

M^3 Scaling Law: Optimizing Multi-Epoch, Multi-Lingual, and Multi-Stage Training for Low-Resource Language Models

Kosuke Akimoto and Taiki Miyagawa and Masafumi Oyamada
NEC Corporation

{kosuke_a, miyagawataik, oyamada}@nec.com

Abstract

In this paper, we study a fundamental design problem in pretraining Large Language Models (LLMs) for low-resource language regimes, focusing on cross-family language pairs in which English serves as the high-resource language. Existing works adopt multi-epoch, multi-lingual, and multi-stage training to utilize the limited target-language corpus efficiently, but no prior scaling law can compare recipes spanning these approaches under the same compute budget C and target-language corpus size D_T , leaving the optimal training setup unclear. To address this gap, we propose the M^3 Scaling Law, a unified predictive model parameterized by the model scale, the number of target-corpus epochs k , the average target-language ratio r , and the final-stage target-language ratio r_f , which places monolingual single-stage, multi-lingual single-stage, and multi-lingual multi-stage recipes on a single target-language validation-loss surface. Across three language pairs (English paired with Japanese, Indonesian, and Swahili), it extrapolates to unseen hyperparameter regions more accurately than existing scaling laws. Using M^3 as a surrogate objective, we derive two practical guidelines for low-resource LLM pretraining: (i) as D_T decreases, the optimal recipe shifts directly from monolingual single-stage to multi-lingual two-stage training at a compute-budget-dependent threshold, with multi-lingual single-stage never optimal in our experimental grid; and (ii) the optimal number of epochs collapses onto a single curve in the scarcity variable $D_T/D^*(C)$, where $D^*(C) \propto C^{\alpha/(\alpha+\beta)}$ is the monolingual compute-optimal corpus size.

1 Introduction

In this paper, we study a fundamental design problem in pretraining Large Language Models (LLMs) for low-resource language regimes, focusing on cross-family language pairs in which English serves as the high-resource language and the

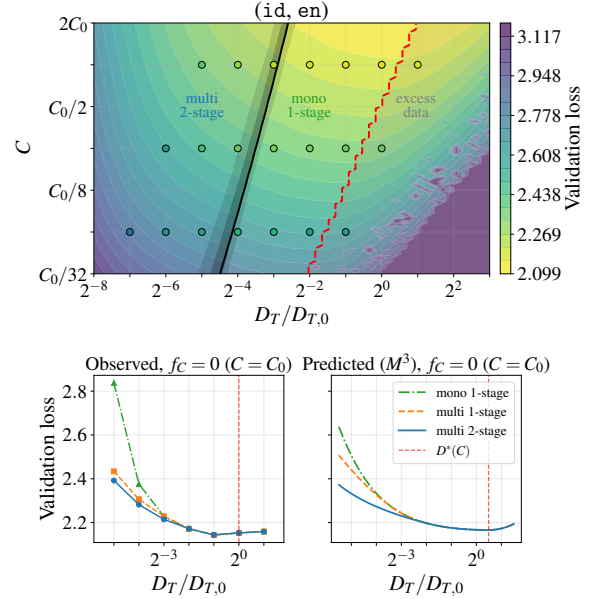


Figure 1: **Top:** Predicted minimum validation loss landscape over (D_T, C) for the (Indonesian, English) pair, with three regime regions separated by the compute-optimal data size $D^*(C)$ (red dashed) and the approach boundary (black solid) at which the optimal training approach switches from mono 1-stage to multi 2-stage as D_T decreases. Circles show observed best loss at each experimental (D_T, C) point, colored by the same scale as the contours. Shading around the approach boundary shows its bootstrap variability (Appendix D.4). **Bottom left:** Observed minimum validation loss for each training approach at $C = C_0$. **Bottom right:** Corresponding predictions by the M^3 Scaling Law.

target language is moderately low-resource. Given a fixed compute budget C and a fixed amount of available target-language corpus D_T , the practitioner faces a training-recipe design space whose axes include the model scale together with the hyperparameters of three well-known approaches: **multi-epoch training** (Muennighoff et al., 2023) repeats the target corpus k times; **multi-lingual training** (Scao et al., 2023) mixes high-resource data with target-language ratio r ; and **multi-stage**

	model scale	#epochs	1st #token	1st lang. ratio	2nd #token	2nd lang. ratio	single- stage	two- stage
(Kaplan et al., 2020)	★	✗	★	✗	—	—	★	✗
(Hoffmann et al., 2022)	★	★*	★	✓*	—	—	★	✗
(Muennighoff et al., 2023)	★	★	★	—	—	—	★	✗
(Bryant and Liu, 2026)	★	✗ [◊]	✓	✓	—	—	✓	✗
(Chang et al., 2024)	★	✗	✗	★	—	—	★	✗
(Faysse et al., 2025)	★	✗	★	★	—	—	★	✗
(He et al., 2025)	★	✗	★	★	—	—	★	✗
(Shukor et al., 2025) [§]	★	★	★	★	—	—	★	✗ [†]
(Longpre et al., 2026)	★	★	★	★	—	—	★	✗
(Sedova et al., 2026)	★	★	★	★	—	—	★	✗
(Ibrahim et al., 2024)	✓	✗	—	—	✓	✓	✗ [△]	✓
(Que et al., 2024) [§]	★	✗	—	—	★	★	✗	★
(Zhang et al., 2024) [§]	★	✗	★	—	★	✗	✗	★
(Liew and Kato, 2026) [§]	★	✗	★	—	★	✗	✗	★
(Seto et al., 2026) [§]	★	✗	★	—	★	★	✗	★
(Goffinet et al., 2025)	★	✗	★	—	★	★	✗	★
(Baek et al., 2026) [§]	✓	✓	★	★	✗ [#]	✗	★	★
(Wang et al., 2025) [§]	★	✗	★	✗	★	★	★	★
Ours	★	★	★ [‡]	★ [‡]	★ [‡]	★	★	★

★: covered by a proposed scaling law. [§]Focusing on adaptation to non-language domains. *Epochs and language ratio impacts studied independently. [◊]Only one number of epochs per corpus budget tested. [△]Only tested one baseline single-stage setup mixing all English and German corpus. [†]Only monolingual single-epoch continual pretraining from a fixed multi-lingual base model; second-stage hyperparameters are not explicitly explored. [#]Finetuning length is implicitly determined by early stopping rather than explicitly analyzed. ^{||}Only one of these parameters is covered at a time. [‡]These variables are not directly included in the proposed M^3 Scaling Law, but are reflected through the average language ratio and total number of tokens.

Table 1: Scope of hyperparameters explored in existing works and our study.

training (Ibrahim et al., 2024) varies r across stages.

Existing scaling laws cannot rank competing recipes under the same (C, D_T) . As shown in Table 1, prior laws cover only restricted slices of this design space: single-stage laws (Hoffmann et al., 2022; Longpre et al., 2026; Sedova et al., 2026) do not extend to multi-stage training, continual pretraining laws (Que et al., 2024; Goffinet et al., 2025; Zhang et al., 2024) lack a common objective with single-stage recipes, and even laws nominally spanning both settings (Baek et al., 2026; Wang et al., 2025) leave orthogonal axes uncovered such as second-stage length or multi-epoch training. As a result, no current law can compare, under the same (C, D_T) , recipes that differ along multiple axes at once, such as whether one should repeat the target corpus, mix high-resource data throughout training, or use high-resource data early and concentrate target-language data in the final stage.

We address this gap with the M^3 **Scaling Law**, a unified predictive model for target-language validation loss parameterized by (M, D_T, k, r, r_f) , where r and r_f are the average and final-stage target-language ratios. As we show empirically, the latter two together summarize the effect of multi-stage schedules. The M^3 Scaling Law maps different training recipes into a single loss-prediction surface through effective model size M' , effective

data size D' , and a dual power-law factor in (r, r_f) . This makes monolingual single-stage, multi-lingual single-stage, and multi-lingual multi-stage recipes directly comparable under fixed (C, D_T) .

We contribute in two layers. First, as empirical modeling components: (i) in multi-stage training, the target-language loss follows a dual power law in (r, r_f) , largely insensitive to intermediate-stage ratios, extending He et al. (2025); (ii) in monolingual multi-epoch training, the effective model size saturates more strongly with k , captured by a k -dependent saturation parameter $R_M^*(k)$ that recovers Chinchilla at $k=1$; (iii) in multi-lingual multi-epoch training, the (k, r) interaction is modeled by a language-ratio-dependent weight on high-resource tokens, extending Longpre et al. (2026).

Second, using the M^3 Scaling Law as a surrogate objective, we then derive two recipe-selection rules¹: (a) as D_T decreases, the optimal approach transitions directly from monolingual single-stage to multi-lingual two-stage at a compute-budget-dependent threshold, with multi-lingual single-stage never optimal in our experimental grid; (b) the optimal number of epochs collapses onto a single curve in $D_T/D^*(C)$, where $D^*(C)$ is the monolingual compute-optimal corpus size.

¹Both rules minimize the predicted target-language validation loss, not downstream task performance.

2 Experimental Setups

2.1 Problem Setup

In this paper, we focus on pretraining LLMs from scratch for a low-resource target language under a fixed compute budget C and a fixed amount of available target-language corpus of size D_T . In addition to the target-language corpus, we assume the availability of a high-resource language corpus.

To investigate the optimal training setup for each budget constraint (C, D_T) , we pretrained LLMs with various training setups for each (C, D_T) , spanning multiple training approaches, namely, multi-epoch, multi-lingual, and multi-stage training. We adopt target-language validation loss on a held-out corpus as our primary metric, following common practice in scaling law research (e.g., [Hoffmann et al., 2022](#)) and prior findings that validation loss correlates with downstream task performance (e.g., [Chen et al., 2025](#)); see Appendix H for a preliminary correlation analysis on Japanese benchmarks.²

In our experiments, we cover combinations of the following three well-known approaches for LLM pretraining for low-resource language: **Multi-epoch training** ([Muennighoff et al., 2023](#)) repeats the target-language corpus k times during training, assuming abundant high-resource data (no repetition). **Multi-lingual training** mixes the target-language corpus with a high-resource subset, with target language ratio $r = kD_T/D$, where D is the total number of training tokens. **Two-stage training** changes target language ratio from r_1 (first stage) to r_2 (second stage).

2.2 Languages and Training Configuration

In our experiments, we used three language pairs with English as the high-resource language: Japanese, Indonesian, and Swahili, all from different language families. Japanese and Indonesian, though not typically low-resource, enable systematic analysis across data regimes through downsampling their corpora. Swahili complements these with verification on actually lower-resource language, although its limited availability restricts experiments to smaller data settings. We used C4 ([Raffel et al., 2020](#)) for English and mC4 ([Xue et al., 2021](#)) for the target languages, both from Common Crawl with consistent preprocessing except for language-specific ones, minimizing the impact

²Due to compute constraints, we ran a single training run per setup.

of data quality variations on our analysis. We respectively used their original train/validation splits with the BLOOM tokenizer ([Scao et al., 2023](#)).

We followed the training configuration of [Bi et al. \(2024\)](#): a decoder-only LLaMA architecture ([Touvron et al., 2023](#)) with context length $S = 4096$, and their compute-dependent power-law schedules for the learning rate, $\eta = 0.3118 \cdot C^{-0.1250}$, and the batch size, $B = 0.2920 \cdot C^{0.3271}$. We adopted these schedules as given and did not search over the learning rate or the batch size ourselves. We measured model scale by non-embedding FLOPs per token $M = 72nd_{\text{model}}^2 + 12nd_{\text{model}}S$, where n , d_{model} , and S are the number of layers, the size of hidden states, and the sequence length, respectively. For two-stage training, we re-warmed the learning rate at the start of the second stage to the first-stage maximum, as [Ibrahim et al. \(2024\)](#) found that re-warming is effective for adaptation to new data in the later stage. See Appendix C for detailed training configurations.

2.3 Hyperparameter Search Space

To systematically explore the search space across multiple orders of magnitude, we parametrized each training setup by integer factors (f_r, f_M, f_k, f_C) defining the language ratio $r = \frac{1}{2^{f_r}}$, model scale $M = \frac{M_0}{2^{f_M}}$, number of epochs $k = 2^{f_k}$, and compute budget $C = 2^{f_C} \cdot C_0$. The resulting target language corpus size is $D_T = 2^{-f_r+f_M-f_k+f_C} \cdot D_{T,0}$, so setups sharing the same (f_C, f_D) with $f_D = -f_r + f_M - f_k + f_C$ have identical (C, D_T) . The reference values $(C_0, D_{T,0}, M_0)$ follow the monolingual compute-optimal scaling law of [Bi et al. \(2024\)](#). See Appendix C.3 for the exact values and the grid of factors explored (Table 5). A setup involves multi-epoch training when $f_k > 0$ and multi-lingual training when $f_r > 0$.

For two-stage training, we additionally specified the stage-wise language ratios (r_1, r_2) with $r_1 < r_2$, and used the average ratio $r = s_1r_1 + s_2r_2$ (which coincides with kD_T/D from the single-stage definition) to compute the stage proportions via $s_1 = \frac{r_2-r}{r_2-r_1}$ and $s_2 = 1 - s_1$. We grid-searched $r_1 \in \{1/2, 1/4, 1/8, 1/16, 1/32, 0\}$ and $r_2 \in \{1/4, 1/2, 3/4, 1\}$, combined with the same (f_r, f_M, f_k, f_C) grid as in single-stage training.

3 Scaling Law Analysis

In this section, we integrate and extend existing scaling laws to propose the M^3 Scaling Law, a unified scaling law that jointly covers multi-epoch, multi-lingual, and multi-stage training.

Notation. Throughout §3, D' and M' denote the effective data and model sizes, whose definitions are progressively extended across subsections (Muennighoff in Eqs. 4–5; ATLAS in Eq. 6; the M^3 Scaling Law in Eqs. 11–12). Within each subsection, D' and M' refer to the most recent definition unless otherwise noted. See Appendix B for a full summary of notation.

3.1 Background

Our approach builds on two existing scaling laws. First, He et al. (2025) showed that in single-stage single-epoch multi-lingual training, the target language validation loss follows a power law with respect to the target language ratio r :

$$L_{\text{He}}(M, D, r) = L_{\text{base}}(M, D) \cdot \mathcal{R}(r), \quad (1)$$

$$L_{\text{base}}(M, D) = \frac{A}{M^\alpha} + \frac{B}{D^\beta} + E, \quad (2)$$

where $D = kD_T/r$ is the total number of training tokens, L_{base} is the Chinchilla scaling law (Hoffmann et al., 2022) with coefficients A , B , exponents α , β , and irreducible loss E , $\mathcal{R}(r) = r^{-\gamma}$ is the language ratio factor, and γ is the power-law exponent.

Second, Muennighoff et al. (2023) proposed the following scaling law for monolingual single-stage multi-epoch training, replacing M and D in L_{base} with effective sizes M' and D' :

$$L_{\text{Mu}}(M', D') = L_{\text{base}}(M', D'), \quad (3)$$

$$D' = D_T \cdot h(R_D; R_D^*), \quad (4)$$

$$M' = U_M \cdot h(R_M; R_M^*), \quad (5)$$

where $h(R; R^*) = 1 + R^* \cdot (1 - \exp(-R/R^*))$ is a saturation function with R^* controlling the rate of saturation. Here, $R_D = k - 1$ is the data repetition count, $U_M = \min(G^{(\alpha+\beta)/\alpha} D_T^{\beta/\alpha}, M)$ with $G = (\alpha A / \beta B)^{1/(\alpha+\beta)}$ is the compute-optimal model size for D_T under single-epoch training, and $R_M = M/U_M - 1$ is the excess model size. Longpre et al. (2026) extended this scaling law to the multi-lingual setting by adding the effective data size of other languages to D' with a constant weight

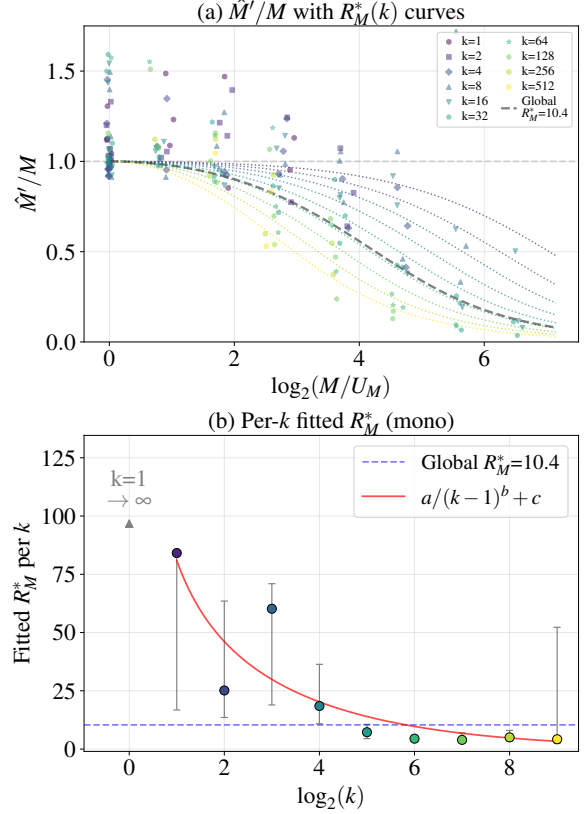


Figure 2: Empirical analysis of effective model size M' on Japanese monolingual data. **(a)** $\hat{M}'/M$ vs. $\log_2(M/U_M)$ with per- k $R_M^*(k)$ curves (dotted, colored by k) and the global constant R_M^* (black dashed). **(b)** Per- k fitted R_M^* , showing a decreasing trend from ∞ at $k = 1$ to a finite floor c ; the red curve shows the power-law fit $a/(k-1)^b + c$. Error bars in (b) show 95% bootstrap confidence intervals (200 resamples).

for each language. In our setup, where the high-resource language is not repeated, this reduces to:

$$D' = D_T \cdot h(R_D; R_D^*) + g \cdot D_{\text{high}}, \quad (6)$$

where $D_{\text{high}} = kD_T(1 - r)/r$ denotes the number of high-resource language tokens used during training, and g is a constant weight.

3.2 Extending Effective Model Size of Muennighoff Model

The Muennighoff form $M' = U_M \cdot h(R_M; R_M^*)$ uses a single R_M^* shared across all cases, but we find that the strength of the saturation in fact depends on number of epochs k . To isolate the behavior of M' , we invert L_{Mu} for each data point and compute estimated effective model size $\hat{M}'$ from the observed loss L_{actual} as $\hat{M}' = (A/(L_{\text{actual}} - B/D'^\beta - E))^{1/\alpha}$ with (A, B, α, β, E) fit and fixed from the monolingual $k \leq 4$ subset and D' computed via h (Eq. 4) with R_D^* obtained by fitting L_{Mu}

to all monolingual data while keeping the above parameters fixed. As shown in Figure 2(a), while $\hat{M}'/M$ has a decreasing trend with M/U_M as the Muennighoff form predicts, the trend also depends on k . For small k , e.g., $k = 1, 2$, $\hat{M}'/M$ shows a small decay, while for larger k , $\hat{M}'/M$ decreases faster at smaller M/U_M . This suggests saturation of M' also depends on k in addition to M/U_M .

Refitting R_M^* independently for each k on the monolingual data confirms this trend (Figure 2(b)). The estimated R_M^* has a trend to decrease and saturate to a finite floor as k increases. We propose to model this behavior with

$$R_M^*(k) = \frac{a}{(k-1)^b} + c, \quad (7)$$

which diverges as $k \rightarrow 1$ and converges to c as $k \rightarrow \infty$.

This form has desirable limiting behavior: as $k \rightarrow 1$, $R_M^* \rightarrow \infty$ and $h(R_M; R_M^*) \rightarrow 1 + R_M = M/U_M$, yielding $M' \rightarrow M$ and recovering Chinchilla at $k = 1$. This fixes an issue of the original Muennighoff form, where a constant R_M^* predicts $M' < M$ whenever $M > U_M$, deviating from Chinchilla even without multi-epoch effects.

3.3 Dual Power Law for Multi-Stage Training

Here, we extend the language ratio factor $\mathcal{R}$ of the He model (Eq. 1) to multi-stage training. We first analyze two-stage as a pilot case.

3.3.1 Pilot Analysis for Two-Stage Training

We fitted L_{base} (Eq. 2) to monolingual single-epoch data and computed $L_{\text{actual}}/L_{\text{base}}(M, D)$ as an empirical proxy for $\mathcal{R}$ on $k = 1$ two-stage setting. Figure 3(a) plots this ratio against r for both single-stage (black circles) and two-stage (colored symbols) setups, averaged over (M, D_T) .

As shown in the figure, the single-stage data follow the power law $r^{-\gamma}$ ($\gamma = 0.1070$), consistent with He et al. (2025). On the other hand, two-stage settings systematically exhibit lower ratios at matched r , and those sharing the same r_2 tend to trace a common slope differing from $-\gamma$. On the other hand, Figure 3(b) shows that r_1 has a much weaker effect than r and r_2 , with only a slight increase at lower r_1 ³.

3.3.2 Proposed Dual Power Law

Based on observations in §3.3.1, we propose the following dual power law model for the language

ratio factor that unifies single-stage and multi-stage training:

$$\mathcal{R}_{\text{dual}}(r, r_f) = \begin{cases} r^{-\gamma} & \text{(single-stage)} \\ r_f^{-\gamma} \cdot \left(\frac{r}{r_f}\right)^{-\gamma_2} & \text{(multi-stage)} \end{cases} \quad (8)$$

where r_f is the final-stage language ratio, γ is shared with single-stage training, and γ_2 is an additional exponent corresponding to the common slope of multi-stage training. Note that when $r = r_f$ (i.e., effectively single-stage training), the multi-stage formula reduces to $r_f^{-\gamma} = r^{-\gamma}$, so the two cases are consistent.

As shown in Figure 3(a), fixing γ to the value obtained from single-stage data and fitting γ_2 yields $\gamma_2 = 0.0515$, which provides a good fit to the two-stage data. The fact that $\gamma_2 < \gamma$ indicates that two-stage training mitigates the loss increase associated with decreasing r compared to single-stage training, suggesting the effectiveness of concentrating target language data in the second stage.

3.3.3 Verification of Dual Power Law for Multi-Stage Training

The dual power law (Eq. 8) suggests that $\mathcal{R}$ depends only on the average language ratio r and the final-stage language ratio r_f , independent of intermediate stage ratios. To verify that this behavior generalizes to multi-stage training, we conducted three-stage training experiments.

More specifically, we trained three-stage models on (Japanese, English) at $k = 1$, sweeping r , M , D_T , and the final-stage ratio r_3 , together with the stages 1–2 average r_{12} . At each matched (r_3, r, r_{12}, M, D_T) , we additionally compared three internal split patterns: **(A)** (r_1, r_2) bracketing r_{12} , and **(B,C)** $r_1 = 0$ with r_2 set to the smallest or largest valid value above r_{12} . Since exhaustive enumeration of all valid (r_1, r_2) combinations is infeasible, these three patterns were chosen as representative configurations spanning the range of feasible splits. See Appendix C.4 for implementation details.

Figure 3(a) shows that three-stage setups also follow the dual power law with the same γ and γ_2 fitted on two-stage data (r_3 taking the role of r_f), and panels (c,d) confirm that the ratio is insensitive to r_{12} and the internal split (r_1, r_2) . These results suggest that the same dual power law $\mathcal{R}_{\text{dual}}(r, r_f)$ governs multi-stage training in general, depending only on the average and final-stage ratios.

³See Appendix E.4 for a quantitative analysis

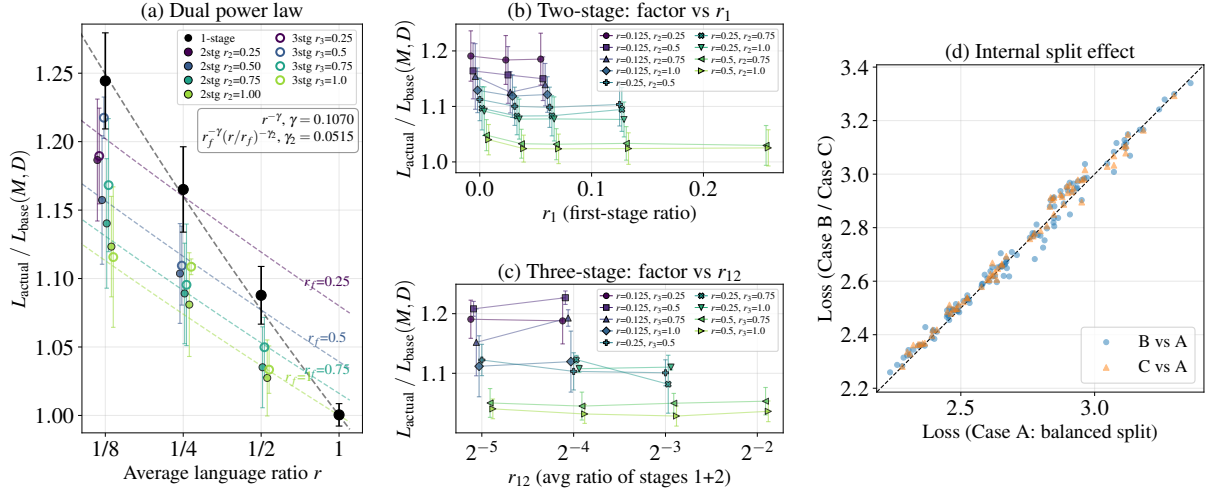


Figure 3: Verification of the dual power law for multi-stage training ((Japanese, English) pair, $k = 1$ only). (a) Ratio $L_{\text{actual}}/L_{\text{base}}(M, D)$ vs. average language ratio r . Filled and hollow symbols denote two-stage and three-stage setups, respectively, colored by final-stage ratio r_f (r_2 or r_3). Dashed lines show $r_f^{-\gamma}(r/r_f)^{-\gamma_2}$ fitted on **two-stage data**. (b, c) $L_{\text{actual}}/L_{\text{base}}(M, D)$ vs. first-stage ratio r_1 (two-stage) and vs. stages 1–2 average ratio r_{12} (three-stage), showing near-flat trends. (d) Loss comparison across three (r_1, r_2) patterns (A, B, C) at matched (r_3, r_{12}, r, M, D_T), confirming negligible effect of the internal split. Error bars show standard deviations across different (M, D_T) configurations sharing the same ratio settings; points are jittered horizontally for visibility.

