SCALING LAWS FOR MIXTURE PRETRAINING UNDER DATA CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

As language models scale, the amount of data they require grows – yet many target data sources, such as low-resource languages or specialized domains, are inherently limited in size. A common strategy is to mix this scarce but valuable target data with abundant generic data, which presents a fundamental trade-off: too little target data in the mixture underexposes the model to the target domain, while too much target data repeats the same examples excessively, yielding diminishing returns and eventual overfitting. We study this trade-off across more than 2,000 language-model training runs spanning multiple model and target dataset sizes, as well as several data types, including multilingual, domain-specific, and quality-filtered mixtures. Across all settings, we find that repetition is a central driver of target-domain performance, and that mixture training tolerates much higher repetition than single-source training: scarce target corpora can be reused 15–20 times, with the optimal number of repetitions depending on the target data size, compute budget, and model scale. Next, we introduce a repetition-aware mixture scaling law that accounts for the decreasing value of repeated target tokens and the regularizing role of generic data. Optimizing the scaling law provides a principled way to compute effective mixture configurations, yielding practical mixture recommendations for pretraining under data constraints.

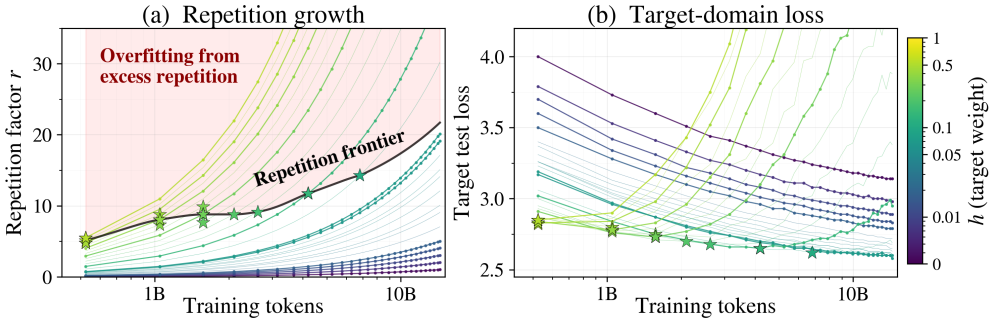


Figure 1: Repetition dynamics for 143M model, 50M German tokens mixed with English, h defining the weight of German data in the training mix. **(a)** Repetition factor r grows with training tokens; beyond the repetition frontier, loss begins to increase. **(b)** German validation loss vs. training tokens. Stars indicate overfitting onset.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable performance across a wide range of language understanding, mathematics, science, and knowledge-intensive tasks (Luong & Lockhart, 2025; Woodruff et al., 2026; Singh et al., 2025; Anthropic, 2026). While much of this success can be attributed to large scale pretraining corpora exceeding trillions of tokens (AI et al., 2025; Li et al., 2024), many language model pretraining scenarios involve data that cannot be freely scaled, including low-resource languages, specialized domains, and curated datasets, which offer far less unique data. During pretraining, this data is mixed with abundant generic data, such as pairing low-resource language text with English text, or a specific math corpus with generic web text.

However, mixing introduces a new trade-off. The limited data must be repeated for enough of the total training to provide sufficient target-domain signal, but high repetition leads to memorization

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and eventual overfitting. Figure 1 illustrates this challenge for a 143M model with 50M German tokens mixed with English. Higher fractions of German data start overfitting (Figure 1b) as the number of repetitions crosses a frontier (Figure 1a). This raises the question: *how do model scale, data size, and repetition jointly shape the outcome of mixture pretraining when target data in the mixture is constrained?*

Prior work has studied these two dimensions separately. Muennighoff et al. (2023) derive scaling laws for data-constrained training, showing that up to 4 repetitions over a fixed monolithic dataset are nearly as valuable as unique data. However, this result, widely adopted as a practical ceiling for data repetition, applies to single-source training where *all* tokens are repeated. In mixture pretraining, only the constrained domain is repeated while the generic domain continues to supply fresh tokens, which is a structurally different regime. Conversely, data mixing laws (Ye et al., 2025; Xie et al., 2023; Shukor et al., 2025) optimize the composition of training mixtures to maximize downstream performance, but assume each data source has unlimited data. Our work sits at the intersection: we study how to optimally mix data-constrained sources with abundant ones, jointly optimizing mixture weights and repetition — a practical setting that arises in many real-world pretraining scenarios.

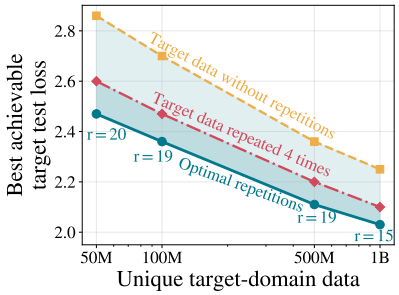


Figure 2: Best achievable German test loss by German data size (539M model).

Our contributions are:

- A systematic empirical study of over 2000 training runs spanning model sizes from 101M to 1.2B parameters and different target data sizes and types: multilingual (German, French, Swahili), multi-domain (maths, scientific papers, Wikipedia), and quality-filtered subsets.
- Empirical findings on repetition in data mixtures: repeated tokens follow diminishing returns predictably across all data types; optimal repetition scales with target dataset size and compute budget; and larger models consistently extract more from limited data despite overfitting faster. Crucially, abundant generic data regularizes learning and allows for far higher repetition than reported for single-source training without performance degradation, with optimal repetitions for target task performance reaching 15–20 times (Figure 2).
- A scaling law for data-constrained mixtures that predicts target-domain loss as a function of target data size, mixture ratio, and model size. We demonstrate that the law holds across all settings in our study (multilingual, multi-domain, and quality-filtered data), can be fitted at small scales and then extrapolates to larger scales, and can be used to estimate accurately the optimal number of repetitions, where adaptations of prior single-source and mixture laws fail. Also, we extend our experiments to the case of two scarce target domains.

Together, our findings pave the way to more principled mixture selection when some domains have limited data.

2 Related Work

Neural scaling laws The study of scaling laws for language models was popularized by Kaplan et al. (2020), who showed that loss decreases as a power law in model size, dataset size, and compute. Hoffmann et al. (2022) refined these findings with the Chinchilla scaling laws, establishing compute-optimal training recipes that balance model size and data. Relating to our work, Muennighoff et al. (2023) generalize Chinchilla to a single data-constrained domain, finding that repeating data up to approximately four epochs causes negligible degradation, with meaningful gains extending to roughly 16 epochs and ending at 40. They propose scaling laws that model the diminishing value of repeated tokens. Subsequent work has extended scaling laws to downstream task performance (Wei et al., 2022), mixture-of-experts architectures (Abnar et al., 2025; Ludziejewski et al., 2024; Wang et al., 2024), continued pretraining (Que et al., 2024; Liew & Kato, 2025; Seto et al., 2026), finetuning (Béthune et al., 2025; Zhang et al., 2024), and data quality (Chang et al., 2024). Our work builds on this line of research but addresses a regime none of it covers: single-source data-constrained scaling repeats every token (Muennighoff et al., 2023), while mixture laws assume no component

is constrained (Ye et al., 2025; Shukor et al., 2025). We characterize the under-explored regime in between, where only one component of a data mixture is unlimited while others are constrained, and tolerable repetition is several times what single-source results would predict.

Data mixing for language models Training on a mixture of data domains has become standard, with corpora aggregated from multiple web sources (Gao et al., 2020; Soboleva et al., 2023; Soldaini et al., 2024) with varying data size per domain, and recent models are trained on carefully tuned domain mixtures (Bakouch et al., 2025; OLMo et al., 2025). Prior work also focuses on optimizing domain weights using distributionally robust optimization (Xie et al., 2023; Fan et al., 2024), predicting weights by regressing from smaller model training runs (Liu et al., 2025), or by discovering domains through clustering the pretraining corpus (Diao et al., 2025; Grangier et al., 2025). The domain mixture scaling law (DMSL) framework (Ye et al., 2025; Shukor et al., 2025) models how per-domain loss depends on mixture weights and total compute. In practice, repetition in such mixtures is often capped by hand: OLMo et al. (2025) avoid repeating any domain more than $\sim$4–7$\times$, adding token constraints on top of unconstrained mixing-law methods (Liu et al., 2025). Our repetition-aware scaling law extends DMSL with the effect of data repetition: instead of a global repetition cap, it predicts the optimal repetition per constrained domain. Finally, Jeha et al. (2026) compare data mixing against hyperparameter tuning; our focus is instead on setting the mixture optimally, which eliminates the need for per-configuration sweeps.

Data-constrained pretraining Several works study data-constrained pretraining. Hernandez et al. (2022) propose parametric models for the value of repeated data in multi-epoch training, and Goyal et al. (2024) compare repeating high-quality data against training on fresh lower-quality data. When high-quality data for the target domain is constrained, synthetic data has been used as an alternative to repetition: Gunasekar et al. (2023) pretrained on synthetic "textbook quality" data, and Maini et al. (2024) demonstrate that rephrasing web data offer alternatives to repeating data. However, synthesis requires a generator faithful to the target distribution, which is often unavailable precisely when data is scarcest. For multilingual settings, Chung et al. (2023) studies sampling strategies that account for repetition of tail languages with unconstrained head languages, and Seto et al. (2025) show that auxiliary-language data can partially substitute for constrained target-language data in bilingual pretraining, with diminishing returns from model scaling. Similarly, mixing low-resource languages with high-resource languages is a common strategy (Joshi et al., 2020). We instead study settings where data is constrained in at least one target domain and unconstrained in others, and provide a recipe for determining the target domain loss, tangential to approaches that enlarge the target set or add generic data.

3 Methodology: Mixture Training and Datasets

We consider a mixture pretraining setup with two data sources, where one *target* source has limited size and a *generic* source provides effectively unlimited data. Let D_{target} be the target data size, i.e., the number of unique target-domain tokens, D_{total} the total number of training tokens (i.e., the total compute budget), $h \in [0, 1]$ the fraction of D_{total} devoted to the target domain. The number of repetitions r is the number of times the unique target tokens are repeated throughout training:

$$r = h \cdot D_{\text{total}}/D_{\text{target}} \quad . \tag{1}$$

For a fixed compute budget D_{total} and data pool D_{target}, increasing the target weight h increases the repetition factor r, and vice versa. For example, with a total budget of $D_{\text{total}} = 10$B tokens and $D_{\text{target}} = 50$M available target tokens, $h = 0.2$ gives $r = 40$, but if we have 1B available target tokens, then $r = 2$, as the same mixture weight means different amounts of repetition (see Figure 3d for the visualization). Data sources are mixed at the sequence level: each training sequence comes from either the target or the generic pool, in proportion h to $1 - h$; r is thus the average number of times a target token is seen, with individual documents seen more or fewer times. The generic domain is assumed abundant enough to never repeat.

To test whether the repetition-diversity trade-off generalizes, we consider three data-constrained scenarios commonly encountered in practice.

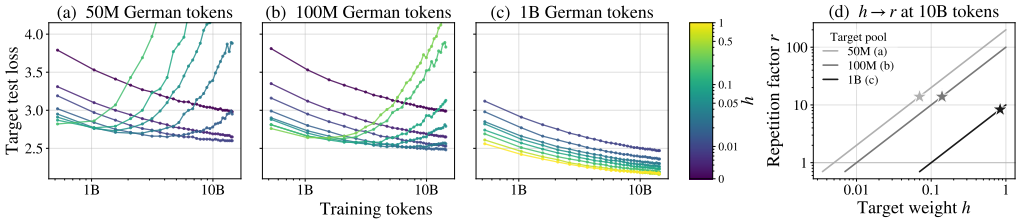


Figure 3: **(a–c)** German validation loss vs. total training tokens for the 143M model across three target data budgets, each curve a different target weight h. **(d)** How h maps to repetition r (Eq. 1) at a fixed 10B-token budget; stars mark the empirically best h.

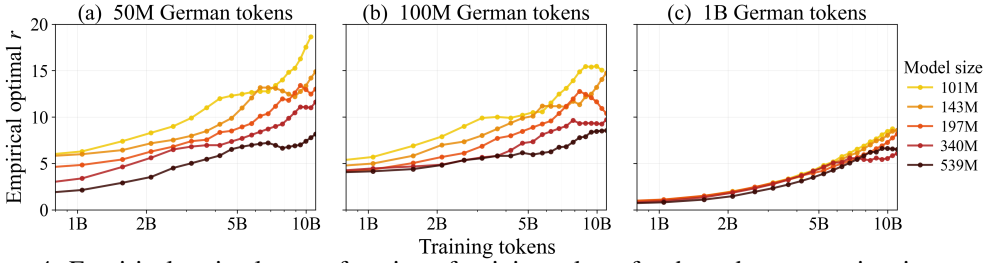


Figure 4: Empirical optimal r as a function of training tokens for three data constraint sizes across model sizes. The x-axis is the total budget D_{total} across both domains and each panel title is the fixed unique-target pool D_{target}, so a run at target weight h makes $r = h \cdot D_{\text{total}}/D_{\text{target}}$ passes over the pool. Optimal repetition grows steadily with the training budget.

Multilingual The target domain is German text from FineWeb2 (Penedo et al., 2025), artificially constrained to 50M, 100M, 500M, or 1B tokens[1], as well as a control unlimited setting with $D_{\text{target}} = \infty$. The generic domain is English web text from FineWeb (Penedo et al., 2024), treated as effectively unlimited. To verify that findings are not language-specific, we also run 50M, 100M, and 500M data size experiments with French and Swahili (which is typologically distant from English, has far less web data and thus provides a naturally low-resource test case) as target languages, also derived from FineWeb2. Evaluation is done on a held-out set of FineWeb2.

Multi-domain The target domain is OpenWebMath (Paster et al., 2024), a collection of mathematical web pages, constrained to 10M, 100M, or 1B tokens. The generic domain is a subset of DCLM (Li et al., 2024), a large-scale curated English web corpus. We additionally run three-domain experiments using Wikipedia and peS2o (Soldaini & Lo, 2023), a corpus of scientific papers derived from Semantic Scholar, both sourced from the OLMo data mix (OLMo et al., 2024), with dataset sizes ranging from 10M–500M and 50M–2.5B, respectively. As the generic domain, we use Fineweb (Penedo et al., 2024) and Nemotron-CC (Su et al., 2025). Evaluation is done on held-out target sets.

Quality We score DCLM (Li et al., 2024) documents using the original fasttext quality classifier[2] and select the top $\{1, 5, 10, 20\}\%$ of the DCLM base corpus as the *high-quality* (HQ) target set, with the full DCLM base data as the *low-quality* generic set. Unlike the different domain and language experiments, exploring data quality allows us to explore the trade-off of amount of data from the data constrained domain, and closeness to the target domain. Stricter filtering yields a lower set of HQ data and thus more repetition at a given h, but the data is expected to be closer on average to the target set. In contrast, a lower filter percent will yield more data, but will be further from the target set. Evaluation is performed on the data used to train the original classifier.

Experiments use GPT-2-style autoregressive decoder-only Transformer language models spanning sizes from 101M to 1.2B non-embedding parameters (see Appendix B for full architecture specifications). Models are trained for approximately $100 \times N$ tokens, where N is the non-embedding parameter count, a sufficient horizon to observe both the benefits and eventual degradation from data repetition. We use a constant learning rate after warmup, as required by the analysis design.[3]

[1] Across all our experimental setups, each smaller set is a subset of a larger one.

[2] `https://huggingface.co/mlfoundations/fasttext-oh-eli5`

[3] We verify our conclusions and scaling law under a more realistic cosine-decay setup (Appendix B): the optimal repetition shifts 13–20% lower, while the qualitative picture and the scaling-law fit remain unchanged.

4 EMPIRICAL FINDINGS

We present results primarily through the German-English bilingual setup due to space constraints, simulating the low-resource scenario with German. Results for other languages and domains are consistent and discussed in Appendix C. We extend our experiments to multiple constrained target domains in Section 7.

Data Repetition Leads to Predictable Overfitting Figure 3 shows target-domain loss for the 143M model across three target dataset sizes, with each curve corresponding to a different target weight h (and, consequently, the amount of target data repetition r; see Eq. 1). When h forces high repetition, loss first improves then rises sharply, reflecting overfitting. Target datasets of larger size delay this degradation because the same h produces less repetition. Crucially, the same target weight can be safe or harmful depending on the size of the available pool: h values that overfit at 50M target tokens are harmless at 1B, since the same h there produces far fewer repetitions. What matters is thus not h in isolation but the repetition it induces, which motivates formulating the scaling law of Section 5 in terms of repetition-aware effective data.

