansion strategies (repetition and paraphrase); others such as self-distillation or synthetic structured data (Lee et al., 2026) remain unexplored, as do combinations of strategies that may yield higher effective η than any single one.

Acknowledgments

DAM acknowledges support from the Kempner Institute, FAS Dean's Competitive Fund for Promising Scholarship, Aramont Fellowship Fund, and the NSF AI-SDM Institute (Grant No. IIS-2229881). Additionally, we thank Antonio Torralba for helpful discussions and guidance on this project.

References

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model, 2025. URL https://arxiv.org/abs/2502.02737.

Reyhane Askari-Hemmat, Mohammad Pezeshki, Elvis Dohmatob, Florian Bordes, Pietro Astolfi, Melissa Hall, Jakob Verbeek, Michal Drozdzal, and Adriana Romero-Soriano. Improving the Scaling Laws of Synthetic Data with Deliberate Practice, February 2025. URL http://arxiv.org/abs/2502.15588. arXiv:2502.15588 [cs].

Tamay Besiroglu, Ege Erdil, Matthew Barnett, and Josh You. Chinchilla scaling: A replication attempt. *arXiv preprint arXiv:2404.10102*, 2024.

Dan Busbridge, Amitis Shidani, Floris Weers, Jason Ramapuram, Etai Littwin, and Russ Webb. Distillation Scaling Laws, July 2025. URL http://arxiv.org/abs/2502.08606. arXiv:2502.08606 [cs].

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

DatologyAI, Pratyush Maini, Vineeth Dorna, Parth Doshi, Aldo Carranza, Fan Pan, Jack Urbanek, Paul Burstein, Alex Fang, Alvin Deng, Amro Abbas, Brett Larsen, Cody Blakeney, Charvi Bannur, Christina Baek, Darren Teh, David Schwab, Haakon Mongstad, Haoli Yin, Josh Wills, Kaleigh Mentzer, Luke Merrick, Ricardo Monti, Rishabh Adiga, Siddharth Joshi, Spandan Das, Zhengping Wang, Bogdan Gaza, Ari Morcos, and Matthew Leavitt. BeyondWeb: Lessons from Scaling Synthetic Data for Trillion-scale Pretraining, August 2025. URL http://arxiv.org/abs/2508.10975. arXiv:2508.10975 [cs].

Samir Yitzhak Gadre, Georgios Smyrnis, Vaishaal Shankar, Suchin Gururangan, Mitchell Wortsman, Rulin Shao, Jean Mercat, Alex Fang, Jeffrey Li, Sedrick Keh, Rui Xin, Marianna Nezhurina, Igor Vasiljevic, Jenia Jitsev, Luca Soldaini, Alexandros G. Dimakis, Gabriel Ilharco, Pang Wei Koh, Shuran Song, Thomas Kollar, Yair Carmon, Achal Dave, Reinhard Heckel, Niklas Muennighoff, and Ludwig Schmidt. Language models scale reliably with over-training and on downstream tasks, 2024. URL https://arxiv.org/abs/2403.08540.

Kimia Hamidieh, Lester Mackey, and David Alvarez-Melis. Domain-aware scaling laws uncover data synergy. *arXiv preprint arXiv:2607.11052*, 2026.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, March 2015. URL http://arxiv.org/abs/1503.02531. arXiv:1503.02531 [stat].

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, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models, March 2022. URL http://arxiv.org/abs/2203.15556. arXiv:2203.15556 [cs].

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, 2017.

Feiyang Kang, Newsha Ardalani, Michael Kuchnik, Youssef Emad, Mostafa Elhoushi, Shubhabrata Sengupta, Shang-Wen Li, Ramya Raghavendra, Ruoxi Jia, and Carole-Jean Wu. Demystifying Synthetic Data in LLM Pre-training: A Systematic Study of Scaling Laws, Benefits, and Pitfalls, October 2025. URL http://arxiv.org/abs/2510.01631. arXiv:2510.01631 [cs].

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, January 2020. URL http://arxiv.org/abs/2001.08361. arXiv:2001.08361 [cs].