3.4 Modeling Interaction of Multi-Epoch and Multi-Lingual Training

For multi-epoch multi-lingual training, we consider $L = L_{\text{base}}(M', D') \cdot \mathcal{R}_{\text{dual}}(r, r_f)$ with M' from Eq. (5) and D' extending the multi-lingual form of Longpre et al. (2026) (Eq. 6). The (k, r) dependence can be carried by either R_M^* (through M') or the weight g on D_{high} ; preliminary experiments indicated that placing the dependence solely on g yields the best M -axis extrapolation at a minor cost in k -axis accuracy, while involving R_M^* degrades M -axis extrapolation. We therefore adopt the g -only route, keeping R_M^* as a single global constant. As epochs increase, the effective weight of high-resource tokens is expected to decay due to target-language data repetition; however, when r is small, repeated target-language tokens constitute a smaller fraction of the total training data, weakening this decay. At $r = 0$ no data is repeated and g should remain 1. Based on this hypothesis, we promote g to a function of (r, R_D):

$$g(r, R_D; R_{D, \text{high}}^*, \psi) = (1 - r)^\psi + [1 - (1 - r)^\psi] \exp\left(-\frac{R_D}{R_{D, \text{high}}^*}\right), \quad (9)$$

which gives $g = 1$ at $R_D = 0$ (recovering He et al. (2025)) and $g \rightarrow (1 - r)^\psi$ as $R_D \rightarrow \infty$. Although this means the monolingual k -dependence $R_M^*(k)$

obtained in §3.2 is not used here, it is incorporated in a separate monolingual variant introduced in §3.5. See Appendix E.5 for a more detailed analysis of the interaction between multi-epoch and multi-lingual training.

3.5 Proposed Model: M^3 Scaling Law

Combining the extensions in §3.2, §3.3, and §3.4, we propose the M^3 **Scaling Law**, a unified scaling law that jointly captures multi-epoch, multi-lingual, and multi-stage training:

$$L_{M^3}(M, D_T, k, r, r_f) = \left(\frac{A}{M'^\alpha} + \frac{B}{D'^\beta} + E \right) \cdot \mathcal{R}_{\text{dual}}(r, r_f), \quad (10)$$

where $\mathcal{R}_{\text{dual}}$ is the dual power law defined in Eq. (8). The effective data and model sizes are

$$D' = D_T \cdot h(R_D; R_D^*) + g(r, R_D) \cdot D_{\text{high}}, \quad (11)$$

$$M' = U_M \cdot h(R_M; R_M^*), \quad (12)$$

where R_D, R_M, U_M, G , and D_{high} are as defined in Eqs. (4)–(6), h is the saturation function of Eq. (5), g follows Eq. (9), and R_M^* is a single global constant shared across all (k, r). This form has 11 free parameters: $A, B, \alpha, \beta, E, R_D^*, R_{D, \text{high}}^*, \psi, R_M^*, \gamma$, and γ_2 .

Model	#P	C	M	D_T	D	r	k	Avg
(a) All data (1-stage + 2-stage)								
<i>Baseline models</i>								
Chinchilla	5	.45 _{.00}	-.25 _{.00}	.57 _{.00}	.07 _{.00}	.38 _{.00}	-.06 _{.00}	.19 _{.00}
He	6	.42 _{.03}	-.04 _{.03}	.78 _{.03}	-.26 _{.19}	.45 _{.07}	-.60 _{.06}	.13 _{.06}
Muennighoff	7	.58 _{.01}	.03 _{.04}	.79 _{.02}	.06 _{.18}	-.48 _{.09}	.18 _{.02}	.19 _{.05}
ATLAS	7	.52 _{.01}	.57 _{.01}	.74 _{.02}	-.77 _{.10}	.54 _{.02}	.52 _{.01}	.35 _{.02}
Sedova	9	.78 _{.00}	.56 _{.01}	.80 _{.00}	.48 _{.01}	.75 _{.01}	.41 _{.00}	.63 _{.00}
Bryant	8	.52 _{.01}	-.25 _{.06}	.58 _{.00}	.20 _{.06}	.29 _{.00}	.05 _{.03}	.23 _{.02}
<i>Proposed model and ablation</i>								
M^3 Scaling Law	11	.81_{.01}	.53 _{.02}	.85 _{.07}	.68_{.10}	.76 _{.03}	.55 _{.02}	.70_{.02}
–Dual PL	10	.76 _{.01}	.51 _{.02}	.79 _{.06}	.49 _{.10}	.64 _{.03}	.57_{.01}	.63 _{.02}
–g	9	.73 _{.02}	.58_{.01}	.87_{.06}	.20 _{.23}	.77_{.03}	.42 _{.03}	.59 _{.05}
(b) 1-stage data only								
<i>Baseline models</i>								
Chinchilla	5	.32 _{.00}	-.04 _{.00}	.28 _{.00}	.28 _{.00}	-.08 _{.00}	-.21 _{.00}	.09 _{.00}
He	6	.18 _{.03}	-.03 _{.00}	.89_{.03}	-.39 _{.17}	.44 _{.07}	-.18 _{.05}	-.02 _{.05}
Muennighoff	7	.56 _{.01}	.34 _{.02}	.75 _{.02}	.09 _{.11}	-.42 _{.11}	.26 _{.01}	.26 _{.04}
ATLAS	7	.58 _{.00}	.53 _{.01}	.70 _{.01}	-.32 _{.06}	.27 _{.03}	.46 _{.01}	.37 _{.01}
Sedova	9	.71 _{.03}	.64 _{.11}	.86 _{.01}	.68 _{.12}	.67 _{.06}	.25 _{.10}	.64 _{.05}
Bryant	8	.47 _{.01}	-.11 _{.03}	.33 _{.01}	.31 _{.02}	-.39 _{.01}	-.02 _{.01}	.10 _{.01}
<i>Proposed model</i>								
M^3 Scaling Law	11	.79_{.01}	.66_{.01}	.88 _{.07}	.74_{.09}	.74_{.06}	.50_{.01}	.72_{.03}
(c) Multi-epoch only (monolingual, 1-stage)								
<i>Baseline models</i>								
Muennighoff	7	.78 _{.02}	.75 _{.02}	.90 _{.02}	.90 _{.00}	—	.52 _{.01}	.77 _{.01}
ATLAS	7	.67 _{.00}	.41 _{.01}	.90_{.02}	.85 _{.00}	—	.48 _{.01}	.66 _{.01}
Sedova	9	.80 _{.01}	.52 _{.14}	.88 _{.04}	.90 _{.02}	—	.36 _{.02}	.69 _{.03}
Bryant	8	.80 _{.00}	.59 _{.03}	.89 _{.00}	.87 _{.00}	—	.20 _{.01}	.67 _{.01}
<i>Proposed model and ablation</i>								
$M^3 + R_M^*(k)$	9	.89_{.01}	.78_{.01}	.89 _{.02}	.93_{.00}	—	.53_{.08}	.81_{.02}
M^3 Scaling Law	7	.78 _{.02}	.75 _{.02}	.90 _{.02}	.90 _{.00}	—	.52 _{.01}	.77 _{.01}

Model	#P	C	M	D_T	D	r	Avg
(d1) Multi-lingual only (single-epoch): 1-stage test							
<i>Baseline models</i>							
He	6	.49 _{.02}	.10 _{.06}	.83 _{.03}	.15 _{.05}	-.43 _{.02}	.23 _{.02}
He + Dual PL	7	.95 _{.01}	.74 _{.04}	.97_{.00}	.86_{.01}	.81 _{.02}	.86 _{.01}
Sedova	9	.65 _{.01}	.38 _{.02}	.75 _{.00}	.32 _{.03}	-.47 _{.07}	.33 _{.02}
<i>Proposed model and ablation</i>							
M^3 Scaling Law	8	.96_{.00}	.81_{.01}	.97 _{.01}	.77 _{.02}	.82_{.01}	.87_{.01}
–Dual PL	7	.55 _{.02}	.28 _{.09}	.77 _{.04}	-.07 _{.05}	-.34 _{.00}	.24 _{.03}
(d2) Multi-lingual only (single-epoch): 2-stage test							
<i>Baseline models</i>							
He	6	.70 _{.02}	.18 _{.10}	.87 _{.02}	.03 _{.13}	.69 _{.01}	.50 _{.02}
He + Dual PL	7	.76 _{.03}	.26 _{.08}	.94 _{.01}	.40 _{.14}	.76 _{.01}	.62 _{.02}
Sedova	9	.91_{.00}	.76_{.03}	.94 _{.00}	.50_{.01}	.86_{.02}	.79_{.01}
<i>2-stage-only baselines[†]</i>							
D-CPT Law	8	(.53 _{.02})	(.11 _{.00})	(.84 _{.01})	(.71 _{.01})	(.81_{.00})	(.60 _{.01})
PTPP-F1	10	(.59_{.01})	(.45 _{.01})	(.90 _{.01})	(.89 _{.01})	(.74 _{.01})	(.71 _{.01})
PTPP-F2	10	(.52 _{.03})	(.10 _{.01})	(.86 _{.00})	(.71 _{.06})	(.80 _{.00})	(.60 _{.01})
PTPP-F3	12	(.59 _{.02})	(.46_{.02})	(.91_{.01})	(.89_{.02})	(.73 _{.01})	(.72_{.01})
Zhang	6	(-.61 _{2.38})	(-1.19 _{.39})	(.55 _{.26})	(-1.29 _{.14})	(.44 _{.49})	(-.42 _{.67})
<i>Proposed model and ablation</i>							
M^3 Scaling Law	8	.85 _{.01}	.48 _{.02}	.95_{.01}	.21 _{.18}	.80 _{.01}	.66 _{.03}
–Dual PL	7	.84 _{.01}	.55 _{.04}	.90 _{.01}	-.30 _{.16}	.77 _{.01}	.55 _{.03}
(d3) Multi-lingual only (single-epoch): both stages							
<i>Baseline models</i>							
He	6	.60 _{.01}	.14 _{.06}	.85 _{.01}	.09 _{.09}	.13 _{.01}	.36 _{.01}
He + Dual PL	7	.85 _{.02}	.50 _{.05}	.96 _{.01}	.63_{.08}	.79 _{.01}	.74 _{.01}
Sedova	9	.78 _{.00}	.57 _{.02}	.84 _{.00}	.41 _{.02}	.20 _{.04}	.56 _{.01}
<i>Proposed model and ablation</i>							
M^3 Scaling Law	8	.90_{.00}	.64_{.02}	.96_{.00}	.49 _{.10}	.81_{.01}	.76_{.02}
–Dual PL	7	.69 _{.01}	.41 _{.06}	.83 _{.02}	-.18 _{.10}	.22 _{.00}	.39 _{.03}

Table 2: Extrapolation accuracy (Test R^2) of baseline and proposed scaling law models across Japanese, Indonesian, and Swahili. *Left: (a)* All (1-stage + 2-stage) data. *(b)* 1-stage data only. *(c)* Multi-epoch only: monolingual ($r=1$) data; the r axis is not applicable. *Right: (d1, d2)* Multi-lingual only: single-epoch ($k=1$) data evaluated on 1-stage and 2-stage test sets separately; the k axis is not applicable. *(d3)* Both stages (macro-average of (d1) and (d2)). [†] 2-stage-only models are trained on 2-stage data only; scores in parentheses are not directly comparable to other models trained on the combined data. Each column shows the R^2 macro-averaged over held-out splits and languages along the corresponding axis (see Appendix D.2.3 for details). Avg is the unweighted mean across applicable axes. #P denotes the number of free parameters. We repeat every scaling law fit 5 times with different random initialisations; each cell reports the mean over these fits, with the standard deviation as a subscript. **Bold** marks the best mean in a column.

We also consider a variant that incorporates the k -dependent $R_M^*(k)$ from Eq. (7) for monolingual multi-epoch settings ($r = 1$). In this case, $D_{\text{high}} = 0$ and g does not participate, so the interaction between multi-epoch and multi-lingual training need not be considered and $R_M^*(k)$ can be incorporated without degrading M -axis extrapolation. We denote this variant $M^3 + R_M^*(k)$ and evaluate it in the monolingual multi-epoch setting. To summarize the role of R_M^* across the paper: the per- k form $R_M^*(k)$ from Eq. (7) is used only in this monolingual variant; in the main M^3 Scaling Law, R_M^* is a single constant.

3.6 Evaluation of M^3 Scaling Law

We evaluate the proposed M^3 Scaling Law against existing scaling laws—Chinchilla (Hoffmann et al., 2022), He (He et al., 2025), Muennighoff (Muennighoff et al., 2023), ATLAS (Longpre et al., 2026), Sedova (Sedova et al., 2026), and Bryant (Bryant

and Liu, 2026)—under four settings (Table 2): (a) all setups combined; (b) single-stage setups only, the regime targeted by most prior work; (c) monolingual ($r=1$) multi-epoch setups, providing a focused comparison against methods that target data repetition (Muennighoff et al., 2023; Sedova et al., 2026; Bryant and Liu, 2026); and (d) multi-lingual single-epoch ($k=1$) setups, where models are fit on the combined 1-stage and 2-stage training data and evaluated separately on 1-stage (d1) and 2-stage (d2) test sets, as well as their macro-average (d3). For reference, we additionally include methods focusing on continual pretraining, the D-CPT Law (Que et al., 2024), PTPP (Goffinet et al., 2025), and Zhang et al. (2024), fit on 2-stage training data only and evaluated on the 2-stage test set (d2).

Extrapolation accuracy is measured by the coefficient of determination (R^2) on a held-out test set. To avoid sensitivity to any single train-test split, we construct multiple splits along each of the

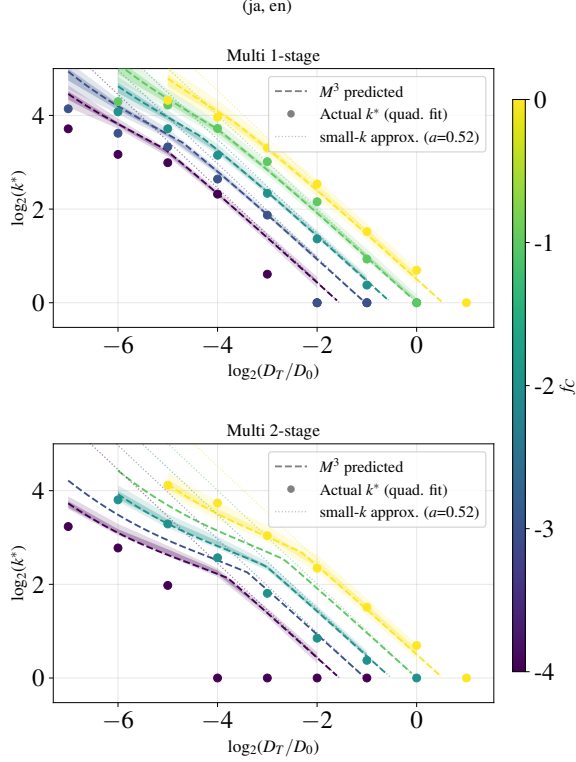


Figure 4: Optimal number of epochs k^* vs. target language corpus size D_T for (Japanese, English), shown for **multi 1-stage** (top) and **multi 2-stage** (bottom). Dashed lines show k^* predicted by numerically minimizing the M^3 Scaling Law at each (C, D_T) ; filled circles show empirical k^* estimated by quadratic fitting of $L^*(C, D_T, k)$ in $\log_2 k$. Dotted lines show the small- k approximation $k^* \approx D^*(C)/D_T$. Shading around the predicted curves shows their bootstrap variability (Appendix D.4). Color encodes compute budget C via $f_C = \log_2(C/C_0)$.

training setup variables (C, M, D_T, D, r, k) , average R^2 over splits within each language, and report the unweighted mean across languages (see Appendix D.2 for details).

The M^3 Scaling Law achieves the best average R^2 across settings (a)–(c). In setting (d), it achieves the highest accuracy on the combined 1-stage and 2-stage evaluation (d3), outperforming all baselines, which demonstrates that the dual power law enables strong extrapolation on both single-stage and two-stage data simultaneously. On the 2-stage test set alone (d2), the M^3 Scaling Law achieves comparable performance to methods specifically designed for continual pretraining, despite being trained on the combined 1-stage and 2-stage data rather than 2-stage data alone.

Ablations clarify the contribution of each extension. The dual power law is essential for jointly

achieving good extrapolation on both single-stage and two-stage data: in setting (d), models that incorporate it (He + Dual PL and M^3) consistently outperform their counterparts without it (He and M^3 –Dual PL). In the monolingual setting (c), the $M^3 + R_M^*(k)$ variant—which incorporates the k -dependent $R_M^*(k)$ into M' —achieves the best accuracy, exceeding all baselines. Finally, removing the high-resource weight g ($-g$ in part (a)) degrades extrapolation along the D and k axes, while slightly improving M -axis extrapolation.

4 Predicting Optimal Training Setups with M^3 Scaling Law

4.1 Optimal Training Approach

Since empirically $\gamma_2 < \gamma$, the dual power law (Eq. 8) at fixed r is minimized at $r_f = 1$, giving $r^{-\gamma_2}$ for two-stage training, which is lower than the single-stage value $r^{-\gamma}$ at any $r < 1$ under matched (M, D_T, k) .

To examine which category is optimal under realistic constraints, we numerically minimize the M^3 Scaling Law over training setups within each of three categories—**mono 1-stage**, **multi 1-stage**, and **multi 2-stage**—for each (C, D_T) . As shown in Figure 1, mono 1-stage is optimal at large D_T and multi 2-stage at small D_T , while multi 1-stage is never optimal in either regime. The same pattern holds for the empirically optimal setups across all languages and (C, D_T) in our grid search.

4.2 Optimal Number of Epochs

The M^3 Scaling Law predicts that the optimal number of epochs depends on (C, D_T) only through the scarcity variable $D_T/D^*(C)$, where $D^*(C) \propto C^{\alpha/(\alpha+\beta)}$ is the monolingual compute-optimal corpus size. Specifically, under the compute constraint $C = MD$ with $D = kD_T/r$ and for a fixed ratio schedule $\mathbf{r} = (r, r_1, r_2)$ specifying the average and per-stage target-language ratios, one obtains

$$k_{\mathbf{r}}^*(C, D_T) = K_{\mathbf{r}} \left(\frac{D_T}{D^*(C)} \right) \quad (13)$$

for some function $K_{\mathbf{r}}^4$. The relation is exact for fixed $\mathbf{r}$ (covering **mono 1-stage**) and approximate under joint optimization of $\mathbf{r}$ (**multi 2-stage**). In the small- k regime where the saturations of M' and D' are negligible, $K_{\mathbf{r}}$ further reduces to $k_{\mathbf{r}}^*(C, D_T) \approx D^*(C)/D_T$, predicting a unit-slope line passing

⁴The proof is given in Appendix E.6.3

through $k^* = 1$ at $D_T = D^*(C)$ in the log-log plot.

Numerically minimizing the M^3 Scaling Law over k separately for **multi 1-stage** and **multi 2-stage**, the resulting log-log curves of k^* vs. D_T at different C are nearly parallel with uniform log-scale spacing (Figure 4), consistent with the predicted collapse along $D_T/D^*(C)$ and matching $k^* = D^*(C)/D_T$ in the small- k regime. Empirical estimates from quadratic fits of $L^*(C, D_T, k)$ in $\log_2 k$ broadly follow these predictions, with non-negligible scatter likely reflecting the limited k -grid resolution and residual fitting error⁵.

5 Related Work

Continual pretraining, initially proposed for encoder-only models (Chau et al., 2020; Muller et al., 2021), has been increasingly adopted for low-resource language LLM training (Luukkonen et al., 2023; Lin et al., 2024; Fujii et al., 2024); our findings corroborate its benefit while identifying when it becomes optimal. As summarized in Table 1, prior scaling law studies each cover a subset of the axes relevant to low-resource LLM pretraining, but none jointly addresses multi-epoch, multi-lingual, and multi-stage training. Studies on complementary hyperparameters such as batch size, learning rate, vocabulary size, compression rate of tokenizer, and training data quality (Bi et al., 2024; Hu et al., 2024b; Tao et al., 2024; Limisiewicz et al., 2026; Zhou et al., 2026) are orthogonal to our work.

6 Conclusion

We proposed the M^3 **Scaling Law**, a unified scaling law jointly covering multi-epoch, multi-lingual, and multi-stage training for low-resource language LLM pretraining. It extrapolates more accurately than existing scaling laws across three language pairs, and reveals that the optimal training approach is always either monolingual single-stage or multi-lingual two-stage—never multi-lingual single-stage—and that the optimal number of epochs is governed by the scarcity variable $D_T/D^*(C)$.

Limitations

Evaluation Metrics: Our evaluation of LLM performance is limited to the validation loss of the language modeling task on the low-resource target

language (§2.1), and we do not directly evaluate downstream task performance. Our training setups are relatively small-scale, and many downstream tasks exhibit near-zero performance at this scale due to the emergent nature of LLM capabilities (Wei et al., 2022), making direct analysis on downstream metrics infeasible. Nevertheless, validation loss has been shown to correlate with downstream task performance in prior work (Brandfonbrener et al., 2024; Chen et al., 2025; Du et al., 2024; Gadre et al., 2025; Thrush et al., 2025), and our preliminary analysis (Appendix H) is consistent with this finding, supporting validation loss as a reasonable proxy.

Beyond predicting aggregate target-language validation loss, a natural extension is to ask how training recipes affect different capabilities of LLMs. Validation loss aggregates two qualitatively different capabilities: language-general abilities such as reasoning, which are expected to transfer well across languages, and language-specific knowledge, which transfers less. Depending on which capability a downstream application emphasizes, the effect of multi-lingual or multi-stage training on transfer may differ from what aggregate validation loss suggests. Disentangling how training recipes differentially affect these capabilities, and constructing scaling laws at the capability level, is a natural direction for future work—one for which the M^3 Scaling Law could serve as a starting point.

Scope of Hyperparameters and Corpora: While our experiments cover a broader scope of hyperparameters than prior work relevant to low-resource language LLM pretraining, we could not exhaustively explore all training hyperparameters due to compute constraints. Specifically, we did not search over batch size or learning rate; instead, we used the compute-dependent values optimized in Bi et al. (2024). We also used C4 (Raffel et al., 2020) and mC4 (Xue et al., 2021) as our corpora throughout, chosen for their consistent pre-processing across languages and wide adoption in prior scaling law studies (Hoffmann et al., 2022; Muennighoff et al., 2023), and did not consider variations in target language corpus quality, the use of LLM-rephrased or synthetically generated data, or more recent multi-lingual corpora such as MADLAD-400 (Kudugunta et al., 2023) and CulturaX (Nguyen et al., 2024). This choice reflects our intent to fix reasonable defaults along these axes and concentrate experimental resources on the multi-epoch, multi-lingual, and multi-stage

⁵See Appendix E.6 for the derivation and fit details.

axes that are central to low-resource language LLM pretraining.

A consequence of these fixed defaults is that some aspects of recipe design fall outside our analysis. The values from [Bi et al. \(2024\)](#) are optimized for single-stage training, and different values may be optimal for two-stage training. Similarly, since C4 and mC4 share consistent preprocessing across languages, our analysis cannot characterize how asymmetries in corpus quality between high- and low-resource languages—common in practice—affect the optimal training recipe. Concurrent to our work, [Zhou et al. \(2026\)](#) model this quality axis directly, jointly capturing the effects of data quality and repetition through an information-centric scaling law. Extending scaling-law modeling to jointly cover corpus quality, data selection, and synthetic data generation, and to study how these axes interact with the multi-lingual and multi-stage training central to our work, is a natural direction for future research. The M^3 Scaling Law offers a natural starting point for such investigations.