Mixture Training Unlocks High Repetition Tolerance Figure 4 shows the optimal repetition factor (i.e., the value of r that minimizes target loss) as a function of training tokens. Across all settings, optimal r increases steadily with the training budget, reaching up to 15–20 depending on model size and dataset size, substantially exceeding the < 4 epochs widely adopted as a practical ceiling for single-source data-constrained training (Muennighoff et al., 2023) as well as the $\sim$4–7$\times$ caps applied by hand when assembling multi-domain mixtures (OLMo et al., 2025). The difference stems from the generic domain, which provides a continuous supply of fresh tokens that effectively regularizes training and allows the model to absorb far more target-domain repetition before degradation sets in. Since r results from h, dataset size, and training budget (Eq. 1), the target weight needed to reach optimal r varies, e.g., from 9.5% (101M) to 1.9% (539M) for the 50M dataset.

Larger Models Overfit Faster, Yet Still Win Figure 5 shows target-domain loss across five model sizes at two fixed target weights ($h = 9\%$ and $h = 20\%$) with 100M German tokens. Dashed lines show the best-loss envelope, the minimum achievable loss at each training step across all h values. At $h = 9\%$, larger models plateau earlier while smaller models are still improving. At $h = 20\%$, the larger models eventually overfit (increasing target loss), while the smallest models plateau or still improve. This reflects the fact that larger models fit the data more quickly. Yet in every case, larger models reach lower minima before degradation sets in, and the *best-loss* envelope consistently favors larger models at every training budget. Thus, we can conclude that *scaling up remains beneficial* even under tight data constraints, but the optimal operating window narrows with model size. This also remains consistent with Figure 4, where the optimal repetition decreases with model size.

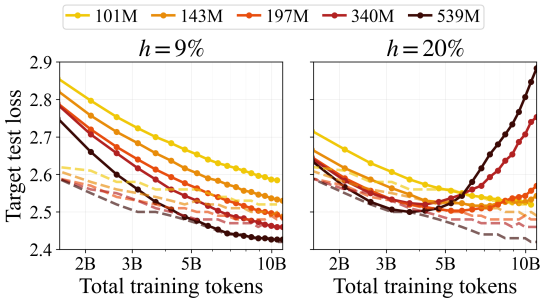


Figure 5: Target-domain loss across model sizes at 9% and 20% target weights for fixed 100M German pool size. Dashed lines show the best-loss envelope across all h values.

Predictable Scaling Figure 7 shows *best* achievable loss as a function of unique data available among all weights h and total training tokens, for different model sizes. We see that these curves are remarkably regular: scaling up model size, or the number of available target data (e.g., by collecting fresh target data) both have a very predictable effect on the loss. We also evaluated the trained models on various benchmarks and verified that these gains and findings are not confined to loss only: model downstream performance improves with both target pool size and model size in the same way, while generic-domain accuracy declines as more of the budget is allocated to the target domain (see Appendix M).

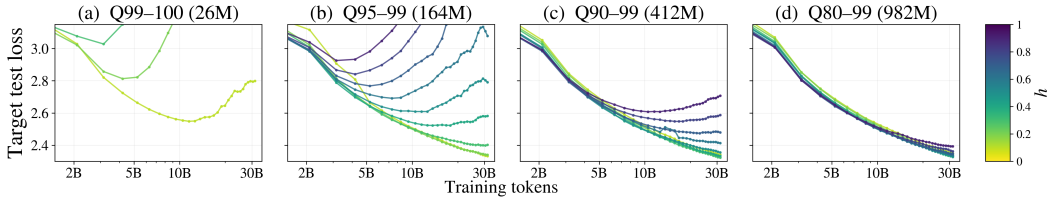


Figure 6: Loss curves for four quality bands ranging from 26M tokens to 982M, and ordered by increasing dataset size. Broader filters enable higher target weights without overfitting.

Broad Target Data Filters Can Beat More Repetitions The preceding experiments treat the domain dataset as fixed and study how repetition degrades loss. In some settings, however, the dataset size is a design choice as one can choose how to classify documents into the domain of interest. For example, using a quality filter at varying quality levels determines how much data remains; a practitioner can trade quality for quantity by adjusting the threshold.

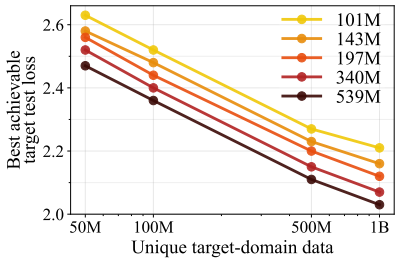


Figure 7: Best achievable target loss by dataset size across different model sizes with optimal repetition.

We use the Quality Data Mixture to determine whether accepting lower quality data to reduce repetition yields better outcomes than repeating the highest quality data[4]. To ground the experiment, we first confirm with a high quality target dataset of the 99th percentile and show loss curves at three dataset sizes (26M, 132M, 264M tokens) (see Appendix G). At the smallest scale, only $h \leq 0.1$ avoids overfitting, and larger sets progressively unlock higher sustainable target weights. This establishes the baseline that for a fixed quality threshold, the repetition penalty imposes the same constraint on the target loss.

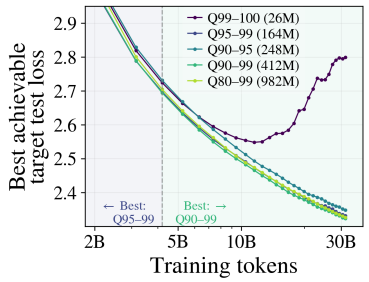


Figure 8: Best loss by quality band. Dashed line marks the crossover where Q90-99 overtakes Q95-99.

Next, we vary the quality filter threshold while holding the data source fixed. We fix the data constrained set such that the top 1% (Q99-100) is 26M tokens, and add increasing amounts of data to the data constrained dataset at thresholds of 5%, 10%, and 20% up to 982M tokens. Results are in Figure 6. Relaxing the quality threshold and accepting slightly lower-quality data in exchange for a larger set *consistently outperforms* repeating a narrow high-quality slice. At Q99-100, only $h = 0.1$ avoids overfitting at up to 10B tokens of training, while Q90-99 supports weights up to $h = 0.4$ and Q80-99 shows nearly all weights monotonically decreasing.

The best quality set also depends on the training budget as we further observe that the best quality filter threshold changes at around 5B tokens as seen in Figure 8, where the top 5% high quality documents (Q95-99) perform best due to higher per-token quality, but the top 10% set (Q90-99) overtakes it as training progresses and the smaller dataset saturates. By 30B tokens, Q90-99 achieves loss 2.33 versus 2.80 for Q99-100 due to the repetition penalty. This demonstrates that while repetition is an important factor, when high-quality data is scarce, broadening the filter is preferable to over repeating data.

5 Repetition-Aware Mixture Scaling Law

The goal of this section is to quantify the empirical findings of Section 4. We use scaling laws (Kaplan et al., 2020), which, in their original forms, are simple power laws that allow to predict the loss of a model from its number of parameters N and total number of tokens it has been trained on D_{total}. In our setup, both the target data size D_{target} and the target weight h influence the loss on the target

[4]For many data constrained domains, selecting the amount to filter out is tunable. We study data quality as existing work gives us an already established way of easily varying the amount of data and relevance to the target domain (Li et al., 2024)

domain. Our objective is to obtain a simple and predictive formula for $L_{\text{target}}(N, D_{\text{total}}, D_{\text{target}}, h)$. We focus on the practically relevant regime where the target data is fully consumed at least once, i.e., $r = h \cdot D_{\text{total}}/D_{\text{target}} \geq 1$. When $r < 1$, target tokens are never repeated and mixing reduces to standard mixture selection, as explored, e.g., in (Shukor et al., 2025). In the following, we write scaling laws learned parameters in blue.

Effective data The core building block of our law is the *effective data* D_{eff}, which accounts for the diminishing value of repeated tokens. Following Muennighoff et al. (2023), we define the effective contribution of the *target* domain as:

$$D_T = D_{\text{target}}\big(1 + \rho(r)\big)\,, \qquad \text{with} \quad \rho(r) = r_1\Big(1 - e^{-(r-1)/r_1}\Big)\,, \tag{2}$$

where r_1 is a parameter that controls the effect of repeated data. For small $r \ll r_1 + 1$, we have $\rho(r) \approx r - 1$ so that $D_T \approx r \cdot D_{\text{target}}$: each pass counts fully. For large r, D_T saturates at $(1 + r_1)\, D_{\text{target}}$, reflecting the diminishing value of further repetitions. The total effective data is:

$$D_{\text{eff}} = (1 - h)\, D_{\text{total}} + \tau\, D_T\,, \tag{3}$$

where the $(1 - h)\, D_{\text{total}}$ term corresponds to the number of tokens seen from the generic dataset and τ controls the relative value of target-domain tokens compared to generic-domain tokens. This formulation directly encodes two observations from Section 4. The saturation of $\rho(r)$ at r_1 is the quantitative form of the diminishing value of repeated target tokens: for $r \gg r_1$, further passes add essentially nothing to D_{eff}. Conversely, the unsaturated $(1 - h)\, D_{\text{total}}$ term formalizes why mixture training tolerates far more repetition than single-source training (Figure 4): generic tokens never saturate, so they keep D_{eff} growing even when the target contribution has plateaued.

Scaling law formulas We propose two loss formulas, for fixed and variable model size:

$$L_{\text{fix}} = E + \frac{A}{D_{\text{eff}}^{\alpha}} + \gamma\, h\,, \qquad L_{\text{size}} = E + \frac{C}{N^{\beta}} + \frac{B\, N^{\delta}}{D_{\text{eff}}^{\alpha}} + \gamma\, h\,. \tag{4}$$

In L_{fix}, the irreducible loss E is a constant baseline; the data term $A/D_{\text{eff}}^{\alpha}$ is a Chinchilla-style power-law relating loss to effective data. The weight penalty $\gamma\, h$ is a linear cost on the target weight: it is the opportunity cost of the mixture — weight given to the target domain is weight not spent on fresh generic tokens (see Appendix K for more details). L_{size} additionally includes a Chinchilla-style capacity term C/N^{β} and a data-size coupling N^{δ} ($\delta > 0$) (Hoffmann et al., 2022): the capacity term captures the intrinsic capability gain from scale, while the coupling means that for the same D_{eff}, the data-limited loss component is amplified at larger N, encoding the fact that larger models yield better losses but overfit faster (Figure 5). At fixed N, L_{size} reduces exactly to L_{fix} with $A = B\, N^{\delta}$ and $E_{\text{fix}} = E + C/N^{\beta}$, ensuring consistency. L_{fix} has six fitted parameters (E, A, α, r_1, τ, γ); L_{size} has nine (E, C, β, B, δ, α, r_1, τ, γ).

Fitting Given a collection of training runs indexed by i, each characterised by a total token budget $D_{\text{total},i}$, a target weight h_i, an available target dataset $D_{\text{target},i}$, and optionally a model size N_i, we observe the resulting loss on the target domain ℓ_i. The parameters θ of the scaling law are estimated by minimising the reweighted Huber loss

$$\hat{\theta} = \arg\min_{\theta} \sum_i \omega_i \cdot \mathcal{H}\big(\ell_i - L_\theta(D_{\text{total},i},\, h_i,\, D_{\text{target},i},\, N_i)\big)\,, \tag{5}$$

where $\mathcal{H}$ is the Huber loss and the weights $\omega_i = \max(r_i \cdot h_i, \epsilon)$ with $\epsilon = 0.01$ emphasise the high-repetition, high-fraction regime: h alone would under-weight high-repetition runs at moderate fractions, while r alone would under-weight high-fraction runs with large datasets (where r is low). The qualitative ranking of methods is unchanged under alternative weights; see Appendix L. Following Shukor et al. (2025), we minimise Eq. 5 using basin-hopping optimisation with 100 random restarts to avoid poor local minima in the non-convex loss landscape.

6 Scaling Law Results

We evaluate the scaling law on experimental setups described in Section 3. For each setup, L_{fix} is fitted independently per model size, and L_{size} is fitted jointly across all model sizes. To test extrapolation for L_{fix}, we fit on the first 50% of training steps and test on second half. For L_{size}, we fit on all but the largest model scale, and test on the held-out largest model scale.

Table 1: Test weighted R^2. **Left:** fixed-size formulas, fitted independently per model size. **Right:** multi-size formulas, fitted on smaller model sizes and evaluated on the held-out largest model.

	German	Maths	Quality	Wiki/peS2o		German	Maths
L_{fix}	**0.95**	**0.88**	**0.71**	**0.80**	L_{size}	**0.65**	**0.73**
Repetition-agnostic	0.78	0.78	0.14	0.72	Repetition-agnostic+N	0.59	0.71
Utility decay	0.72	0.55	−0.64	0.79	Utility decay+N	0.56	0.69
Domain-agnostic	−40.7	−0.49	−2.19	−1.16	Domain-agnostic+N	−0.23	−0.77

Baselines We compare against three other scaling laws formulas inspired by the literature. *Repetitions-agnostic* (Shukor et al., 2025) replaces D_{eff} with $(1-h)\,D_{\text{total}} + \tau\, h\, D_{\text{total}}$, treating repeated tokens as unique. *Domain-agnostic* (Muennighoff et al., 2023) uses a single saturating function of total tokens without distinguishing domains. *Utility decay* (Goyal et al., 2024) models repetition through a decaying exponent on the data term. See Appendix J for details.

Optimal mixture prediction Table 1 reports the weighted coefficient of determination (wR^2) on the held-out test split (train wR^2 in Appendix K), where each observation is weighted by $\max(r \cdot h, \epsilon)$ to emphasise the high-repetition regime. L_{fix} consistently achieves the best test wR^2, extrapolating beyond the fitting range. *Repetitions-agnostic* fits the training range reasonably but fails to extrapolate to high repetitions. *Domain agnostic* fails completely: treating all tokens as interchangeable does not capture the loss dynamics. *Utility decay* performs honorably on several datasets.

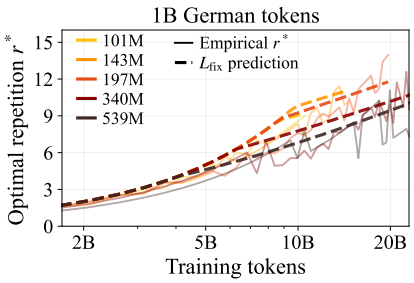


Figure 9: Predicted vs. empirical optimal repetition r^* for the German target dataset of 1B tokens. Solid: empirical optimum; dashed: L_{fix} prediction.

The primary use of the scaling law is to predict the optimal target weight h^* for a given total token budget D_{total} and available target domain dataset D_{target}. Once the scaling law parameters are estimated, we can simply solve

$$h^* = \arg\min_{h \in [0,1]} L_{\text{fix}}(D_{\text{total}},\, h,\, D_{\text{target}}) \tag{6}$$

by grid search over h. Figure 9 shows an example of empirical vs. estimated optimal repetition.

To quantify the practical cost of suboptimal mixture predictions, we measure the fraction of training tokens *wasted* by following each formula's recommendation instead of the oracle (Table 2). At a given budget D_{total}, the formula recommends a fraction h^*_{pred}; we interpolate the empirical loss at that fraction and find the budget D' at which the run using the empirically best fraction at each step first reached the same loss. The wasted fraction $(D_{\text{total}} - D')/D_{\text{total}}$ directly measures how much compute could have been saved with perfect knowledge of the optimal mixture. Detailed results are in Appendix K.

Table 2: Fraction of training tokens wasted by following each formula's predicted optimal mixture instead of the oracle.

	Median	Mean	p90
L_{fix}	**26**	**34**	**76**
Rep-agn.	88	73	99
Util. dec.	31	41	95
Dom.-agn.	47	59	98

7 Multiple Domain Experiments

In order to test whether our findings and scaling law formula extend to settings with more than one constrained domain, we set up experiments with a mixture of three domains: one unconstrained generic source and two constrained target domains.[5] We use peS2o and Wikipedia as targets, while the generic data is sourced either from FineWeb or from Nemotron. We vary the dataset sizes across three configurations: 5× less, base, and 5× more data. Our primary metric is the average loss across both target domains, analogous to evaluating on a test set that combines text from both domains.

[5]This is close to practical settings: for example, in the Dolma 3 mix (OLMo et al., 2025), three of six sources are repeated (up to ∼4.5×) while the remaining three are consumed in under one epoch — i.e., a few constrained domains alongside effectively unlimited generic data.

