HOW TEXT QUALITY INTERVENTIONS RESHAPE NEURAL SCALING LAWS FOR LLMS: EMPIRICAL STUDY

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural scaling laws are widely used for performance projection and resource planning, yet their sensitivity to data quality interventions remains poorly understood. We present the first large-scale empirical study of how interventions—deduplication, heuristic filtering, and LLM-guided rewriting—reshape scaling behavior in large language model training. Using QualityPajama, a suite of 23 systematically curated datasets, we train over 2,000 models (100M–8B parameters, 100M–200B tokens) to measure how text quality interventions affects scaling-law parameters and compute-optimal design decisions. While prior studies have shown that model architecture primarily shifts coefficients, we demonstrate that data interventions shift both coefficients and exponents, fundamentally changing the fitted scaling laws in ways not anticipated by existing theory. We show that data quality ranking is scale and resource-dependent. Compute-optimal token–to-parameter ratios vary by orders of magnitude across interventions, revealing a fundamental data quality–quantity trade-off in scaling. These findings pave the way for deeper theoretical understanding of scaling laws, establish scaling-law analysis as a principled framework for data strategy evaluation and ranking, and motivate a data-quality–aware approach to scaling next-generation LLMs.

1 INTRODUCTION

While nearly all large language models are trained on similar sources of text—web data— the key differentiating factor among state-of-the-art models lies in the quality of their pre-training and post-training data. However, data quality itself remains an elusive and context-dependent concept—what constitutes "high quality" can vary with downstream use case, compute scale, and resource constraints. This raises the question: *can neural scaling laws offer a principled framework for ranking data quality across scales?*

Neural scaling laws are empirical relationships that describe how model performance improves as a function of resource investment - typically the number of parameters and training tokens. A growing body of empirical Hestness et al. (2017); Johnson & Nguyen (2017); Rosenfeld et al. (2019); Kaplan et al. (2020); Hernandez et al. (2021); Ghorbani et al. (2021); Ardalani et al. (2022); Hoffmann et al. (2022); Alabdulmohsin et al. (2022); Aghajanyan et al. (2023); Isik et al. (2024); Zhang et al. (2024) and theoretical Sharma & Kaplan (2022); Bahri et al. (2024); Brill (2024); Hutter (2021); Michaud et al. (2023); Dohmatob et al. (2024b); Dębowski (2023); Dohmatob et al. (2024a) work has shown that pre-training loss follows a power-law trend with respect to these axes. Neural scaling laws have been central to the development of large language models (LLMs), informing decisions about model scaling, data scaling, and compute allocation, while also serving as a key tool for return-on-investment (ROI) analysis and capability forecasting Hestness et al. (2019); Hoiem et al. (2021); Mahmood et al. (2022); Alabdulmohsin et al. (2022). However, despite their widespread adoption, the impact of data quality on scaling laws remains poorly understood.

A prominent example of this uncertainty is the ongoing debate over the discrepancy between Kaplan's Kaplan et al. (2020) and Hoffman's Hoffmann et al. (2022) prediction of the compute-optimal token-to-parameter ratio (21 vs. 1) Porian et al. (2024); Pearce & Song (2024); Bi et al. (2024). Recent work speculates that differences in training data may have played a role in this divergence Bi et al. (2024). While prior theoretical works Sharma & Kaplan (2022); Bahri et al. (2024); Brill

(2024); Hutter (2021); Michaud et al. (2023); Dohmatob et al. (2024b); Dębowski (2023); Dohmatob et al. (2024a) have linked the power-law *exponents* to properties of the data manifold and the Zipfian distribution of input tokens, the impact of *text quality* interventions on these underlying structures remains poorly understood. Furthermore, prior work overlooks how data quality influences **other components** of the scaling law—namely, the coefficients and asymptotic loss terms—which, as we will show, play a critical role in shaping loss behavior at today's compute scales. Moreover, most theoretical predictions isolate a single exponent (either model or data) while holding the other in the infinite limit. As we demonstrate, understanding the **joint fit** is essential, as the components often move in opposing directions to control loss trajectory, revealing important trade-offs induced by data quality shifts. Although prior empirical work has explored the effects of synthetic noise, data source composition, and filtering algorithms in domains such as machine translation Bansal et al. (2022) and image classification Bahri et al. (2024), to the best of our knowledge, there is no systematic study examining how *text-specific interventions*—such as filtering, deduplication, rephrasing and mixing synthetic and natural data—impact the components of neural scaling laws in LLM pretraining.