Scope of Languages: Our experiments used English as the high-resource language paired with three target languages from different language families: Japanese, Indonesian (relatively high-resource), and Swahili (lower-resource). This selection reflects three deliberate choices. First, we prioritized pairs from different language families, since cross-family transfer is weaker and deviates more from the monolingual setting, making such pairs more informative for studying how multi-lingual and multi-stage training depart from single-language behavior. Second, we fixed English as the high-resource language because English is by far the most common choice in practice for the high-resource side of low-resource LLM pretraining (e.g., [Fujii et al., 2024](#)). Third, we focused on moderately low-resource languages rather than truly low-resource ones, since the latter typically lack sufficient corpus volume to support scaling-law analysis at all; building understanding from moderately low-resource settings is a natural first step. As a result, generalization to within-family pairs, non-English high-resource languages, or truly low-resource languages with orders of magnitude less data is not directly verified by our experiments.

A related limitation is that our analysis models only the target-language validation loss and does not model the high-resource language loss. This reflects our problem setup, which focuses on improv-

ing LLM capabilities for the low-resource target language; improvements on the high-resource side are out of scope. However, applications such as machine translation require capabilities in both languages, and a recipe that minimizes target-language loss alone may not be Pareto-optimal across both. Extending the M^3 Scaling Law to jointly predict losses on both languages, and applying it across more diverse language pairs, is a natural direction for future work. The M^3 Scaling Law already exposes modeling primitives such as the dual power law exponent γ_2 that characterize previously uncharacterized aspects of cross-lingual transfer, and broader language coverage in such future studies would deepen the understanding of how these quantities vary across language pairs.

Scope of Scales and Configurations: Our experiments span $C_0/16 \leq C \leq C_0 = 10^{18}$ FLOPs and model scales up to 9.39×10^8 non-embedding FLOPs per token. While our evaluation requires extrapolation to up to four times larger compute, eight times larger models, and sixteen times larger target corpora than the training split (Appendix D.2), extrapolation to billion-parameter models is not verified. Verifying the predicted recipes at such scales, by training them alongside competing recipes under the same (C, D_T) and comparing downstream task performance, is an important direction for future work. We also fixed the architecture, tokenizer, and corpora (LLaMA-style models, the BLOOM tokenizer, and C4/mC4), assuming that these axes shift only the fitted parameters rather than the functional form of the scaling law.

Acknowledgements

We used AI assistants (including large language models) during this research in the following ways: (i) grammar and style polishing and translation assistance for the manuscript text; (ii) drafting prose from author-provided outlines or bullet points, which the authors then reviewed and revised; (iii) assisting the implementation of experimental and visualization code; and (iv) suggesting draft outlines for mathematical proofs of propositions formulated by the authors. All AI-generated content—including manuscript text, code, and proof drafts—was reviewed and verified by the authors, who take full responsibility for the final content of this paper.

References

- Christina Baek, Ricardo Pio Monti, David Schwab, Amro Abbas, Rishabh Adiga, Cody Blakeney, Maximilian Böther, Paul Burstein, Aldo Gael Carranza, Alvin Deng, Parth Doshi, Vineeth Dorna, Alex Fang, Tony Jiang, Siddharth Joshi, Brett W. Larsen, Jason Chan Lee, Katherine L. Mentzer, Luke Merrick, and 15 others. 2026. [The finetuner’s fallacy: When to pretrain with your finetuning data](#). *Preprint*, arXiv:2603.16177.
- Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiusi Du, Zhe Fu, Huazuo Gao, Kaige Gao, Wenjun Gao, Ruiqi Ge, Kang Guan, Daya Guo, Jianzhong Guo, Guangbo Hao, Zhewen Hao, and 67 others. 2024. [Deepseek llm: Scaling open-source language models with longtermism](#). *Preprint*, arXiv:2401.02954.
- David Brandfonbrener, Nikhil Anand, Nikhil Vyas, Eran Malach, and Sham Kakade. 2024. [Loss-to-loss prediction: Scaling laws for all datasets](#). *Preprint*, arXiv:2411.12925.
- Christopher M. Bryant and Hao Liu. 2026. [Practical scaling laws: Converting compute into performance in a data-constrained world](#). *Preprint*, arXiv:2605.09189.
- Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K. Bergen. 2024. [When is multilinguality a curse? language modeling for 250 high- and low-resource languages](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4074–4096, Miami, Florida, USA. Association for Computational Linguistics.
- Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. [Parsing with multilingual BERT, a small corpus, and a small treebank](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. Association for Computational Linguistics.
- Yangyi Chen, Binxuan Huang, Yifan Gao, Zhengyang Wang, Jingfeng Yang, and Heng Ji. 2025. [Scaling laws for predicting downstream performance in LLMs](#). *Transactions on Machine Learning Research*.
- Zhengxiao Du, Aohan Zeng, Yuxiao Dong, and Jie Tang. 2024. [Understanding emergent abilities of language models from the loss perspective](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 53138–53167. Curran Associates, Inc.
- Manuel Faysse, Patrick Fernandes, Nuno M. Guerreiro, António Loison, Duarte M. Alves, Caio Corro, Nicolas Boizard, João Alves, Ricardo Rei, Pedro H. Martins, Antoni Bigata Casademunt, François Yvon, André F. T. Martins, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2025. [Croissantllm: A truly bilingual french-english language model](#). *Preprint*, arXiv:2402.00786.
- Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. 2024. [Continual pre-training for cross-lingual LLM adaptation: Enhancing japanese language capabilities](#). In *First Conference on Language Modeling*.
- 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, Luca Soldaini, Jenia Jitsev, Alex Dimakis, Gabriel Ilharco, Pang Wei Koh, Shuran Song, and 6 others. 2025. [Language models scale reliably with over-training and on downstream tasks](#). In *The Thirteenth International Conference on Learning Representations*.
- Etienne Goffinet, Shane Bergsma, Avraham Sheinin, Natalia Vassilieva, Shaheer Muhammad, Preslav Nakov, and Gurpreet Gosal. 2025. [Ptp-aware adaptation scaling laws: Predicting domain-adaptation performance at unseen pre-training budgets](#). *Preprint*, arXiv:2510.23198.
- Yifei He, Alon Benhaim, Barun Patra, Praneetha Vadamandu, Sanchit Ahuja, Parul Chopra, Vishrav Chaudhary, Han Zhao, and Xia Song. 2025. [Scaling laws for multilingual language models](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4257–4273, Vienna, Austria. Association for Computational Linguistics.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, and 3 others. 2022. [Training compute-optimal large language models](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 30016–30030. Curran Associates, Inc.
- Shengding Hu, Xin Liu, Xu Han, Xinrong Zhang, Chaoqun He, Weilin Zhao, Yankai Lin, Ning Ding, Zebin Ou, Guoyang Zeng, Zhiyuan Liu, and Maosong Sun. 2024a. [Predicting emergent abilities with infinite resolution evaluation](#). In *The Twelfth International Conference on Learning Representations*.
- Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, and 6 others. 2024b. [Minicpm: Unveiling the potential of small language models with scalable training strategies](#). *Preprint*, arXiv:2404.06395.
- Peter J Huber. 1964. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101.

- Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats Leon Richter, Quentin Gregory Anthony, Eugene Belilovsky, Timothée Lesort, and Irina Rish. 2024. [Simple and scalable strategies to continually pre-train large language models](#). *Transactions on Machine Learning Research*.
- Ai Ishii, Naoya Inoue, Hisami Suzuki, and Satoshi Sekine. 2024. [JEMHopQA: Dataset for Japanese explainable multi-hop question answering](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9515–9525, Torino, Italia. ELRA and ICCL.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling laws for neural language models](#). *Preprint*, arXiv:2001.08361.
- Diederik P. Kingma and Jimmy Ba. 2017. [Adam: A method for stochastic optimization](#). *Preprint*, arXiv:1412.6980.
- Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. [Madlad-400: A multilingual and document-level large audited dataset](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 67284–67296. Curran Associates, Inc.
- Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. [JGLUE: Japanese general language understanding evaluation](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2957–2966, Marseille, France. European Language Resources Association.
- Seng Pei Liew and Takuya Kato. 2026. [Reusing over-trained language models saturates scaling](#). *Preprint*, arXiv:2510.06548.
- Tomasz Limisiewicz, Artidoro Pagnoni, Srini Iyer, Mike Lewis, Sachin Mehta, Alisa Liu, Margaret Li, Gargi Ghosh, and Luke Zettlemoyer. 2026. [Compute optimal tokenization](#). *Preprint*, arXiv:2605.01188.
- Peiqin Lin, Shaoxiong Ji, Jörg Tiedemann, André F. T. Martins, and Hinrich Schütze. 2024. [Mala-500: Massive language adaptation of large language models](#). *Preprint*, arXiv:2401.13303.
- LLM-jp. 2024. [Llm-jp: A cross-organizational project for the research and development of fully open japanese llms](#). *Preprint*, arXiv:2407.03963.
- Shayne Longpre, Sneha Kudugunta, Niklas Muennighoff, I-Hung Hsu, Isaac Rayburn Caswell, Alex Pentland, Sercan O Arik, Chen-Yu Lee, and Sayna Ebrahimi. 2026. [ATLAS: Adaptive transfer scaling laws for multilingual pretraining, finetuning, and decoding the curse of multilinguality](#). In *The Fourteenth International Conference on Learning Representations*.
- Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Nouamane Tazi, Teven Scao, Thomas Wolf, Osmo Suominen, Samuli Sairanen, Mikko Merioksa, Jyrki Heinonen, Aija Vah-tola, and 2 others. 2023. [FinGPT: Large generative models for a small language](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726, Singapore. Association for Computational Linguistics.
- Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. [Scaling data-constrained language models](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. [When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.
- Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. [CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.
- Jorge Nocedal. 1980. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782.
- Haoran Que, Jiaheng Liu, Ge Zhang, Chenchen Zhang, Xingwei Qu, Yinghao Ma, Feiyu Duan, Zhiqi Bai, Jiakai Wang, Yuanxing Zhang, Xu Tan, Jie Fu, Jiamang Wang, Lin Qu, Wenbo Su, and Bo Zheng. 2024. [D-CPT law: Domain-specific continual pre-training scaling law for large language models](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Teven Le Scao, Angela Fan, Christopher Akiki, El-lie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas

- Muennighoff, Albert Villanova del Moral, and 373 others. 2023. [Bloom: A 176b-parameter open-access multilingual language model](#). *Preprint*, arXiv:2211.05100.
- Anastasiia Sedova, Skyler Seto, Natalie Schluter, and Pierre Ablin. 2026. [Scaling laws for mixture pretraining under data constraints](#). *Preprint*, arXiv:2605.12715.
- Satoshi Sekine. 2003. Development of a Question Answering System Focused on an Encyclopedia [百科事典を対象とした質問応答システムの開発]. In *Proceedings of the Ninth Annual Meeting of the Association for Natural Language Processing*. In Japanese.
- Skyler Seto, Pierre Ablin, Anastasiia Filippova, Jiayuan Ye, Louis Béthune, Angelos Katharopoulos, and David Grangier. 2026. [Optimal splitting of language models from mixtures to specialized domains](#). In *Forty-third International Conference on Machine Learning*.
- Mustafa Shukor, Louis Béthune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. 2025. [Scaling laws for optimal data mixtures](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Masatoshi Suzuki, Jun Suzuki, Koji Matsuda, Kyosuke Nishida, and Naoya Inoue. 2020. JAQKET: Construction of a Japanese QA dataset on the subject of quizzes [JAQKET:クイズを題材にした日本語QAデータセットの構築]. In *Proceedings of the Twenty-Sixth Annual Meeting of the Association for Natural Language Processing*. In Japanese.
- Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. 2024. [Scaling laws with vocabulary: Larger models deserve larger vocabularies](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Tristan Thrush, Christopher Potts, and Tatsunori Hashimoto. 2025. [Improving pretraining data using perplexity correlations](#). In *The Thirteenth International Conference on Learning Representations*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#). *Preprint*, arXiv:2302.13971.
- Xingjin Wang, Howe Tissue, Lu Wang, Linjing Li, and Daniel Dajun Zeng. 2025. [Learning dynamics in continual pre-training for large language models](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 64525–64548. PMLR.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. [Emergent abilities of large language models](#). *Transactions on Machine Learning Research*. Survey Certification.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Chengyin Xu, Kaiyuan Chen, Xiao Li, Ke Shen, and Chenggang Li. 2026. [Unveiling downstream performance scaling of LLMs: A clustering-based perspective](#). In *The Fourteenth International Conference on Learning Representations*.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. [mT5: A massively multilingual pre-trained text-to-text transformer](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.
- Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. [When scaling meets llm finetuning: The effect of data, model and finetuning method](#). In *International Conference on Learning Representations*, volume 2024, pages 44694–44713.
- Weidong Zhou, Fengze Liu, Binbin Liu, Ping Guo, Zijun Wang, Bingni Zhang, Yifan Zhang, Yifeng Yu, Xiaohuan Zhou, and Taifeng Wang. 2026. [Infolaw: Information scaling laws for large language models with quality-weighted mixture data and repetition](#). In *Forty-third International Conference on Machine Learning*.

A Code and Data Availability

The code and data used in this work are being prepared for public release. Once the release is complete, a link will be posted in the comments field of the arXiv version of this paper (<https://arxiv.org/abs/2410.12325>).

B Notation

Table 3 summarizes all symbols used in this paper.

Algorithm 1 Determine Batch Size

```
1: procedure DETERMINEBATCHSIZE( $C, d_{\text{model}}, n_{\text{layer}}$ )
2:   seq_length  $\leftarrow$  4096
3:   complexity  $\leftarrow n_{\text{layer}} \cdot d_{\text{model}}^2$ 
4:   if complexity  $< 0.1 \times 10^8$  then
5:     local_batch_size  $\leftarrow$  4
6:   else if complexity  $< 0.5 \times 10^8$  then
7:     local_batch_size  $\leftarrow$  2
8:   else if complexity  $< 1.1 \times 10^8$  then
9:     local_batch_size  $\leftarrow$  1
10:  end if
11:  optimal_local_batch_size  $\leftarrow \lfloor 0.292 \cdot C^{0.3271} / (\text{seq\_length} \cdot 8) \rfloor$ 
12:  if optimal_local_batch_size  $<$  local_batch_size then
13:    local_batch_size  $\leftarrow$  optimal_local_batch_size
14:    accumulation_step  $\leftarrow$  1
15:  else
16:    accumulation_step  $\leftarrow \lfloor \text{optimal\_local\_batch\_size} / \text{local\_batch\_size} \rfloor$ 
17:  end if
18:  batch_size  $\leftarrow \text{local\_batch\_size} \cdot 8 \cdot \text{accumulation\_step}$ 
19:  return batch_size
20: end procedure
```

C Implementation Details of LLM Pretraining and Evaluation

C.1 Model Pretraining

Throughout our experiments, we adopted the training configuration of Bi et al. (2024). More specifically, we trained decoder-only language models with LLaMA architecture (Touvron et al., 2023) and set context length $S = 4096$. We measured the model scale by non-embedding FLOPs/token, $M = 72nd_{\text{model}}^2 + 12nd_{\text{model}}S$, where n , d_{model} , and S are the number of layers, the size of hidden states, and the sequence length respectively. To improve the efficiency of our experiments by reusing intermediate checkpoints, we used their multi-step learning rate scheduler, in which we decreased the learning rate to 31.6% and 10% after processing 80% and 90% of the total steps of each training stage, respectively. For learning rate η and batch size $\mathcal{B}$, we adopted their power-law relationships, $\eta = 0.3118 \cdot C^{-0.1250}$ and $\mathcal{B} = 0.2920 \cdot C^{0.3271}$, which were fitted through extensive experiments across different compute budgets C to achieve near-optimal performance in (Bi et al., 2024).⁶ Following findings by Ibrahim et al. (2024) that the learning rate of continual pretraining should be re-warmed to that of pretraining, we set the same maximum learning rate for both the first and second

⁶We added minor adjustment to $\mathcal{B}$ to improve training efficiency in some cases, which is detailed in Appendix C.

stages for two-stage training.

We used the Transformers library (Wolf et al., 2020) (ver. 4.37.2, Apache-2.0 license) in our experiments. We used LlamaForCausalLM as the model architecture, and set the standard deviation for parameter initialization to 0.006. Configurations of the model shape is shown in Table 4. Based on the analysis by Kaplan et al. (2020), we only used those model shapes with aspect ratio $\frac{d_{\text{model}}}{n_{\text{layers}}}$ between 30 and 150. We used Adam optimizer (Kingma and Ba, 2017) and set its parameter to $(\beta_1, \beta_2, \epsilon) = (0.9, 0.95, 10^{-8})$. We set the number of warmup steps to 500, the weight decay to 0.1 and gradient clipping at a norm of 1. We used Trainer class to train our models, with all unspecified parameters left at their default values. To improve efficiency of training, we used the batch size adjusted by Algorithm 1. We used a server with eight A100 80GB GPUs to train our models.

For multi-epoch training, the order of the repeated data was randomized with different seeds for each epoch.

C.2 Corpus and Tokenizer

We retrieved C4 (Raffel et al., 2020) and mC4 (Xue et al., 2021) from <https://huggingface.co/datasets/allenai/c4>. We only used ja, id, and sw subsets of mC4 and did not use ja-Latn subsets. To reduce the cost of the evaluation, we respectively subsampled 5%, and 6.6% of the validation corpora for Japanese, and Indonesian, while using the full validation corpus for Swahili.

We tokenized the corpora with BLOOM tokenizer. BLOOM tokenizer is available at <https://huggingface.co/bigscience/tokenizer> under the BigScience RAIL License v1.0. Note that English, Indonesian, and Swahili were used to train the BLOOM tokenizer, while Japanese was not used to train it.

C.3 Training Hyperparameters

Based on the scaling law of the optimal model scale and data allocation by Bi et al. (2024), we set the reference compute budget C_0 , target language corpus budget $D_{T,0}$, and model scale M_0 as follows:

$$\begin{aligned} C_0 &= 10^{18}, \\ D_{T,0} &= 5.8316 \cdot C_0^{0.4757}, \\ M_0 &= \frac{C_0}{D_{T,0}}. \end{aligned} \tag{14}$$

Single-stage Training: The hyperparameters of single-stage training are the constant language ratio r , the model scale M , and the number of target language epochs k .

We parametrized each single-stage training setup by four integer factors, (f_r, f_M, f_k, f_C) , where $f_* \in \mathbb{Z}$, and set each of its hyperparameters and compute C and D_T as follows:

$$\begin{aligned} r &= \frac{1}{2^{f_r}}, & M &= \frac{M_0}{2^{f_M}}, \\ k &= 2^{f_k}, & C &= 2^{f_C} \cdot C_0, \\ D_T &= 2^{-f_r+f_M-f_k+f_C} \cdot D_{T,0}. \end{aligned} \quad (15)$$

The training setup involved multi-epoch training when $f_k > 0$ and multi-lingual training when $f_r > 0$. Multiple training setups with the same (C, D_T) were created by keeping f_C and $f_D = -f_r + f_M - f_k + f_C$ constant. We list the search spaces for those factors in Table 5.

Two-stage Training: For each stage of two-stage training, we used a different constant language ratio, r_1 (for the first stage) and r_2 (for the second stage), where $r_1 < r_2$. Using the proportions of the first and second stage training, s_1 and s_2 , relative to the total length of the training, we define the average language ratio as $r = s_1 r_1 + s_2 r_2$ by slightly abusing notation. Instead of explicitly setting s_1 and s_2 , we used r to compute $s_1 = \frac{r_2 - r}{r_2 - r_1}$ and $s_2 = 1 - s_1$.

We parametrized each two-stage training setup by $(r_1, r_2, f_r, f_M, f_k, f_C)$, and used Eq. (15) in §C.3 to set r , M , and k , and compute C and D_T . We grid-searched over $r_1 \in \{1/2, 1/4, 1/8, 1/16, 1/32, 0\}$, $r_2 \in \{1/4, 1/2, 3/4, 1\}$, and the same search space of other factors as shown in Table 5.

C.4 Details of Three-Stage Training Experiments

In three-stage training experiments, we trained models with randomly sampled three-stage training setups (r_3, r, M, D_T) on the (Japanese, English) pair under single-epoch training ($k = 1$), spanning $C \in \{C_0/16, \dots, C_0\}$, average ratios $r \in \{1/2, 1/4, 1/8\}$, and final-stage ratios $r_3 \in \{0.25, 0.5, 0.75, 1.0\}$ ($r_3 > r$) with M and D_T drawn from the same search space as the two-stage experiments. To isolate the effects of the intermediate ratios, for each (r_3, r, M, D_T) we varied r_1 , r_2 , and their average r_{12} and observed the resulting performance changes. Specifically, we sampled $r_{12} < r$ from $\{1/4, 1/8, 1/16, 1/32\}$, and for each

(r_3, r, r_{12}) compared three representative (r_1, r_2) patterns: **(A)** the closest valid (r_1, r_2) bracketing r_{12} , approximating a near-constant two-stage schedule; **(B)** $r_1 = 0$ with r_2 set to the smallest valid candidate above r_{12} ; and **(C)** $r_1 = 0$ with r_2 set to the largest valid candidate between r_{12} and r_3 .

The stage proportions s_1, s_2, s_3 for three-stage training were determined hierarchically, extending the inverse-computation approach used for two-stage training (§2.2). We first split the total training into a combined first-and-second stage block and the third stage using the outer constraint $r = s_{12} r_{12} + s_3 r_3$:

$$s_{12} = \frac{r_3 - r}{r_3 - r_{12}}, \quad s_3 = 1 - s_{12}, \quad (16)$$

where r_{12} is the average language ratio over stages 1 and 2. We then split s_{12} into stages 1 and 2 using the inner constraint $r_{12} = (s_1/s_{12})r_1 + (s_2/s_{12})r_2$:

$$s_1 = s_{12} \cdot \frac{r_2 - r_{12}}{r_2 - r_1}, \quad s_2 = s_{12} - s_1. \quad (17)$$

For each combination of (r_3, r_{12}) , we collected the set of (f_M, C) pairs that were feasible for all three patterns (A, B, and C) described in §3.3, pooled across all C values, and randomly sampled up to 10 pairs (with a fixed random seed). Pattern C was omitted when r_2 coincided with that of pattern B, since the two patterns become identical in that case. In total, we conducted 307 three-stage training runs (pattern A: 117, B: 125, C: 65).

D Implementation Details of Scaling Law Analysis

D.1 Scaling Law Model Fitting

In this paper, following Hoffmann et al. (2022), we fit a scaling law model $L(s_i; \theta)$ parameterized by θ to the observed validation loss L_i of an LLM trained under setting s_i (e.g., data size D_T , model size M , language ratio r , etc.), by minimizing the Huber loss (Huber, 1964) using the L-BFGS algorithm (Nocedal, 1980):

$$\min_{\theta} \sum_i \text{Huber}_{\delta}(\log L(s_i; \theta) - \log L_i), \quad (18)$$

where we used $\delta = 0.001$. The parameter bounds during optimization were set as shown in Table 6. To mitigate sensitivity to initial values, we randomly sampled initial values of θ_i within the ranges

shown in Table 6 (uniformly in log scale for A , B , and r_0 , and uniformly in linear scale for other parameters), independently ran the optimization 50 times, and adopted the best fit result.

D.2 Scaling Law Model Evaluation

The extrapolation performance of scaling law models can vary depending on the choice of train-test split. To avoid relying on a single arbitrary split, we evaluate extrapolation performance along multiple axes of the training setups. Specifically, for each axis, we define a threshold that partitions the data into a training set and a held-out test set. Each model is fit on the training set and evaluated by the coefficient of determination (R^2) on the test set, defined as

$$R^2 = 1 - \frac{\sum_{i \in \mathcal{T}} (L_i - \hat{L}_i)^2}{\sum_{i \in \mathcal{T}} (L_i - \bar{L}_{\mathcal{T}})^2}, \quad (19)$$

where $\mathcal{T}$ is the test set, L_i and $\hat{L}_i$ are the observed and predicted validation losses, and $\bar{L}_{\mathcal{T}} = \frac{1}{|\mathcal{T}|} \sum_{i \in \mathcal{T}} L_i$ is the test set mean. Note that R^2 is computed in the original (non-log) scale and can be negative when predictions are worse than predicting the test set mean. We require both the training and test sets to contain at least 10 data points, and splits that do not meet this criterion are skipped.

D.2.1 Held-Out Splits for Combined Evaluation

For the main evaluation on all single-stage and two-stage data, we define 18 train-test splits along six axes of the training setups (Table 7). The test set in each split contains data from regions that are of practical interest for extrapolation: larger compute budgets, larger models, larger corpora, lower or higher target language ratios, and higher epoch counts. To avoid dependence on any single threshold, we use multiple thresholds per axis. Note that for Swahili, the available target language corpus is limited to $f_D \leq -3$, so no data points fall into the test set of the three D_T -axis splits (DT_ge0, DT_ge-1, DT_ge-2); these splits are therefore omitted for Swahili.

D.2.2 Held-Out Splits for Two-Stage Evaluation

For multi-lingual single-epoch evaluation, of the 18 splits in Table 7, we use 13 for this evaluation (Table 8). From the r -axis, only r_le0.125 and r_ge0.5 are retained; r_le0.25 is excluded because the two-stage data has only $r \in \{0.125, 0.25, 0.5\}$,

and training set with $r > 0.25$ would contain only a single r level ($r = 0.5$), preventing continual-pretraining baselines trained only on two-stage data from identifying the effect of r . Similarly, r_ge1 is excluded because no monolingual ($r = 1$) data exists in the two-stage experiments. All k -axis splits are omitted since this evaluation is restricted to $k = 1$.