Konwoo Kim, Suhas Kotha, Percy Liang, and Tatsunori Hashimoto. Pre-training under infinite compute, September 2025. URL http://arxiv.org/abs/2509.14786. arXiv:2509.14786 [cs].

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 785–794, 2017.

Dan Lee, Seungwook Han, Akarsh Kumar, and Pulkit Agrawal. Training language models via neural cellular automata, 2026. URL https://arxiv.org/abs/2603.10055.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. RegMix: Data mixture as regression for language model pre-training. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2407.01492.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Justin Lovelace, Christian Belardi, Srivatsa Kundurthy, Shriya Sudhakar, and Kilian Q. Weinberger. Prescriptive scaling laws for data constrained training, 2026. URL https://arxiv.org/abs/2605.01640.

Pratyush Maini, Skyler Seto, He Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the Web: A Recipe for Compute and Data-Efficient Language Modeling, January 2024. URL http://arxiv.org/abs/2401.16380. arXiv:2401.16380 [cs].

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*, 2018.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling Data-Constrained Language Models, June 2025. URL http://arxiv.org/abs/2305.16264. arXiv:2305.16264 [cs].

Joel Niklaus, Guilherme Penedo, Hynek Kydlicek, Elie Bakouch, Lewis Tunstall, Ed Beeching, Thibaud Frere, Colin Raffel, Leandro von Werra, and Thomas Wolf. The synthetic data playbook: Generating trillions of the finest tokens, 2026.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, December 2025. URL http://arxiv.org/abs/2512.13961. arXiv:2512.13961 [cs].

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*, 2016.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124.

Jaime Sevilla, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos. Compute trends across three eras of machine learning. *arXiv preprint arXiv:2202.05924*, 2022. URL https://arxiv.org/abs/2202.05924.

Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Position: Will we run out of data? Limits of LLM scaling based on human-generated data. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 49523–49544. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/villalobos24a.html.

Sang Michael Xie, Hieu Pham, Xinyun Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S. Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://arxiv.org/abs/2305.10429.

Tingkai Yan, Haodong Wen, Binghui Li, Kairong Luo, Wenguang Chen, and Kaifeng Lyu. Larger Datasets Can Be Repeated More: A Theoretical Analysis of Multi-Epoch Scaling in Linear Regression, November 2025. URL http://arxiv.org/abs/2511.13421. arXiv:2511.13421 [cs].

Zitong Yang, Aonan Zhang, Hong Liu, Tatsunori Hashimoto, Emmanuel Candès, Chong Wang, and Ruoming Pang. Synthetic bootstrapped pretraining, December 2025. URL http://arxiv.org/abs/2509.15248. arXiv:2509.15248 [cs].

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Yunhua Zhou, Yuhao Zhan, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance, 2024. URL https://arxiv.org/abs/2403.16952.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

A Implementation Details and Ablations for the CD-Scaling Fit

This appendix supplies the implementation details, the outlier-trimming sweep, the functional-form ranking, and the per-term ablation of R^* referenced from Sections 4.3 and 4.4. The main text reports the canonical $k = 15$ row and the chosen exp-sat $R^*(D, N)$ form; the tables below show every cell of both sweeps so the choices can be audited.

A.1 Implementation details

Log-space objective. We work in log-loss space using the numerically stable LSE-equivalent reformulation of Eq. (2) (Hoffmann et al., 2022):

$$\log \mathcal{L}(N, D, D') \;=\; \mathrm{LSE}\big(e,\; a - \alpha \log N,\; b - \beta \log(D + \eta \cdot D')\big), \tag{6}$$

algebraically identical to Eq. (2) with $(E, A, B) = (e^e, e^a, e^b)$ but numerically stable. The fit minimises Huber loss on log-residuals with $\delta = 0.1$, which gives quadratic weight to typical residuals while suppressing the influence of stragglers an order of magnitude larger.

Optimiser and initialisation. L-BFGS with strong-Wolfe line search. We initialize from a log-spaced grid over the parameters, find the best in-sample seed, and report the optimum. Across 30+ random seeds the optimum is reproducible to four significant figures.