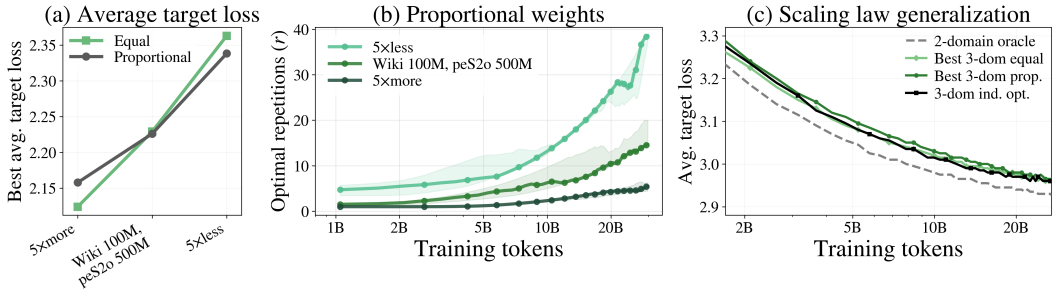


Figure 10: Multi-domain experiments (101M model, Wikipedia + peS2o + English generic data: FineWeb in (a), Nemotron in (b, c)). **(a)** Best achievable average target loss under equal vs. proportional weighting across three set-size configurations. **(b)** Optimal repetitions under proportional weighting with 10% compute confidence band (shaded). **(c)** Independently optimizing r per domain using the bilingual scaling law outperforms the best grid-searched proportional weighting.

Weighting Strategies First, we compared two weighting strategies: *equal weighting* ($h_{\text{wiki}} = h_{\text{peS2o}}$) and *proportional weighting*, where fractions are set proportional to dataset size ($h_{\text{wiki}}/h_{\text{peS2o}} = D_{\text{wiki}}/D_{\text{peS2o}}$). Our experiments demonstrate (Figure 10a) that by average loss, proportional weighting outperforms equal weighting when data is scarce, while equal weighting is better in the data-rich regime. The two strategies cross near the base configuration, suggesting that proportional weighting becomes increasingly beneficial as repetition grows. For the rest of this section, we focus on the proportional weighting; the results for equal weighting are in Appendix I.

Empirical Findings The key findings from the bilingual experiments carry over to the multi-domain setting. Figure 10b shows that optimal repetition grows steadily with training tokens across all set-size configurations, with smaller sets requiring substantially higher repetition: the 5×less variant reaches $r > 30$ while 5×more stays below $r \approx 7$. This mirrors the bilingual experiments: the generic domain continues to regularize training even with multiple constrained domains present, enabling high repetition tolerance. Importantly, the choice of r need not be exact. We quantify this through a 10% compute confidence band (shaded regions in Figure 10b): at each checkpoint, we identify all values of r that achieve loss no worse than the best loss achievable with 10% less compute. The bands are wide, particularly for smaller sets, indicating that even rough estimates of optimal r yield near-optimal target loss.

Scaling Law Generalization The similarity of these findings to the bilingual setting raises the question of whether the scaling law transfers directly to multi-domain mixtures. We test this by setting each domain's repetition factor independently using the optimal r predicted by the bilingual scaling law for the corresponding set size ($r_{\text{wiki}} = 35$, $r_{\text{peS2o}} = 17$). Figure 10c shows that this independently optimized configuration consistently outperforms the best proportional weighting and narrows the gap to the 2-domain oracle (the average loss achieved by training each domain separately with English). While a gap to the oracle remains, this result suggests that costly multi-domain sweeps over all possible combinations of domain weights can be avoided: our two-domain scaling law can independently estimate the optimal r for each constrained domain, and these can be combined into a mixture, replacing an entire experimental grid with a single, better-performing run.

8 Conclusion

We studied mixture pretraining under data constraints across diverse data types: multilingual, multi-domain, and quality-filtered. Our experiments reveal consistent patterns across all settings: repeated target-domain tokens yield diminishing returns, optimal repetition scales predictably with data availability, and larger models extract more from limited data. Crucially, mixture training tolerates substantially higher repetition than single-source training, with generic data acting as an implicit regularizer. We formalized these dynamics in a scaling law that accurately predicts the optimal mixture across experimental setups, outperforming baselines that ignore either repetition or domain structure. In practice, this allows training on scarce target data without expensive sweeps: given only the target pool size and compute budget, the law prescribes the mixture ratio that maximizes target-domain performance in a single run.

AI USE STATEMENT

In this work, we used generative AI tools to draft parts of the paper, to aid and polish the writing, to improve the quality of figures and plots, and to support research execution (specifically to organize and keep track of scaling-law experiments and to implement parts of the code for the data preprocessing, training, and evaluation pipelines). We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology, or interpret results; proofs, translation, and qualitative or thematic data analysis are not applicable to this work. We have reviewed all AI-assisted work. All code generated with the help of AI tools was carefully reviewed by the authors, and its outputs were verified through sanity checks and comparison with expected behavior. All AI-drafted and AI-corrected text was carefully proofread by all authors, and all references were verified manually against their original sources. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Samira Abnar, Harshay Shah, Dan Busbridge, Alaaeldin El-Nouby, Joshua M. Susskind, and Vimal Thilak. Parameters vs flops: Scaling laws for optimal sparsity for mixture-of-experts language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/abnar25a.html.

Essential AI, :, Andrew Hojel, Michael Pust, Tim Romanski, Yash Vanjani, Ritvik Kapila, Mohit Parmar, Adarsh Chaluvaraju, Alok Tripathy, Anil Thomas, Ashish Tanwer, Darsh J Shah, Ishaan Shah, Karl Stratos, Khoi Nguyen, Kurt Smith, Michael Callahan, Peter Rushton, Philip Monk, Platon Mazarakis, Saad Jamal, Saurabh Srivastava, Somanshu Singla, and Ashish Vaswani. Essential-web v1.0: 24t tokens of organized web data, 2025. URL https://arxiv.org/abs/2506.14111.

Anthropic. Introducing claude opus 4.6. *Anthropic Announcements*, 2026. URL https://www.anthropic.com/news/claude-opus-4-6.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, Kashif Rasul, Nathan Habib, Clémentine Fourrier, Hynek Kydlicek, Guilherme Penedo, Hugo Larcher, Mathieu Morlon, Vaibhav Srivastav, Joshua Lochner, Xuan-Son Nguyen, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM3: smol, multilingual, long-context reasoner. https://huggingface.co/blog/smollm3, 2025.

Louis Béthune, David Grangier, Dan Busbridge, Eleonora Gualdoni, Marco Cuturi, and Pierre Ablin. Scaling laws for forgetting during finetuning with pretraining data injection. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=vWMij23BmQ.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 7432–7439. AAAI Press, 2020. doi: 10.1609/AAAI.V34I05.6239. URL https://doi.org/10.1609/aaai.v34i05.6239.

Ernie Chang, Matteo Paltenghi, Yang Li, Pin-Jie Lin, Changsheng Zhao, Patrick Huber, Zechun Liu, Rastislav Rabatin, Yangyang Shi, and Vikas Chandra. Scaling parameter-constrained language models with quality data. In Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 80–97, Miami, Florida, US, November 2024. Association for Com-

putational Linguistics. URL https://aclanthology.org/2024.emnlp-industry.8/.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kXwdL1cWOAi.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Shizhe Diao, Yu Yang, Yonggan Fu, Xin Dong, Dan Su, Markus Kliegl, Zijia Chen, Peter Belcak, Yoshi Suhara, Hongxu Yin, Mostofa Patwary, Yingyan (Celine) Lin, Jan Kautz, and Pavlo Molchanov. Nemotron-climb: Clustering-based iterative data mixture bootstrapping for language model pre-training. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/24a8968affe71ffe4067d022b9d16566-Abstract-Datasets_and_Benchmarks_Track.html.

Simin Fan, Matteo Pagliardini, and Martin Jaggi. DOGE: domain reweighting with generalization estimation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=7rfZ6bMZq4.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Sachin Goyal, Pratyush Maini, Zachary C. Lipton, Aditi Raghunathan, and J. Zico Kolter. Scaling laws for data filtering - data curation cannot be compute agnostic. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 22702–22711. IEEE, 2024. doi: 10.1109/CVPR52733.2024.02142. URL https://doi.org/10.1109/CVPR52733.2024.02142.

David Grangier, Simin Fan, Skyler Seto, and Pierre Ablin. Task-adaptive pretrained language models via clustered-importance sampling. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=p6ncr0eTKE.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need, 2023. URL https://arxiv.org/abs/2306.11644.

Alexander Hägele, Elie Bakouch, Atli Kosson, Loubna Ben Allal, Leandro von Werra, and Martin Jaggi. Scaling laws and compute-optimal training beyond fixed training

durations. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/8b970e15a89bf5d12542810df8eae8fc-Abstract-Conference.html.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Danny Hernandez, Tom Brown, Tom Conerly, Nova DasSarma, Dawn Drain, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Tom Henighan, Tristan Hume, Scott Johnston, Ben Mann, Chris Olah, Catherine Olsson, Dario Amodei, Nicholas Joseph, Jared Kaplan, and Sam McCandlish. Scaling laws and interpretability of learning from repeated data, 2022. URL https://arxiv.org/abs/2205.10487.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Paul Jeha, Anastasiia Sedova, Louis Béthune, Skyler Seto, Jes Frellsen, Pierre Ablin, and Natalie Schluter. Mix, don't tune: Bilingual pre-training outperforms hyperparameter search in data-constrained settings, 2026. URL https://arxiv.org/abs/2605.13225.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. The state and fate of linguistic diversity and inclusion in the NLP world. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 6282–6293, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.560. URL https://aclanthology.org/2020.acl-main.560/.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL https://arxiv.org/abs/2001.08361.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012/.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick Scott Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee F. Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Josh Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah M. Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Raghavi Chandu, Thao Nguyen, Igor Vasiljevic, Sham M. Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alex Dimakis, Yair

Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-lm: In search of the next generation of training sets for language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/19e4ea30dded58259665db375885e412-Abstract-Datasets_and_Benchmarks_Track.html.

Seng Pei Liew and Takuya Kato. From acceleration to saturation: Scaling behavior of bootstrapped language model pretraining. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*, 2025. URL https://openreview.net/forum?id=PhsneSYvWK.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=5BjQOUXq7i.

Jan Ludziejewski, Jakub Krajewski, Kamil Adamczewski, Maciej Pióro, Michal Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygózdz, Piotr Sankowski, Marek Cygan, and Sebastian Jaszczur. Scaling laws for fine-grained mixture of experts. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=yoqdlynCRs.

Thang Luong and Edward Lockhart. Advanced version of gemini with deep think officially achieves gold-medal standard at the international mathematical olympiad. *Google DeepMind Blog*, 2025.

Pratyush Maini, Skyler Seto, Richard Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the web: A recipe for compute and data-efficient language modeling. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14044–14072, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.757. URL https://aclanthology.org/2024.acl-long.757/.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A. Raffel. Scaling data-constrained language models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/9d89448b63ce1e2e8dc7af72c984c196-Abstract-Conference.html.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious, 2024. URL https://arxiv.org/abs/2501.00656.

Team OLMo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal

Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025. URL https://arxiv.org/abs/2512.13961.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In Katrin Erk and Noah A. Smith (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, Berlin, Germany, 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1144. URL https://aclanthology.org/P16-1144/.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=jKHmjlpViu.

Guilherme Penedo, Hynek Kydlícek, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin A. Raffel, Leandro von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/370df50ccfdf8bde18f8f9c2d9151bda-Abstract-Datasets_and_Benchmarks_Track.html.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. FineWeb2: One pipeline to scale them all – adapting pre-training data processing to every language, 2025. URL https://arxiv.org/abs/2506.20920.

Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/b6341525cd84f3be0ef203e4d7cd8556-Abstract-Conference.html.

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. D-CPT law: Domain-specific continual pre-training scaling law for large language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/a4628e9fbd3002a554923642f74d5d6b-Abstract-Conference.html.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 8732–8740. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6399.

Skyler Seto, Maartje Ter Hoeve, Richard He Bai, Natalie Schluter, and David Grangier. Training bilingual LMs with data constraints in the targeted language. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19096–19122, Vienna, Austria, 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.977. URL https://aclanthology.org/2025.findings-acl.977/.

Skyler Seto, Pierre Ablin, Anastasiia Filippova, Jiayuan Ye, Louis Bethune, Angelos Katharopoulos, and David Grangier. Optimal splitting of language models from mixtures to specialized domains, 2026. URL https://arxiv.org/abs/2603.19149.

Mustafa Shukor, Louis Béthune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. Scaling laws for optimal data mixtures. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/bc1d640f841f752c689aae20b31198c1-Abstract-Conference.html.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, CJ Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, DJ Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang, Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang,

Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Y. Guan, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, RJ Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song Mei, Sonya Egoian, Spencer Kim, Spug Golden, SQ Mah, Steph Lin, Stephen Imm, Steve Sharpe, Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer, Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomek Korbak, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. Openai gpt-5 system card, 2025. URL https://arxiv.org/abs/2601.03267.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama. https://cerebras.ai/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B.

Luca Soldaini and Kyle Lo. peS2o (Pretraining Efficiently on S2ORC) Dataset. Technical report, Allen Institute for AI, 2023. ODC-By, https://github.com/allenai/pes2o.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan

Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an open corpus of three trillion tokens for language model pretraining research. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15725–15788, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.840. URL https://aclanthology.org/2024.acl-long.840/.

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-CC: Transforming Common Crawl into a refined long-horizon pretraining dataset. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2459–2475, Vienna, Austria, 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.123. URL https://aclanthology.org/2025.acl-long.123/.

Siqi Wang, Zhengyu Chen, Bei Li, Keqing He, Min Zhang, and Jingang Wang. Scaling laws across model architectures: A comparative analysis of dense and MoE models in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5583–5595, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.319. URL https://aclanthology.org/2024.emnlp-main.319/.

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. Emergent abilities of large language models. *Trans. Mach. Learn. Res.*, 2022, 2022. URL https://openreview.net/forum?id=yzkSU5zdwD.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. In Leon Derczynski, Wei Xu, Alan Ritter, and Tim Baldwin (eds.), *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pp. 94–106, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4413. URL https://aclanthology.org/W17-4413/.

David P. Woodruff, Vincent Cohen-Addad, Lalit Jain, Jieming Mao, Song Zuo, MohammadHossein Bateni, Simina Branzei, Michael P. Brenner, Lin Chen, Ying Feng, Lance Fortnow, Gang Fu, Ziyi Guan, Zahra Hadizadeh, Mohammad T. Hajiaghayi, Mahdi JafariRaviz, Adel Javanmard, Karthik C. S., Ken-ichi Kawarabayashi, Ravi Kumar, Silvio Lattanzi, Euiwoong Lee, Yi Li, Ioannis Panageas, Dimitris Paparas, Benjamin Przybocki, Bernardo Subercaseaux, Ola Svensson, Shayan Taherijam, Xuan Wu, Eylon Yogev, Morteza Zadimoghaddam, Samson Zhou, Yossi Matias, James Manyika, and Vahab Mirrokni. Accelerating scientific research with gemini: Case studies and common techniques, 2026. URL https://arxiv.org/abs/2602.03837.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/dcba6be91359358c2355cd920da3fcbd-Abstract-Conference.html.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=jjCB27TMK3.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp.

4791–4800, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets LLM finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=5HCnKDeTws.

A Experimental Setup

Multilingual setup For each combination of model size and pool size, we sweep h over a fine grid of 19–27 values (see Table 4 for a summary), with denser sampling at low repetition levels where the loss landscape changes most rapidly. The grid spans number of repetitions from below 1 to above 30, though configurations where the resulting h exceeds 1 are excluded. For larger models, the feasible range of h is narrower because the longer training duration ($100 \times N$ tokens) produces higher repetition at the same h; for example, the 539M model on the 50M pool covers r up to approximately 20.

Beyond German and French, which are high-resource Indo-European languages closely related to English, we include Swahili, a Bantu language with limited lexical overlap with English and substantially less available web data, providing a typologically distant and naturally lower-resource test case. Pool sizes are measured in tokens to ensure a fair comparison in terms of training compute and repetition, as each token incurs the same cost and is counted equally in the number of repetitions, regardless of language. Due to differing tokenizer fertility, equal token counts might correspond to different amounts of raw text across languages. Evaluation is performed on 10,000 samples from held-out test sets of both FineWeb2 (target language) and FineWeb (English), allowing us to measure both target-domain and generic-domain loss at each checkpoint.