Our work bridges this gap by conducting a large-scale empirical analysis of diverse data quality interventions for pretraining large scale language models and study how they influence **all** components of the scaling law. We introduce a benchmark of 23 curated datasets, each representing a different quality intervention, and train over 100 language models per dataset, totaling more than 2000 model training runs. This extensive experimental design enables us to disentangle the effects of data quality on scaling law components and loss behavior, and propose how to design an effective data quality strategy as we scale.

1.1 Our Contributions

- **QualityPajama Benchmark:** We introduce *QualityPajama*, a benchmark suite of 23 datasets designed to systematically evaluate the impact of diverse text quality interventions on neural scaling behavior in LLMs. (Section 3)
- **Full Scaling Law Decomposition:** We provide the first systematic analysis of how text-quality interventions affect all components of the joint scaling law—not only the exponents. Our results show that stronger filtering does not consistently push components toward more favorable regimes, but instead produces conflicting shifts across parameters. (Section 4)
- **Data-Aware Scaling Strategies:** We show that designing compute-optimal scaling strategies requires careful accounting for data quality, as variation in quality could shift the optimal number of parameters, tokens, and their ratio by couple orders of magnitude.(Section 4.1)
- **Scale- and Resource-Dependent Rankings:** Data quality rankings are not uniform across scales or resource regimes. Strategies that excel at small scales may underperform at larger ones, and the optimal choice depends critically on the constraint (e.g., fixed compute vs. fixed data). Moreover, "scale" can refer to model size, dataset size, or compute budget, and the best intervention differs across these regimes. We recommend using scaling-law curves to rank data quality strategies across different scales and resource constraints, rather than relying on small-scale experiments, which often lead to misleading conclusions. (Section 4.1)
- **Deduplication Efficiency:** We demonstrate that deduplication yields large compute savings that far exceed reductions in data volume (Section 5)
- **PageRank Signals:** While PageRank scores correlate with improved quality, filtering based solely on PageRank does not outperform the unfiltered baseline. (Section 5)
- **Synthetic–Natural Data Mixing:** We show that mixing synthetic and natural data consistently outperforms using either alone, but the optimal mixing ratio evolves as the model and compute scale. (Section 5)

2 Background and Related Work

The study of scaling laws in deep learning has a rich history, with numerous empirical Hestness et al. (2017); Johnson & Nguyen (2017); Rosenfeld et al. (2019); Kaplan et al. (2020); Hernandez et al. (2021); Ghorbani et al. (2021); Ardalani et al. (2022); Hoffmann et al. (2022); Alabdulmohsin et al. (2022); Aghajanyan et al. (2023); Isik et al. (2024); Zhang et al. (2024) and theoretical Sharma & Kaplan (2022); Bahri et al. (2024); Brill (2024); Hutter (2021); Michaud et al. (2023); Dohmatob et al. (2024b); Dębowski (2023); Dohmatob et al. (2024a) investigations into their components. A commonly used form of the scaling law is given by:

$$\text{Loss}(N, D) \sim AD^{-\alpha} + BN^{-\beta} + E$$

where Loss typically represents cross-entropy loss, D denotes data size in tokens, N represents model size in parameters, and α, β, A, B, and E are constants. The terms in this equation capture the effects of finite data, limited model capacity, and the inherent entropy of the underlying phenomenon, respectively.

Although prior work has empirically explored the impact of model architecture Tay et al. (2022), vocabulary size and tokenizer on the components of scaling law Hestness et al. (2017); Kaplan et al. (2020), the impact of data quality on all components of scaling law remains poorly understood. Prior theoretical works on the origin of the power law and its relation to the dimensionality of data manifold Sharma & Kaplan (2022); Bahri et al. (2024) and Zipfian distribution of input data Hutter (2021); Michaud et al. (2023) are perhaps closest to our own. usually under some simplifying assumptions like infinite data size or model size. They particularly make predictions about the exponents of power law but remain silent about other components.

Data Manifold Theory: Data manifold refers to the low-dimensional structure that higher dimensional data lies on. Data manifold theory predicts that exponents of power law are inversely proportional to the data manifold dimension Sharma & Kaplan (2022); Bahri et al. (2024). However, the impact of data quality on data manifold itself is poorly understood. Data quality, particularly text quality, can be characterized across various axes: diversity of topics, grammar complexity, formatting artifacts, information density, factuality, fairness, safety, etc. While prior theoretical work do not discuss the impact of data quality explicitly, their machinery is powerful enough to make predictions. Take removing unstructured noise, like garbled text, it could ostensibly decrease the apparent dimensionality. On the other hand, deduplication could expand the data manifold. While both are different text interventions towards improving quality, one seems to improve the exponent, while the other decreases.