Joint fit. The headline pipeline (Section 4.3) fits all eleven parameters (E, A, B, α, β) + $(\log K, \rho, \sigma)_{\text{rep}}$ + $(\log K, \rho, \sigma)_{\text{para}}$ jointly on the pooled 1-epoch + repetition + paraphrase data via a single Huber LSE with iterative residual trimming. The initialization grid brackets both signs of ρ_{para} and σ_{para} so the optimizer can land on either sign. Across every grid we tried, the optimum lands in the same negative-$(\rho_{\text{para}}, \sigma_{\text{para}})$ basin.

Iterative residual trimming. A handful of noisy small-scale points can dominate the fit, so we drop them using an iterative variant of the residual-based trim of Besiroglu et al. (2024): fit, drop the single worst residual, refit, drop the next worst, and so on for k steps. Applied to the untrimmed fit, the standard Hampel outlier rule (**?**) identifies 14 points as outliers. We round to $k = 15$ for consistency with our sweep grid (Table 2), which trims $\sim$4% of the data. The trimmed points are all small-scale 1-epoch and repetition runs at 14M and 30M, the same class of outliers identified by Besiroglu et al. (2024) for Chinchilla itself.

A.2 Outlier-trimming sweep

Table 2: **Outlier-trimming sweep for the joint fit.** Iterative residual trimming on the pooled ($n = 356$) fit residuals. Per-source RMSEs are on kept points. The canonical $k = 15$ row (bold) is justified in Section A.1.

k	n kept	β	$\log K_{\text{rep}}$	σ_{rep}	$\log K_{\text{para}}$	σ_{para}	1-ep RMSE	rep RMSE	para RMSE
0	356	0.298	15.92	−0.63	31.57	−1.34	0.076	0.046	0.027
5	351	0.371	11.85	−0.44	31.29	−1.32	0.062	0.042	0.024
10	346	0.420	11.15	−0.41	31.08	−1.31	0.052	0.038	0.023
15	**341**	**0.437**	**10.96**	**−0.41**	**31.02**	**−1.33**	**0.043**	**0.035**	**0.024**
20	336	0.456	10.82	−0.42	31.02	−1.34	0.039	0.031	0.024
25	331	0.472	10.71	−0.42	31.03	−1.35	0.035	0.029	0.022
30	326	0.482	10.64	−0.42	31.03	−1.36	0.030	0.027	0.020

A.3 Functional-form ranking for η

We compared nine candidate forms for η on the multi-epoch corpus, holding the backbone (B, β) fixed at the Stage-1 anchors and scoring each form by leave-one-out (LOO) RMSE on $\log \mathcal{L}$ (Table 3). The three forms with a full $R^*(D, N)$ via Eq. (5) (sat, exp-sat, tanh) dominate the rest at every parameter count. We adopt exp-sat $R^*(D, N)$ (close 2nd in LOO behind tanh, $\Delta = 0.001$) because it is the canonical data-repetition form (Muennighoff et al., 2025), has $\eta(0) = 1$ exactly, and approaches the R^* asymptote monotonically from below.

Table 3: **Functional-form ranking for η.** LOO RMSE on $\log \mathcal{L}$ over 108 multi-epoch points (kept after pooled residual trimming), with the backbone (B, β) frozen at the Stage-1 anchors. The bottom three rows use the full $R^*(D, N)$ ansatz from Eq. (5) ($\log R^* = \log K + \rho \log(D/N) + \sigma \log N$). Constant-$R^*$ (single-parameter Muennighoff baseline, $R^* = K$) is rejected by roughly 40% relative LOO at the same parameter count, confirming that the (D, N)-dependence of R^* is load-bearing. We adopt exp-sat $R^*(D, N)$ (row in bold) as our functional form.