Multi-domain setup For the three-domain experiments (Section 7), we train 101M models with a total training budget of 30B tokens (approximately $300 \times N$), increased relative to the bilingual setup to allow sufficient repetition of both target domains. The three pool-size configurations correspond to: 5×less (Wiki 20M, peS2o 100M), base (Wiki 100M, peS2o 500M), and 5×more (Wiki 500M, peS2o 2.5B). We sweep over total target fractions $h_{\text{total}} = h_{\text{wiki}} + h_{\text{peS2o}}$ ranging from 0.02 to 0.60 with step 0.02, under both equal weighting ($h_{\text{wiki}} = h_{\text{peS2o}}$) and proportional weighting ($h_{\text{wiki}}/h_{\text{peS2o}} = D_{\text{wiki}}/D_{\text{peS2o}}$).

Math setup For the OpenWebMath experiments, we sweep h over a linear grid whose density depends on pool size: 26 values for the 10M and 100M pools and 8 values for the 1B pool (60 total per model size). The 805M model uses a denser grid (35, 35, and 20 values for the 10M, 100M, and 1B pools respectively; 90 total), because its longer training horizon (1M vs. 200K steps) produces higher repetition counts that require finer resolution. The h range is adapted to each pool size, spanning up to $h = 0.078$ for the 10M pool, $h = 0.78$ for the 100M pool, and $h = 1.0$ for the 1B pool. The 101M–539M models are trained for 200K steps (13.1B tokens) and the 805M model for 1M steps (65.5B tokens). The resulting tokens-to-parameters ratios range from approximately $130\times$ (101M) down to $24\times$ (539M), with $81\times$ for 805M. These experiments use a SentencePiece tokenizer (Kudo & Richardson, 2018) trained on C4 (Raffel et al., 2020) with a vocabulary of 32,000 tokens, rather than the GPT-2 BPE tokenizer used in other setups. Evaluation is performed on 10,240 sequences (10.5M tokens) held out from OpenWebMath.

Quality setup For the quality filter experiments, we train 101M non-embedding parameter models for 30B tokens, sweeping over $h \in \{0.1, 0.2, \ldots, 1.0\}$. We train models with varying quality pools of primarily 1%, 5%, 10%, and 20%, where the 5%, 10%, and 20% pools exclude the top 1%, to explicitly measure the impact of having lower quality data with less repetition. To test whether our findings hold at larger scale, we additionally train 1.2B parameter models on 30B tokens with four quality pools (Q99–100, Q90–99, Q80–99, and Q50–99), sweeping over $h \in \{0.1, 0.2, \ldots, 1.0\}$. Evaluation is based on the OpenHermes dataset, which is a portion of the classifier training set.

B Training Details

Model architecture All models are GPT-2-style autoregressive decoder-only Transformers with learned positional embeddings. Models are scaled along a depth-based rule where the hidden dimension $d_{\text{model}} = 128 \times L$ for L layers, with an FFN intermediate size of $4 \times d_{\text{model}}$, attention head dimension of 64, and $d_{\text{model}}/64$ attention heads (except the 1.2B models in quality experiments, which use 16 heads of dimension 128). All models use a context length of 1,024 tokens. Further details are provided in Table 3.

Training Multilingual models are trained for approximately $100 \times N$ tokens, where N is the non-embedding parameter count. The three-domain experiments are trained for $300 \times N$, the quality experiments are trained for 30K steps (with batch size of 1024 and context length of 1024); the details about the OpenWebMath experiments are provided above. All models are optimized with Adam (Kingma & Ba, 2015), a constant learning rate following a linear warmup, weight decay of 10^{-2}, gradient clipping at 0.1, and no dropout. The multilingual and Wiki/peS2o experiments use a learning rate of 10^{-3} (selected based on initial experiments for 101M, 143M, and 197M models on English-German data), a warmup of 1% of training steps, a batch size of 256 sequences, and a GPT-2 BPE tokenizer (Radford et al., 2019) with a vocabulary of 50,257 tokens. The quality experiments use a learning rate of 3×10^{-4}, a warmup of 1% of training steps, a batch size of 1,024 sequences, and the same GPT-2 BPE tokenizer. The OpenWebMath experiments use a learning rate of 10^{-4} with μP, a 2,000-step warmup, a batch size of 64 sequences, and a SentencePiece tokenizer (Kudo & Richardson, 2018) trained on C4 (Raffel et al., 2020) with a vocabulary of 32,000 tokens. Each dataset is tokenized separately and concatenated into a single token stream from which training sequences are cut. Mixing operates on the fixed-length sequences, each taken from a single domain, so the realized token-level mixture ratio matches h regardless of the document-length distributions of the two pools. Within a domain, the document order is shuffled. The models are trained using Megatron-LM.

Learning Rate In our main experiments, we use a constant learning rate (after warmup) rather than a cosine or linear decay schedule. Constant or near-constant schedules are common practice in pretraining work (Hägele et al., 2024; Porian et al., 2024); our experimental design also requires it. The central object of analysis is a loss-vs-repetition curve read along a single training run (e.g., Figures 3, 5). A decay schedule couples the learning rate to the training horizon, so checkpoints at different repetition counts would not be directly comparable within a run. Replicating the whole sweep under a decay schedule would require a separate run for each evaluation point, multiplying compute by the number of checkpoints.

Cosine Learning Rate To confirm that our conclusions do not depend on the constant learning rate, we repeat the OpenWebMath + DCLM setup for 101M models on the same three target pools (10M, 100M, and 1B tokens) with the constant learning rate replaced by a warmup followed by cosine decay to 10% of the peak learning rate; everything else is identical to our main experiments. The optimum shifts modestly: the optimal repetition r^* is 13–20% smaller under cosine decay than under a constant schedule, while the qualitative picture is unchanged — r^* still increases steadily with the training budget and still lands well above the single-source $\sim$4-epoch rule of thumb (Muennighoff et al., 2023). The scaling law also still fits: refitting L_{fix} on the cosine runs gives train $wR^2 = 0.978$ and test $wR^2 = 0.805$, in line with the mathematics fits reported in Table 1. The functional form therefore does not depend on the schedule being constant, and r^* remains predictable from small-scale experiments under a realistic schedule, with only a downward correction to its absolute value.

Compute resources All experiments were run on a mix of A100 and H100 GPUs. A single training run requires approximately 40 GPU-hours for the 101M model, 64 for 143M, 100 for 197M, 264 for 340M, 600 for 539M, and 1200 GPU-hours for 805M.

C Consistency of the Findings Across Languages and Domains

The core findings from the German–English bilingual experiments generalize across all settings tested. We replicate the experimental grid for French and Swahili. While French, like German,

Table 3: Model architectures. Parameters refer to non-embedding parameters.

Params	Layers	Hidden	FFN	Heads
101M	8	1024	4096	16
143M	9	1152	4608	18
197M	10	1280	5120	20
340M	12	1536	6144	24
539M	14	1792	7168	28
805M	16	2048	8192	32
1.2B	24	2048	8192	16

Table 4: Main experimental setup summary. D_{target} denotes the range of target data pool sizes; # h is the total number of target fraction configurations per model size.

	N	D_{target}	# h
Multilingual	*German*		
	101M	50M–1B	72
	143M	50M–1B	75
	197M	50M–1B	75
	340M	50M–1B	77
	539M	50M–1B	82
	French		
	101M	50M–500M	63
	143M	50M–500M	64
	197M	50M–500M	63
	340M	50M–500M	55
	539M	50M–500M	57
	Swahili		
	101M	50M–500M	63
	143M	50M–500M	64
	197M	50M–500M	63
	340M	50M–500M	55
	539M	50M–500M	57

	N	D_{target}	# h
Domain	*Math (OpenWebMath)*		
	101M	10M–1B	60
	197M	10M–1B	60
	340M	10M–1B	60
	539M	10M–1B	60
	805M	10M–1B	90
	Wiki + peS2o (3-domain)		
	101M	10M–500M	153
Qual.	*Quality filtering*		
	101M	26M–264M	10
	1.2B	26M–2B	10

is a high-resource Indo-European language closely related to English, Swahili is a Bantu language with agglutinative morphology, limited lexical overlap with English, higher tokenizer fertility, and far less available web data. We observe qualitatively identical behavior in all cases. The same overfitting emerges at high h with small data pools, the same dependence of optimal r on pool size holds, and larger models consistently achieve lower best-case loss despite earlier overfitting onset. We further verify these patterns on non-multilingual but domain setup (Wikipedia, peS2o, OpenWebMath mixed with generic English data), confirming that the repetition dynamics are a general property of mixture training under data constraints rather than an artifact of any particular domain or language pair.

Below we provide results of our experiments organized as follows:

- **Optimal repetition** (Section D): French, Swahili, and OpenWebMath all exhibit the same steady growth of optimal r with training budget, governed by data pool size rather than specific domain.
- **Loss curves** (Section E): Full loss-vs-tokens curves for all model sizes and languages confirm the overfitting at high h with small pools and monotonic improvement with large pools.
- **Cross-model behavior** (Section F): Larger models overfit faster yet achieve lower minima across all languages (confirming the pattern we observed for German).

- **Quality experiments** (Section G): The repetition–diversity trade-off extends to quality-filtered data at multiple data scales.
- **Comparison to unlimited data** (Section H): When target data is unconstrained, mixture training provides no benefit over single-domain training, confirming that our findings are specific to the data-constrained regime.

D Optimal Repetition Across Languages and Domains

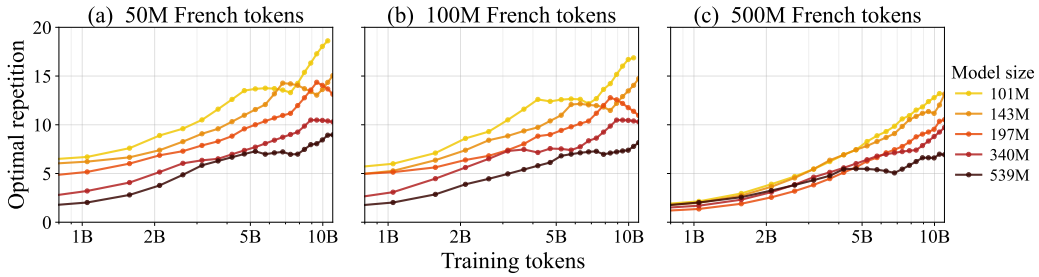


Figure 11: Optimal repetition as a function of training tokens for three data constraint sizes across different model sizes (French).

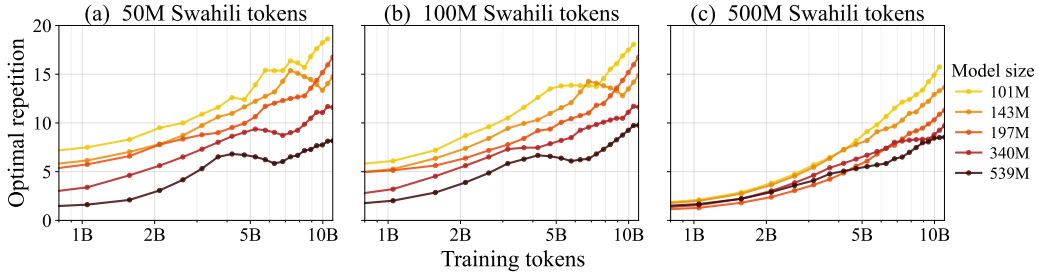


Figure 12: Optimal repetition as a function of training tokens for three data constraint sizes across different model sizes (Swahili).

Figures 11–12 show the optimal number of repetitions as a function of training tokens for French and Swahili, matching the German plot in the main text (Figure 4). The same steady growth of optimal r with training budget is observed across all languages.

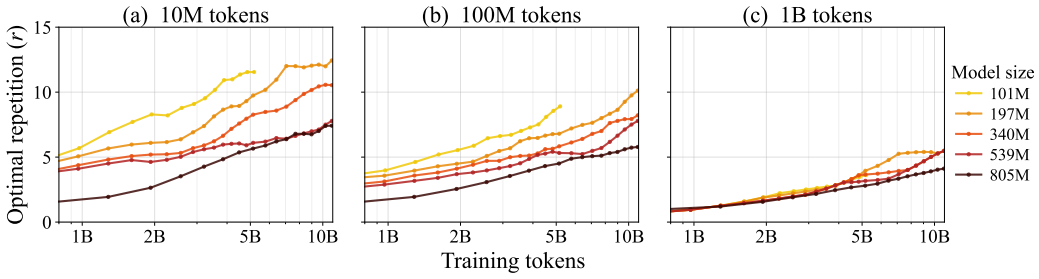


Figure 13: Optimal repetition as a function of training tokens for three data constraint sizes across different model sizes (OpenWebMath).

Figure 13 further extends this analysis to a non-language domain: OpenWebMath mixed with DCLM as the generic source. Despite the very different nature of mathematical text, the same qualitative pattern holds: optimal r increases with training tokens, decreases with pool size, and smaller models tolerate higher repetition (since they are trained on more tokens relative to their capacity). This confirms that the repetition dynamics are a general property of mixture training under data constraints, independent of the specific domain or language.

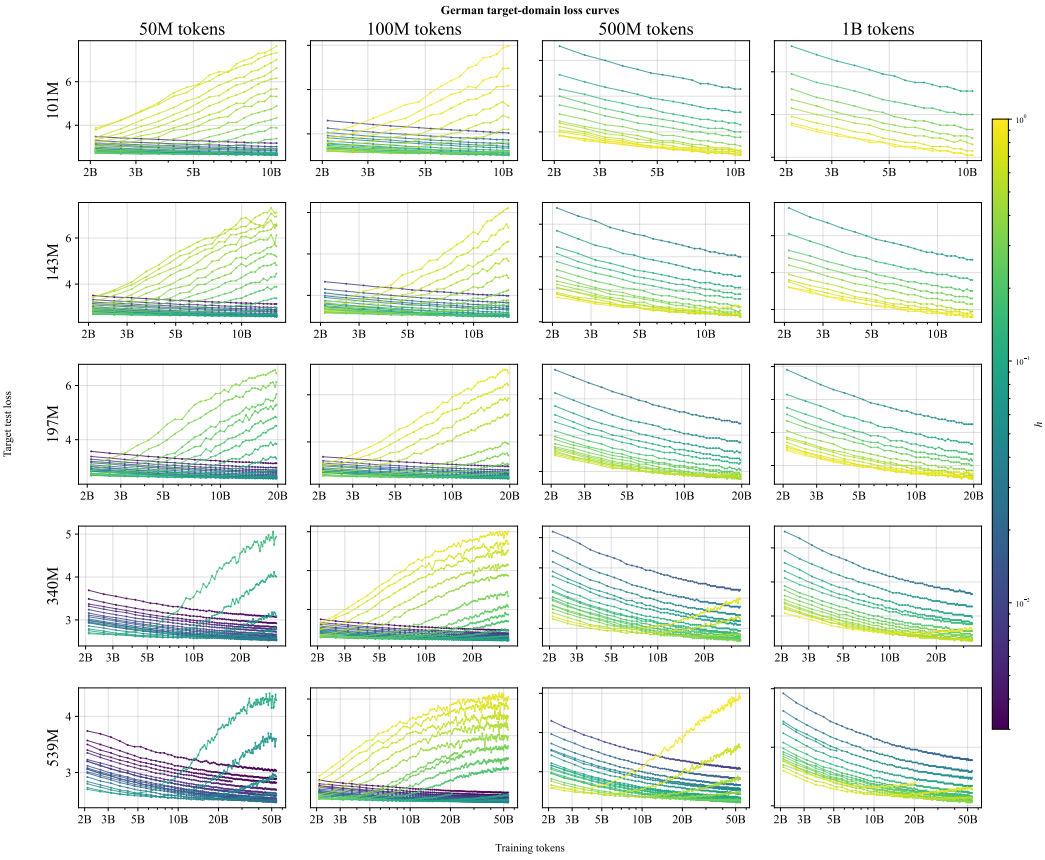


Figure 14: German target-domain loss curves across all model sizes (rows) and data pool sizes (columns).