D.2.3 Aggregation of Test R^2 Across Splits and Languages

For each axis (e.g., C , M), the per-axis R^2 reported in Table 2 is computed in two steps. First, for each language, we average the test R^2 over all held-out splits belonging to that axis (e.g., for the C axis, over C_ge0 and C_ge-1). If a split is skipped for a particular language (e.g., because the training or test set contains fewer than 10 data points) or yields an invalid R^2 (e.g., due to numerical issues in fitting), it is excluded from that language’s average. To ensure a fair comparison, whenever a split is excluded for any model, we exclude it for *all* models in the same evaluation setting, so that all models are compared on the same set of splits. Second, we take the unweighted mean of these per-language averages across all languages. The Avg column is the unweighted mean of the per-axis values.

D.2.4 Two-Phase Fitting

In preliminary experiments, we found that fitting the Chinchilla base parameters (A , B , α , β , E) jointly with the remaining parameters causes the base parameters to drift, substantially degrading extrapolation performance along the M - and D -axis splits. To address this, we adopt a two-phase fitting procedure for all models that build on the Chinchilla formulation:

1. **Phase 1.** Fit the Chinchilla base parameters (A , B , α , β , E) using only monolingual ($r = 1$), low-epoch ($k \leq 4$) single-stage data from the training set.
2. **Phase 2.** Fix the Phase 1 parameters and optimize the remaining parameters (e.g., R_D^* , R_M^* , γ , γ_2) on the full training set.

D.3 Implementation of Baseline Models

This section summarizes the scaling law models used as baselines in Table 2. All models share the Chinchilla base $L_{\text{base}}(M, D) = A/M^\alpha + B/D^\beta + E$ (Eq. 2) and the saturation function $h(R; R^*) = 1 + R^*(1 - \exp(-R/R^*))$ (Eq. 5).

Chinchilla (Hoffmann et al., 2022) (5P). $L = L_{\text{base}}(M, D)$.

He (He et al., 2025) (6P). $L = L_{\text{base}}(M, D) \cdot r^{-\gamma}$.

He + Dual PL (7P). $L = L_{\text{base}}(M, D) \cdot \mathcal{R}_{\text{dual}}(r, r_f)$, where $\mathcal{R}_{\text{dual}}$ is defined in Eq. (8).

Muennighoff (Muennighoff et al., 2023) (7P). $L = L_{\text{base}}(M', D')$ with effective sizes D' and M' as defined in Eqs. (4)–(5).

ATLAS (Longpre et al., 2026) (7P). $L = L_{\text{base}}(M, D')$, where D' is defined in Eq. (6) with constant weight $g = \tau$. Unlike the Muennighoff model, ATLAS does not use the effective model size M' , applying M directly.

Sedova (Sedova et al., 2026) (9P).

$$L = E_s + \frac{C_s}{M^{\beta_s}} + \frac{B_s M^{\delta_s}}{D_{\text{Se}}^{\alpha_s}} + \gamma_s \cdot r,$$

where $D'_{\text{Se}} = D_{\text{high}} + \tau_s \cdot D_T \cdot h(R_D; R_{D,s}^*)$. Since this model does not contain the Chinchilla model as a special case, all 9 parameters are fitted jointly in a single phase, unlike the two-phase procedure used for the other models (§D.2).

Bryant (Bryant and Liu, 2026) (8P).

$$L = E + (L_0 - E) \cdot \frac{h}{1 + h},$$

$$h = \frac{a}{M^\alpha} + \frac{b}{D^\beta} + \frac{c M^\gamma}{D_{\text{uniq}}^\delta},$$

where $D = kD_T/r$ is the total number of tokens seen, $D_{\text{uniq}} = D_T + D_{\text{high}}$ is the number of unique training tokens, and the three terms of h model undercapacity, undertraining, and overfitting, respectively. The uninformed baseline loss $L_0 = \ln V$ is fixed by the tokenizer vocabulary size $V = 250,680$ and is not fitted. Since this model does not contain the Chinchilla model as a special case without approximation, all 8 parameters are fitted jointly in a single phase, unlike the two-phase procedure used for the L_{base} -based models.

The following four two-stage-only baselines also do not contain the Chinchilla model as a substructure and are fitted in a single phase.

D-CPT Law (Que et al., 2024) (8P).

$$L = E + \frac{A}{M^\alpha} + \frac{B \cdot r_f^\nu}{D_2^\beta} + \frac{C_c}{r_f^\gamma},$$

where $D_2 = s_2 D$ is the number of second-stage tokens.

PTPP-F1 (Goffinet et al., 2025) (10P).

$$L = E + \frac{A}{M^\alpha} + \frac{B \cdot r_f^\nu}{D_2^\beta} + \frac{C_c}{r_f^\gamma} + \frac{F}{\text{PTPP}^\xi},$$

where $\text{PTPP} = D_1/N$ is the pretraining tokens per parameter, $D_1 = s_1 D$ is the number of first-stage tokens, and N is the number of model parameters.

PTPP-F2 (Goffinet et al., 2025) (10P). Same as D-CPT Law, but with data exponent modulated by PTPP:

$$\beta_{\text{eff}} = \beta \left(1 - \frac{\lambda \cdot \text{PTPP}^\zeta}{1 + \text{PTPP}^\zeta} \right),$$

and D_2^β in the D-CPT formula is replaced by $D_2^{\beta_{\text{eff}}}$.

PTPP-F3 (Goffinet et al., 2025) (12P). Same as PTPP-F1, but with data exponent modulated by PTPP as in PTPP-F2:

$$L = E + \frac{A}{M^\alpha} + \frac{B \cdot r_f^\nu}{D_2^{\beta_{\text{eff}}}} + \frac{C_c}{r_f^\gamma} + \frac{F}{\text{PTPP}^\xi},$$

where $\beta_{\text{eff}} = \beta \left(1 - \frac{\lambda \cdot \text{PTPP}^\zeta}{1 + \text{PTPP}^\zeta} \right)$, combining the gated data exponent of PTPP-F2 with the additive floor of PTPP-F1.

Zhang (Zhang et al., 2024) (6P). Zhang et al. (2024) propose a multiplicative joint scaling law over D_2 and one other variable (D_1 or M). Here we evaluate a naïve extension to four variables (M , D_1 , D_2 , r_f):

$$L = \frac{A}{M^\alpha \cdot D_1^{\phi_1} \cdot D_2^{\phi_2} \cdot r_f^\gamma} + E.$$

D.4 Bootstrap Uncertainty Bands

Each training setup is trained once, so the fitted loss surface, and the recipes obtained by minimizing it, inherit the run-to-run variation of the observed losses. To quantify this sensitivity, we bootstrap over the single-run results following Hoffmann et al. (2022). For each of 200 replicates, we draw 80% of all loss observations without replacement and refit the M^3 Scaling Law with the same two-phase, multi-start procedure (§D.2.4), keeping the multi-start initialization fixed so that the spread reflects the resampling rather than the parameter initialization. From each replicate we then recompute the quantity of interest, that is, the approach boundary, k^* , M^* , or r^* , by applying the same recipe optimization to the refitted parameters. The shaded bands report the resulting distribution over replicates: the outer band spans the 5th to the 95th percentile and the inner band the 25th to the 75th.

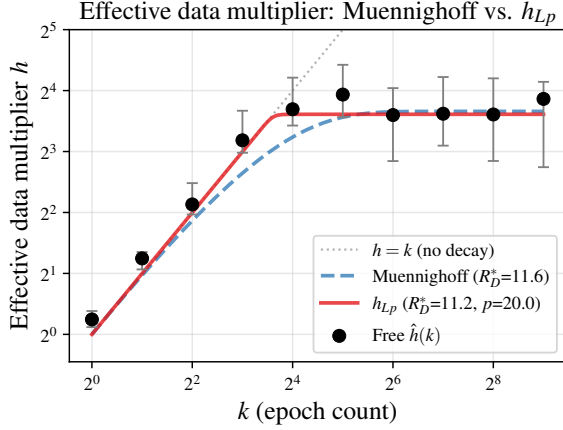


Figure 5: Effective data multiplier comparison on Japanese monolingual data. Black points show independently fitted $\hat{h}(k)$ for each epoch count k ; the Muennighoff form h (blue dashed) saturates too early, while the h_{LP} extension (red solid) with sharpness parameter p more closely tracks the free estimates around the transition region. Error bars show 95% bootstrap confidence intervals (200 resamples).

E Additional Scaling Law Analysis

E.1 Per-Language, Per-Split Evaluation Results

Tables 9–12 provide the per-language, per-split test R^2 values underlying the aggregated results in Table 2. Splits where the training or test set contains fewer than 10 data points are marked “—”. For Swahili, the D_T -axis splits (DT_ge0, DT_ge-1, DT_ge-2) are omitted due to limited corpus availability.

E.2 M^3 Scaling Law Parameters Fitted on All Data

Table 13 shows the M^3 Scaling Law parameters fitted on the full dataset (1-stage + 2-stage) for each language pair.

E.3 Analysis of Extending Effective Data Multiplier

The saturation function $h(R_D; R_D^*)$ in the Muennighoff model (Eq. 4) may not fully capture how the effective data size D' transitions from the linear regime to the saturation regime. To see this, we focus on the Japanese monolingual single-stage setting ($r = 1$). We first fit and fix the Chinchilla base parameters (A, B, α, β, E) of L_{base} (Eq. 2) using the subset with $k \leq 4$. We then model the effective data multiplier in $D' = D_T \cdot \hat{h}$ in three ways, each fit using the Muennighoff model L_{Mu}

(Eq. 3) on all monolingual data: (i) a *discrete* parameterization that fits an independent $\hat{h}_k$ for each k , jointly with a shared R_M^* for M' ; (ii) the parametric *Muennighoff* form $\hat{h} = h(R_D; R_D^*)$ (Eq. 4), fitting R_D^* and R_M^* ; and (iii) the parametric form $\hat{h} = h_{LP}(R_D; R_D^*, p)$ defined below, fitting R_D^* , p , and R_M^* . As shown in Figure 5, the Muennighoff form $h(R_D; R_D^*)$ captures the convergence of the free $\hat{h}(k)$ to a constant in the large- k regime, but does not capture the sharpness of the transition from the linear ($\hat{h} \approx k$) to the saturation regime. A sharper transition can be modeled by introducing a sharpness parameter p :

$$h_{LP}(R_D; R_D^*, p) = 1 + \frac{R_D}{\left(1 + \left(\frac{R_D}{R_D^*}\right)^p\right)^{1/p}}, \quad (20)$$

which preserves the same asymptotic limit $1 + R^*$ as the original h . As shown in Table 14, replacing h with h_{LP} in the M^3 Scaling Law improves test R^2 on the full data (Avg .71 vs. .70), primarily through improved k -axis extrapolation.

However, we do not adopt h_{LP} in the proposed model for two reasons. First, the sharp saturation of h_{LP} introduces non-smooth behavior in the predicted optimal number of epochs k^* (Figure 6): as D_T decreases, k^* rises smoothly until it reaches the saturation knee of h_{LP} , at which point k^* plateaus abruptly before resuming its increase—a pattern not observed in the empirical estimates. Second, the non-smooth transition of h_{LP} seems intuitively unnatural for LLM training, where data repetition effects would be expected to degrade smoothly rather than switching abruptly between a linear and a saturated regime. The original Muennighoff form h , while fitting the per- k data less tightly, produces smoother k^* predictions that better match empirical trends (Figure 4).

E.4 Effect of First-Stage Language Ratio r_1

The dual power law (Eq. 8) formulates the language ratio factor for two-stage training as $\mathcal{R}_{\text{dual}} = r_f^{-\gamma} \cdot (r/r_f)^{-\gamma_2}$, treating the exponent γ_2 as a constant independent of the first-stage language ratio r_1 . As noted in §3.3, Figure 3(b) suggests that r_1 has only a minor effect: lower r_1 (i.e., more high-resource language in the first stage) tends to slightly increase the loss. Here we quantitatively assess whether explicitly modeling this r_1 dependence improves extrapolation accuracy.

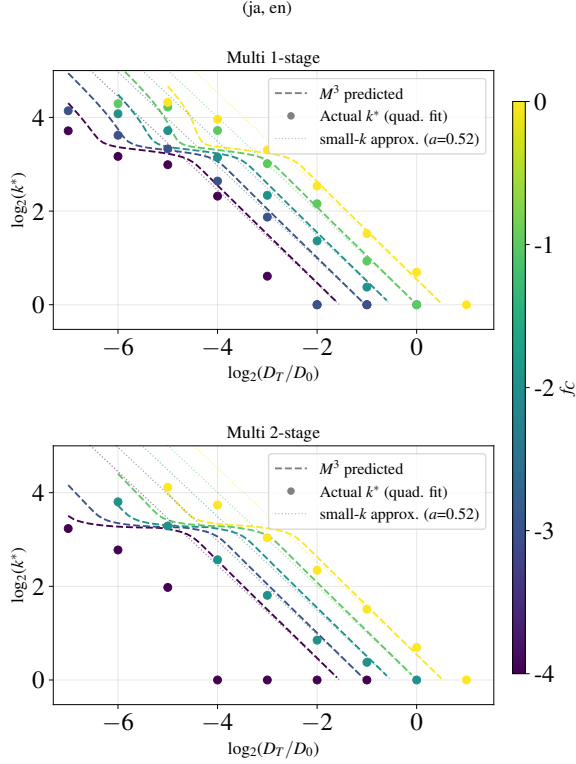


Figure 6: Same as Figure 4 but with h_{Lp} replacing h in the M^3 Scaling Law. The predicted k^* curves (dashed) exhibit non-smooth behavior near the saturation knee of h_{Lp} , unlike the more smooth curves obtained with the original h (Figure 4). Bootstrap bands are not shown here.

We extend γ_2 to depend on r_1 as follows:

$$\tilde{\gamma}_2(r_1) = \gamma_2 + \dot{\gamma}_2 \cdot \ln(r_0 + r_1), \quad (21)$$

where $\dot{\gamma}_2$ controls the strength of r_1 dependence and r_0 is an offset to prevent divergence at $r_1 = 0$. When $\dot{\gamma}_2 = 0$, this reduces to the constant- γ_2 model. The extended dual power law becomes:

$$\mathcal{R}_{\text{dual}}(r, r_1, r_f) = r_f^{-\gamma} \cdot \left(\frac{r}{r_f}\right)^{-\tilde{\gamma}_2(r_1)}. \quad (22)$$

We incorporate this extension into the M^3 Scaling Law (Eq. 10), adding two parameters ($\dot{\gamma}_2$ and r_0) for a total of 13. Table 15 compares the original 11-parameter model against this 13-parameter variant using the same multi-split evaluation protocol described in §3.6.

The r_1 -dependent extension yields no notable improvement in average test R^2 for either language. Based on the result, we conclude that the effect of r_1 on the scaling behavior is negligible at the current experimental scale, and adopt the constant- γ_2 formulation in the M^3 Scaling Law.

E.5 Trade-off Between Modeling Routes for (k, r) Dependence

As discussed in §3.4, the (k, r) dependence of the loss can be carried either through R_M^* (in M') or through the weight g on D_{high} (in D'). We first inspect the per- (k, r) trends of each parameter to motivate the parametric forms, and then report the full fitting experiments behind the trade-off observed between these two routes.

E.5.1 Pilot Analysis: Per- (k, r) Fitting of g and R_M^*

Starting from the multiplicative form of $L = L_{\text{base}}(M', D') \cdot \mathcal{R}_{\text{dual}}(r, r_f)$ with effective sizes $M' = U_M \cdot h(R_M; R_M^*)$ and $D' = D_T \cdot h(R_D; R_D^*) + g \cdot D_{\text{high}}$, we seek to determine how g and R_M^* each depend on (k, r) . We fix the Chinchilla base parameters (A, B, α, β, E) from the monolingual low-epoch ($k \leq 4$) subset and then simultaneously fit the remaining global parameters together with per- (k, r) free parameters on all single-stage data:

Case A. We fix R_M^* as a single shared constant and fit an independent $g_{k,r}$ for each (k, r) with $r < 1$ (at $r = 1$, $D_{\text{high}} = 0$ makes g unidentifiable), jointly with the global parameters R_D^* , R_M^* , and γ .

Case B. We fix $g \equiv 1$ and fit an independent $R_{M,k,r}^*$ for each (k, r) , jointly with the global parameters R_D^* and γ .

Figure 7 shows the results. In Case A (panel (a)), g is close to 1 at $k=1$ across all r , but develops k -dependent variation: for large r ($r=0.5$), g decreases below 1, while for small r ($r=0.125$), g increases well above 1. However, g growing above its $k=1$ baseline of approximately 1 as k increases would counterintuitively imply that repeating data increases the effective data size beyond the non-repeated case. To avoid this, we sought a parametric form that satisfies $g \leq 1$ while allowing the floor to depend on r . This led to the form adopted in Eq. (9), which decays from $g=1$ at $R_D=0$ to an r -dependent floor $(1-r)^\psi < 1$.

In Case B (panel b), R_M^* is large at small k for all r , indicating weak epoch saturation of the effective model size, but decreases with k more rapidly at larger r . This is a natural consequence: when the fraction of repeated target-language tokens is higher (larger r), multi-epoch repetition has a stronger saturation effect on M' . Based on this trend, we model the r -dependence as

$$R_M^*(k, r) = R_M^*(k) \cdot r^{-d}, \quad (23)$$

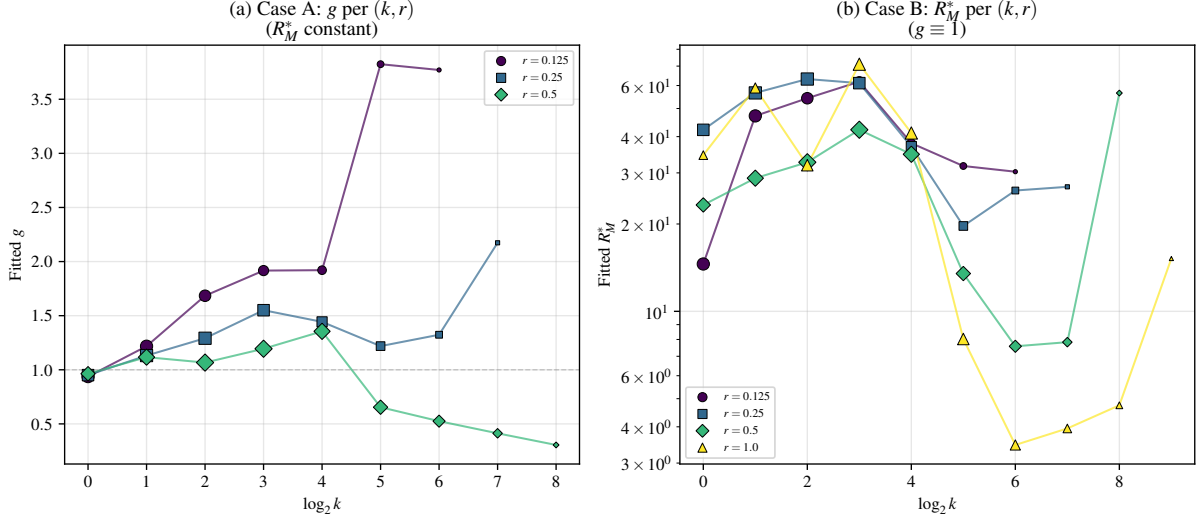


Figure 7: Per- (k, r) fitting of g and R_M^* on (Japanese, English) single-stage data. Marker sizes are proportional to the number of data points for each (k, r) combination. **(a)** Case A: R_M^* is a shared constant; an independent $g_{k,r}$ is fitted for each (k, r) with $r < 1$. At low k , g is close to 1 for all r ; at higher k , g decreases for large r but increases for small r . **(b)** Case B: $g \equiv 1$; an independent $R_{M,k,r}^*$ is fitted for each (k, r) . R_M^* is large at small k across all r , and decreases with k more rapidly at larger r .

where $R_M^*(k)$ is the monolingual form from Eq. (7) and $d > 0$ is a new parameter. This form yields larger R_M^* (weaker saturation) at smaller r , consistent with the observed trend.

E.5.2 Experimental Setup

We compare four combinations of R_M^* and g forms:

- $R_M^*(k, r) + g \equiv 1$: (k, r) dependence is carried entirely by R_M^* , with g fixed to constant. Here, $R_M^*(k, r)$ follows Eq. (23).
- $R_M^*(k, r) + g(r, R_D)$: (k, r) dependence is carried jointly by both routes, with $R_M^*(k, r)$ as above and g following Eq. (9).
- $R_M^*(k) + g(r, R_D)$: R_M^* retains only k -dependence (Eq. 7), with the (k, r) dependence otherwise carried by g .
- $R_M^* = \text{const} + g(r, R_D)$ (**adopted**): R_M^* is a single global constant, with (k, r) dependence carried entirely by g . This is the form adopted in the M^3 Scaling Law (§3.5).

Each variant is fit and evaluated under the same protocol as the main scaling law evaluation (§3.6).

E.5.3 Results

Table 16 summarizes the extrapolation accuracy of each variant. Configurations that place the (k, r) or k dependence on R_M^* (the first three variants) achieve better k -axis extrapolation than the

adopted form, but degrade M -axis extrapolation; the adopted form ($R_M^* = \text{const}$ with $g(r, R_D)$) shows the opposite tendency.

E.5.4 Hypothesis for the Trade-off

We hypothesize that this trade-off stems from an asymmetry between the two routes: R_M^* enters the loss in both monolingual ($r = 1$) and multilingual ($r < 1$) settings, whereas g contributes only through D_{high} and is inactive at $r = 1$. Absorbing (k, r) into R_M^* thus captures the k -dependence from both settings, but couples the D_{high} contribution to the M -axis, which may not transfer to unseen M at $r < 1$ and degrade M -axis extrapolation. Absorbing (k, r) into g instead decouples the (k, r) contribution from the M -axis, yielding more conservative M -axis behavior, at the cost of capturing the k -dependence only in multi-lingual settings and weakening k -axis extrapolation.

E.6 Details of Analysis of Optimal Number of Epochs

E.6.1 Empirical Estimation of k^* via Quadratic Fitting

To estimate the optimal number of epochs $k^*(C, D_T)$ from experimental data, we first compute the minimum validation loss $L^*(C, D_T, k)$ achieved across all training setups sharing the same (C, D_T, k) . For **multi 1-stage**, the minimum is taken over model scale M and r for single-stage

setups; for **multi 2-stage**, it is taken over M , r , r_1 , and r_2 including both single-stage and two-stage setups.

We then fit a quadratic function $L^* = p_0 f_k^2 + p_1 f_k + p_2$ in $f_k = \log_2 k$ to the resulting (f_k, L^*) pairs for each (C, D_T) , and estimate k^* as the vertex of the fitted parabola, i.e., $f_k^* = -p_1/(2p_0)$ and $k^* = 2^{f_k^*}$. This smoothing mitigates the impact of noise and the limited resolution of our k grid ($k \in \{1, 2, 4, \dots\}$). We exclude estimates where (i) the fitted parabola has a maximum rather than a minimum ($p_0 \leq 0$) within the observed range, indicating a noisy fit, or (ii) the vertex lies below the smallest observed f_k and $k=1$ was not tested, as the true optimum cannot be reliably determined. When the vertex falls at or below $f_k=0$ and $k=1$ is included in the search grid, we clamp k^* to 1. Figure 14 and Figure 15 illustrate the quadratic fits for the (Japanese, English) and (Indonesian, English) pairs respectively, showing the fitted curves together with the observed $L^*(C, D_T, k)$ values.

E.6.2 Results for Indonesian

Figure 8 shows the optimal number of epochs analysis for the (Indonesian, English) pair. We do not present optimal epoch analysis for the (Swahili, English) pair, as the available D_T range is too narrow to investigate the dependence of k^* on D_T across compute budgets.

E.6.3 Derivation of Optimal Number of Epochs

We now prove Eq. (13) for the M^3 Scaling Law in Eqs. (10)–(12). Since M is defined as the non-embedding FLOPs per token, the total compute is simply

$$C = MD, \quad D = \frac{kD_T}{r}. \quad (24)$$

In the derivation below, we restrict attention to the fixed- r setting and optimize only over k under the compute constraint. This directly covers single-stage training, for which $r_f = r$; in particular, the monolingual regime corresponds to $r = 1$. We state the argument for an arbitrary fixed r because the same homogeneity proof goes through without modification, but we do not claim here that the globally optimal r remains fixed as D_T changes. Thus, the proof is exact for fixed- r settings (especially mono 1-stage), whereas the collapse observed for jointly optimized multi 2-stage training should be interpreted as empirical.