Form	shape of $\eta \cdot r$	n_{par}	LOO RMSE
constant η	$c\,r$	1	0.035
power in r	$c\,r \cdot r^{-\gamma}$	2	0.032
sat in r	$c\,r/(1 + b\,r)$	2	0.029
exp-decay, $R(D/N)$	$\eta_0\, r \cdot e^{-r/R}$	3	0.026
sat $\times(D/N)$, $b(N)$	$c(D/N)^{-\gamma} r/(1 + b_0(N/N_0)^{\kappa} r)$	4	0.029
exp-sat, $R^*(D/N)$	$R^*(1 - e^{-r/R^*})$	2	0.026
sat, $R^*(D, N)$	$R^*\, r/(R^* + r)$	3	0.021
exp-sat, $R^*(D, N)$ (ours)	$R^*(1 - e^{-r/R^*})$	**3**	**0.020**
tanh, $R^*(D, N)$	$R^* \tanh(r/R^*)$	3	0.019

A.4 Per-term ablation of Eq. (5)

Holding the exp-sat shape of η fixed, we ablate the terms of Eq. (5) to test the necessity of each (Table 4). Each exponent is individually necessary, and Figure 2(a, c) shows why: the D/N- and N-dependence of η are not absorbable into one another.

Table 4: **Per-term ablation of R^*.** Starting from the original parametric form, we remove terms from $\log R^*$ and report the resulting LOO RMSE. Removing either exponent individually degrades the fit, removing both recovers the constant-R^* (Muennighoff et al., 2025) baseline.

Ablation	$\log R^*$	LOO RMSE
original	$\log K + \rho \log(D/N) + \sigma \log N$	**0.020**
$\hookrightarrow$ removing N^{σ}	$\log K + \rho \log(D/N)$	0.026
$\hookrightarrow$ removing $(D/N)^{\rho}$	$\log K + \sigma \log N$	0.025
$\hookrightarrow$ removing both	$\log K$	0.028

A.5 Cross-scale validation across sizes

We now sweep the fit-set cutoff $N_{\max} \in \{30, 60, 100, 190\}$M and predict all held-out points at $N > N_{\max}$ (Table 6).

Held-out RMSE is stable across cutoffs between 0.05 and 0.08, and comparable to the in-sample RMSE at each cutoff. Once $N_{\max} \geq 60$M, the recovered β is within 0.03 of the full-data value. The CD-scaling law therefore extrapolates robustly across an order of magnitude in model size.

Table 5: **Cross-scale validation across fit-set sizes.** We refit on $N \leq N_{\max}$ and predict all held-out $N > N_{\max}$ points. RMSE is reported on $\log \mathcal{L}$.

$N_{\max}$	Fit sizes	n_{fit}	n_{held}	β	in-sample RMSE	held-out RMSE
30M	14M, 30M	103	204	0.34	0.048	0.079
60M	14M, 30M, 60M	173	134	0.35	0.047	**0.049**
100M	14M, 30M, 60M, 100M	180	127	0.35	0.046	**0.050**
190M	14M, 30M, 60M, 100M, 190M	235	72	0.49	0.054	**0.057**

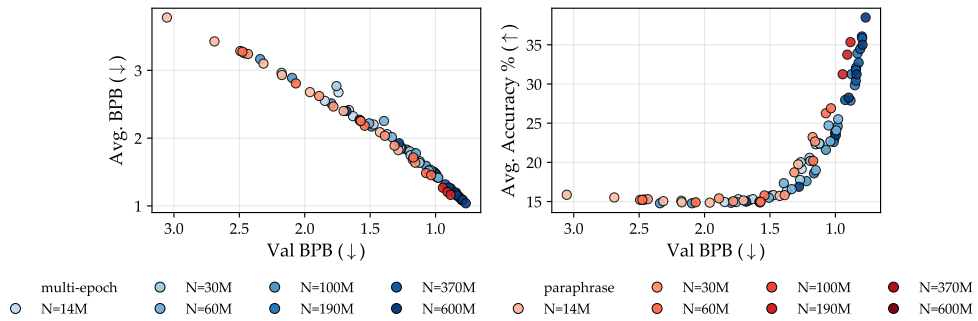


Figure 7: **Downstream performance as a function of validation BPB.** *Left*: average BPB on benchmarks recast as language-modeling tasks (GSM8K, TriviaQA, HumanEval, etc.). *Right*: mean accuracy on LAMBADA, HellaSwag, OpenBookQA, RACE, and SQuAD. In both panels, points from multi-epoch and paraphrase runs collapse onto a single curve, indicating that validation loss is a sufficient statistic for downstream capability regardless of training settings.