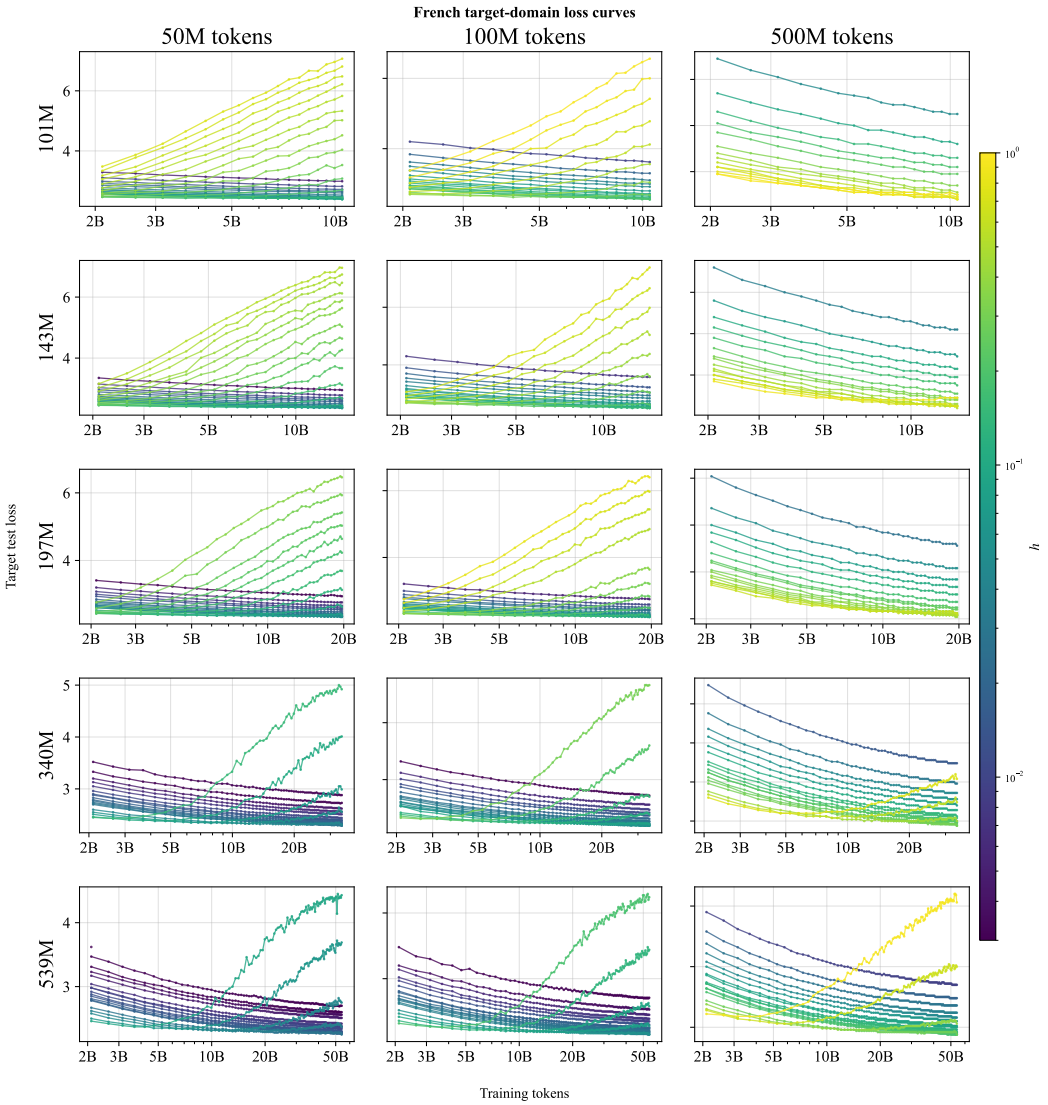


Figure 15: French target-domain loss curves across all model sizes (rows) and data pool sizes (columns).

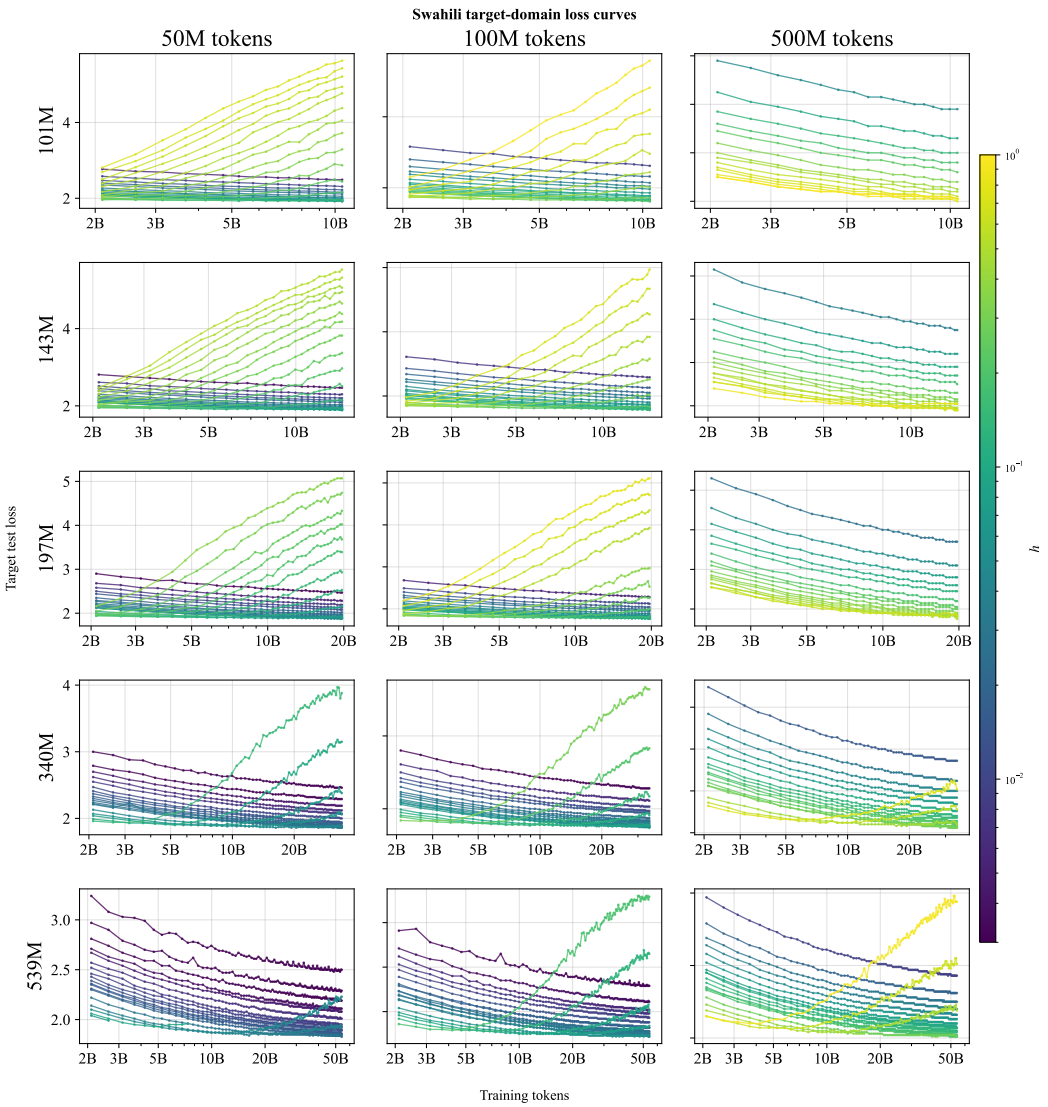


Figure 16: Swahili target-domain loss curves across all model sizes (rows) and data pool sizes (columns).

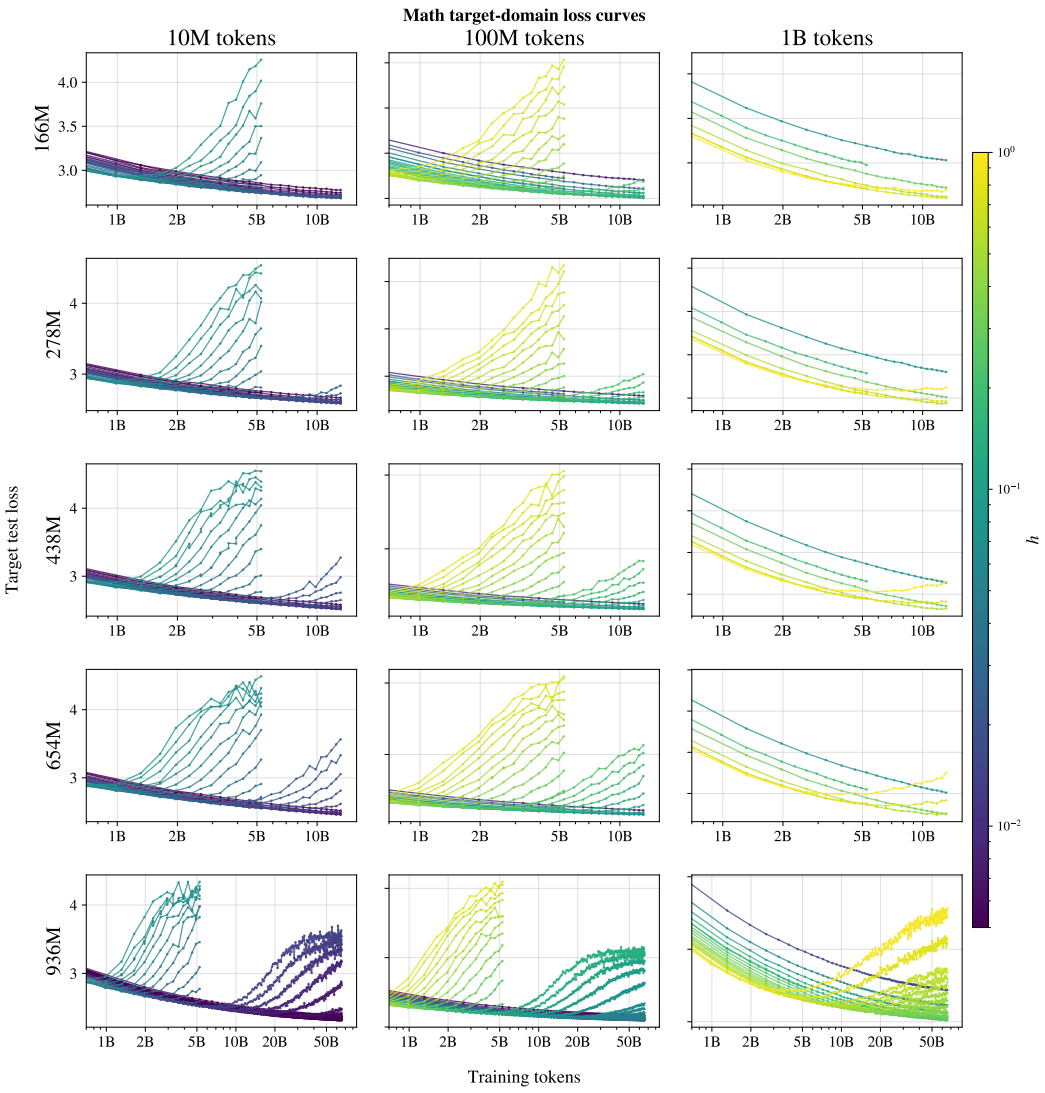


Figure 17: Math (OpenWebMath) target-domain loss curves across all model sizes (rows) and data pool sizes (columns). Legend reports total parameter counts including embeddings (166M, 278M, 438M, 654M, and 936M correspond to the 101M, 197M, 340M, 539M, and 805M non-embedding models in Table 3t).

E Loss Curves Across Languages, Model Sizes, and Domains

Figures 14–16 show target-domain validation loss as a function of training tokens for all model sizes (101M–539M) and all three target languages (German, French, Swahili). Each column corresponds to a different target data pool size; each row to a different model size; each curve within a panel corresponds to a different target weight h (color). The same qualitative pattern(i.e., overfitting learning curves at high h with small pools, while monotonic improvement with large pools) holds consistently across all configurations.

Figure 17 shows the same analysis for OpenWebMath, confirming that the loss curve dynamics extend beyond natural language to mathematical text.

F Cross-Model Loss Curves

Figure 5 shows that larger models overfit faster yet achieve lower best-case loss, demonstrated for German with 100M tokens at $h = 9\%$ and 20%. Here we extend this analysis to five target fractions (5–20%) and verify the pattern across all languages and domains. Figure 18 shows the full German panel, Figures 19–20 show French and Swahili, and Figure 21 shows the OpenWebMath domain. In all cases, the same ordering persists: larger models reach lower loss minima despite earlier overfitting onset, and the best-loss envelope (dashed) consistently favors larger models at every training budget. The pattern is particularly pronounced for the mathematics domain (Figure 21), where the separation between model sizes is larger and the overfitting onset for larger models is sharper.

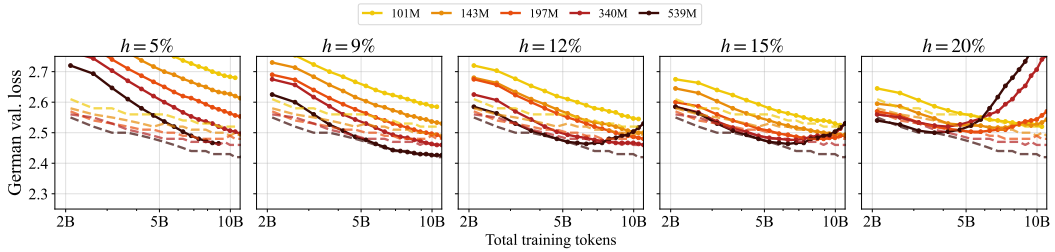


Figure 18: German target-domain loss across model sizes at five target fractions, for the 100M token pool. Dashed lines show the best-loss envelope.

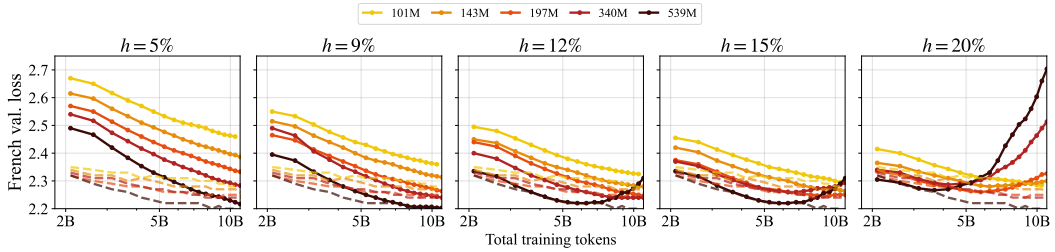


Figure 19: French target-domain loss across model sizes at five target fractions, for the 100M token pool. Dashed lines show the best-loss envelope.

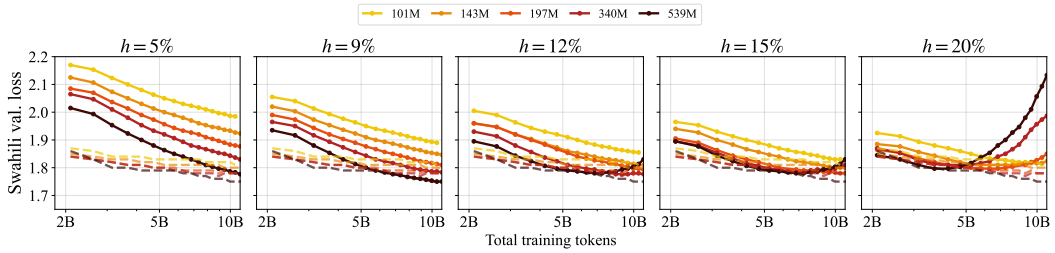


Figure 20: Swahili target-domain loss across model sizes at five target fractions, for the 100M token pool. Dashed lines show the best-loss envelope.