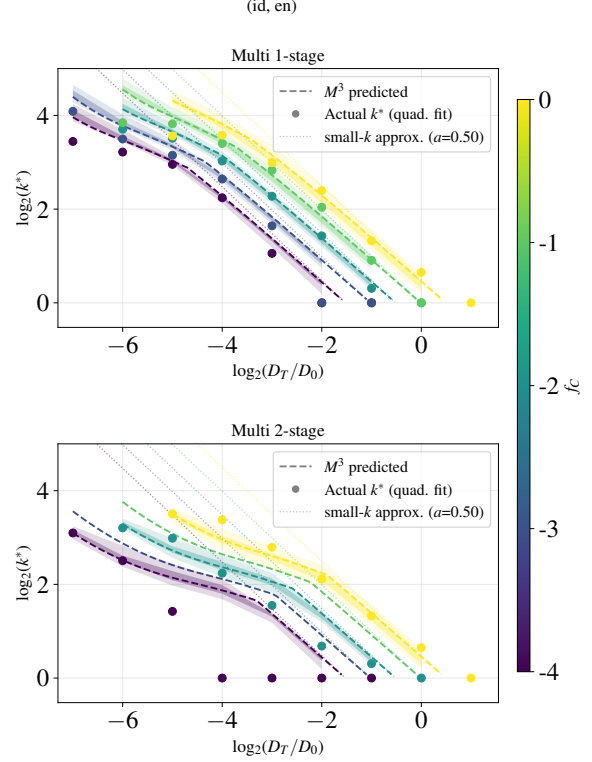


Figure 8: Same as Figure 4 but for the (Indonesian, English) pair.

Proposition. Fix a ratio schedule $\mathbf{r}$. Then, for the M^3 Scaling Law, the argmin over k depends on (C, D_T) only through the scarcity variable $D_T/D^*(C)$, where $D^*(C)$ is the monolingual single-epoch Chinchilla-optimal data size. When the minimizer is unique, this is written as

$$k_{\mathbf{r}}^*(C, D_T) = K_{\mathbf{r}} \left(\frac{D_T}{D^*(C)} \right). \quad (25)$$

Proof. Let

$$x = \frac{D_T}{C^{\alpha/(\alpha+\beta)}}. \quad (26)$$

Then

$$\begin{aligned} D_T &= C^{\alpha/(\alpha+\beta)} x, \\ M &= \frac{Cr}{kD_T} = C^{\beta/(\alpha+\beta)} \frac{r}{kx}. \end{aligned} \quad (27)$$

For brevity, write

$$g_k(r) = g(r, k-1; R_{D, \text{high}}^*, \psi). \quad (28)$$

Using Eq. (11) and $D_{\text{high}} = kD_T(1-r)/r$, the

effective data size becomes

$$\begin{aligned}
D' &= D_T \cdot h(k-1; R_D^*) + g_k(r) \cdot \frac{kD_T(1-r)}{r} \\
&= D_T \left[h(k-1; R_D^*) + \frac{k(1-r)}{r} g_k(r) \right] \\
&\equiv D_T Q_D(k; r) \\
&= C^{\alpha/(\alpha+\beta)} x Q_D(k; r),
\end{aligned} \tag{29}$$

where

$$Q_D(k; r) = h(k-1; R_D^*) + \frac{k(1-r)}{r} g_k(r). \tag{30}$$

Hence

$$\frac{1}{D'^\beta} = C^{-\alpha\beta/(\alpha+\beta)} \frac{1}{x^\beta Q_D(k; r)^\beta}. \tag{31}$$

Next, recall that

$$U_M = \min\left(G^{(\alpha+\beta)/\alpha} D_T^{\beta/\alpha}, M\right), \tag{32}$$

where

$$G = \left(\frac{\alpha A}{\beta B}\right)^{1/(\alpha+\beta)}. \tag{33}$$

Define

$$Q_U(x, k; r) = \min\left(G^{(\alpha+\beta)/\alpha} x^{\beta/\alpha}, \frac{r}{kx}\right). \tag{34}$$

Then

$$\begin{aligned}
U_M &= \min\left(G^{(\alpha+\beta)/\alpha} C^{\beta/(\alpha+\beta)} x^{\beta/\alpha}, \right. \\
&\quad \left. C^{\beta/(\alpha+\beta)} \frac{r}{kx}\right) \\
&= C^{\beta/(\alpha+\beta)} Q_U(x, k; r).
\end{aligned} \tag{35}$$

Therefore,

$$R_M = \frac{M}{U_M} - 1 = \frac{r}{kx Q_U(x, k; r)} - 1, \tag{36}$$

which depends only on (x, k, r) .

Using Eq. (12),

$$\begin{aligned}
M' &= U_M \cdot h(R_M; R_M^*) \\
&= C^{\beta/(\alpha+\beta)} Q_U(x, k; r) h(R_M; R_M^*) \\
&\equiv C^{\beta/(\alpha+\beta)} Q_M(x, k; r),
\end{aligned} \tag{37}$$

where $Q_M(x, k; r)$ is independent of C and D_T . Hence

$$\frac{1}{M'^\alpha} = C^{-\alpha\beta/(\alpha+\beta)} \frac{1}{Q_M(x, k; r)^\alpha}. \tag{38}$$

Substituting the above expressions into Eq. (10) yields

$$\begin{aligned}
L_{M^3} &= \mathcal{R}_{\text{dual}}(r, r_f) \\
&\quad \times \left[C^{-\alpha\beta/(\alpha+\beta)} \Phi(x, k; r) + E \right],
\end{aligned} \tag{39}$$

where

$$\Phi(x, k; r) = \frac{A}{Q_M(x, k; r)^\alpha} + \frac{B}{x^\beta Q_D(k; r)^\beta}. \tag{40}$$

Because $\mathcal{R}_{\text{dual}}(r, r_f) > 0$, $C^{-\alpha\beta/(\alpha+\beta)}$, and E are all independent of k , minimizing L_{M^3} over k is equivalent to minimizing $\Phi(x, k; r)$ over k . Therefore, the argmin over k depends on (C, D_T) only through

$$x = \frac{D_T}{C^{\alpha/(\alpha+\beta)}}. \tag{41}$$

It remains to connect this variable to $D^*(C)$. Under monolingual single-epoch Chinchilla training,

$$L_{\text{base}}\left(\frac{C}{D}, D\right) = A \left(\frac{D}{C}\right)^\alpha + \frac{B}{D^\beta} + E. \tag{42}$$

Its first-order condition is

$$\alpha A C^{-\alpha} D^{\alpha-1} = \beta B D^{-\beta-1}, \tag{43}$$

which gives

$$\begin{aligned}
D^*(C) &= \left(\frac{\beta B}{\alpha A}\right)^{1/(\alpha+\beta)} C^{\alpha/(\alpha+\beta)} \\
&= G^{-1} C^{\alpha/(\alpha+\beta)}.
\end{aligned} \tag{44}$$

Thus,

$$x = \frac{D_T}{C^{\alpha/(\alpha+\beta)}} = G^{-1} \cdot \frac{D_T}{D^*(C)}. \tag{45}$$

Since the multiplicative constant G^{-1} can be absorbed into the definition of the function, the argmin over k depends on (C, D_T) only through $D_T/D^*(C)$. When the minimizer is unique, this proves Eq. (13). $\square$

The same derivation also applies to the monolingual variant $M^3 + R_M^*(k)$, because replacing the constant R_M^* in Eq. (12) by the k -dependent $R_M^*(k)$ in Eq. (7) still does not introduce any direct dependence on C or D_T .

Remark. The proposition above is exact only for fixed ratio schedule $\mathbf{r}$. If $\mathbf{r}$ is jointly optimized together with k , the same collapse does not follow from this argument alone. Therefore, for **multi 1-stage** and **multi 2-stage**, Eq. (13) should be interpreted as an empirical approximation rather than a formal invariance result.

Corollary (small- k regime). Assume that the saturation effects in both D' and M' are negligible. Then, for fixed $\mathbf{r}$, the M^3 Scaling Law reduces to the Chinchilla form with total token count $D = kD_T/r$ multiplied by a constant $\mathcal{R}_{\text{dual}}$ factor, and the optimal number of epochs is approximately proportional to $D^*(C)/D_T$.

Proof. When $k - 1 \ll R_D^*$, the saturation in the target-language effective data is negligible, and

$$h(k - 1; R_D^*) \approx k. \quad (46)$$

Similarly, when $k - 1 \ll R_{D,\text{high}}^*$, the high-resource weight in Eq. (9) satisfies

$$g(r, k - 1; R_{D,\text{high}}^*, \psi) \approx 1. \quad (47)$$

Therefore,

$$\begin{aligned} D' &= D_T \cdot h(k - 1; R_D^*) + g \cdot D_{\text{high}} \\ &\approx kD_T + D_{\text{high}} \\ &= kD_T + \frac{kD_T(1 - r)}{r} \\ &= \frac{kD_T}{r} = D. \end{aligned} \quad (48)$$

If the model-side saturation is also negligible, then

$$M' \approx M. \quad (49)$$

Hence Eq. (10) reduces to

$$L_{M^3} \approx \left(\frac{A}{M^\alpha} + \frac{B}{D^\beta} + E \right) \mathcal{R}_{\text{dual}}(r, r_f). \quad (50)$$

For fixed $\mathbf{r}$, the factor $\mathcal{R}_{\text{dual}}(r, r_f)$ is constant with respect to k . Therefore, minimizing L_{M^3} over k is asymptotically equivalent to minimizing the Chinchilla base loss over the total number of tokens D , whose optimum is $D^*(C)$. Using $D = kD_T/r$, we obtain

$$k_{\mathbf{r}}^*(C, D_T) \approx \frac{r D^*(C)}{D_T}. \quad (51)$$

In particular, for monolingual single-stage training ($r = 1$),

$$k^*(C, D_T) \approx \frac{D^*(C)}{D_T}. \quad (52)$$

This proves the unit-slope asymptote in the log-log plot. $\square$

Remark. Equation (52) is the monolingual special case of Eq. (51). Thus, the statement $k^*(C, D_T) \approx D^*(C)/D_T$ is exact only in the monolingual small- k regime. For fixed multilingual $r < 1$, the corresponding approximation is $k_{\mathbf{r}}^*(C, D_T) \approx r D^*(C)/D_T$. If r is also jointly optimized, the small- k approximation becomes

$$k^*(C, D_T) \approx \frac{r^*(C, D_T) D^*(C)}{D_T}. \quad (53)$$

E.7 Additional Predictions of the M^3 Scaling Law

E.7.1 Optimal Model Scale

Figure 9 shows the optimal model scale M^* predicted by the M^3 Scaling Law as a function of D_T for each compute budget C . We numerically minimized the predicted validation loss over all training approaches (mono 1-stage, multi 1-stage, and multi 2-stage) and their hyperparameters (r, k, r_2) for each (C, D_T) , and report the model scale M that achieved the minimum predicted loss. For comparison, we show the Chinchilla compute-optimal model scale $M^*(C) = G \cdot C^{\beta/(\alpha+\beta)}$, as well as two empirical estimates of M^* from the experimental data. The first estimate (squares) fits a quadratic function in f_M to the minimum validation loss at each model scale: for each (C, D_T) , we compute $\min_{r,k,r_2} L$ at each f_M across all training approaches and setups sharing the same (C, D_T) , fit a quadratic function $L_{\min}(f_M) = p_0 f_M^2 + p_1 f_M + p_2$ to these per- f_M minima, and take the vertex $f_M^* = -p_1/(2p_0)$ as the estimated optimal (clamped to the observed range of f_M). The second estimate (triangles) simply takes the f_M that achieves the lowest minimum validation loss on the discrete experimental grid.

Across all three language pairs and compute budgets, the M^* predicted by the M^3 Scaling Law remains relatively stable as D_T decreases, and the prediction error remains within the experimental grid resolution ($\Delta f_M = 1$) for most (C, D_T) conditions. In the regime where the optimal approach is monolingual single-stage (large D_T), the predicted M^* increases gradually as D_T decreases; after the optimal approach switches to multi-lingual two-stage training, M^* tends to decrease slightly. However, these predicted variations are smaller than the grid resolution of model scales in our experiments, so we cannot empirically confirm this trend.

Overall, these results suggest that, in practice,

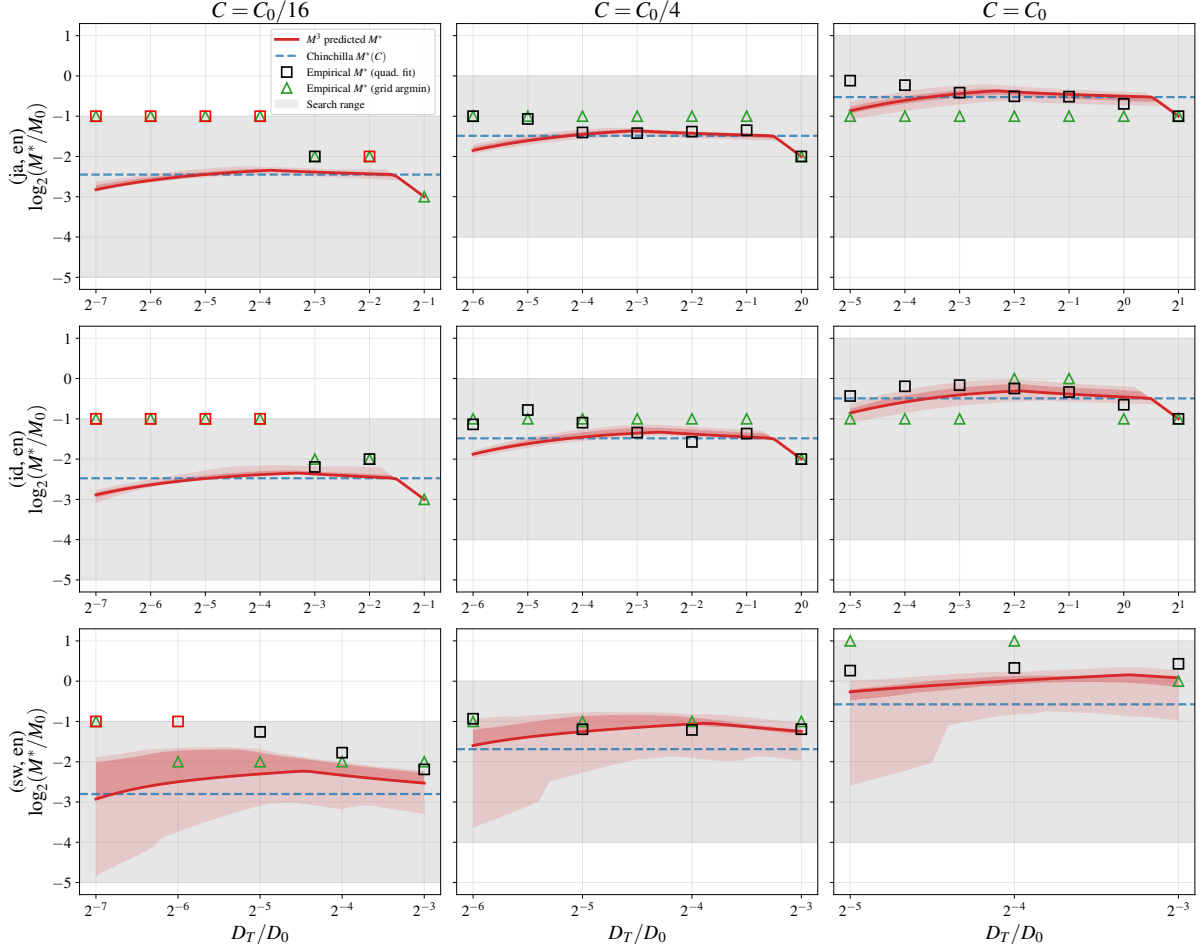


Figure 9: Optimal model scale M^* predicted by the M^3 Scaling Law (red solid) compared with the Chinchilla compute-optimal scale $M^*(C)$ (blue dashed), empirical M^* estimated by quadratic fitting in f_M (squares) and discrete grid argmin (triangles), across three language pairs and three compute budgets. The gray band indicates the range of model scales explored in our experiments. Shading around the predicted curve shows its bootstrap variability (Appendix D.4). Red-bordered squares indicate points where the quadratic fit optimum lies at or beyond the search boundary.

the compute-optimal model scale computed from the standard monolingual data-rich setting serves as a good starting point for low-resource language LLM pretraining, even when D_T is substantially smaller than the compute-optimal data size $D^*(C)$.

E.7.2 Optimal Language Ratio

Figure 10 shows the optimal language ratio r^* predicted by the M^3 Scaling Law for each (C, D_T) . For each condition, we numerically minimized the predicted loss over all training approaches and their hyperparameters (M, k, r_2) , and report the r that achieved the minimum. The horizontal axis is normalized by the compute-optimal data size $D^*(C)$, which is estimated by minimizing the M^3 Scaling Law over M under monolingual single-epoch training ($r = 1, k = 1$) at each C . For comparison, we also plot the empirical r^* obtained by taking the r

value that achieves the minimum validation loss on the discrete experimental grid for each (C, D_T) .

Across all three language pairs, the predicted r^* remains at $r^* = 1$ (monolingual training is optimal) as D_T decreases from $D^*(C)$, until a threshold is reached at which r^* begins to decrease. Below this threshold, the optimal approach is multilingual two-stage training with $r_f = 1$ (§4.1), and $\log_2 r^*$ decreases approximately linearly in $\log_2(D_T/D^*(C))$. The threshold at which this transition occurs approximately aligns across different C when expressed in terms of $D_T/D^*(C)$, but does not collapse perfectly: a residual shift remains that is approximately equally spaced in $\log_2 C$.

The predictions of the M^3 Scaling Law broadly capture the trend of the empirical optimal r^* . However, due to the coarse resolution of the language

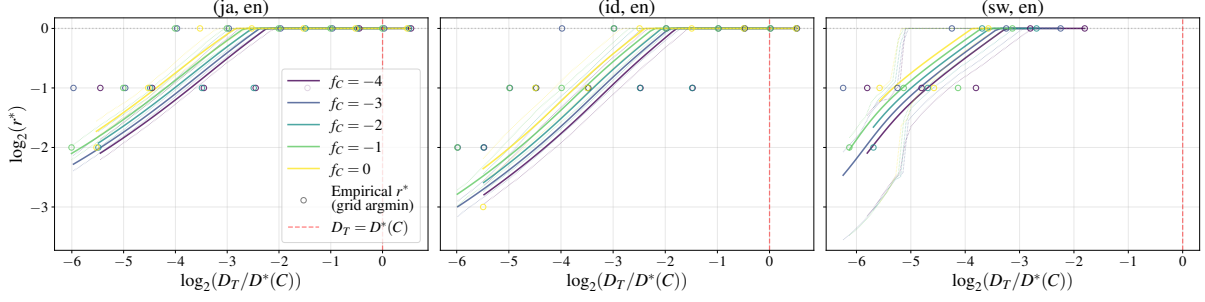


Figure 10: Optimal language ratio r^* predicted by the M^3 Scaling Law (solid lines) and empirical r^* from discrete grid argmin (circles) as a function of $\log_2(D_T/D^*(C))$ for each compute budget C (colored by f_C). Dotted outlines mark the 5th and 95th percentiles of the bootstrap distribution (Appendix D.4). The red dashed line marks $D_T = D^*(C)$.

ratio grid ($r \in \{1, 1/2, 1/4, 1/8\}$) and the limited range of C explored in our experiments, we cannot confirm from the experimental data alone whether the C -dependence of the residual shift of the threshold is a genuine trend or whether the linearity in log-log space holds exactly.

F Recipe Selection from Small-Scale Pilot Experiments

To apply the M^3 Scaling Law to a new language pair, one pretrains a set of small-scale pilot models on that pair, fits the M^3 Scaling Law to their losses, and numerically minimizes the predicted loss over recipes at the budget constraint (C, D_T) of interest. The practical bottleneck is the amount of pilot training required. In this section we ask how far these pilot runs can be reduced, taking the grid of setups trained in our experiments as the pool of pilot candidates. We compare two families of reduction: random reduction, which randomly subsamples the pool down to a fixed fraction of the compute, and rule-based reduction derived from our findings. Each is assessed by the quality of the recipe selected after the reduction.

To reflect the constraints faced in practice, we set the budget constraint of interest to $C = C_0$ with $f_D \in \{0, -1, -2\}$, and restrict the pilot runs to setups that are at least four times cheaper in compute ($f_C \leq -2$) and whose target-language corpus is no larger than D_T . Let L_{sub} denote the M^3 Scaling Law fitted to a subset of the pilot runs, and L_{full} the one fitted to all of our data. We define

the recipe-selection regret as

$$\theta_{\text{sub}} = \arg \min_{\theta} L_{\text{sub}}(C, D_T, \theta), \quad (54)$$

$$\text{regret} = L_{\text{full}}(C, D_T, \theta_{\text{sub}}) - \min_{\theta} L_{\text{full}}(C, D_T, \theta). \quad (55)$$

The recipe θ optimized here is the triple (M, r, r_f) of the model scale, the average target-language ratio, and the final-stage ratio. Our experimental grid is coarse, with factor-of-two spacing, and each setup is a single run, so the observed losses are noisy. To measure the gap to the best recipe over a continuous recipe space, we therefore use the loss surface of the M^3 Scaling Law fitted to all data, rather than the observed losses, as the reference in the regret. We repeat each fit with ten random seeds for the rule-based reductions, and with two seeds for each of twenty subsamples for the random reductions.

Random reduction selects pilot setups until their total compute reaches a target fraction of that of the full pool. We set the target fraction to 5, 10, 20, and 40%, and compare three selection schemes. **Uniform** draws setups uniformly at random. **Stratified** treats each (f_C, f_D) cell as a stratum, allocates the reduced compute budget across the cells in proportion to their total compute, and draws uniformly at random within each cell. **Balanced** greedily selects setups one at a time: at each step it picks the setup that minimizes the total number of times its five factor levels $(f_C, f_D, f_r, f_M, f_k)$ have already been selected, with ties broken uniformly at random. This keeps the marginal distribution of each factor as close to uniform as possible, but does not control the interactions between factors.

Rule-based reduction instead restricts the pool to the region in which the optimal recipe is expected

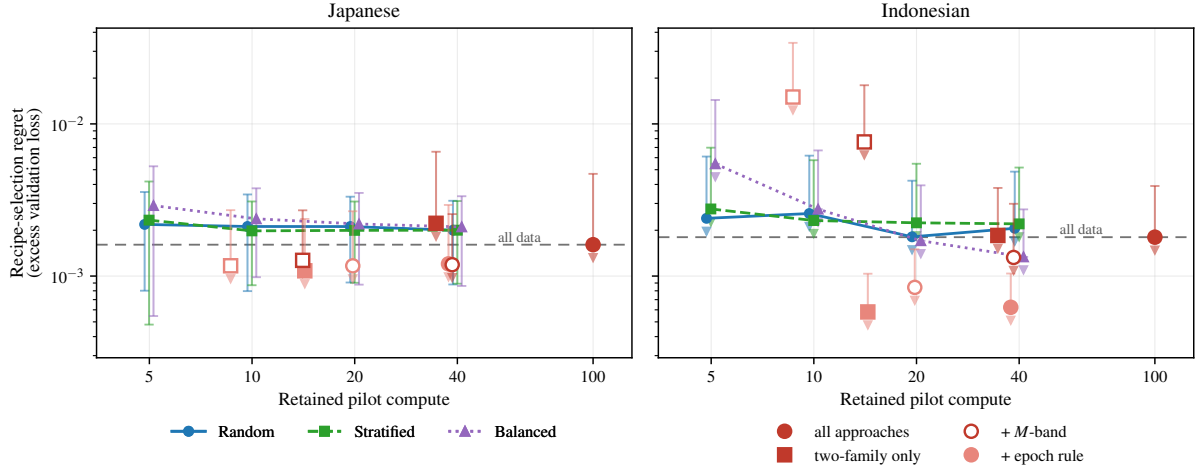


Figure 11: Recipe-selection regret against the pilot compute retained after the reduction, for (Japanese, English) and (Indonesian, English). Lines show random reduction (uniform / stratified / balanced); markers show rule-based reduction, where the shape encodes the candidate set (circle: all approaches, square: **two** only), an open marker indicates the **mband** rule, and a lighter color indicates the **epoch** rule. The gray dashed line marks the level attained with the full pool of pilot candidates. Error bars are ± 1 standard deviation, pooled over the three budget constraints, the fitting seeds, and the subsamples; a downward tick indicates that the lower end of the interval falls below the axis, i.e., the regret is not distinguishable from zero. The points of the random reduction are offset horizontally for legibility. Both axes are logarithmic.

to lie, based on our findings. We evaluate three binary rules and all eight combinations of them. **two** restricts the pilot candidates to mono 1-stage and multi 2-stage with $r_f = 1$ (§4.1). **mband** restricts the model scale to between $1/2$ and 2 times the Chinchilla-optimal scale $M^*(C)$. **epoch** restricts the retained runs according to the ratio $\rho = D^*(C)/D_T$ (§4.2): for $\rho \leq 4$, only mono 1-stage runs with $\rho/2 \leq k \leq 2\rho$; for $\rho > 4$, mono 1-stage and 2-stage runs with $4 \leq k \leq \rho$. These bounds follow from the approximation $k^* \approx D^*(C)/D_T$. For $\rho \leq 4$, where k^* empirically follows this approximation, we keep k within a factor of two of ρ , and restrict the runs to mono 1-stage because the gain from multi-lingual data is small in this data-rich regime. For $\rho > 4$, where k^* falls progressively below ρ , we take $k \leq \rho$ as the upper bound, and, since k^* increases monotonically in ρ , we take $k^* \approx 4$ at $\rho = 4$ as the lower bound.