Zipfian Distribution Theory: Zipf's law is another empirical observation that explains word frequencies follow a power-law in their rank. It shows up not only in word frequencies, but also in n-gram distributions Ha et al. (2009), sentence structures, and higher-level concepts Michaud et al. (2023). Prior work conjectures that if input data follows a Zipfian distribution, the Zipf's exponent correlates with the power law exponent Hutter (2021); Michaud et al. (2023). However, much like data manifold theory, the impact of data quality interventions on Zipfian distribution are not quite predictable. While some data intervention techniques, like synthetic data generation cuts off the heavy tail of the input distribution, other intervention techniques like deduplication flattens the head of the curve. This implies that Zipfian slope gets steeper for synthetic data but flatter for deduplicated data. We will see later in Section 4, these predictions are not always consistent with empirical observations as it is not easy to predict how data quality interventions influence distribution.

Effective Tokens and Utility-Based Scaling Laws: Prior work has examined how to incorporate data quality into scaling law formulations. Chang et al. (2024) focus only on the data axis, proposing to replace dataset size D with an *effective* variant, but leaving other components of the law unchanged. Muennighoff et al. (2023) extend this idea to both model size and dataset size, introducing effective formulations N' and D', though their analysis is tailored to the setting of repeated epoching rather than data interventions. Goyal et al. (2024) similarly reinterpret the data exponent β in terms of *effective utility*. These approaches capture aspects of data efficiency but treat quality as primarily modifying D or β, overlooking its broader influence on parameter coefficient and exponents, or irreducible loss. By contrast, we show that data interventions perturb *all* components of the joint scaling law fit. Most recently, Shukor et al. (2025) proposed a "full" scaling law for data mixtures, which is closest in spirit to our work. Their focus is on mixture composition as the intervention, whereas we analyze heuristic filtering and synthetic data rewrites, broadening the range of data-centric interventions studied under scaling laws. Overall, our work is the first to demonstrate that text quality interventions affect *all* components of the scaling law, not just the data dimension, providing a more complete picture of how quality reshapes scaling dynamics and offering practical guidance for data-centric scaling strategies.

Synthetic Data Scaling Laws Fan et al. (2024) studied the impact of synthetic images on scaling laws, particularly on data exponent. Qin et al. (2025) examined how generator model size influences scaling laws on downstream tasks for LLMs. In contrast, we study upstream loss and investigate how mixing synthetic and natural data shapes scaling behavior in LLM.

Dynamic Data Intervention and Non-Power-Law Scaling Sorscher et al. (2022) show that, with adaptive data pruning during training, it is possible to surpass standard power-law scaling and approach exponential improvements. In contrast, our work assumes interventions are applied once prior to pre-training, rather than adaptively adjusting throughout training.

Post-Training Data Quality Recent work has investigated the role of data quality over quantity in post-training alignment, showing that even small high-quality datasets improve performance (Zhou et al., 2023; Xia et al., 2024).

Text Quality Interventions can be characterized across multiple axes, including information entropy, topical diversity, grammar complexity, formatting artifacts, factuality, fairness, and safety. The exact definition of text quality typically varies by downstream usecase. In this work, we focus on the impact of text quality on upstream loss. Broadly, data quality can be manipulated through three strategies: **filtering**, which removes low-quality or undesired content using heuristic or model-based approaches Raffel et al. (2020); Lee et al. (2021); **mixing**, which rebalances data distributions or adds high-quality subsets Li et al. (2024); Shukor et al. (2025); and **synthetic generation**, which uses LLMs to clean or augment existing content. These approaches have informed the design of many recent LLM training corpora, including RedPajama AI (2023), Dolma Soldaini et al. (2024), RefinedWeb Penedo et al. (2023), FineWeb Penedo et al. (2024), DCLM Li et al. (2024).

3 QUALITYPAJAMA

We introduce QualityPajama, a benchmark suite of 23 datasets derived from Common Crawl, each reflecting a distinct level of data quality and intervention. The suite spans a broad spectrum of data quality techniques, including 14 filtered datasets and 9 synthetically curated datasets, for training large language models. Table 1 summarizes the interventions used in each category. Additional details regarding dataset construction and design choices can be found in Appendix.