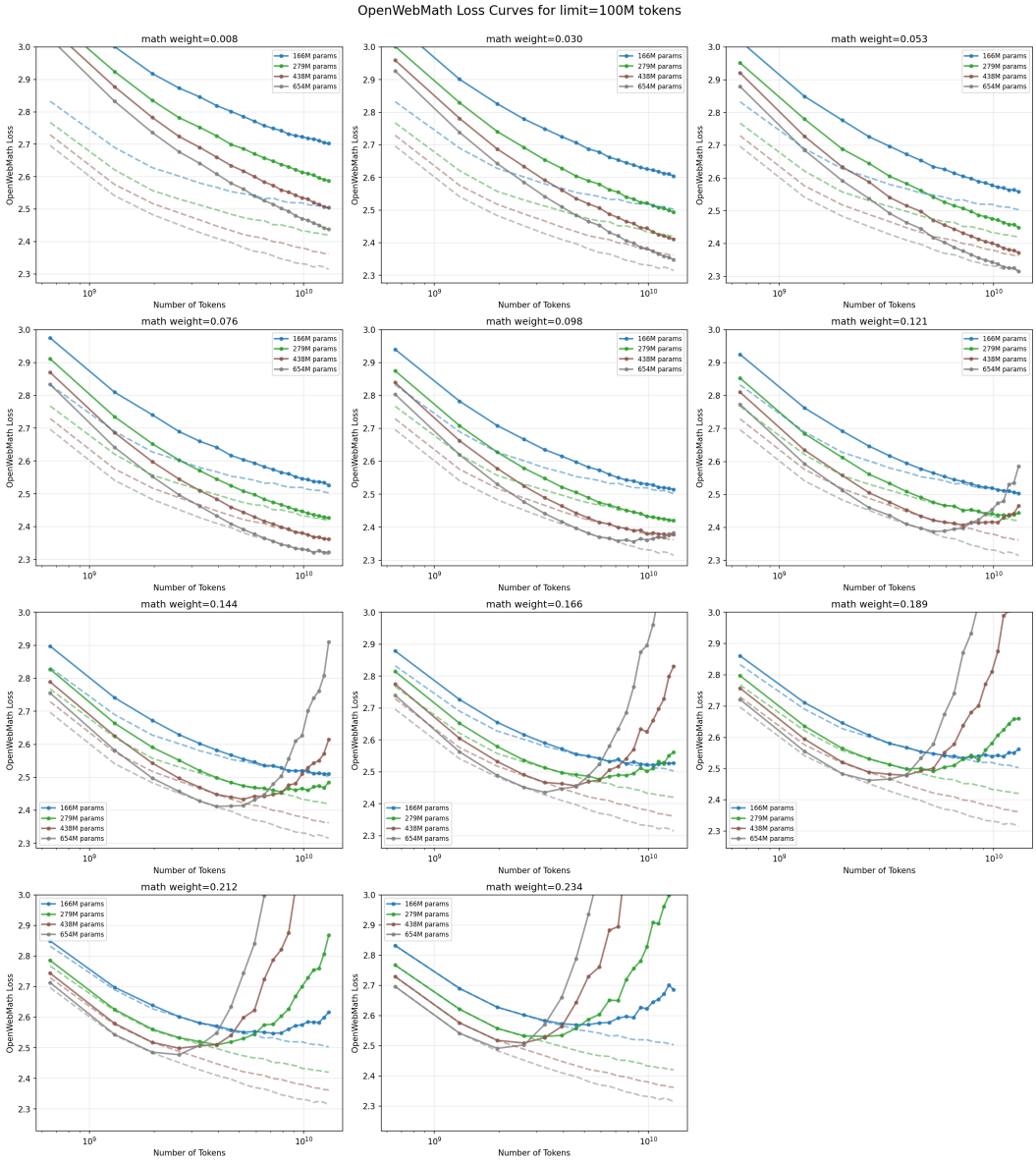


Figure 21: OpenWebMath target loss across model sizes at different mixture fractions h, for the 100M token pool. Dashed lines show the best-loss envelope across all h values. Legend reports total parameter counts including embeddings (166M, 279M, 438M, and 654M correspond to the 101M, 197M, 340M, and 539M non-embedding models in Table 3).

G Quality Experiments: Additional Target Data Scales and Overlapping Quality Pools

To ground the experiments, we first confirm that a data constrained fixed quality level follows the same pattern as multilingual data. Figure 22 shows Q99–100 loss curves at three pool sizes (26M, 132M, 264M tokens). At the smallest scale, only $h \leq 0.1$ avoids overfitting, and larger pools progressively unlock higher sustainable target weights. This establishes the baseline that for a fixed quality threshold, the repetition penalty imposes the same constraint on the target loss.

Section 4 further establishes that when training with a data-constrained high-quality domain, relaxing the quality filter to increase pool size consistently outperforms repeating the highest-quality slice starting from the 26M 99th percentile data pool. A crossover at 5B tokens shows the optimal

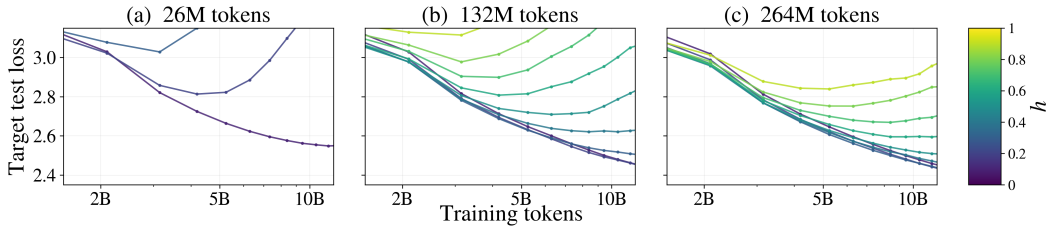
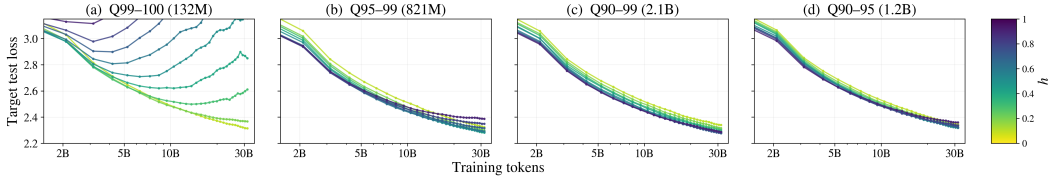


Figure 22: Loss vs. training tokens for Q99–100 at three data pool sizes. Each line corresponds to a target weight h. Smaller pools force lower weights to avoid overfitting from repetition.

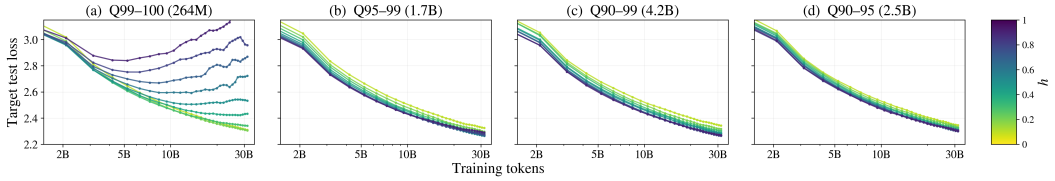
threshold shifts from Q95–99 to Q90–99 as training progresses. Here we present an expanded experimental suite, including additional data scales that support and extend these findings. We conduct three families of experiments, all using 101M-parameter models trained on a two-domain mixture of high-quality (HQ) filtered text and unlimited general DCLM. Each experiment sweeps target weight $h \in \{0.1, 0.2, \ldots, 1.0\}$.

Non-overlapping quality buckets at multiple data scales We define non-overlapping quality bands (Q90–99, Q95–99, Q99–100) and train at three data scales:

- **1**×: pools range from 26M to 982M tokens including severe data constraint, and high repetition especially for high quality data.
- **5**×: pools range from 132M to 2.1B tokens. This imposes a moderate constraint where the highest quality buckets have high repetition counts, but lower quality buckets do not.
- **10**×: pools range from 264M to 4.2B tokens. Most quality buckets avoid significant repetition, and only the highest quality bucket still overfits with too much repetition.



(a) Pool sizes: Q99–100 (132M), Q95–99 (821M), Q90–99 (2.1B), Q90–95 (1.2B).

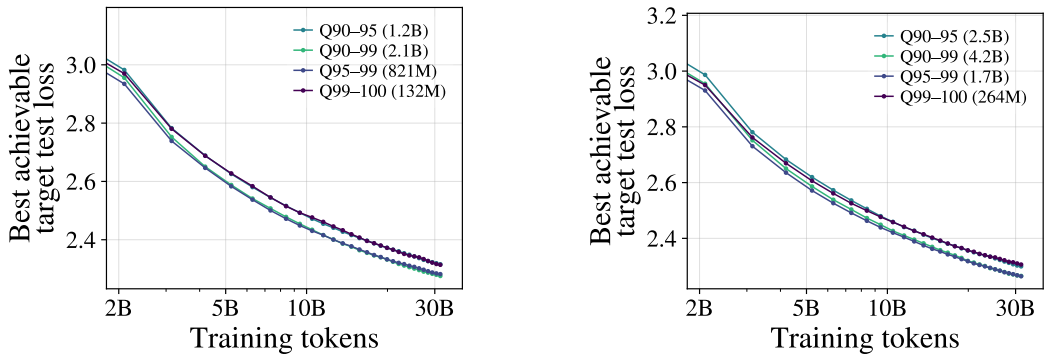


(b) Pool sizes: Q99–100 (264M), Q95–99 (1.7B), Q90–99 (4.2B), Q90–95 (2.5B).

Figure 23: Loss curves for four quality bands at increasing data scales. Each line corresponds to a mixture weight h. Broader quality filters enable higher mixture weights without overfitting, with the effect diminishing as pool size increases.

These experiments disentangle quality from quantity by ensuring no data overlap between bands, and reveal how the quality–quantity tradeoff evolves with different data constrained data scales. Scaling up from the 26M target data pool to 5× and 10× the pool size confirms the main findings and reveals how the quality–quantity tradeoff evolves with data availability. At 5×, Q95–99 dominates for the first ∼14B tokens before Q90–99 overtakes it (Figure 23a). The same crossover is observed at 1× but shifted later due to the larger pool delaying saturation. At 10×, Q95–99 remains the best band throughout 30B tokens of training (Figure 23b): with data pools exceeding 1.7B tokens, no band experiences meaningful repetition at reasonable values of h, and higher per-token quality is the dominant factor. This demonstrates that the quality–quantity tradeoff is scale-dependent: at

small scales, breadth wins; at large scales, quality wins; and the crossover token count increases predictably with pool size. We include results with Q90-95 as a control to show that simply increasing the amount of data but decreasing quality performs worse. The best-loss per quality over all h is shown in Figure 24, and confirms this quantitatively: at 5× starting from 132M target data size, a crossover from Q95–99 to Q90–99 occurs at ∼14B tokens (later than the ∼5B crossover at 1×), while at 10× Q95–99 leads throughout. As pool size grows, the quality–quantity tradeoff vanishes and per-token quality is most important.



(a) 5× data scale.

(b) 10× data scale.

Figure 24: Best achievable target loss by quality band at (a) 5× and (b) 10× target data scales. Unlike the 1× setting (Figure 8), at larger pool sizes the curves largely overlap, confirming that the quality-quantity tradeoff diminishes when pools are large enough to avoid severe repetition.

Experiments with overlapping buckets We use the full untruncated pool at each quality threshold (Q80–100: 5B, Q90–100: 2B, Q95–100: 954M, Q99–100: 132M) and train for 10B tokens. In this experiment, the quality bands are (e.g., all contain the Q99-100 data), representing the practical approach of setting a single lower-bound quality threshold. The results are consistent. The 95th percentile, Q95–100, achieves the best loss consistent with the Q95-99 results for up to 10B tokens in Figure 25.

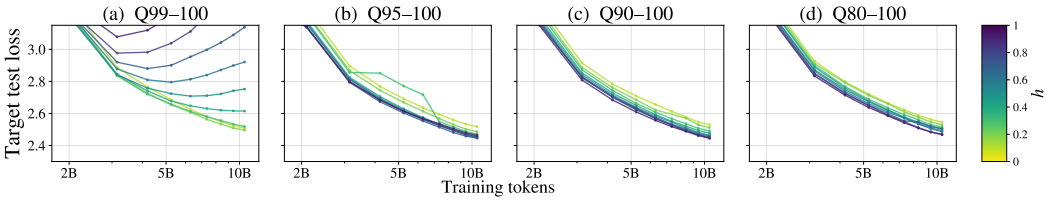


Figure 25: Loss curves for inclusive quality thresholds (each band includes all data above the given percentile). Pool sizes increase from left to right as broader thresholds include more data.

Equal-size data constrained size experiments We truncate each quality band (Q80–100, Q90–100, Q95–100, Q99–100) to approximately the same pool size (∼132–139M tokens) and train for 10B tokens. This isolates the effect of per-token quality by controlling for pool size as all bands have approximately the same number of repetitions. Figure 26 shows results when the data constrained pools are equal size. With the impact of repetition equalized across quality, the ordering is strict: Q99–100 > Q95–100 > Q90–100 > Q80–100 at all training steps.

Quality experiments at 1.2B models To test whether the quality-filtering conclusions hold at larger scale, we train 1.2B-parameter models on the quality mixtures for 30B tokens across the full target-weight grid, and evaluate them on eight benchmarks. We also add a broader tier, Q50–99, to the three used in the main text.

Table 5: Quality mixtures at 1.2B parameters and 30B training tokens. **Top:** target loss. **Bottom:** average accuracy over eight benchmarks (ARC-Challenge, ARC-Easy, BoolQ, HellaSwag, MMLU, PIQA, SciQ, WinoGrande).

	Target weight h							
Quality tier	0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0
Target loss								
Q50–99 (2B)	**1.87**	1.88	**1.88**	**1.89**	**1.90**	**1.91**	**1.96**	**2.05**
Q80–99 (982M)	**1.87**	**1.87**	**1.88**	1.92	1.98	2.14	2.37	2.94
Q90–99 (412M)	**1.87**	1.91	2.10	2.28	2.49	2.82	3.61	6.07
Q99–100 (26M)	2.44	2.35	2.36	2.36	2.40	2.45	2.70	12.5
Downstream accuracy (%)								
Q50–99	**55.5**	**55.3**	**54.6**	**54.8**	**54.3**	**53.8**	**53.1**	**52.4**
Q80–99	55.4	55.0	**54.6**	53.5	**54.3**	51.8	49.9	47.7
Q90–99	54.3	53.6	52.2	50.3	48.7	47.5	45.9	43.9
Q99–100	48.9	47.8	47.4	45.3	46.0	44.8	43.8	39.3

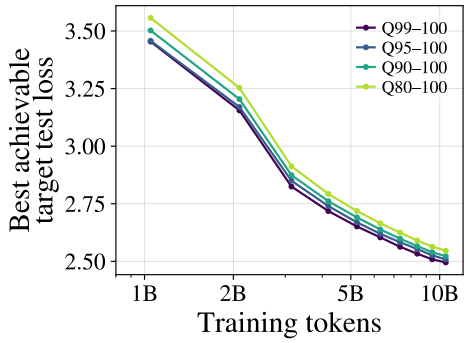


Figure 26: Best achievable target loss with all quality bands truncated to the same pool size. With repetition equalized, higher per-token quality consistently achieves lower loss.

The results of the 1.2B models align with those reported in Section 4, both in loss and in downstream accuracy averaged over the eight benchmarks (ARC-Challenge (Clark et al., 2018), ARC-Easy (Clark et al., 2018), BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), MMLU (Hendrycks et al., 2021), PIQA (Bisk et al., 2020), SciQ (Welbl et al., 2017), WinoGrande (Sakaguchi et al., 2020); see Table 5) and broken down per benchmark (Figure 27). First, broadening the filter beats a narrow slice, which is reflected in both loss and downstream: broader pools outperform narrower ones at every target weight, and their advantage grows with h. The loss gap between the broadest and narrowest tier grows from 0.57 at $h = 0.1$ to 10.5 at $h = 1.0$, and the downstream gap from 6.6 to 13.1 accuracy points. The narrowest tier of the highest quality, Q99–100, is the weakest in both metrics at every weight tested. Second, raising the target weight degrades performance: $h = 0.1$ is optimal downstream for all four tiers, and at $h = 1.0$ each tier loses between 3.1 and 10.4 points relative to its own optimum. Per benchmark, the tier ordering is clearest on ARC-Easy, HellaSwag, MMLU, PIQA and SciQ, where broader pools lead at every weight; on ARC-Challenge, BoolQ and WinoGrande the curves are noisier at this scale.

Final remarks When pool size is equalized, quality improves validation loss. When pools vary naturally, the repetition penalty from small high-quality pools outweighs the quality advantage. The non-overlapping band experiments isolate the mechanism: it is the data in broader bands (Q90–99 beyond Q95–99) that helps at larger training budgets, not merely reduced repetition of the top slice. The scale dependence of the crossover has a practical implication that for any given training budget and data pool, there exists an optimal quality threshold that balances per-token quality against repetition. Our scaling law captures this tradeoff enabling practitioners to extrapolate which data pool to use and for how many repetitions.