The monolingual, low-epoch runs ($r = 1$, $k \leq 4$) used in Phase 1 of the fitting procedure (§D.2.4) account for only a small fraction of the total compute of the pool, but without them the Chinchilla base parameters cannot be estimated and the fit cannot be carried out. Removing them also tended to increase the regret in preliminary experiments. We therefore retain these runs under every scheme in the comparison below.

Figure 11 plots the regret against the pilot com-

pute retained after the reduction. Under both random and rule-based reduction, the regret stays at the level attained with the full pool, even at the smallest retained fraction we tested: 5% of the total compute for random reduction, and about 9% for the most aggressive combination of rules. The differences between the reduction schemes, as well as those from the full pool, mostly lie within the error bars, and we do not identify a scheme that is consistently better than the others. The pilot runs can therefore be reduced substantially without degrading the quality of the recipe selection, and our results suggest that simple random sampling suffices as the reduction scheme, as long as the Phase 1 runs are retained.

G Run-to-Run Variation of the Validation Loss

Because of compute constraints, every setup in our experiments is trained once, so each observed validation loss carries the run-to-run variation induced by the randomness of training. To measure the size of this variation directly, we selected a subset of the setups in our search space and retrained each of them several times, changing only the random seed.

The subset consists of 30 setups, 15 single-stage and 15 two-stage, selected so as not to concen-

trate in any particular region of the search space; they are listed in Table 18. The same 30 setups were trained for both the (Japanese, English) and the (Indonesian, English) pair. Only the random seed differs between the runs of a setup, and it determines the model initialization together with the order and sampling of the training data. The validation set is identical across runs, so what we measure is the variation on the training side alone. Each setup was trained 5 times.

For each setup we computed the standard deviation s of the validation loss across its runs and the coefficient of variation $CV = s/\bar{L}$, where $\bar{L}$ is the mean loss over those runs. Table 17 aggregates these per-setup values by group, and Table 18 reports them for every setup.

The median CV is 0.35% over all setups, which corresponds to a standard deviation of 0.010. It stays at a similar level for both language pairs and for both single-stage and two-stage setups.

H Downstream Task Performance

H.1 Justification of Using Validation Loss as the Primary Metric

While the most important aspect of LLM applications is downstream task performance, our choice of validation loss on language modeling tasks as our primary evaluation metric is motivated by three key factors: (1) due to the emergent nature of LLM capabilities (Wei et al., 2022), analyzing scaling behaviors of downstream task performance with smaller-scale experiments (10M parameters, 200M tokens) is challenging and still under research (Hu et al., 2024a), (2) validation loss has been shown to correlate well with downstream task performance in multiple studies (Brandfonbrener et al., 2024; Chen et al., 2025; Du et al., 2024; Gadre et al., 2025; Thrush et al., 2025), making it a reliable proxy for downstream capabilities despite some variance observed in models with similar validation losses (Xu et al., 2026), and (3) it has been widely adopted in existing works that analyze scaling behaviors of LLMs (Kaplan et al., 2020; Hoffmann et al., 2022; Bi et al., 2024). More complete analysis including comprehensive downstream task analysis remains an important avenue for future work. See §H.2 for our preliminary empirical analysis.

H.2 Preliminary Attempts for Evaluating Downstream Task Performance

While analyzing downstream model performance for our smaller-scale models is challenging due to the emergent nature of LLM capabilities (Wei et al., 2022), we conducted a preliminary analysis to investigate relationship between validation loss and downstream task performance.

We evaluated Japanese LLMs trained in our experiments on the following Japanese downstream task benchmarks⁷:

- Question Answering: JAQKET (Suzuki et al., 2020), JEMHopQA (Ishii et al., 2024), and NIILC (Sekine, 2003)
- Reading Comprehension: JSQuAD (Kurihara et al., 2022)

We also evaluated the models on other natural language inference, entity linking, and commonsense reasoning tasks in `llm-jp-eval`, but exact match scores of these tasks were almost consistently 0% due to the small scale of our models, showcasing difficulties of downstream task evaluation and making these tasks unsuitable for our analysis. We used the implementation of `llm-jp-eval` (v1.4.1, Apache-2.0 license) (LLM-jp, 2024) and set the maximum sequence length to 4096, the number of few shot examples to 16, and the maximum number of samples to 100. We report character F1 scores for these tasks.

As shown in Table 19, we observed moderate negative correlations between validation loss and task performance for most tasks, with a correlation coefficient of -0.696 for the average score across all tasks. While NIILC showed relatively weak correlation (-0.117), other tasks demonstrated moderate negative correlations, ranging from -0.378 to -0.611. This result suggests that improvements in validation loss generally correspond to better downstream task performance.

Furthermore, as shown in Figure 12, both single-stage and two-stage training approaches exhibit similar patterns in the relationship between validation loss and downstream task performance. This similarity supports our comparative analysis of different training approaches based on validation loss.

It is important to note that this correlation analysis was conducted using relatively small-scale

⁷JAQKET, JEMHopQA, NIILC, and JSQuAD are licensed under the CC BY-SA 4.0 license.

models with limited absolute performance on these tasks. While our results suggest the validity of using validation loss as a proxy metric for model comparison, future work with larger-scale models would be necessary for a more precise analysis of this relationship.

I Full Experimental Results

Figure 13 shows the minimum validation loss on the target language achieved by different LLM training approaches across various budget constraints (C, D_T) .

J License Notice

This paper contains information from C4 and mC4 corpus⁸ which is made available under the ODC Attribution License.

⁸Available at <https://huggingface.co/datasets/allenai/c4>.

Compute, data, and model sizes	
C	Compute budget (FLOPs; $C=MD$)
C_0	Reference compute budget (10^{18})
D	Total training tokens (kD_T/r)
$D_T, D_{T,0}$	Target-language corpus size; reference
D_{high}	High-resource tokens ($kD_T(1-r)/r$)
D_1, D_2	Stage-1/2 tokens (s_1D, s_2D)
M, M_0	Model scale (non-emb. FLOPs/token); ref.
N	Model parameters (PTPP baselines only)
S	Sequence/context length ($= 4096$)
$n, d_{\text{model}}, n_{\text{heads}}$	Layers, hidden dim, attention heads
Training-recipe hyperparameters	
k	Number of target-corpus epochs
r	Average target-language ratio (kD_T/D)
r_1, r_2, r_3	Per-stage target-language ratios
r_f	Final-stage target-language ratio
r_{12}	Average ratio over stages 1–2
s_1, s_2, s_3	Stage proportions rel. to total length
s_{12}	Combined proportion of stages 1–2 (s_1+s_2)
$\mathbf{r}$	Ratio schedule (r, r_1, r_2)
$\eta, \mathcal{B}$	Learning rate, batch size
Search-grid factors	
f_C	Compute factor: $C=2^{f_C} \cdot C_0$
f_M	Model-scale factor: $M=M_0/2^{f_M}$
f_k	Epoch factor: $k=2^{f_k}$
f_r	Language-ratio factor: $r=1/2^{f_r}$
f_D	Target-corpus factor: $f_D=-f_r+f_M-f_k+f_C, \quad D_T=2^{f_D} \cdot D_{T,0}$
Chinchilla base scaling law	
$L_{\text{base}}(M, D)$	$A/M^\alpha + B/D^\beta + E$
A, B	Coefficients of model-/data-size terms
α, β	Exponents of model/data size
E	Irreducible loss
G	$(\alpha A/\beta B)^{1/(\alpha+\beta)}$
$D^*(C)$	Compute-optimal corpus size, $G^{-1}C^{\alpha/(\alpha+\beta)}$
$M^*(C)$	Compute-optimal model scale, $G \cdot C^{\beta/(\alpha+\beta)}$
Effective sizes and saturation	
D', M'	Effective data/model size
$\hat{M}'$	Empirically estimated M' (inverted from loss)
U_M	Compute-optimal model size at D_T
R_D	Data repetition count ($k-1$)
R_M	Excess model size (M/U_M-1)
$h(R; R^*)$	Saturation function, $1+R^*(1-e^{-R/R^*})$
R_D^*, R_M^*	Saturation constants for D', M'
$R_M^*(k)$	k -dep. form (mono): $a/(k-1)^b+c$
a, b, c	Parameters of $R_M^*(k)$
$R_{D,\text{high}}^*$	Saturation const. for D_{high} in g
h_{Lp}	Sharper saturation variant (Appendix)
p	Sharpness parameter in h_{Lp}

Language-ratio factor	
$\mathcal{R}(r)$	He ratio factor, $r^{-\gamma}$
$\mathcal{R}_{\text{dual}}(r, r_f)$	Dual power law (single-/multi-stage)
γ	Exponent for language ratio (single-stage)
γ_2	Exponent for r/r_f (multi-stage)
$\hat{\gamma}_2(r_1)$	r_1 -dependent extension (Appendix)
$\hat{\gamma}_2, r_0$	Slope and offset in $\hat{\gamma}_2$
High-resource token weight	
g	Weight on D_{high} in D'
$g(r, R_D)$	Functional form of g : $(1-r)^\psi + [1-(1-r)^\psi] \exp(-R_D/R_{D,\text{high}}^*)$
ψ	Exponent controlling lower bound of g
τ	Constant weight in ATLAS baseline
Loss-prediction models	
L	Target-language validation loss
$L_{\text{He}}, L_{\text{Mu}}$	He / Muennighoff scaling laws
L_{M^3}	Proposed M^3 Scaling Law
L_{actual}	Observed validation loss
$L^*(C, D_T, k)$	Min. loss over remaining hyperparams
Optimization targets	
$k^*, k_{\mathbf{r}}^*$	Optimal epochs; at fixed schedule $\mathbf{r}$
$K_{\mathbf{r}}$	Collapse function: $k_{\mathbf{r}}^*=K_{\mathbf{r}}(D_T/D^*)$
r^*, M^*	Optimal ratio; optimal model scale
x	Scarcity variable, $D_T/C^{\alpha/(\alpha+\beta)}$
Φ, Q_D, Q_U, Q_M	Helpers in optimal- k derivation
Model fitting and evaluation	
θ, s_i	Scaling-law params; training setting of run i
$L_i, \hat{L}_i$	Observed / predicted loss for run i
$\mathcal{T}, \bar{L}_{\mathcal{T}}$	Test set; test-set mean loss
R^2	Coefficient of determination
Huber $_{\delta}$	Huber loss ($\delta=0.001$)
p_0, p_1, p_2	Quadratic-fit coefficients for k^*
ρ	Pearson correlation (downstream)
Baseline-only symbols	
Sedova: $E_s, C_s, B_s, \alpha_s, \beta_s, \delta_s, \gamma_s, \tau_s, R_{D,s}^*, D_{\text{sc}}^*$	
D-CPT: C_c, ν	
PTPP: $\text{PTPP}=D_1/N$; F, ξ (floor coeff./exp.); $\lambda, \zeta, \beta_{\text{eff}}$	
Zhang: ϕ_1, ϕ_2 (per-stage data exponents)	
Optimizer (Appendix)	
$\beta_1, \beta_2, \epsilon$	Adam parameters

Table 3: Summary of notation.

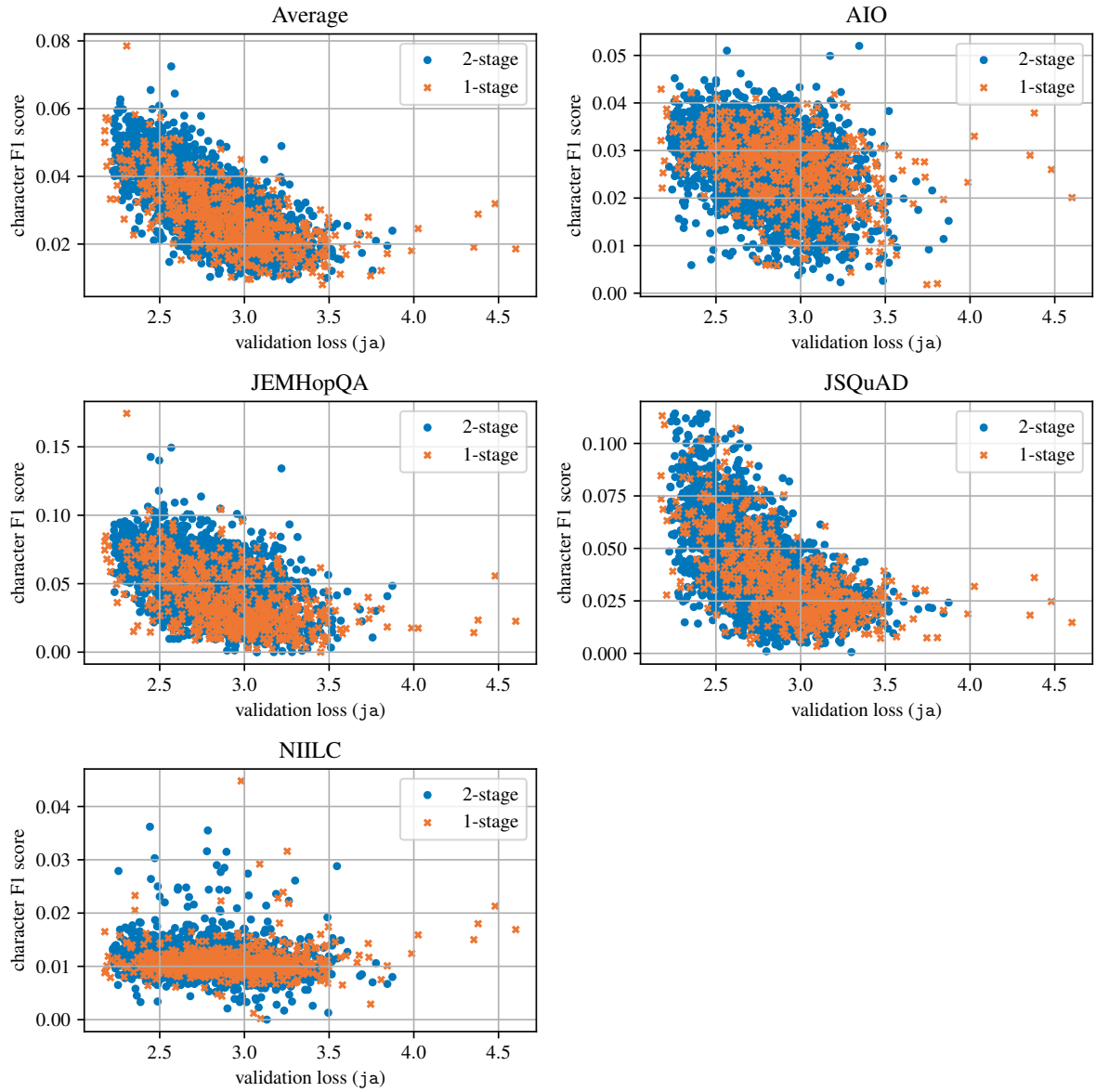


Figure 12: Relationship between validation loss and downstream task performance (character F1 scores). Different colors represent single-stage (orange) and two-stage (blue) training approaches, showing similar correlation patterns regardless of the training strategy.

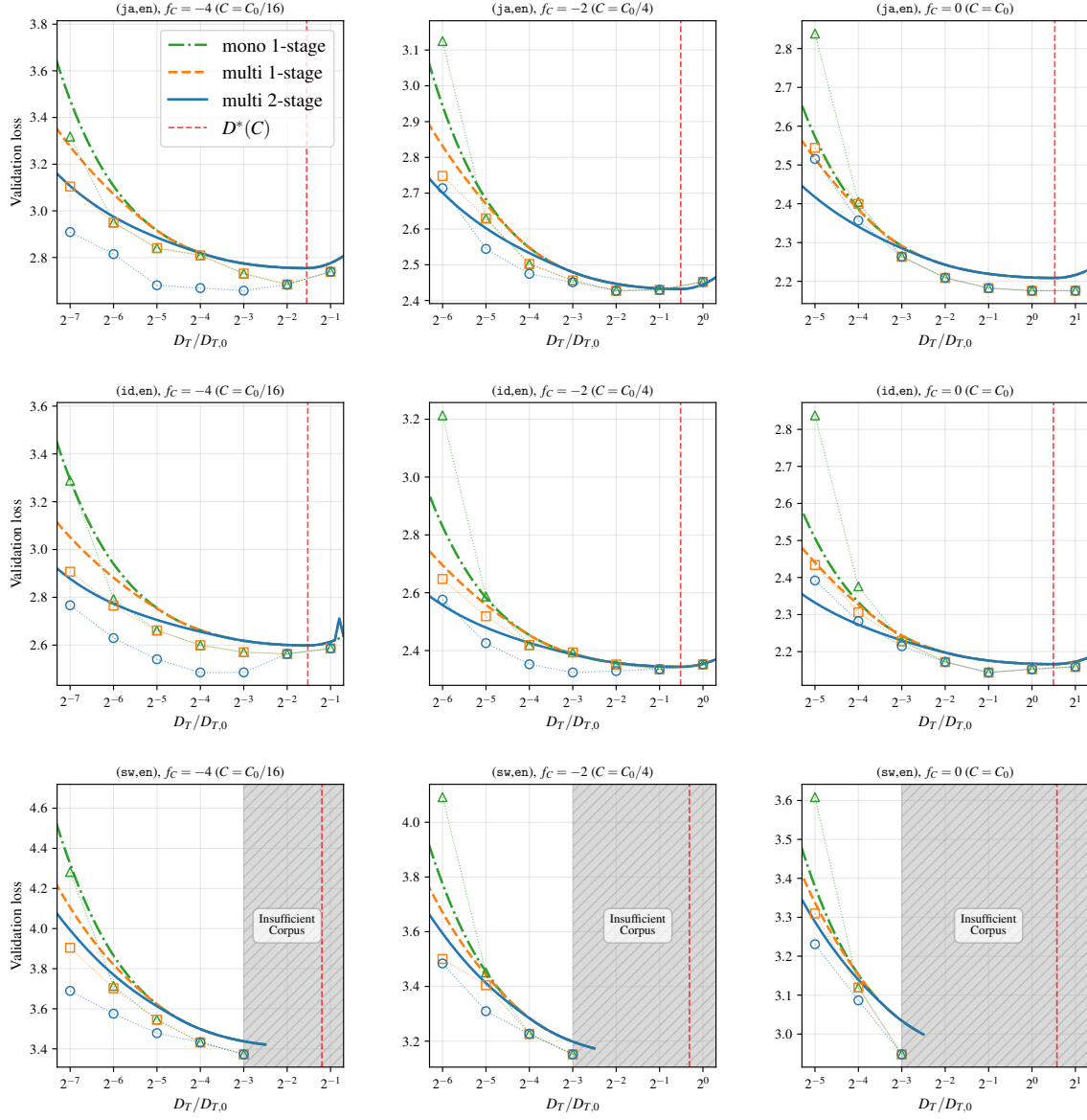


Figure 13: Minimum validation loss for each training approach across various target language corpus sizes D_T and compute budgets C , for all language pairs. Smooth curves show M^3 Scaling Law predictions; dotted lines with hollow markers show corresponding observed values. The red dashed vertical lines represent the compute-optimal data size $D^*(C)$.

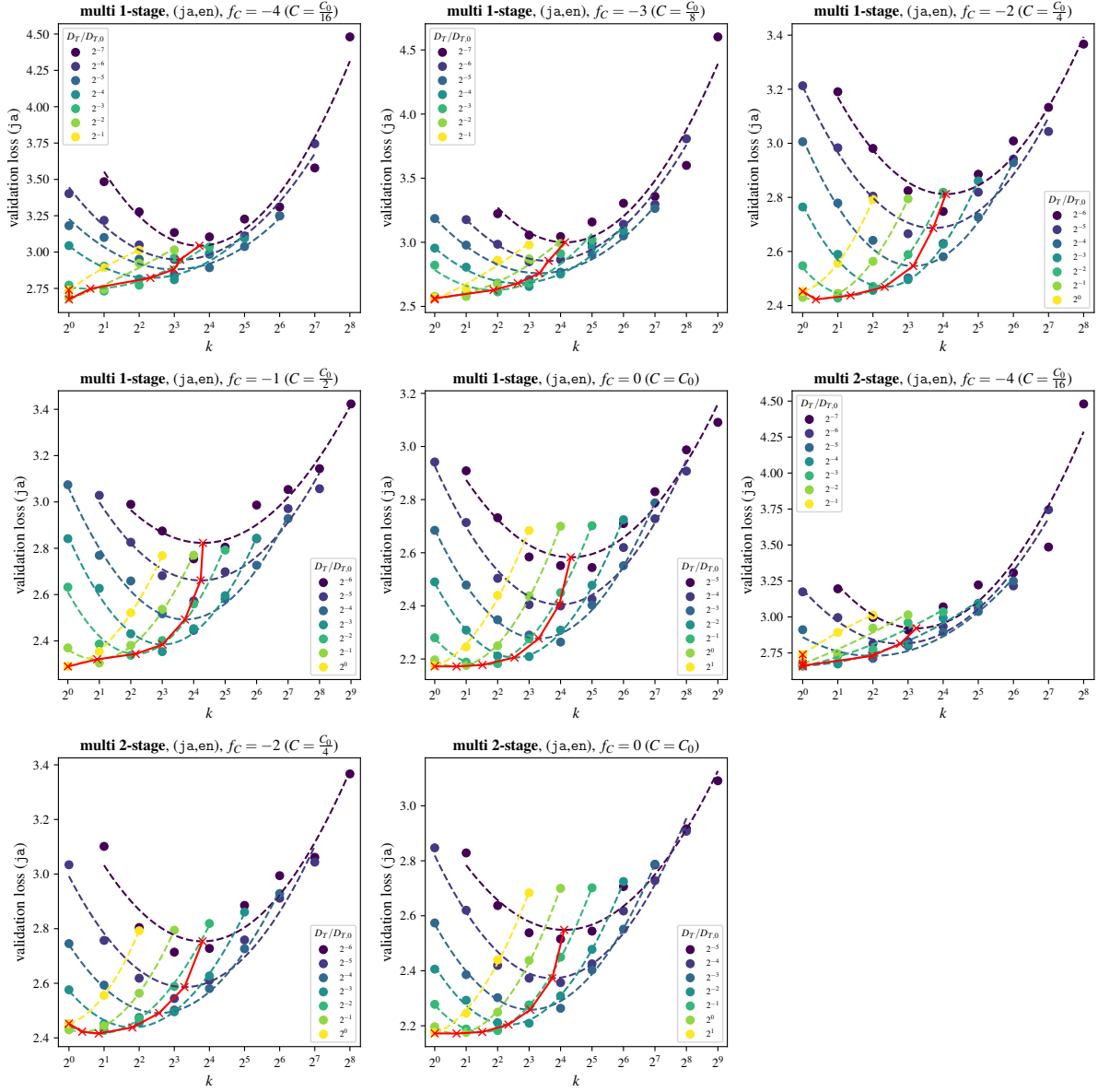


Figure 14: Change of minimum target language validation loss $L^*(C, D_T, k)$ by number of epochs k (x-axis) for each target language corpus size D_T (color) (Japanese, English) pair. Each dashed line represents a quadratic function of $f_k = \log_2 k$ fitted to estimate $L^*(C, D_T, k)$. The red solid line connects minimum points of the fitted curves.