4 IMPACT OF DATA QUALITY ON SCALING LAW COMPONENTS

We aim to understand how data quality affects scaling law components, whether predictable patterns emerge under quality interventions, and how these insights can guide effective data curation.

Figure 1 visualize the impact of text quality interventions, particularly heuristic-based filtering and synthetic data generation, on components of neural scaling laws, namely α, β, A, B and E. Each line in the radial plot represents a different training set, while the radial axis displays various validation sets. It is apparent from these results that all components are sensitive to training set quality as well as validation set quality.

Sequential Application of Data Filters and Effects on Scaling Components We apply a series of data filters sequentially and extract intermediate datasets at each stage to conduct scaling law analysis. The order in which filters are applied is indicated in the legend. Interestingly, the trajectory of changes in scaling law components does not necessarily follow the order of interventions. Take α for example: it increases after removing NSFW content (red to green), but decreases after filtering garbled text (green to yellow). It decreases further after removing pages with low PageRank scores (yellow to blue), but then increases again after deduplication (blue to orange). As shown in Appendix A, these dynamics are not always consistent with predictions from Zipf's law or the data manifold hypothesis, showcasing the limitations of the current theory.

Component-Wise Correlations We examine whether scaling law components exhibit consistent patterns under data quality changes—for example, whether improving quality increases the model