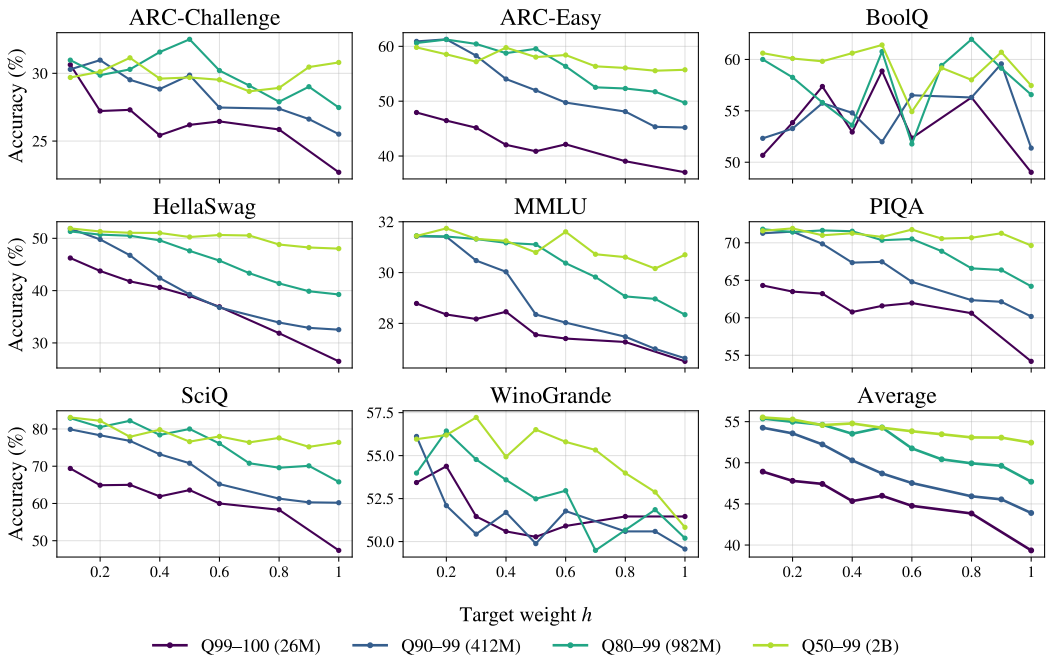


Figure 27: Downstream accuracy against target weight h for the 1.2B quality-mix models at 30B training tokens, per benchmark and averaged. We report length-normalised accuracy where the benchmark provides it and plain accuracy otherwise (BoolQ, WinoGrande).

H Comparison to Unlimited Target Data

To establish an upper bound on target-domain performance, we train all model sizes with a $\sim$60B German token pool – large enough that repetition never occurs. Figure 28 shows target loss for each mixture fraction h in this unlimited setting. Without data constraints, higher target fractions monotonically achieve lower target loss at every training budget: there is no overfitting, no U-shaped behavior, and no quality–quantity tradeoff.

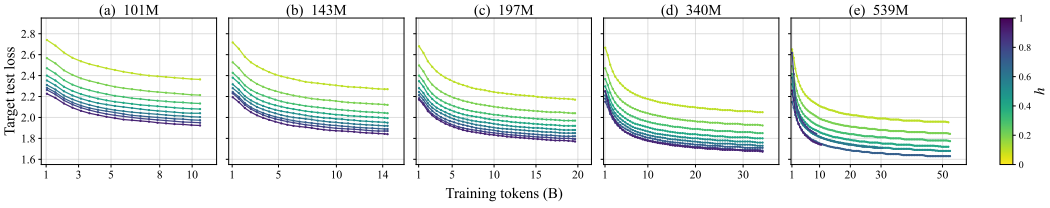


Figure 28: German target loss with unlimited data ($\sim$60B pool, $r < 1$ throughout) for all mixture fractions h. Without data constraints, higher target fractions always achieve lower target loss, and no overfitting occurs.

Figure 29 then compares the best achievable loss under constrained pools (50M–1B) against this unlimited baseline. The gap between each constrained curve and the unlimited line quantifies the cost of the data constraint. Constrained pools plateau as training progresses while the unlimited baseline continues to improve, confirming that the performance ceiling is imposed by data exhaustion rather than model capacity. The gap widens with training budget, underscoring the importance of optimal mixture design when target data is scarce.

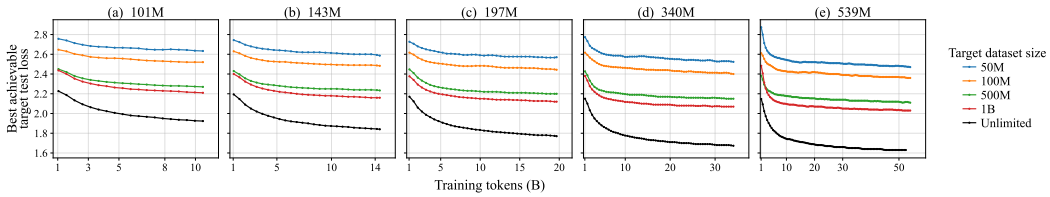


Figure 29: Best achievable German target loss vs. training tokens for each model size, comparing constrained data pools against an unlimited (∼60B) baseline where target data is never repeated.

I Analysis of Three-Domain Experiments

Section 7 presents results under proportional weighting; here we report the complementary equal-weighting experiments ($h_{\text{wiki}} = h_{\text{peS2o}}$).

Figure 30 shows optimal repetition under equal weighting with the 10% compute confidence band. The core pattern from the proportional case carries over: optimal r grows steadily with training budget, and smaller data pools require substantially higher repetition before the optimum is reached. Under equal weighting, the smaller domain (Wiki) is repeated more aggressively than under proportional weighting, as equal weights allocate the same weight to both domains regardless of pool size, and the smaller pool exhausts its unique tokens earlier. For example, at 10B training tokens in the base configuration (Wiki 100M, peS2o 500M), equal weighting reaches $r \approx 14$ while proportional weighting stays at $r \approx 6$ (measured with respect to the Wiki pool). The confidence bands remain wide, confirming that approximate repetition estimates suffice for near-optimal performance under either strategy.

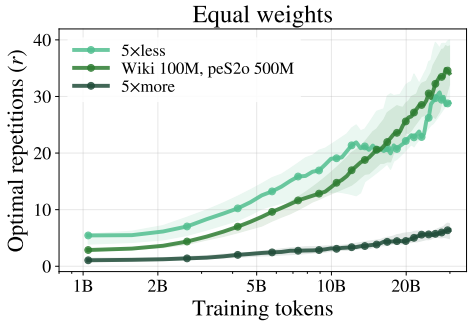


Figure 30: Optimal repetitions under equal weighting with 10% compute confidence band (shaded) in the three-domain setup (101M model).

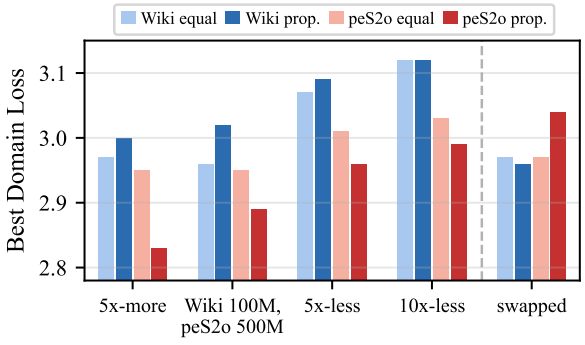


Figure 31: Per-domain best loss under equal vs. proportional weighting across five pool-size configurations.

Additionally, we examined how the choice of weighting strategy affects each target domain individually. Beyond the configurations reported in the main text, we include a 10×less experiment

(Wiki 10M, peS2o 50M) to test both strategies under even more severe data constraints. Figure 31 breaks down the per-domain loss under both strategies across all five configurations. Proportional weighting consistently favors the larger pool (peS2o), achieving lower peS2o loss at the expense of Wiki. Equal weighting reverses this tradeoff: it assigns more relative weight to the smaller domain, improving Wiki loss but sacrificing peS2o performance. The effect is most pronounced when pool sizes differ substantially: in the 5×more configuration, proportional weighting achieves 0.12 lower peS2o loss than equal weighting, while equal weighting gains only 0.03 on Wiki. The *swapped* configuration (where Wiki becomes the larger pool) confirms this pattern: when pool sizes are reversed, proportional weighting now favors Wiki instead.

Figure 32 shows the best achievable per-domain loss throughout training for all four pool-size configurations.

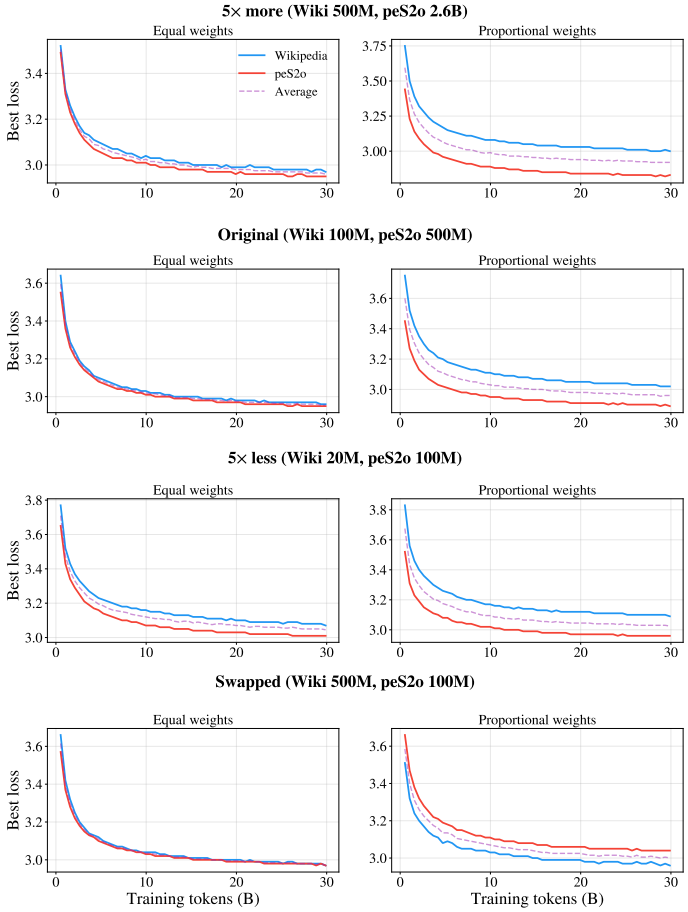


Figure 32: Best achievable per-domain loss throughout training under equal (left) and proportional (right) weighting for four pool-size configurations. Blue: Wikipedia; red: peS2o; dashed purple: average.

J Baseline Scaling Law Formulas

We describe the baseline formulas used in Section 6. All formulas predict the target-domain loss as a function of the total tokens D_{total}, the target weight h, the number of repetitions $r = h\,D_{\text{total}}/D_{\text{target}}$, and optionally the model size N.

Repetition-agnostic (Shukor et al., 2025). This baseline treats all tokens as unique regardless of repetition:

$$L = E + \frac{A}{D_{\text{eff}}^{\alpha}} + \gamma\, h\,, \qquad D_{\text{eff}} = (1-h)\,D_{\text{total}} + \tau\, h\, D_{\text{total}}\,. \tag{7}$$

The effective data is a weighted sum of generic and target tokens with no repetition discounting—repeated target tokens are valued the same as fresh ones. Parameters: E, A, α, τ, γ (5 parameters).

Domain-agnostic (Muennighoff et al., 2023). This baseline uses a single saturating function of total tokens without distinguishing domains:

$$L = E + A\, D_{\text{eff}}^{\alpha}, \qquad D_{\text{eff}} = C\big(1 - e^{-\mu\, R}\big), \tag{8}$$

where $C = (1-h)\, D_{\text{total}} + h\, D_{\text{total}}/r$ is the total number of unique tokens and $R = D_{\text{total}}/C$ is the overall repetition ratio. Parameters: E, A, α (< 0), μ (4 parameters).

Utility decay (Goyal et al., 2024). This baseline models repetition through an exponentially decaying exponent:

$$L = E + a\, D_{\text{total}}^{\,b_{\text{eff}}}, \qquad b_{\text{eff}} = (1-h)\, b_0 + h\, b_1\, \delta^{r-1}, \tag{9}$$

where $\delta = 0.5^{1/\tau}$ is a half-life parameter. As repetitions increase, the exponent on the target-domain contribution decays toward zero, reducing the data-scaling benefit. Parameters: E, a, b_0, b_1, τ (5 parameters).

Multi-size extensions. Each baseline extends to variable model size by adding a capacity term C/N^{β} (and for the repetition-agnostic and domain-agnostic formulas, a data-size coupling N^{δ}), following the same pattern as L_{size} (Eq. 4).

K Extended Scaling Law Results

This section reports the full results underlying Section 6, including the train/test breakdown for the loss prediction tables, the per-dataset optimal-mixture prediction errors, and the size-extrapolation wasted-token metric.

Loss prediction: train and test wR^2 (fixed-size formula). Table 6 reports train and test wR^2 on all four experimental setups for the fixed-size formula L_{fix} and the three baselines. Train corresponds to the first 50% of training steps, test to the held-out second half.

Table 6: Train and test weighted R^2 (wR^2) for the fixed-size formula L_{fix}, fitted independently per model size (or per quality level / domain). Each observation is weighted by $\max(r \cdot h, \epsilon)$ to emphasise the high-repetition regime. Train: first 50% of steps; test: second half.

	German		Maths		Quality		Wiki/peS2o	
	Train	Test	Train	Test	Train	Test	Train	Test
L_{fix}	**0.85**	**0.95**	**0.99**	**0.88**	**0.91**	**0.71**	0.68	**0.80**
Rep-agnostic	0.81	0.78	0.95	0.78	0.86	0.14	0.67	0.72
Utility decay	0.52	0.72	0.77	0.55	0.94	−0.64	**0.69**	0.79
Domain-agnostic	0.03	−40.7	0.10	−0.49	0.46	−2.19	0.42	−1.16

Loss prediction: train and test wR^2 (size extrapolation). Table 7 reports the same metric for the multi-size formula L_{size}, fitted on all model sizes except the largest and evaluated on the held-out largest model.

Optimal mixture prediction error. For each experiment (a specific model size and target dataset size) and each held-out training checkpoint D_{total}, we compare the predicted h^* to the empirically best weight h^*_{emp} at that checkpoint. Table 8 reports the median of $|\log_{10}(h^*_{\text{pred}}) - \log_{10}(h^*_{\text{emp}})|$ per dataset over all such pairs on held-out test steps. A value of 0.1 corresponds to a $\sim 1.26\times$ multiplicative error in the predicted weight.

Necessity of the weight penalty γh. In order to directly test the necessity of the γh term, we refit L_{fix} with γ fixed to zero. Held-out wR^2 drops in every setup (Table 9), and the optimal mixture prediction error on German (Table 8) triples, from 0.067 to 0.198.

Table 7: Size extrapolation: weighted R^2 (wR^2) for the multi-size formula L_{size}, fitted on smaller model sizes and evaluated on the held-out largest model (539M for German, 805M for Maths). Train: wR^2 on fitting sizes; Test: wR^2 on held-out size.

	German (539M)		Maths (805M)	
	Train	Test	Train	Test
L_{size}	**0.95**	**0.65**	**0.97**	**0.73**
Rep-agnostic+N	0.88	0.59	0.94	0.71
Utility decay+N	0.76	0.56	0.85	0.69
Domain-agnostic+N	0.25	−0.23	0.46	−0.77

Table 8: Optimal mixture prediction on held-out test steps: median $|\log_{10}(h^*_{\text{pred}}) - \log_{10}(h^*_{\text{emp}})|$ (lower is better). A value of 0.1 corresponds to a $\sim$1.26$\times$ error in the predicted fraction.

	Ger.	Math	Qual.	Wiki	Avg
L_{fix} (ours)	**0.07**	**0.13**	0.47	**0.10**	**0.19**
Rep-agnostic	0.60	1.01	0.61	0.50	0.68
Utility decay	0.10	0.20	**0.38**	0.12	0.20
Domain-agn.	2.57	1.99	5.30	3.07	3.23
L_{size} (ours)	**0.15**	0.18	—	—	0.17
Rep-agn.+N	0.73	1.44	—	—	1.09
Util. dec.+N	**0.15**	**0.17**	—	—	**0.16**
Dom.-agn.+N	3.27	1.56	—	—	2.42

How much fitting data is needed. Our main results fit L_{fix} on the first 50% of training steps and test on the second half. To verify how much data is actually needed to obtain a reliable fit, we halve it to the first 25% of steps, keeping the evaluation unchanged, i.e., always the same held-out second half. Table 10 reports the effect across all four setups: halving the fitting data preserves the mixture recommendation on German, Maths and Wiki/peS2o while degrading the predicted loss value (mildly on German and substantially on the other datasets). We then additionally repeat the fit at the first 12.5%, 25% and 37.5% of steps on German, and run the same sweep for the three baselines. Figure 33 shows the result on German, with exact values in Table 11. Reducing the fitting data degrades every formula, but our L_{fix} is the best of the four at every fraction on both metrics and the only one with positive held-out wR^2 throughout, despite having more parameters than each baseline. The fitting-data requirement is therefore a property of the setups rather than of our parametrisation.