B Additional Experiment Details and Results

B.1 Cross-scale validation across sizes

We now sweep the fit-set cutoff $N_{\max} \in \{30, 60, 100, 190\}$M and predict all held-out points at $N > N_{\max}$ (Table 6).

Held-out RMSE is stable across cutoffs between 0.05 and 0.08, and comparable to the in-sample RMSE at each cutoff. Once $N_{\max} \geq 60$M, the recovered β is within 0.03 of the full-data value. The CD-scaling law therefore extrapolates robustly across an order of magnitude in model size.

Table 6: **Cross-scale validation across fit-set sizes.** We refit on $N \leq N_{\max}$ and predict all held-out $N > N_{\max}$ points. RMSE is reported on $\log \mathcal{L}$.

$N_{\max}$	Fit sizes	n_{fit}	n_{held}	β	in-sample RMSE	held-out RMSE
30M	14M, 30M	103	204	0.34	0.048	0.079
60M	14M, 30M, 60M	173	134	0.35	0.047	**0.049**
100M	14M, 30M, 60M, 100M	180	127	0.35	0.046	**0.050**
190M	14M, 30M, 60M, 100M, 190M	235	72	0.49	0.054	**0.057**

B.2 Loss to Downstream Transfer

B.3 Loss-to-Downstream Transfer

So far we have compared validation loss across training settings. We now ask whether validation loss remains a valid predictor of downstream capability across data-expansion strategies and model scales.

We group benchmarks into LM-based and accuracy-based tasks. Benchmarks such as GSM8K (Cobbe et al., 2021), TriviaQA (Joshi et al., 2017), and HumanEval (Chen et al., 2021) yield near-random accuracy at small-to-medium scale. Following Gadre et al. (2024), we recast them as language-modeling tasks and report bits-per-byte (BPB) of the gold response given the prompt, which stays predictive even when accuracy is near-trivial. For accuracy-based tasks, we report mean accuracy across LAMBADA (Paperno et al., 2016), HellaSwag (Zellers et al., 2019), OpenBookQA (Mihaylov et al., 2018), RACE (Lai et al., 2017), and SQuAD (Rajpurkar et al., 2018). We plot both against validation BPB in Figure 7.

In both cases, runs from all strategies and scales fall onto a single curve as a function of validation BPB alone, which confirms that validation loss predicts downstream capability independent of training setting.

In Figure 8, and Figure 9, we report loss-to-downstream for each benchmark.

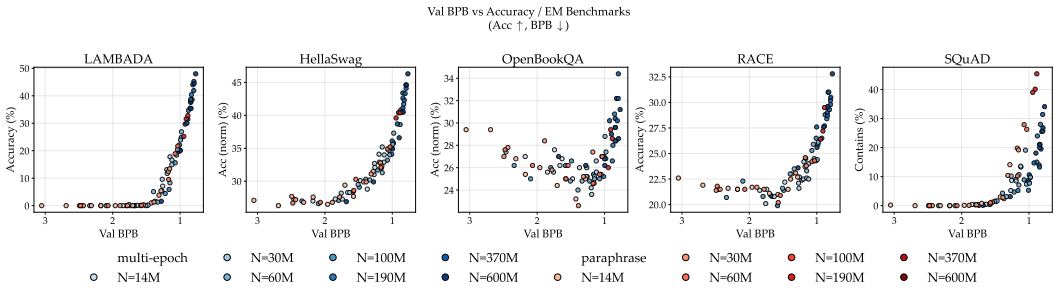


Figure 8: **Per-task breakdown of accuracy-based benchmarks versus validation BPB.** Each subplot shows one task against validation BPB, decomposing the accuracy side of Figure 7. Cool hues mark multi-epoch runs and warm hues mark paraphrase runs, with color denoting model size N (14M to 600M). On every task, runs of all strategies and scales fall onto a single monotonic curve in validation BPB.