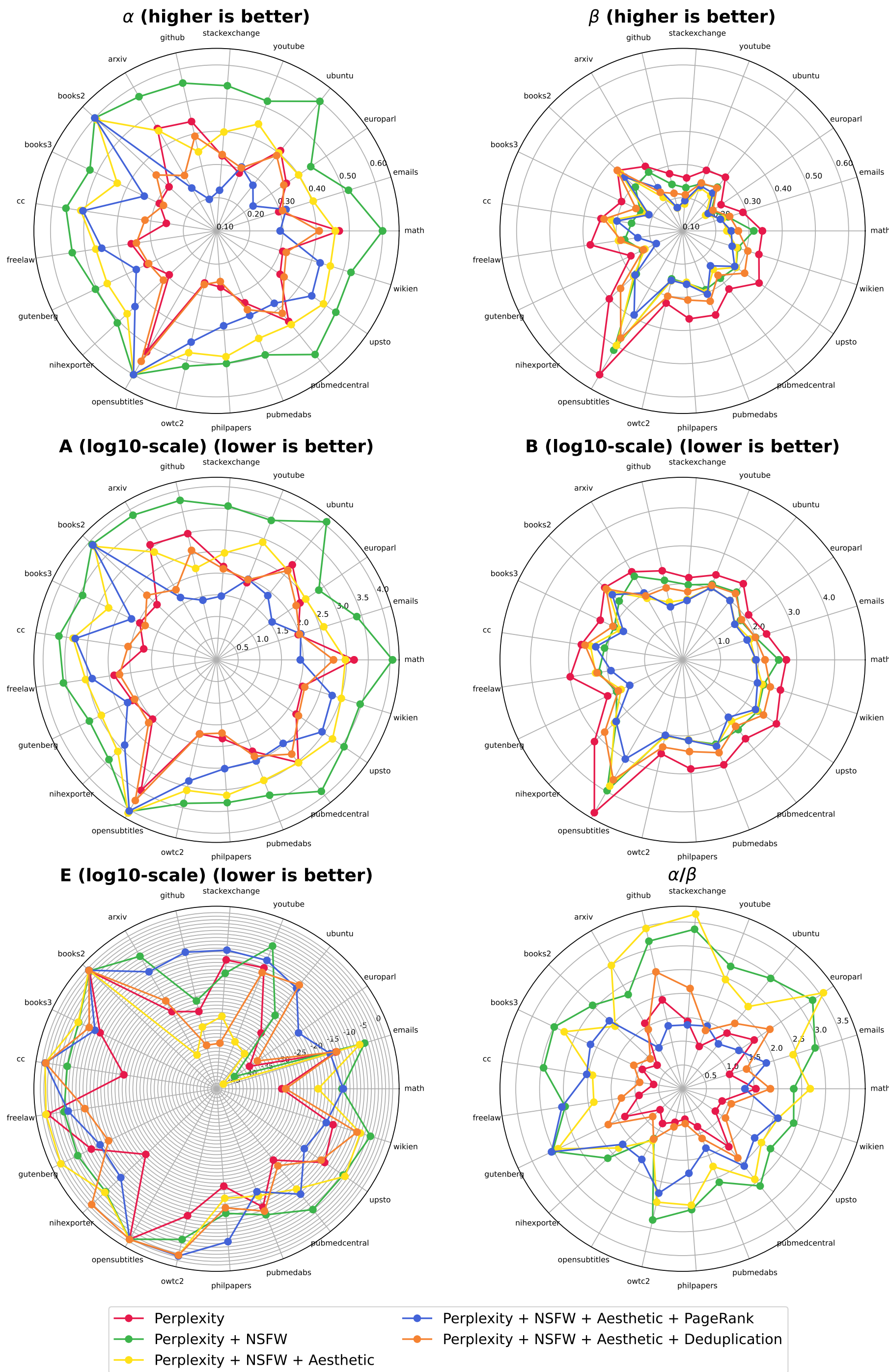


Figure 1: **How Data Filtering Affects Scaling Law Components.** Different colored lines represent different data-quality interventions, while the radial axes show results across different validation sets. Stronger filtering does not uniformly improve scaling-law components: while some parameters move toward more favorable regimes, others degrade, highlighting tensions between different components of scaling law.

Table 1: Summary of QualityPajama dataset interventions.

Category	Description	Abbrev./Variants
Heuristic-based Filters (14 variants)		
NSFW Filtering	Removes documents containing offensive or inappropriate content.	`nsfw`
Aesthetic Filters	Filters out text with undesirable patterns (e.g., "lorem ipsum", inline code, or high alphanumeric ratios > 0.8).	`aesthetic`
PageRank Filtering	Partitions pages into low/medium/high/unknown based on PageRank score. Thresholds are set to the 33rd and 67th percentiles of the score distribution of all pages in the PageRank table. Page et al. (1999).	`high_pr`, `med_pr`, `low_pr`, `no_pr`
Deduplication	Fuzzy deduplication using MinHashLSH Leskovec et al. (2020) with different similarity thresholds; selects lowest perplexity document from near-duplicate clusters.	`deduped_0.7`, `deduped_0.8`, `deduped_0.9`, `deduped_1.0`
Grammar Complexity	Filters based on average sentence length as a proxy for syntactic richness, with thresholds at 10 tokens for short text and 25 tokens for medium text	`short_text`, `medium_text`, `long_text`
Synthetic Curation (9 variants)		
High Quality Rephrasing (HQ)	LLM rewrites documents to be clearer and more coherent Maini et al. (2024). Mixtures denote the percentage of synthetic vs. natural data (CC).	`HQ100`, `HQ67-CC33`, `HQ33-CC67`.
Question Answering Rephrasing (QA)	LLM converts documents into conversational QA pairs.	`QA100`, `QA67-CC33`, `QA33-CC67`
Textbook-style Rephrasing (TB)	Converts documents into textbook-style chapters using structured prompting (inspired by Phi models Li et al. (2023); Javaheripi et al. (2023); Abdin et al. (2024)).	`TB100`, `TB67-CC33`,`TB33-CC67`

Table 2: **Can Scaling Components Reliably Rank Data Interventions?** We report average Spearman correlations across validation sets for each scaling law component. Moderate values (0.3–0.5) suggest that component-based rankings are only partially preserved across validation sets; higher values suggest reliable ordering. Results suggest that such metrics may not reliably rank natural data interventions. In contrast, rankings for synthetically curated datasets show strong consistency, suggesting scaling components are more reliable for evaluating synthetic data strategies.

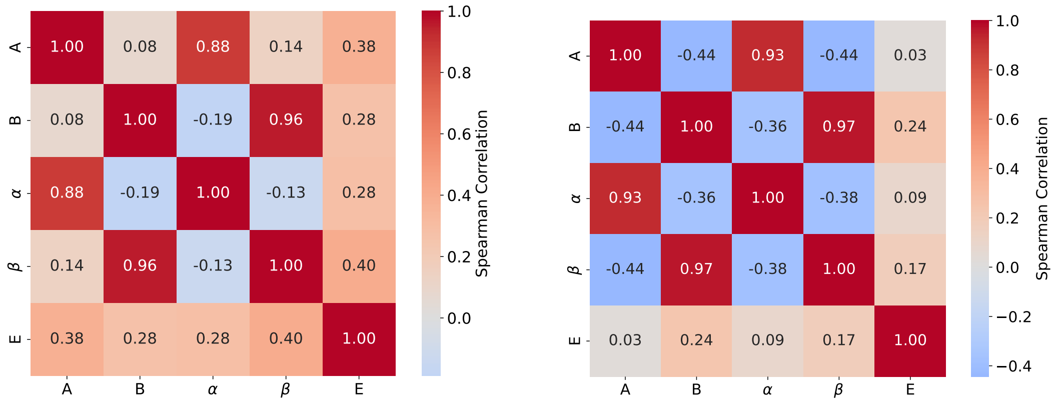
Data Interventions	**A**	**B**	α	β	**E**
All heuristic filters	0.45	0.34	0.46	0.32	0.34
All synthetic data	0.81	0.91	0.76	0.91	0.54