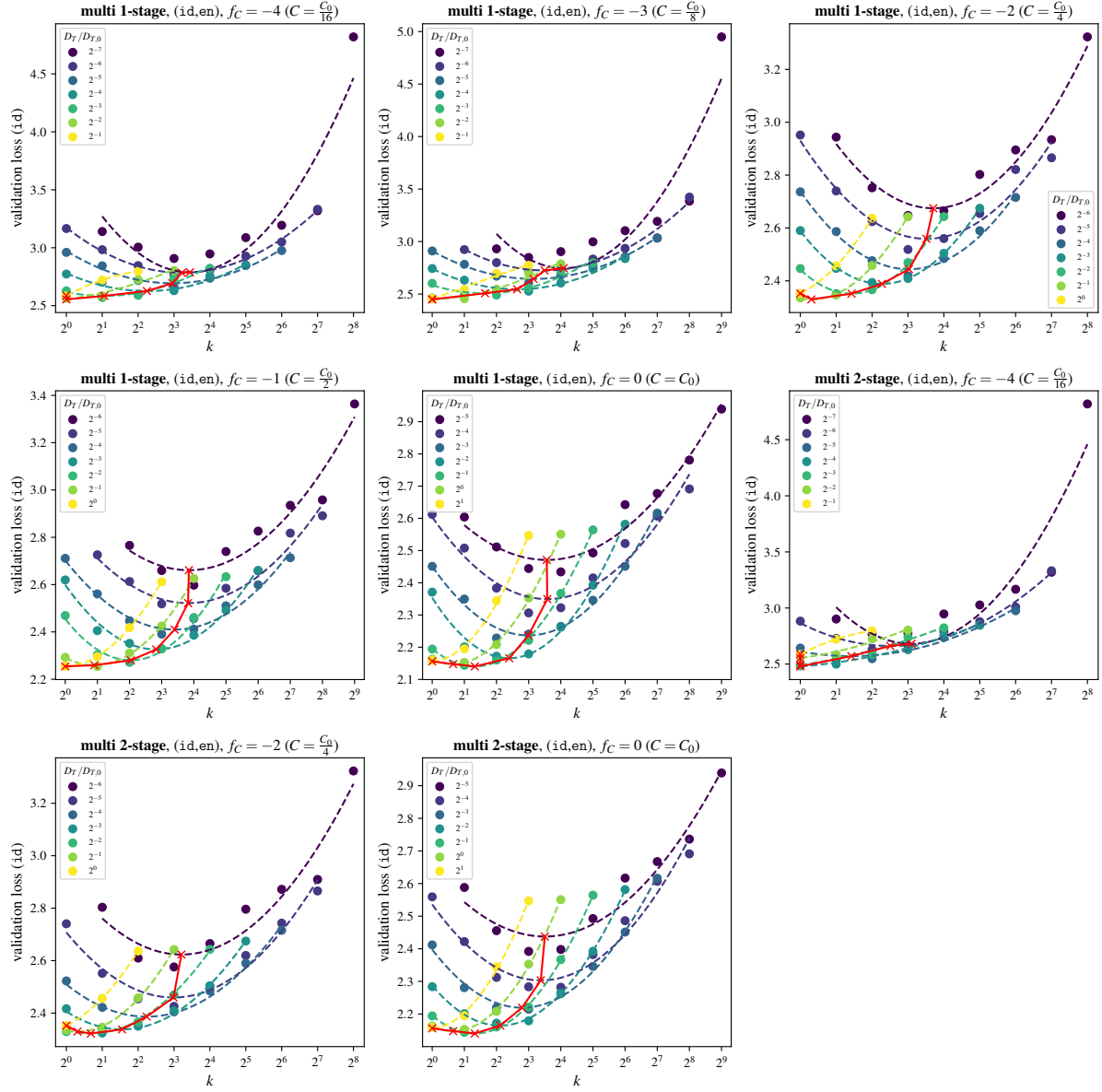


Figure 15: Change of minimum target language validation loss $L^*(C, D_T, k)$ by number of epochs k (x-axis) for each target language corpus size D_T (color) ((Indonesian, English) pair). Each dashed line represents a quadratic function of $f_k = \log_2 k$ fitted to estimate $L^*(C, D_T, k)$. The red solid line connects minimum points of the fitted curves.

f_M	n_{layers}	n_{heads}	d_{model}	M
5	2	4	128	1.49×10^7
4	4	4	128	2.99×10^7
3	4	7	224	5.85×10^7
2	4	12	384	1.18×10^8
1	8	12	384	2.36×10^8
0	8	39	624	4.70×10^8
-1	16	39	624	9.39×10^8

Table 4: The number of layers n_{layers} , the number of heads n_{heads} , and the dimension of hidden states d_{model} for each model scale factor f_M .

f_C	f_r	f_M	f_k	f_D
0	[0,4)	[-1,5)	[0,10)	[-5,2)
-1		[0,5)		[-6,1)
-2				
-3		[1,6)		[-7,0)
-4				

Table 5: Search space of factors (f_r, f_M, f_k) for each compute factor f_C . We grid-searched a set of factors, (f_r, f_M, f_k, f_C) , within those ranges and selected it for the experiment if its corresponding $f_D = -f_r + f_M - f_k + f_C$ is within the specified range. For the (Swahili, English) pair, we limit the search to $f_D \leq -3$ due to limited corpus availability.

Symbol	Description	Bounds	Initial Sampling Range
<i>Parameters of Chinchilla Model (Hoffmann et al., 2022)</i>			
$A^\dagger$	Coefficient for model size term	$[10^{-6}, 10^6]$	$[10^{-2}, 10^4]$
$B^\dagger$	Coefficient for data size term	$[10^{-6}, 10^6]$	$[10^{-2}, 10^4]$
α	Exponent for model size	$[0.1, 2.0]$	$[0.1, 0.8]$
β	Exponent for data size	$[0.01, 5.0]$	$[0.1, 0.8]$
E	Irreducible loss	$[0.001, 10.0]$	$[1.0, 5.0]$
<i>Parameters of Muennighoff Model (Multi-epoch) (Muennighoff et al., 2023)</i>			
R_D^*	Saturation constant for effective data	$[0.1, 200.0]$	$[1.0, 100.0]$
R_M^*	Saturation constant for effective model size	$[0.1, 100.0]$	$[0.5, 50.0]$
$R_{D,\text{high}}^*$	Saturation constant for high-resource language tokens in g (Eq. 9)	$[0.1, 200.0]$	$[1.0, 100.0]$
ψ	Exponent for lower bound of g (Eq. 9)	$[0.01, 10.0]$	$[0.1, 5.0]$
<i>Parameters of He Model (Multi-lingual) (He et al., 2025)</i>			
γ	Exponent for language ratio	$[0.001, 1.0]$	$[0.01, 0.5]$
<i>Parameters of Searched Two-stage Models</i>			
γ_2	Exponent for language ratio in stage 2	$[0.001, 1.0]$	$[0.01, 0.5]$
$r_0^\dagger$	Offset for language ratio (effective language ratio at $r = 0$)	$[10^{-6}, 1.0]$	$[10^{-4}, 0.1]$
$\dot{\gamma}_2$	Slope of r_1 -dependent γ_2 (Eq. 21)	$[-0.5, 0.5]$	$[-0.2, 0.2]$

Table 6: Parameter bounds and initial value sampling ranges used in scaling law model fitting. Bounds define the hard constraints during L-BFGS optimization, while sampling ranges define the narrower region from which initial values are sampled. † Sampled uniformly in log scale while other parameters are sampled uniformly in linear scale.

Axis	Split name	Test condition	Extrapolation direction
C (compute budget)	C_ge0	$f_C \geq 0$	Larger compute
	C_ge-1	$f_C \geq -1$	
M (model scale)	M_le-1	$f_M \leq -1$	Larger model
	M_le0	$f_M \leq 0$	
D_T (target corpus)	M_le1	$f_M \leq 1$	Larger target tokens
	DT_ge0	$f_D \geq 0$	
	DT_ge-1	$f_D \geq -1$	
D (total tokens)	DT_ge-2	$f_D \geq -2$	Larger total tokens
	D_ge34	$\log_2 D \geq 34$	
	D_ge33	$\log_2 D \geq 33$	
	D_ge32	$\log_2 D \geq 32$	
r (language ratio)	r_le0.125	$r \leq 0.125$	Lower target language ratio
	r_le0.25	$r \leq 0.25$	
	r_ge0.5	$r \geq 0.5$	Higher target language ratio
	r_ge1	$r \geq 1.0$	
k (epochs)	k_ge32	$k \geq 32$	Higher epoch
	k_ge64	$k \geq 64$	
	k_ge128	$k \geq 128$	

Table 7: Train-test split definitions for the combined (1-stage + 2-stage) evaluation. Data points satisfying the test condition form the test set; the remainder forms the training set.

Axis	Splits used
C (compute budget)	C_ge0, C_ge-1
M (model scale)	M_le-1, M_le0, M_le1
D_T (target corpus)	DT_ge0, DT_ge-1, DT_ge-2
D (total tokens)	D_ge34, D_ge33, D_ge32
r (language ratio)	r_le0.125, r_ge0.5

Table 8: Train-test split definitions for the multi-lingual single-epoch evaluation (13 splits). Two r -axis splits and all k -axis splits from Table 7 are excluded; see text for rationale.

Model	Lang	C		M			D_T			D			r				k		
		C_{ge0}	C_{ge-1}	M_{le-1}	M_{le0}	M_{le1}	DT_{ge0}	DT_{ge-1}	DT_{ge-2}	D_{ge34}	D_{ge33}	D_{ge32}	r_{le125}	r_{le25}	r_{ge5}	r_{ge1}	k_{ge32}	k_{ge64}	k_{ge128}
Chinchilla	ja	.39 _{.00}	.44 _{.00}	-.44 _{.00}	-.51 _{.00}	-.12 _{.00}	.53 _{.00}	.57 _{.00}	.53 _{.00}	-.38 _{.00}	.31 _{.00}	.21 _{.00}	.25 _{.00}	.31 _{.00}	.42 _{.00} *	.32 _{.00} *	.16 _{.00}	-.01 _{.00}	-.11 _{.00}
Chinchilla	id	.42 _{.00}	.39 _{.01}	-.41 _{.00}	-.47 _{.00}	-.09 _{.00}	.61 _{.00}	.63 _{.00}	.55 _{.00}	-.40 _{.00}	.29 _{.00}	.09 _{.00}	.40 _{.00}	.41 _{.00}	.34 _{.00} *	.09 _{.00} *	-.17 _{.00}	-.27 _{.00}	-.30 _{.00}
Chinchilla	sw	.55 _{.00}	.51 _{.00}	-.02 _{.00}	-.12 _{.00}	.09 _{.00} *	—	—	—	-.34 _{.00}	.44 _{.00}	.40 _{.01}	.47 _{.00}	.47 _{.01}	.37 _{.00} *	.08 _{.00} *	.18 _{.00}	.02 _{.00}	-.06 _{.00}
He	ja	.58 _{.01}	.46 _{.04}	-.78 _{.04}	-.39 _{.02}	-.07 _{.03}	.84 _{.01}	.75 _{.01}	.65 _{.16}	-.11 _{.04}	.46 _{.02}	.64 _{.02}	.55 _{.03}	.63 _{.02}	—	—	-.04 _{.01}	-.28 _{.02}	-.40 _{.04}
He	id	.50 _{.01}	.37 _{.03}	-.29 _{.11}	-.13 _{.06}	-.34 _{.06}	.81 _{.01}	.81 _{.02}	.80 _{.02}	-.04 _{.02}	.43 _{.02}	.56 _{.00}	.70 _{.00}	.70 _{.00}	—	—	-.31 _{.00}	-.45 _{.00}	-.49 _{.00}
He	sw	.39 _{.07}	.22 _{.10}	.63 _{.01}	.48 _{.03}	—	—	—	—	-.375 _{.123}	-.65 _{.28}	.10 _{.14}	.66 _{.01}	-.50 _{.35}	—	—	-.67 _{.14}	-.15 _{.19}	-.159 _{.25}
Muennighoff	ja	.69 _{.04}	.61 _{.01}	.31 _{.02}	-.16 _{.21}	.14 _{.08}	.86 _{.01}	.81 _{.01}	.63 _{.09}	.31 _{.07}	.56 _{.01}	.67 _{.00}	-.112 _{.08}	-.49 _{.05}	—	—	.12 _{.01}	-.01 _{.03}	.37 _{.01}
Muennighoff	id	.56 _{.01}	.49 _{.04}	-.99 _{.41}	.03 _{.10}	.39 _{.04}	.88 _{.00}	.83 _{.02}	.72 _{.06}	.43 _{.01}	.68 _{.00}	.69 _{.02}	-.111 _{.11}	-.50 _{.06}	—	—	-.09 _{.01}	-.07 _{.03}	.15 _{.01}
Muennighoff	sw	.61 _{.05}	.54 _{.00}	-.03 _{.28}	.37 _{.19}	—	—	—	—	-.308 _{.125}	-.13 _{.26}	.42 _{.11}	.07 _{.20}	.27 _{.18}	—	—	.26 _{.10}	.40 _{.01}	.44 _{.01}
ATLAS	ja	.57 _{.02}	.49 _{.03}	.50 _{.01}	.48 _{.01}	.51 _{.02}	.77 _{.00}	.78 _{.01}	.54 _{.11}	-.65 _{.12}	.12 _{.04}	.53 _{.02}	.44 _{.01}	.55 _{.01}	—	—	.57 _{.01}	.61 _{.01}	.74 _{.00}
ATLAS	id	.62 _{.00}	.58 _{.02}	.70 _{.01}	.57 _{.02}	.30 _{.02}	.83 _{.00}	.80 _{.01}	.70 _{.06}	-.62 _{.06}	.35 _{.04}	.65 _{.01}	.59 _{.00}	.63 _{.00}	—	—	.42 _{.00}	.45 _{.00}	.55 _{.00}
ATLAS	sw	.43 _{.03}	.43 _{.02}	.66 _{.04}	.73 _{.02}	—	—	—	—	-.660 _{.79}	-.93 _{.10}	.19 _{.04}	.48 _{.06}	.53 _{.04}	—	—	.43 _{.02}	.42 _{.01}	.50 _{.02}
Sedova	ja	.84 _{.00}	.78 _{.01}	.90 _{.00}	.36 _{.06}	.51 _{.00}	.82 _{.00}	.84 _{.00}	.82 _{.00}	.27 _{.01}	.68 _{.00}	.78 _{.01}	.75 _{.00}	.73 _{.01}	.60 _{.02} *	.48 _{.00} *	.61 _{.01}	.50 _{.00}	.43 _{.00}
Sedova	id	.78 _{.01}	.69 _{.00}	.74 _{.01}	.20 _{.06}	.37 _{.06}	.79 _{.00}	.80 _{.00}	.71 _{.01}	.10 _{.00}	.64 _{.00}	.73 _{.00}	.76 _{.00}	.72 _{.02}	.57 _{.00} *	.34 _{.00} *	.38 _{.00}	.29 _{.01}	.26 _{.01}
Sedova	sw	.85 _{.00}	.74 _{.01}	.67 _{.07}	.63 _{.04}	.51 _{.08} *	—	—	—	-.21 _{.10}	.62 _{.00}	.73 _{.03}	.78 _{.00}	.79 _{.03}	.46 _{.03} *	.28 _{.00} *	.50 _{.01}	.38 _{.01}	.37 _{.01}
Bryant	ja	.44 _{.01}	.49 _{.01}	-.57 _{.39}	-.32 _{.04}	-.09 _{.03}	.54 _{.01}	.58 _{.00}	.54 _{.01}	-.35 _{.05}	.33 _{.01}	.41 _{.13}	.21 _{.00}	.08 _{.04}	.36 _{.06} *	.43 _{.04} *	.18 _{.02}	.05 _{.03}	.03 _{.11}
Bryant	id	.47 _{.01}	.52 _{.04}	-.35 _{.04}	-.08 _{.20}	-.62 _{.14} *	.62 _{.01}	.64 _{.00}	.57 _{.00}	-.30 _{.29}	.40 _{.08}	.40 _{.11}	.35 _{.00}	.26 _{.01}	.38 _{.00} *	.29 _{.11} *	-.09 _{.04}	-.11 _{.07}	.00 _{.13}
Bryant	sw	.58 _{.02}	.62 _{.01}	-.11 _{.02}	-.01 _{.09}	-.47 _{.04} *	—	—	—	-.21 _{.17}	.50 _{.05}	.59 _{.09}	.43 _{.00}	.38 _{.01}	.34 _{.11} *	.08 _{.00} *	.23 _{.04}	.10 _{.05}	.05 _{.05}
M^3	ja	.81 _{.02}	.81 _{.03}	.17 _{.07}	.41 _{.03}	.42 _{.03}	.97 _{.00}	.88 _{.00}	.62 _{.23}	.81 _{.02}	.87 _{.01}	.89 _{.02}	.75 _{.02}	.73 _{.01}	—	—	.56 _{.03}	.65 _{.01}	.73 _{.01}
M^3	id	.79 _{.01}	.76 _{.02}	.62 _{.04}	.60 _{.03}	.28 _{.14}	.96 _{.00}	.93 _{.02}	.73 _{.31}	.44 _{.02}	.83 _{.01}	.87 _{.00}	.85 _{.00}	.71 _{.01}	—	—	.48 _{.01}	.49 _{.01}	.54 _{.01}
M^3	sw	.86 _{.01}	.81 _{.01}	.74 _{.04}	.79 _{.02}	—	—	—	—	-.14 _{.66}	.70 _{.14}	.82 _{.08}	.86 _{.08}	.66 _{.12}	—	—	.59 _{.04}	.50 _{.07}	.44 _{.08}
M^3 - Dual PL	ja	.77 _{.02}	.71 _{.04}	.17 _{.08}	.30 _{.04}	.33 _{.03}	.90 _{.00}	.83 _{.00}	.56 _{.22}	.47 _{.01}	.71 _{.02}	.81 _{.02}	.62 _{.01}	.56 _{.01}	—	—	.54 _{.02}	.67 _{.03}	.73 _{.02}
M^3 - Dual PL	id	.77 _{.01}	.71 _{.02}	.67 _{.03}	.59 _{.04}	.23 _{.15}	.89 _{.00}	.87 _{.02}	.68 _{.28}	.10 _{.01}	.69 _{.02}	.81 _{.01}	.70 _{.00}	.54 _{.01}	—	—	.51 _{.01}	.51 _{.01}	.55 _{.01}
M^3 - Dual PL	sw	.85 _{.01}	.78 _{.01}	.74 _{.02}	.79 _{.02}	—	—	—	—	-.48 _{.65}	.57 _{.14}	.76 _{.08}	.78 _{.07}	.65 _{.08}	—	—	.58 _{.06}	.52 _{.06}	.47 _{.06}
M^3 - g	ja	.79 _{.02}	.79 _{.03}	.32 _{.06}	.43 _{.03}	.44 _{.04}	.97 _{.00}	.88 _{.00}	.72 _{.15}	.70 _{.01}	.79 _{.02}	.84 _{.03}	.74 _{.02}	.76 _{.01}	—	—	.55 _{.02}	.60 _{.01}	.69 _{.01}
M^3 - g	id	.75 _{.01}	.73 _{.01}	.68 _{.02}	.67 _{.03}	.35 _{.12}	.96 _{.00}	.92 _{.02}	.75 _{.32}	.68 _{.01}	.77 _{.04}	.82 _{.01}	.85 _{.00}	.79 _{.01}	—	—	.40 _{.00}	.40 _{.01}	.48 _{.01}
M^3 - g	sw	.65 _{.06}	.67 _{.07}	.71 _{.07}	.81 _{.01}	—	—	—	—	-.319 _{.158}	-.10 _{.32}	.49 _{.13}	.84 _{.07}	.61 _{.09}	—	—	.29 _{.08}	.16 _{.07}	.17 _{.10}

Table 9: Per-language, per-split test R^2 for (a) All data (1-stage + 2-stage), corresponding to Table 2(a). Each cell reports the mean over the repeated fits, with the standard deviation as a subscript. Splits marked with * are excluded from the aggregation in Table 2 because at least one model failed to fit on that split for at least one seed.

Model	Lang	C		M				D_T			D			r				k		
		C_{ge0}	C_{ge-1}	M_{le-1}	M_{le0}	M_{le1}	DT_{ge0}	DT_{ge-1}	DT_{ge-2}	D_{ge34}	D_{ge33}	D_{ge32}	r_{le125}	r_{le25}	r_{ge5}	r_{ge1}	k_{ge32}	k_{ge64}	k_{ge128}	
Chinchilla	ja	.29 _{.00}	.34 _{.01}	-.09 _{.00}	-.17 _{.00}	.21 _{.00}	.23 _{.00}	.28 _{.00}	.24 _{.00}	-.00 _{.00}	.37 _{.00}	.54 _{.00}	-.54 _{.00}	-.25 _{.00}	.14 _{.00} *	.26 _{.00} *	.05 _{.00}	-.10 _{.00}	-.22 _{.00}	
Chinchilla	id	.32 _{.01}	.25 _{.00}	-.08 _{.00}	-.19 _{.01}	.07 _{.00}	.31 _{.00}	.33 _{.00}	.30 _{.00}	-.00 _{.00}	.32 _{.00}	.38 _{.00}	-.06 _{.00}	-.02 _{.01}	.18 _{.00} *	.16 _{.00} *	-.26 _{.00}	-.38 _{.00}	-.50 _{.00}	
Chinchilla	sw	.38 _{.01}	.33 _{.00}	-.00 _{.00}	-.11 _{.01}	.00 _{.01} *	—	—	—	-.02 _{.00}	.40 _{.01}	.53 _{.00}	.19 _{.00}	.20 _{.00}	.18 _{.00} *	.14 _{.00} *	-.04 _{.00}	-.19 _{.00}	-.26 _{.01}	
He	ja	.58 _{.01}	.39 _{.02}	.23 _{.02}	-.12 _{.01}	.08 _{.02}	.96 _{.00}	.88 _{.01}	.74 _{.14}	.25 _{.01}	.45 _{.03}	.46 _{.02}	.49 _{.04}	.41 _{.03}	—	—	-.62 _{.01}	-.94 _{.02}	-.114 _{.04}	
He	id	.39 _{.00}	.14 _{.01}	.05 _{.02}	-.10 _{.02}	-.25 _{.03}	.96 _{.00}	.93 _{.02}	.89 _{.03}	.36 _{.01}	.42 _{.04}	.36 _{.00}	.83 _{.01}	.77 _{.02}	—	—	-.66 _{.01}	-.84 _{.01}	-.101 _{.01}	
He	sw	-.14 _{.10}	-.29 _{.11}	.01 _{.05}	-.14 _{.02}	—	—	—	—	-.399 _{.101}	-.141 _{.34}	-.41 _{.19}	.69 _{.08}	-.56 _{.35}	—	—	-.128 _{.11}	-.174 _{.15}	-.241 _{.23}	
Muennighoff	ja	.70 _{.02}	.59 _{.03}	.53 _{.01}	.37 _{.06}	.48 _{.04}	.76 _{.01}	.82 _{.00}	.66 _{.07}	-.02 _{.07}	.47 _{.01}	.70 _{.00}	-.95 _{.11}	-.34 _{.06}	—	—	.36 _{.01}	.36 _{.02}	.45 _{.01}	
Muennighoff	id	.57 _{.01}	.42 _{.03}	-.14 _{.16}	.26 _{.02}	.35 _{.06}	.81 _{.00}	.81 _{.02}	.65 _{.08}	-.06 _{.02}	.53 _{.01}	.65 _{.00}	-.77 _{.15}	-.63 _{.03}	—	—	.01 _{.04}	.07 _{.00}	.11 _{.00}	
Muennighoff	sw	.60 _{.03}	.47 _{.04}	.36 _{.10}	.43 _{.06}	—	—	—	—	-.181 _{.63}	-.09 _{.24}	.45 _{.12}	-.06 _{.29}	.21 _{.24}	—	—	.33 _{.04}	.30 _{.04}	.32 _{.02}	
ATLAS	ja	.66 _{.02}	.56 _{.02}	.77 _{.01}	.53 _{.01}	.53 _{.02} *	.71 _{.01}	.78 _{.00}	.60 _{.07}	-.48 _{.06}	.21 _{.01}	.62 _{.00}	.02 _{.03}	.39 _{.01}	—	—	.57 _{.01}	.57 _{.01}	.67 _{.01}	
ATLAS	id	.66 _{.00}	.53 _{.01}	.56 _{.02}	.42 _{.01}	.18 _{.01}	.74 _{.00}	.75 _{.02}	.61 _{.05}	-.81 _{.04}	.27 _{.01}	.64 _{.00}	.24 _{.01}	.47 _{.00}	—	—	.35 _{.01}	.36 _{.01}	.46 _{.01}	
ATLAS	sw	.56 _{.00}	.49 _{.02}	.64 _{.02}	.56 _{.02}	—	—	—	—	-.320 _{.34}	-.50 _{.10}	.36 _{.10}	.11 _{.08}	.36 _{.08}	—	—	.40 _{.01}	.35 _{.01}	.43 _{.02}	
Sedova	ja	.81 _{.04}	.76 _{.06}	.80 _{.06}	.36 _{.16}	.67 _{.24}	.92 _{.01}	.91 _{.01}	.88 _{.01}	.78 _{.14}	.74 _{.09}	.78 _{.00}	.66 _{.01}	.67 _{.01}	-.74 _{.47} *	.30 _{.01} *	.26 _{.06}	.27 _{.11}	.34 _{.19}	
Sedova	id	.75 _{.03}	.67 _{.07}	.94 _{.01}	.62 _{.25}	.52 _{.17}	.86 _{.02}	.81 _{.01}	.76 _{.05}	.55 _{.30}	.70 _{.15}	.68 _{.06}	.64 _{.05}	.59 _{.10}	-.48 _{.26} *	.14 _{.01} *	.11 _{.09}	.11 _{.17}	.31 _{.23}	
Sedova	sw	.75 _{.06}	.52 _{.20}	.64 _{.17}	.62 _{.22}	.47 _{.32} *	—	—	—	.43 _{.65}	.71 _{.20}	.76 _{.12}	.79 _{.02}	.69 _{.24}	-.72 _{.21} *	-.06 _{.03} *	.21 _{.24}	.34 _{.26}	.33 _{.29}	
Bryant	ja	.33 _{.04}	.38 _{.05}	-.05 _{.08}	.06 _{.07}	-.22 _{.06}	.26 _{.02}	.31 _{.02}	.28 _{.01}	-.06 _{.07}	.40 _{.02}	.58 _{.02}	-.82 _{.02}	-.56 _{.06}	-.29 _{.18} *	.25 _{.03} *	.08 _{.04}	.06 _{.04}	-.07 _{.05}	
Bryant	id	.47 _{.01}	.52 _{.03}	.09 _{.03}	.41 _{.02}	-.137 _{.23}	.37 _{.02}	.39 _{.01}	.34 _{.02}	-.04 _{.11}	.43 _{.02}	.57 _{.04}	-.39 _{.01}	-.37 _{.02}	-.28 _{.03} *	.17 _{.03} *	-.13 _{.02}	-.12 _{.05}	-.14 _{.04}	
Bryant	sw	.52 _{.02}	.57 _{.02}	-.28 _{.04}	.37 _{.04}	-.183 _{.09} *	—	—	—	-.18 _{.06}	.46 _{.03}	.61 _{.01}	-.04 _{.03}	-.18 _{.04}	-.03 _{.20} *	.14 _{.00} *	.09 _{.01}	.03 _{.06}	.03 _{.01}	
M^3	ja	.88 _{.02}	.84 _{.01}	.77 _{.02}	.66 _{.01}	.66 _{.02}	.98 _{.00}	.91 _{.00}	.79 _{.11}	.80 _{.02}	.94 _{.00}	.88 _{.00}	.64 _{.03}	.66 _{.02}	—	—	.58 _{.01}	.62 _{.02}	.72 _{.01}	
M^3	id	.80 _{.00}	.70 _{.01}	.74 _{.02}	.60 _{.01}	.46 _{.05}	.98 _{.00}	.94 _{.02}	.69 _{.42}	.74 _{.00}	.90 _{.01}	.82 _{.01}	.87 _{.00}	.81 _{.02}	—	—	.45 _{.00}	.42 _{.01}	.49 _{.01}	
M^3	sw	.81 _{.02}	.71 _{.01}	.72 _{.07}	.67 _{.05}	—	—	—	—	.17 _{.54}	.60 _{.20}	.78 _{.10}	.80 _{.16}	.69 _{.18}	—	—	.41 _{.08}	.41 _{.06}	.43 _{.06}	