Table 10: Reducing the fitting data from the first 50% of training steps to the first 25%.

	Test wR^2			
Fit on	Ger.	Math	Qual.	Wiki
first 25%	0.861	0.415	−9.88	−0.078
first 50%	0.951	0.875	0.708	0.797
	Optimal-mixture error			
first 25%	0.081	0.134	1.338	0.082
first 50%	0.067	0.126	0.474	0.099

Size extrapolation: wasted tokens. A stronger test of the multi-size formula is whether it can predict the behaviour of a model size never seen during fitting. We fit L_{size} and all multi-size baselines on all model sizes *except the largest* and evaluate exclusively on the held-out largest model (539M for German, 805M for Maths). Table 12 reports the weighted median wasted token fraction on the held-out largest model for each setup. L_{size} achieves a test wR^2 of 0.65 on German and 0.73 on Maths (Table 7), demonstrating that the capacity and data-size coupling terms extrapolate mean-

Table 9: wR^2 for L_{fix} with and without the weight penalty.

	Ger.	Math	Qual.	Wiki
L_{fix}	**0.951**	**0.875**	**0.708**	**0.797**
$\gamma = 0$	0.941	0.808	0.497	0.789

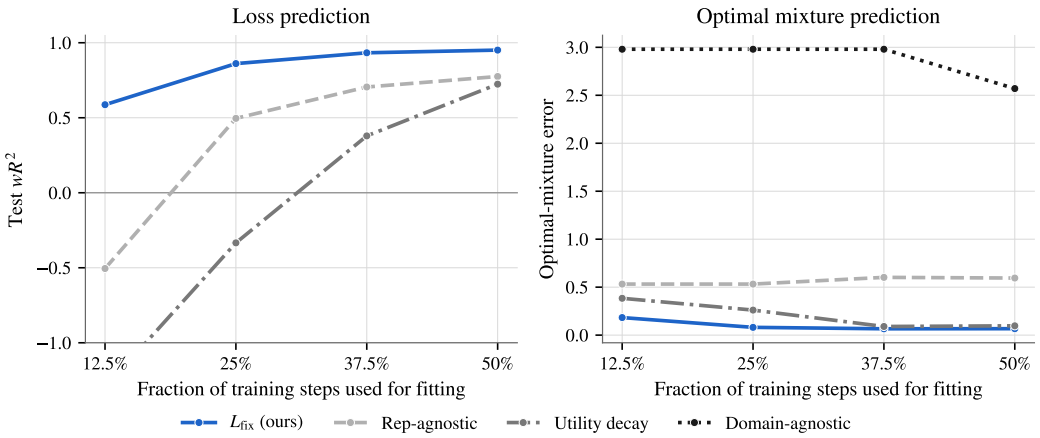


Figure 33: Fitting-data requirement on German. **Left:** held-out wR^2 (higher is better) as a function of the fraction of training steps used for fitting. Domain-agnostic is off-scale at every fraction. **Right:** optimal-mixture error (lower is better). Only L_{fix} remains predictive at the smallest fitting fractions.

Table 11: Fitting-data requirement on German: held-out wR^2 (higher is better) and optimal-mixture error (lower is better) as a function of the fraction of training steps used for fitting.

	Test wR^2			
Fit on	L_{fix} (ours)	Rep-agnostic	Utility decay	Domain-agnostic
first 12.5%	**0.587**	−0.505	-8.3×10^5	−153
first 25%	**0.861**	0.496	−0.334	−2.23
first 37.5%	**0.933**	0.705	0.379	−2.96
first 50%	**0.951**	0.775	0.724	−40.7
	Optimal-mixture error			
first 12.5%	**0.183**	0.532	0.384	2.98
first 25%	**0.081**	0.532	0.261	2.98
first 37.5%	**0.066**	0.602	0.090	2.98
first 50%	**0.067**	0.595	0.097	2.57

ingfully to larger scales. The wasted token metric is even more discriminating: most baselines fail catastrophically on German, producing predictions outside the empirical range (counted as 100%). L_{size} wastes 58% on German and 24% on Maths, the only formula that avoids catastrophic failure on both. Results for French and Swahili are reported in Appendix K.

Table 12: Size extrapolation: weighted median wasted token fraction (lower is better). Predictions outside the empirical weight range are counted as 100%.

	German (539M)	Maths (805M)
L_{size}	**58%**	**24%**
Rep-agnostic+N	100%	97%
Utility decay+N	100%	22%
Domain-agnostic+N	100%	84%

How many model sizes are needed. L_{size} is fitted on all model sizes except the largest. Table 13 shortens the ladder to the three smallest sizes (101M, 143M, 197M), evaluating on the same held-out 539M model. Test wR^2 drops only slightly on all three languages, so three model sizes still support extrapolation to an unseen larger one. Fewer is not possible: a power law in N is not identifiable from two sizes.

Table 13: Shortening the fitting ladder for L_{size}: test wR^2 on the held-out 539M model when fitting on the three smallest model sizes instead of four.

Fit on	German	French	Swahili
$k = 3$ (101M, 143M, 197M)	0.59	0.88	0.64
$k = 4$	0.65	0.93	0.69

Limitations. The scaling law is descriptive: it summarises observed training dynamics but does not prescribe an optimal allocation from first principles. Size extrapolation is encouraging on both German ($wR^2 = 0.65$ at 539M) and Maths ($wR^2 = 0.73$ at 805M), though the wasted-token metric reveals that predictions remain imperfect at unseen scales. The quality-filtering setup is harder to predict than language/domain splits (test $wR^2 = 0.71$, log-error $= 0.47$), likely because quality tiers interact with the model in ways not fully captured by the effective data abstraction. We mix target and generic data uniformly at the sample level and do not claim this is the optimal placement: uniform mixing is where mixture weight and repetition separate cleanly, and every comparison we report holds it fixed, so our findings characterise this regime specifically. Staged alternatives (for example, placing target data in a mid-training phase, in a later pretraining phase, or during learning-rate annealing) are left to future work. Finally, our experiments span $\sim$100M–900M parameters; whether the fitted exponents extrapolate to multi-billion-parameter models remains an open question.

Size Extrapolation: French and Swahili We repeat the size-extrapolation experiment of Section 6 on French and Swahili. For each language, we fit the multi-size formulas on model sizes 101M–340M and evaluate on the held-out 539M model. Table 14 reports weighted R^2 (test / train).

On French, L_{size} achieves a test wR^2 of 0.93, even exceeding its train wR^2 of 0.92, which indicates that the 539M model's behavior is well predicted by trends from smaller scales. On Swahili, extrapolation is harder: L_{size} achieves 0.69 on the held-out 539M model, while the utility decay baseline reaches 0.71. The domain-agnostic baseline yields near-zero or negative wR^2 on both languages.

For wasted tokens, all formulas (including L_{size}) produce predictions outside the empirical weight range for French and Swahili at 539M, resulting in $\geq 94\%$ waste. This reflects the narrower range of target fractions tested at this scale for these languages rather than a fundamental failure of the formula.

Table 14: Size extrapolation on French and Swahili: weighted R^2 (test / train) when fitting on 101M–340M and evaluating on 539M.

	French (539M)	Swahili (539M)
L_{size} (9p)	**0.93 / 0.92**	0.69 / **0.92**
Rep-agnostic+N (8p)	0.87 / 0.90	0.65 / 0.90
Utility decay+N (7p)	0.69 / 0.62	**0.71** / 0.69
Domain-agnostic+N (7p)	-0.54 / 0.08	-0.00 / 0.02

L Robustness to Weighting Scheme

We verify that the qualitative ranking of formulas is robust to the choice of observation weights in the wR^2 metric. Table 15 reports test wR^2 on the German dataset under four weighting schemes: $\max(r \cdot h, \epsilon)$ (our default), r only, h only, and uniform (standard R^2). The ranking $L_{\text{fix}} >$ rep-agnostic $>$ utility decay $>$ domain-agnostic is preserved across all schemes, confirming that the advantage of our formula is not an artifact of the weighting.

M Downstream Benchmark Evaluation

We evaluate downstream task performance for the best-loss configurations from Figure 7 (one checkpoint per model size and data pool combination, selected at the training step that minimizes German

Table 15: Test wR^2 on German under different weighting schemes. The formula ranking is stable across all choices.

	$r \cdot h$ (default)	r	h	Uniform
L_{fix} (6p)	**0.95**	**0.94**	**0.74**	**0.88**
Rep-agnostic (5p)	0.78	0.84	0.45	0.84
Utility decay (5p)	0.72	0.84	0.36	0.72
Domain-agnostic (4p)	<-1	<-1	<-1	<-1

test loss). We use seven benchmarks covering reasoning, commonsense, and language understanding: ARC-Challenge and ARC-Easy (Clark et al., 2018) (science question answering), HellaSwag (Zellers et al., 2019) (commonsense sentence completion), LAMBADA (Paperno et al., 2016) (long-range word prediction), PIQA (Bisk et al., 2020) (physical intuition reasoning), SciQ (Welbl et al., 2017) (science multiple-choice), and WinoGrande (Sakaguchi et al., 2020) (coreference resolution). Each benchmark is evaluated in both English (original) and German (translated), allowing us to assess both target-domain and non-target-domain capabilities from the same checkpoint.

Tables 16 and 17 report accuracy on all benchmarks, and Figure 34 further visualize it. German benchmark performance improves consistently with both model size and target data pool size, confirming that the perplexity gains translate to downstream improvements. English performance decreases as more training budget is allocated to German (higher h), reflecting the expected trade-off between target and non-target domains. Larger models partially mitigate this cost: the 539M model retains 48.6% English average even at the highest German allocation, compared to 39.7% for 101M.

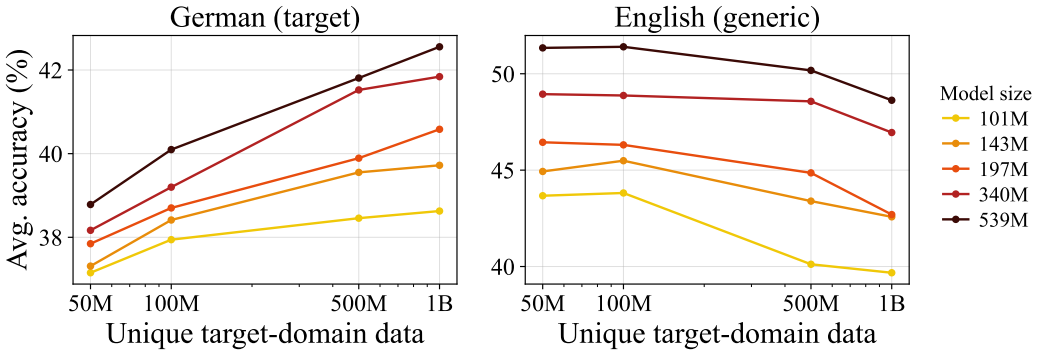


Figure 34: Downstream benchmark accuracy averaged over seven tasks at the best target-loss checkpoint, as a function of the unique target data available. German accuracy improves with both target pool size and model size, mirroring the loss results, while English accuracy declines as more of the budget is spent on the target domain. Per-benchmark results are in Appendix M.

Table 16: German downstream benchmark accuracy (%) at the best target-loss checkpoint. More target data consistently improves German performance across all model sizes.

Model	Target tokens	ARC-C	ARC-E	HSwag	Lambada	PIQA	SciQ	Wino	**Avg**
101M	50M	24.0	31.0	28.0	13.7	52.1	60.4	50.8	37.2
101M	100M	24.5	31.2	28.4	15.2	53.0	63.2	50.2	37.9
101M	500M	23.4	32.6	29.8	17.6	55.3	60.1	50.3	38.5
101M	1B	25.7	30.9	29.8	19.1	57.0	59.1	48.9	38.6
143M	50M	23.2	29.9	28.6	15.3	52.4	61.7	50.1	37.3
143M	100M	23.7	31.9	28.7	16.3	54.4	62.3	51.5	38.4
143M	500M	23.7	33.5	29.8	19.3	56.5	63.2	50.8	39.6
143M	1B	24.3	33.1	30.5	20.7	56.0	62.8	50.7	39.7
197M	50M	24.1	31.0	28.9	15.4	53.7	62.1	49.7	37.8
197M	100M	24.2	32.7	29.6	16.9	53.9	63.2	50.5	38.7
197M	500M	24.1	32.3	31.2	20.0	57.6	62.7	51.3	39.9
197M	1B	24.2	35.3	31.2	20.8	57.5	62.2	52.9	40.6
340M	50M	24.2	31.2	29.2	15.7	52.8	62.6	51.4	38.2
340M	100M	25.0	33.0	30.4	18.4	53.7	64.2	49.7	39.2
340M	500M	24.8	33.8	32.2	24.1	57.6	65.3	53.0	41.5
340M	1B	25.3	36.1	32.9	24.6	57.2	64.3	52.5	41.8
539M	50M	24.2	31.0	29.8	19.0	52.9	64.6	50.0	38.8
539M	100M	26.1	31.8	31.6	19.7	54.7	65.3	51.5	40.1
539M	500M	25.1	35.3	34.3	23.6	57.4	64.5	52.5	41.8
539M	1B	25.2	37.3	35.1	24.3	59.3	65.2	51.4	**42.6**

Table 17: English downstream benchmark accuracy (%) at the same checkpoints. More target data (higher h) reduces English performance, reflecting the trade-off between target and non-target domains.

Model	Target tokens	ARC-C	ARC-E	HSwag	Lambada	PIQA	SciQ	Wino	**Avg**
101M	50M	23.1	40.2	32.9	31.4	64.7	62.7	50.7	43.7
101M	100M	23.1	40.2	32.6	30.8	64.9	65.3	49.8	43.8
101M	500M	21.3	35.3	28.8	24.6	61.6	57.8	51.5	40.1
101M	1B	21.8	35.5	28.0	22.5	60.7	57.8	51.5	39.7
143M	50M	23.2	40.4	34.6	36.6	65.2	64.0	50.5	44.9
143M	100M	24.6	41.2	34.9	32.5	66.5	67.7	51.0	45.5
143M	500M	23.3	38.8	32.2	29.9	64.6	66.0	48.9	43.4
143M	1B	22.4	38.9	30.3	28.8	63.0	62.8	51.9	42.6
197M	50M	24.7	41.3	37.9	35.5	67.3	66.6	51.9	46.4
197M	100M	23.9	41.3	36.8	35.8	67.7	67.4	51.2	46.3
197M	500M	23.4	41.0	35.0	32.0	65.8	66.1	50.7	44.9
197M	1B	22.1	38.6	31.3	29.4	63.0	62.7	51.8	42.7
340M	50M	24.2	45.2	41.9	39.8	68.9	69.7	52.9	48.9
340M	100M	24.3	44.6	42.3	40.6	69.2	69.9	51.2	48.9
340M	500M	25.4	44.2	40.1	40.0	68.5	70.1	51.7	48.6
340M	1B	24.3	42.3	38.5	38.5	66.2	67.6	51.3	47.0
539M	50M	26.4	46.6	46.5	45.4	70.3	71.4	52.8	**51.4**
539M	100M	24.7	47.5	46.5	45.4	69.5	72.4	53.7	**51.4**
539M	500M	24.6	46.6	45.2	43.1	69.2	70.1	52.4	50.2
539M	1B	24.3	45.4	42.3	42.1	67.5	69.0	49.7	48.6

N LIMITATIONS

We did not tune training hyperparameters jointly with the mixture and repetition settings: the learning rate was selected once on small-scale runs in the unconstrained setting and held fixed across all mixture configurations. It is possible that interactions between learning rate, batch size, or schedule and the number of repetitions could shift the optimal operating points we report. Additionally, all experiments use a single architecture family (GPT-2-style decoder-only transformers); whether the observed scaling behavior transfers to other architectures remains an open question. We leave both explorations to future work. Our largest model is 1.2B parameters; while we observe consistent trends across model sizes suggesting the scaling law extrapolates, verifying this at scales of tens or hundreds of billions of parameters would require substantially more compute than available for this study.