exponent (α) and decreases the data exponent (β). While Figure 1 suggests such trends qualitatively, we quantify them via Spearman correlations (Figures 2a and 2b). The strongest, most stable correlations across all validation sets are: $A \propto \alpha$ and $B \propto \beta$.

Sensitivity to Validation Sets We examine whether data intervention rankings are consistent across validation sets. Table 2 reports average Spearman correlations per component. While filters in Figure 1 show high consistency, rankings across all 14 heuristic filters exhibit only moderate correlation (0.3–0.5), suggesting that filter rankings are only partially preserved across validation sets. **This indicates that scaling law behavior is not independent of the validation set for *naturally* curated datasets.** On the contrary, for synthetically curated datasets, we see a strong correlation across validation sets. **This indicates that scaling law behavior is less sensitive to validation set for *synthetically* curated datasets.**

4.1 Interpretations

Designing Compute-Optimal Scaling Strategy Requires Accounting for Data Quality: Prior work has shown that compute-optimal design decisions depend on the scaling law components: α, β, A, and B. Since data quality influences these parameters, it directly affects compute-optimal



(a) Heuristic filters

(b) Synthetically curated

Figure 2: **How Scaling Law Components Co-Vary with Data Quality Intervention?**. We observe strong monotonic correlations between A and α, and between B and β. For synthetic data, there are also notable negative correlations between α and β, and between A and B.

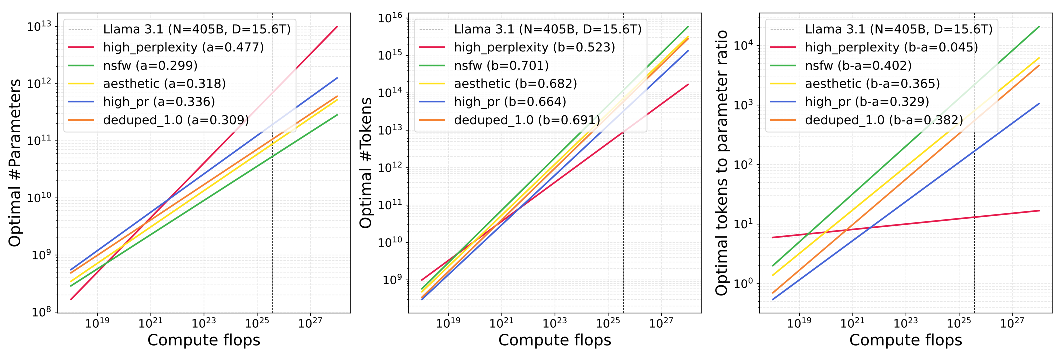


Figure 3: **How Data Quality Influences Scaling Strategy?** Given $N_{\text{opt}} \propto C^a$, $D_{\text{opt}} \propto C^b$, and $D_{\text{opt}}/N_{\text{opt}} \propto C^{b-a}$, where $a = \beta/(\alpha+\beta)$ and $b = \alpha/(\alpha+\beta)$. **(Left)** shows how optimal model size scales with compute. At today's compute budget (dashed line), the best and worst data interventions differ by over an order of magnitude in optimal model size. **(Right)** shows the variation in token-to-parameter ratio, where interventions differ by up to two orders of magnitude at the same compute scale.

choices. Figure 3 illustrates how the compute-optimal number of tokens, number of parameters, and their respective ratio (a proxy for sample efficiency) scale with available compute and vary with data quality intervention. Notably, at today's compute scale (indicated by the dashed line), the optimal design point can differ significantly across—by up to 14× for the number of parameters, 13× for the number of tokens, and an astonishing 182× for the token-to-parameter ratio. These results highlight the critical role of data quality in determining efficient scaling strategies, underscoring the need to account for quality variations when designing large-scale training runs.