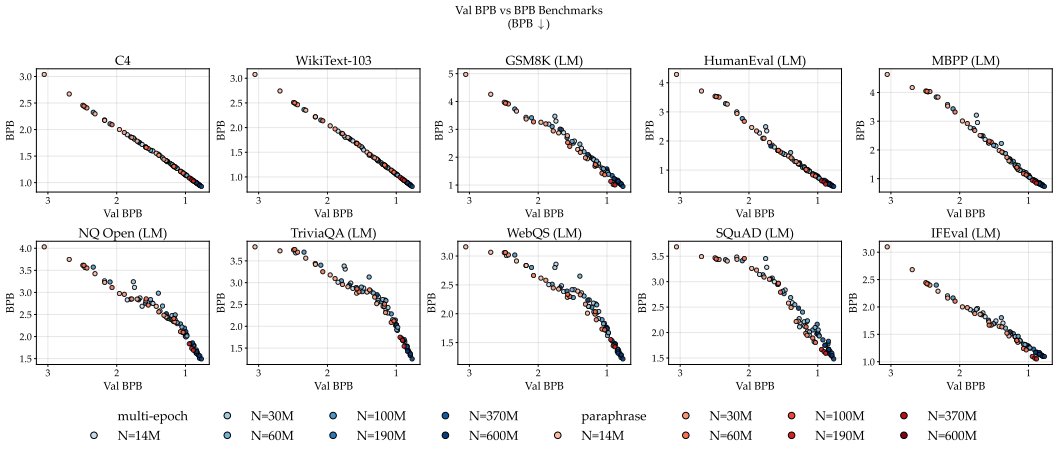


Figure 9: **Per-task breakdown of BPB-based benchmarks versus validation BPB.** Each subplot shows one task (recast as language modeling and scored in BPB) against validation BPB, decomposing the BPB side of Figure 7. Cool hues mark multi-epoch runs and warm hues mark paraphrase runs, with color denoting model size N (14M to 600M). On every task, runs of all strategies and scales fall onto a single monotonic curve in validation BPB.

B.4 Optimal Scaling Path for 370M model

In Figure 10, we visualize the optimal scaling path for 370M model and show that the scaling law path predicted by CD-scaling laws also follow closely with the empirical optimal path.

B.5 Training Run Details

Hyperparameter Sweep. We visualize the hyperparamter sweep grid and validation loss for 30M, 370M models. Lower epoch requires smaller weight-decay and as we increase data repetition, higher weight decay is necessary. For larger epochs, 30M model requires WD as high as 1.6, which is consistent with observations made in (Kim et al., 2025).

Other Hyperparameters. While we sweep learning rate and weight decay for our training runs, we list the rest of hyperparameter in Table 7.

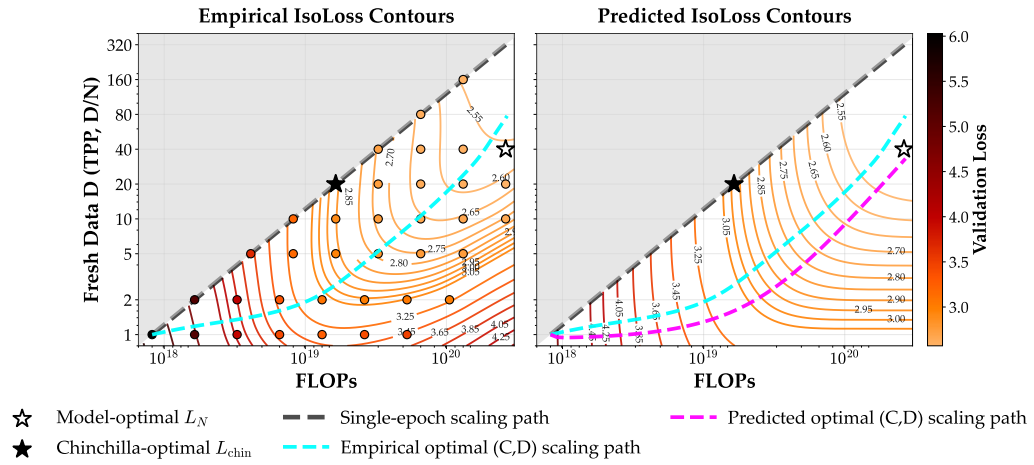


Figure 10: **Optimal compute-data allocation on (C, D) loss surfaces for $N = 370M$.** Replicating results for Figure 4.