Model	Lang	C		M			D_T			D			k		
		C_{ge0}	C_{ge-1}	M_{le-1}	M_{le0}	M_{le1}	DT_{ge0}	DT_{ge-1}	DT_{ge-2}	D_{ge34}	D_{ge33}	D_{ge32}	k_{ge32}	k_{ge64}	k_{ge128}
Muennighoff	ja	.83 _{.01}	.82 _{.01}	—	.74 _{.01}	.73 _{.01}	.96 _{.01}	.90 _{.00}	.79 _{.09}	—	.94 _{.01}	.91 _{.00}	.57 _{.01}	.68 _{.02}	.76 _{.01}
Muennighoff	id	.83 _{.01}	.72 _{.06}	—	.79 _{.01}	.73 _{.04}	.97 _{.00}	.92 _{.02}	.84 _{.07}	—	.93 _{.01}	.83 _{.01}	.38 _{.01}	.38 _{.03}	.52 _{.03}
Muennighoff	sw	.78 _{.02}	.68 _{.06}	—	.76 _{.05}	—	—	—	—	—	.93 _{.01}	.87 _{.00}	.38 _{.01}	.41 _{.06}	.57 _{.03}
ATLAS	ja	.75 _{.00}	.71 _{.01}	—	.37 _{.01}	.46 _{.01}	.96 _{.01}	.90 _{.00}	.81 _{.09}	—	.86 _{.01}	.89 _{.00}	.59 _{.01}	.60 _{.01}	.69 _{.00}
ATLAS	id	.69 _{.00}	.59 _{.02}	—	.41 _{.02}	.15 _{.00}	.96 _{.01}	.92 _{.02}	.88 _{.03}	—	.78 _{.01}	.78 _{.01}	.38 _{.00}	.37 _{.01}	.44 _{.02}
ATLAS	sw	.71 _{.03}	.57 _{.02}	—	.52 _{.03}	—	—	—	—	—	.93 _{.01}	.88 _{.01}	.37 _{.01}	.38 _{.01}	.50 _{.02}
Sedova	ja	.84 _{.01}	.82 _{.02}	—	.56 _{.11}	.56 _{.23}	.95 _{.01}	.93 _{.04}	.80 _{.12}	—	.94 _{.02}	.89 _{.01}	.45 _{.02}	.37 _{.01}	.47 _{.09}
Sedova	id	.82 _{.03}	.77 _{.03}	—	.83 _{.05}	-.42 _{.96}	.91 _{.01}	.85 _{.10}	.82 _{.06}	—	.94 _{.01}	.80 _{.08}	.03 _{.25}	.19 _{.05}	.40 _{.17}
Sedova	sw	.82 _{.03}	.75 _{.02}	—	.78 _{.08}	.12 _{.30} *	—	—	—	—	.95 _{.01}	.89 _{.03}	.28 _{.03}	.45 _{.14}	.58 _{.09}
Bryant	ja	.80 _{.01}	.80 _{.00}	—	.67 _{.07}	.59 _{.03}	.96 _{.01}	.95 _{.01}	.90 _{.01}	—	.88 _{.02}	.85 _{.01}	-.19 _{.02}	.25 _{.01}	.33 _{.04}
Bryant	id	.83 _{.01}	.78 _{.00}	—	.84 _{.01}	-.15 _{.15}	.93 _{.01}	.90 _{.02}	.69 _{.02}	—	.91 _{.01}	.77 _{.00}	-.20 _{.04}	.31 _{.01}	.30 _{.09}
Bryant	sw	.83 _{.00}	.78 _{.01}	—	.80 _{.02}	.22 _{.10} *	—	—	—	—	.94 _{.01}	.89 _{.01}	-.06 _{.03}	.48 _{.03}	.58 _{.02}
$M^3+R_M^*(k)$	ja	.88 _{.03}	.89 _{.02}	—	.63 _{.01}	.78 _{.01}	.96 _{.01}	.91 _{.00}	.75 _{.10}	—	.95 _{.00}	.94 _{.01}	.63 _{.05}	.64 _{.09}	-.15 _{.33}
$M^3+R_M^*(k)$	id	.91 _{.01}	.86 _{.01}	—	.80 _{.01}	.83 _{.01}	.97 _{.00}	.93 _{.01}	.85 _{.05}	—	.92 _{.00}	.94 _{.00}	.66 _{.12}	.58 _{.09}	.85 _{.00}
$M^3+R_M^*(k)$	sw	.94 _{.01}	.89 _{.00}	—	.81 _{.01}	—	—	—	—	—	.91 _{.02}	.96 _{.01}	.67 _{.25}	.09 _{.79}	.85 _{.06}
M^3	ja	.83 _{.01}	.82 _{.01}	—	.74 _{.01}	.73 _{.01}	.96 _{.01}	.90 _{.00}	.79 _{.09}	—	.94 _{.01}	.91 _{.00}	.57 _{.01}	.68 _{.02}	.76 _{.01}
M^3	id	.83 _{.01}	.72 _{.06}	—	.79 _{.01}	.73 _{.04}	.97 _{.00}	.92 _{.02}	.84 _{.07}	—	.93 _{.01}	.83 _{.01}	.38 _{.01}	.38 _{.03}	.52 _{.03}
M^3	sw	.78 _{.02}	.68 _{.06}	—	.76 _{.05}	—	—	—	—	—	.93 _{.01}	.87 _{.00}	.38 _{.01}	.41 _{.06}	.57 _{.03}

Table 11: Per-language, per-split test R^2 for (c) 1-stage monolingual ($r=1$) data only, corresponding to Table 2(c). The r axis is not applicable. Each cell reports the mean over the repeated fits, with the standard deviation as a subscript. Splits marked with * are excluded from the aggregation in Table 2 because at least one model failed to fit on that split for at least one seed.

Model	Lang	C		M			D _T			D			r		
		C _{ge0}	C _{ge-1}	M _{le-1}	M _{le0}	M _{le1}	DT _{ge0}	DT _{ge-1}	DT _{ge-2}	D _{ge34}	D _{ge33}	D _{ge32}	r _{le125}	r _{ge5}	
Full models (trained on 1-stage + 2-stage, evaluated on 1-stage test)															
He	ja	.53 _{.03}	.41 _{.01}	—	.10 _{.09}	—	.84 _{.06}	—	—	—	—	.21 _{.11}	-.41 _{.02}	—	
He	id	.57 _{.04}	.45 _{.02}	—	.11 _{.03}	—	.81 _{.02}	—	—	—	—	.09 _{.05}	-.46 _{.03}	—	
He	sw	—	—	—	—	—	—	—	—	—	—	—	—	—	
He + Dual PL	ja	.95 _{.02}	.92 _{.02}	—	.71 _{.07}	—	.98 _{.01}	—	—	—	—	.87 _{.02}	.78 _{.03}	—	
He + Dual PL	id	.96 _{.01}	.95 _{.01}	—	.77 _{.02}	—	.97 _{.01}	—	—	—	—	.84 _{.02}	.84 _{.00}	—	
He + Dual PL	sw	—	—	—	—	—	—	—	—	—	—	—	—	—	
Sedova	ja	.77 _{.00}	.60 _{.01}	—	.33 _{.04}	.06 _{.07} *	.77 _{.00}	.80 _{.01} *	.83 _{.04} *	—	—	.33 _{.06}	-.55 _{.11}	-.67 _{.26} *	
Sedova	id	.72 _{.00}	.51 _{.01}	—	.44 _{.01}	-.12 _{.12} *	.73 _{.01}	.82 _{.00} *	.82 _{.02} *	—	—	.32 _{.03}	-.38 _{.04}	-.60 _{.26} *	
Sedova	sw	—	—	—	—	.76 _{.03} *	—	—	—	—	—	—	.04 _{.17} *	—	
M ³	ja	.98 _{.00}	.94 _{.00}	—	.76 _{.01}	—	.97 _{.01}	—	—	—	—	.78 _{.04}	.82 _{.02}	—	
M ³	id	.96 _{.00}	.95 _{.01}	—	.85 _{.02}	—	.96 _{.01}	—	—	—	—	.76 _{.02}	.82 _{.01}	—	
M ³	sw	—	—	—	—	—	—	—	—	—	—	—	—	—	
M ³ – Dual PL	ja	.63 _{.05}	.49 _{.03}	—	.23 _{.09}	—	.76 _{.08}	—	—	—	—	-.07 _{.11}	-.36 _{.01}	—	
M ³ – Dual PL	id	.60 _{.04}	.48 _{.02}	—	.32 _{.09}	—	.77 _{.02}	—	—	—	—	-.06 _{.04}	-.31 _{.01}	—	
M ³ – Dual PL	sw	—	—	—	—	—	—	—	—	—	—	—	—	—	
Full models (evaluated on 2-stage test) and 2-stage-only models [†]															
M ³	ja	.85 _{.01}	.83 _{.03}	.60 _{.02}	.41 _{.04}	—	.97 _{.01}	—	—	-.12 _{.88}	.52 _{.22}	.81 _{.06}	.76 _{.01}	—	
M ³	id	.86 _{.01}	.86 _{.01}	.43 _{.02}	.50 _{.04}	—	.93 _{.01}	—	—	-1.40 _{.40}	.60 _{.06}	.86 _{.01}	.85 _{.00}	—	
M ³	sw	—	—	—	—	—	—	—	—	—	—	—	—	—	
D-CPT [†]	ja	.72 _{.01}	.72 _{.01}	.16 _{.02}	-.30 _{.00}	-.04 _{.00}	.79 _{.02}	.85 _{.02}	.87 _{.03}	.56 _{.03}	.72 _{.01}	.77 _{.01}	.79 _{.00}	.90 _{.00}	
D-CPT [†]	id	.72 _{.01}	.72 _{.01}	.06 _{.04}	-.29 _{.01}	-.58 _{.03}	.80 _{.01}	.89 _{.02}	.88 _{.02}	.57 _{.03}	.83 _{.01}	.81 _{.01}	.82 _{.00}	.91 _{.00}	
D-CPT [†]	sw	.14 _{.06}	.14 _{.06}	.53 _{.01}	.73 _{.00}	.71 _{.01}	—	—	—	—	—	—	.84 _{.01}	.58 _{.01}	
PTPP-F1 [†]	ja	.75 _{.01}	.75 _{.01}	.45 _{.00}	.27 _{.01}	.41 _{.08}	.90 _{.01}	.91 _{.01}	.89 _{.02}	.72 _{.01}	.88 _{.02}	.89 _{.00}	.71 _{.05}	.64 _{.01}	
PTPP-F1 [†]	id	.76 _{.01}	.76 _{.01}	.48 _{.01}	.33 _{.03}	.14 _{.01}	.91 _{.02}	.93 _{.01}	.88 _{.03}	.95 _{.01}	.96 _{.01}	.91 _{.01}	.72 _{.01}	.71 _{.01}	
PTPP-F1 [†]	sw	.25 _{.02}	.25 _{.02}	.37 _{.04}	.76 _{.01}	.81 _{.01}	—	—	—	—	—	—	.80 _{.03}	.84 _{.01}	
PTPP-F2 [†]	ja	.72 _{.01}	.72 _{.01}	.12 _{.05}	-.29 _{.02}	-.04 _{.00}	.81 _{.05}	.89 _{.02}	.88 _{.03}	.59 _{.11}	.71 _{.05}	.79 _{.05}	.79 _{.02}	.90 _{.01}	
PTPP-F2 [†]	id	.71 _{.00}	.71 _{.00}	.03 _{.02}	-.31 _{.03}	-.59 _{.00}	.83 _{.02}	.90 _{.02}	.87 _{.04}	.50 _{.21}	.81 _{.02}	.84 _{.03}	.82 _{.01}	.91 _{.00}	
PTPP-F2 [†]	sw	.13 _{.07}	.13 _{.07}	.52 _{.02}	.73 _{.01}	.68 _{.03}	—	—	—	—	—	—	.83 _{.01}	.57 _{.02}	
PTPP-F3 [†]	ja	.75 _{.01}	.75 _{.01}	.45 _{.01}	.27 _{.02}	.45 _{.07}	.91 _{.03}	.93 _{.02}	.91 _{.02}	.76 _{.06}	.90 _{.03}	.89 _{.01}	.73 _{.03}	.63 _{.00}	
PTPP-F3 [†]	id	.77 _{.01}	.77 _{.01}	.48 _{.01}	.34 _{.02}	.22 _{.10}	.92 _{.02}	.94 _{.01}	.89 _{.01}	.92 _{.05}	.96 _{.00}	.92 _{.01}	.74 _{.05}	.69 _{.00}	
PTPP-F3 [†]	sw	.24 _{.07}	.24 _{.07}	.38 _{.04}	.75 _{.01}	.78 _{.05}	—	—	—	—	—	—	.77 _{.04}	.83 _{.01}	
Zhang [†]	ja	-.27 _{.151}	-.27 _{.151}	-2.18 _{.24}	-.97 _{.48}	-.48 _{.19}	.66 _{.01}	.53 _{.45}	.39 _{.40}	-4.00 _{.33}	-.24 _{.02}	.67 _{.01}	.59 _{.04}	.68 _{.02}	
Zhang [†]	id	-.24 _{.152}	-.24 _{.152}	-3.85 _{.35}	-1.43 _{.21}	-.79 _{.18}	.73 _{.03}	.58 _{.41}	.41 _{.30}	-4.82 _{.52}	-.05 _{.03}	.72 _{.02}	.35 _{.69}	.32 _{.83}	
Zhang [†]	sw	-1.33 _{.412}	-1.33 _{.412}	-.59 _{.256}	-.24 _{.68}	-.21 _{.33}	—	—	—	—	—	—	.42 _{.35}	.29 _{.110}	

Table 12: Per-language, per-split test R^2 for (d) Multi-lingual single-epoch ($k=1$) evaluation, corresponding to Table 2(d1, d2). The k axis is not applicable. [†]2-stage-only models trained and tested on 2-stage data only. Each cell reports the mean over the repeated fits, with the standard deviation as a subscript. Splits marked with * are excluded from the aggregation in Table 2 because at least one model failed to fit on that split for at least one seed.

Parameter	Japanese	Indonesian	Swahili
A	11577.3	6197.1	461.8
B	17875.0	16593.2	6775.8
α	0.550	0.526	0.368
β	0.510	0.518	0.462
E	1.704	1.759	2.312
R_D^*	10.43	7.83	5.85
$R_{D,\text{high}}^*$	63.89	35.25	26.67
ψ	2.717	4.269	10.000
R_M^*	22.12	26.78	100.00
γ	0.0762	0.0569	0.0755
γ_2	0.0277	0.0182	0.0529

Table 13: M^3 Scaling Law parameters fitted on the full dataset for each language pair.

Data	D' form	#P	C	M	D_T	D	r	k	Avg
<i>All data (1-stage + 2-stage)</i>									
h_{Lp}	12	.82 _{.01}	.55 _{.02}	.85 _{.07}	.67 _{.09}	.79 _{.03}	.60 _{.02}	.71 _{.02}	
h (adopted)	11	.81 _{.01}	.53 _{.02}	.85 _{.07}	.68 _{.10}	.76 _{.03}	.55 _{.02}	.70 _{.02}	
<i>1-stage monolingual ($r=1$), with $R_M^*(k)$</i>									
h_{Lp}	10	.89 _{.01}	.78 _{.01}	.89 _{.02}	.94 _{.00}	—	.53 _{.08}	.81 _{.01}	
h (adopted)	9	.89 _{.01}	.78 _{.01}	.89 _{.02}	.93 _{.00}	—	.53 _{.08}	.81 _{.02}	
<i>1-stage monolingual ($r=1$), constant R_M^*</i>									
h_{Lp}	8	.80 _{.01}	.78 _{.02}	.88 _{.03}	.90 _{.01}	—	.58 _{.01}	.79 _{.01}	
h (adopted)	7	.77 _{.02}	.75 _{.02}	.90 _{.02}	.90 _{.00}	—	.52 _{.01}	.77 _{.01}	

Table 14: Effect of replacing the Muennighoff form h with h_{Lp} (Eq. 20). On the full data, h_{Lp} improves Avg R^2 by $+.016 \pm .004$ (mean $\pm$ s.d. over the 5 repeated fits), but we adopt h for smoother optimal- k^* predictions; see text. Evaluation follows the same protocol as Table 2.

Model	#P	ja	id
M^3 ($\dot{\gamma}_2 = 0$)	11	0.702 _{.012}	0.691 _{.023}
$M^3 + r_1\text{-dep. } \gamma_2$	13	0.703 _{.012}	0.681 _{.040}

Table 15: Average test R^2 across multi-split evaluation (setting (a) of Table 2) with and without r_1 -dependent γ_2 , evaluated on Japanese (ja) and Indonesian (id). The extended model adds $\dot{\gamma}_2$ and r_0 to the M^3 Scaling Law. Each cell reports the mean over the repeated fits, with the standard deviation as a subscript.

Variant	#P	C	M	D_T	D	r	k	Avg
$R_M^*(k, r) + g \equiv 1$	12	.84 _{.02}	.38 _{.03}	.88 _{.03}	.62 _{.11}	.79 _{.01}	.66 _{.10}	.70 _{.04}
$R_M^*(k, r) + g(r, R_D)$	14	.84 _{.01}	.38 _{.02}	.88 _{.04}	.80 _{.04}	.78 _{.01}	.70 _{.04}	.73 _{.01}
$R_M^*(k) + g(r, R_D)$	13	.84 _{.01}	.35 _{.03}	.88 _{.04}	.73 _{.07}	.79 _{.01}	.63 _{.04}	.70 _{.01}
$R_M^* = \text{const} + g(r, R_D)$ (adopted)	11	.81 _{.01}	.53 _{.02}	.85 _{.07}	.68 _{.10}	.76 _{.03}	.55 _{.02}	.70 _{.02}
$R_M^* = \text{const} + g \equiv 1$	9	.73 _{.02}	.58 _{.01}	.87 _{.06}	.20 _{.23}	.77 _{.03}	.42 _{.03}	.59 _{.05}

Table 16: Trade-off between modeling routes for (k, r) dependence. Each row shows the test R^2 (macro-averaged over held-out splits and languages) for a different attribution of the (k, r) interaction to R_M^* (in M') vs. g (in D'). Evaluation follows the same protocol as Table 2(a), including the mean over repeated fits with the standard deviation as a subscript.

Group	#setups	#runs	CV%				s
			median	5%	95%	max	
All	60	300	0.35	0.14	1.29	3.33	0.0102
Japanese, single-stage	15	75	0.42	0.18	1.95	3.33	0.0124
Japanese, two-stage	15	75	0.33	0.15	0.73	0.92	0.0112
Indonesian, single-stage	15	75	0.31	0.15	1.07	1.28	0.0100
Indonesian, two-stage	15	75	0.32	0.11	1.26	1.55	0.0083

Table 17: Run-to-run variation of the validation loss. For each setup we train the same configuration with several random seeds and compute the standard deviation s across runs and the coefficient of variation $CV = s/\bar{L}$. Each row summarises those per-setup values over the setups in that group: the median, the 5th and 95th percentiles, and the maximum of CV, and the median of s in nats. #setups is the number of distinct configurations and #runs the number of training runs behind them.

ID	Stage	Setup							Japanese			Indonesian			
		f_C	f_M	f_k	f_r	r_1	r_f	n	L	s	CV%	n	L	s	CV%
S1	1	-4	5	7	1	—	—	5	3.650	0.1216	3.33	5	3.358	0.0223	0.66
S2	1	-4	1	0	3	—	—	5	3.406	0.0185	0.54	5	3.161	0.0308	0.97
S3	1	-4	1	4	0	—	—	5	3.291	0.0287	0.87	5	3.300	0.0143	0.43
S4	1	-4	5	1	3	—	—	5	3.472	0.0065	0.19	5	3.197	0.0100	0.31
S5	1	-4	4	1	0	—	—	5	2.885	0.0059	0.20	5	2.707	0.0058	0.21
S6	1	-2	2	1	3	—	—	5	2.943	0.0053	0.18	5	2.701	0.0032	0.12
S7	1	-2	4	8	0	—	—	5	3.365	0.0266	0.79	5	3.295	0.0203	0.62
S8	1	-2	0	3	0	—	—	5	2.763	0.0215	0.78	5	2.645	0.0051	0.19
S9	1	-2	4	1	1	—	—	5	2.912	0.0112	0.39	5	2.733	0.0077	0.28
S10	1	-2	4	6	2	—	—	5	3.176	0.0101	0.32	5	2.950	0.0068	0.23
S11	1	0	3	5	3	—	—	5	2.947	0.0071	0.24	5	2.721	0.0048	0.18
S12	1	0	-1	0	0	—	—	5	2.314	0.0164	0.71	5	2.215	0.0200	0.90
S13	1	0	4	9	0	—	—	5	3.113	0.0124	0.40	5	2.934	0.0180	0.61
S14	1	0	-1	0	3	—	—	5	2.884	0.0390	1.35	5	2.617	0.0336	1.28
S15	1	0	3	2	0	—	—	5	2.430	0.0101	0.42	5	2.344	0.0037	0.16
T1	2	-4	2	0	3	0	1	5	3.088	0.0200	0.65	5	2.804	0.0179	0.64
T2	2	-4	5	7	1	2^{-2}	0.75	5	3.493	0.0112	0.32	5	3.320	0.0157	0.47
T3	2	-4	5	5	3	2^{-5}	0.25	5	3.454	0.0124	0.36	5	3.178	0.0103	0.32
T4	2	-4	1	0	1	2^{-2}	1	5	2.942	0.0180	0.61	5	2.741	0.0426	1.55
T5	2	-4	5	1	1	0	0.75	5	3.052	0.0042	0.14	5	2.827	0.0032	0.11
T6	2	-2	3	1	3	0	0.75	5	2.814	0.0051	0.18	5	2.585	0.0028	0.11
T7	2	-2	4	7	1	2^{-2}	1	5	3.148	0.0168	0.53	5	3.008	0.0043	0.14
T8	2	-2	0	0	1	2^{-3}	0.75	5	2.758	0.0121	0.44	5	2.551	0.0291	1.14
T9	2	-2	2	5	1	0	1	5	3.115	0.0082	0.26	5	3.172	0.0046	0.15
T10	2	-2	0	0	3	2^{-4}	0.25	5	3.145	0.0289	0.92	5	2.794	0.0232	0.83
T11	2	0	4	2	2	2^{-3}	0.5	5	2.856	0.0045	0.16	5	2.679	0.0083	0.31
T12	2	0	-1	0	1	0	1	5	2.406	0.0080	0.33	5	2.305	0.0077	0.33
T13	2	0	-1	0	3	0	0.25	5	2.791	0.0090	0.32	5	2.554	0.0065	0.26
T14	2	0	4	8	1	0	1	5	3.014	0.0160	0.53	5	2.840	0.0102	0.36
T15	2	0	2	3	3	0	1	5	2.579	0.0043	0.17	5	2.419	0.0052	0.22

Table 18: Run-to-run variation of the validation loss for each measured setup. The same 30 setups are trained for both language pairs. n is the number of training runs, $\bar{L}$ their mean validation loss, s the standard deviation across runs, and CV% the ratio $s/\bar{L}$ in percent. r_1 and r_f are the target-language ratios of the first and final stage of two-stage training.

Task	ρ
JAQKET	-0.378
JEMHopQA	-0.555
JSQuAD	-0.611
NIILC	-0.117
Average	-0.696

Table 19: Pearson correlation coefficients (ρ) between validation loss and downstream task performance (character F1 scores). All tasks show negative correlation, indicating that lower validation loss tends to correspond to better task performance. Note that “Average” represents the correlation with the mean F1 score across all tasks, not the mean of individual correlation coefficients.