Tension Among Scaling Law Components: Data interventions do not uniformly shift all components of the scaling law in a direction that reduces loss. We observe that the coefficients A and B are positively correlated with their corresponding exponents α and β, respectively. This coupling creates a tension in how different components influence performance. While increasing the exponents α and β typically leads to improved scaling and lower loss, increases in the coefficients A and B have the opposite effect, raising the loss. As a result, interventions that improve one component may simultaneously degrade another.

One may argue that in the trade-off between exponent and coefficient, the exponent should dominate, since its effect is exponential while the coefficient scales only linearly. While this may hold

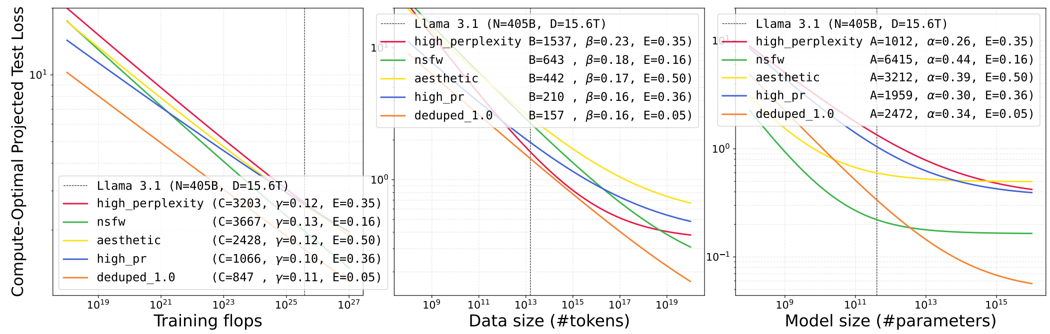


Figure 4: **How Does the Optimal Data Quality Strategy Change with Scale and Resource Constraints?** Compute-scaling law (left), data-scaling law (middle), and model-scaling law (right) curves show that no single data strategy remains optimal across all scales. The optimal choice shifts as the resource scale changes and also depends on which resource is constrained: model size, data size, or compute budget.

asymptotically at extremely large scale, Figure 4 shows that the tension persists even at today's compute scale (e.g., 10^{24} FLOPs). This persistence may be due to the fact that the coefficients A and B vary across several orders of magnitude, while the exponents α and β remain relatively small, limiting their ability to compensate.

Notably, in synthetically curated datasets, we observe a negative correlation between α and β, suggesting that improvements in model scaling efficiency may come at the expense of data scaling efficiency. Such opposing forces highlight the complex and sometimes counteractive nature of data quality interventions on loss behavior. This underscores the need to analyze all components of the scaling law jointly, rather than relying on any single metric to assess data quality improvements.

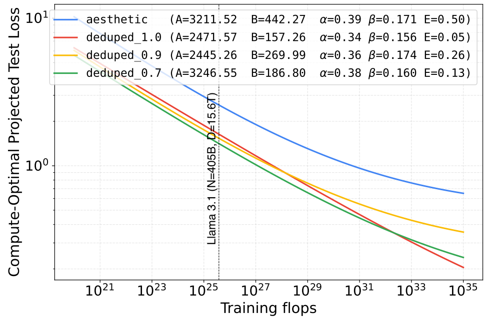
Data Quality Rankings Vary with Scale We observe frequent crossovers between scaling curves for different data interventions (Figure 4), indicating that a dataset which minimizes loss at small scale may be outperformed by another at larger scale. This shift in relative performance highlights the risk of extrapolating small-scale experimental results to large-scale settings. Consequently, conclusions drawn from limited-scale experiments may not generalize to high-compute regimes, and data quality strategies should be validated at or near the intended scale of deployment to ensure their effectiveness holds under real-world training budgets.

The Best Data Quality Strategy Depends on Your Resource Constraint In addition to being scale-dependent, the "best" data quality strategy depends on the specific resource constraint, as shown in Figure 4. For instance, if the goal is to identify the most efficient dataset under a fixed compute budget, compute scaling provides the most relevant lens. However, if the constraint lies in model size or available training tokens, the conclusions may differ. Therefore, practitioners should be mindful of their primary resource constraint when evaluating or selecting data quality strategies, as the optimal choice is inherently constraint-dependent.