Table 7: Pretraining hyperparameters (OLMo2-1B, 50B tokens).

Category	**Hyperparameter**	**Value**
Data	Global batch size (sequences)	512
	Gradient accumulation steps	64
Optimization	Learning rate	tuned
	Optimizer	AdamW
	(β_1, β_2)	$(0.9,\ 0.95)$
	Weight decay	tuned
	Gradient clip	1.0
LR Schedule	Schedule	cosine with warmup
	Warmup tokens	1B
	Min LR ratio (α_f)	0.1
	Units	tokens
Regularization	Precision	BF16 (AMP)
	Softmax auxiliary loss	✓
	Auxiliary loss multiplier	1×10^{-5}

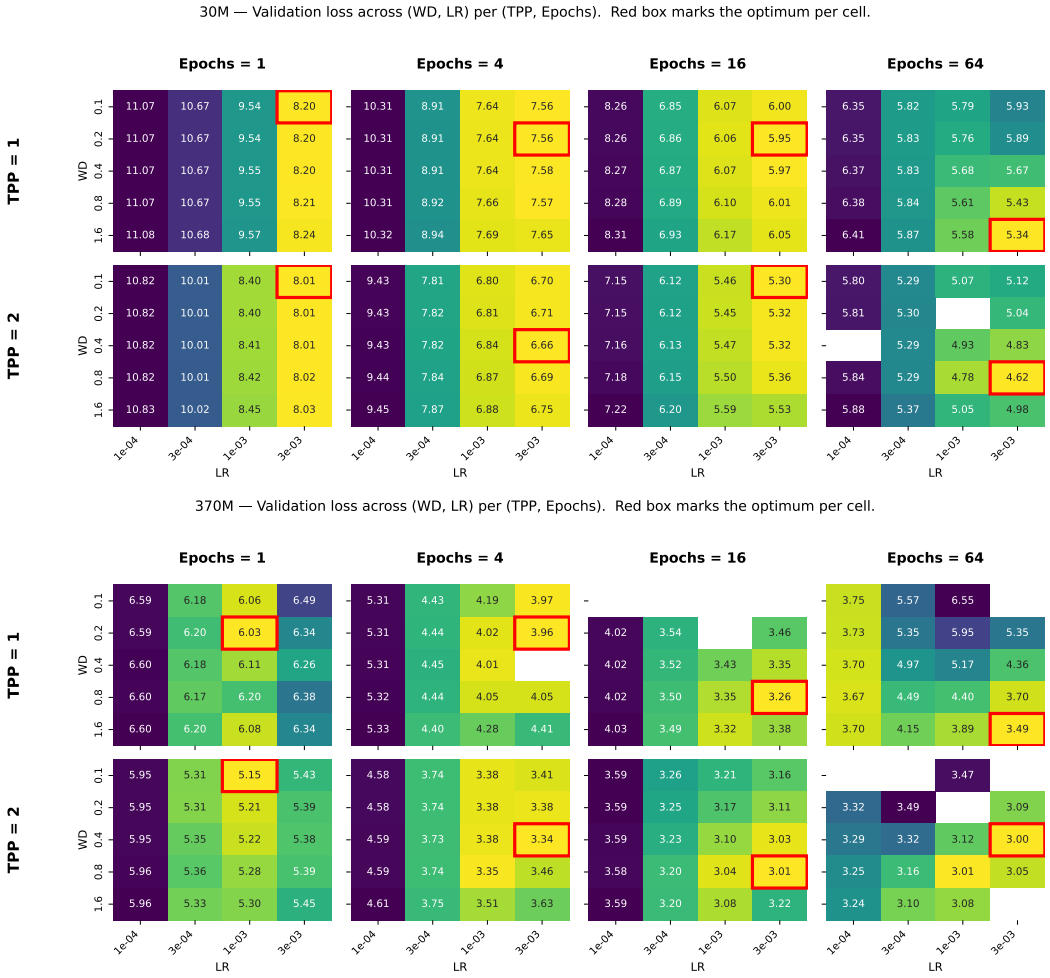


Figure 11: **Hyperparameter grid example for 30M and 370M model.** More data repetition requires much higher weight decay especially for small models.