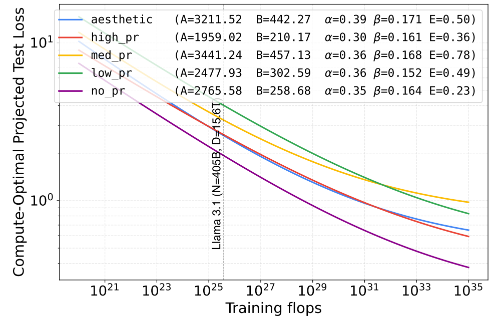
5 Data Quality Intervention Comparisons Through Compute-Efficiency Lens

How aggressive should deduplication be? Is there a diminishing return in compute efficiency as we dudupe more aggressively? Is PageRank a useful signal for filtering? Is it more or less compute efficient to train with synthetic data? Do improvements in compute-efficiency merely reflect reductions in data volume, or can they go further? To address these questions, we analyze compute scaling laws under various data quality interventions in Figure 5.

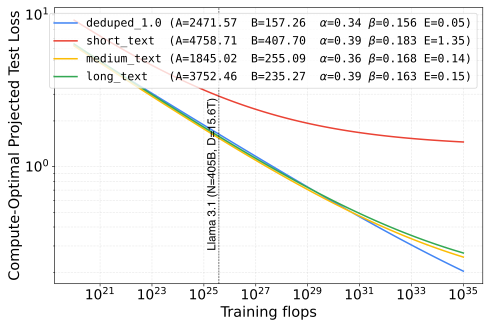
- **Deduplication:** Fuzzy deduplication offers substantial compute savings that far exceed reductions in dataset size. For example, exact deduplication reduces data volume to 83% of



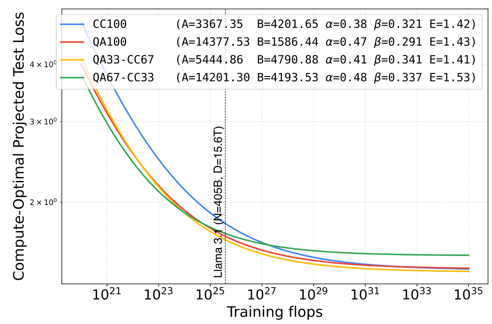
(a) Deduplication



(b) PageRank Filtering



(c) Sentence Length Filtering



(d) Synthetic–Natural Mixing

Figure 5: Compute scaling law results for various data quality interventions.

its original size yet yields a $100\times$ gain in compute efficiency. Fuzzier approaches perform even better: `dedupe_0.7` requires approximately $3\times$ less compute than `dedupe_0.9`, $10\times$ less than exact deduplication, and $300\times$ less than no deduplication (Figure 5a).

- **PageRank Filtering:** While a higher PageRank correlates with improved quality (`high_pr` $>$ `med_pr` $>$ `low_pr`), filtering strictly by high PageRank does not outperform the baseline. In contrast, including pages not found in the ranking table (`no_pr`) results in significantly greater compute efficiency—likely due to recency effects (Figure 5b).
- **Synthetic–Natural Mixing:** Mixing synthetic and natural data consistently outperforms using either alone, but the optimal mixing ratio evolves with compute scale (Figure 5d).

6 Discussion

Summary: We set out to analyze the impact of text quality interventions, particularly heuristic-based filtering and LLM-guided data rewrite, on the components of neural scaling laws in training large language models. To enable this study, we developed QualityPajama, a benchmark suite of 23 systematically constructed text datasets spanning a range of quality levels and interventions, from filtering to deduplication to paraphrasing and synthetic curation built on top of Common Crawl dataset. We found that: (1) all components of the scaling law are sensitive to data quality (2) data intervention rankings are not preserved across scales; (3) the decision on how to scale model size and data size with increased compute budget is heavily influenced by data quality; (4) data intervention impact on compute saving goes far beyond the reduction in data volume; and (5) mixing synthetic and natural data outperforms using either alone, though the optimal ratio is scale dependent.

Ethical Considerations A potential negative societal impact of this work is that data interventions may unintentionally amplify biases or lead to unfair outcomes for certain groups. While our analysis shows how interventions affect scaling-law parameters, scaling laws alone should not be treated as a sufficient basis for data strategy decisions. Broader evaluations—including fairness, representational balance, and downstream task impacts—are necessary to ensure that improvements in efficiency do not come at the cost of equity or inclusiveness.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Armen Aghajanyan, Lili Yu, Alexis Conneau, Wei-Ning Hsu, Karen Hambardzumyan, Susan Zhang, Stephen Roller, Naman Goyal, Omer Levy, and Luke Zettlemoyer. Scaling laws for generative mixed-modal language models. In *International Conference on Machine Learning*, pp. 265–279. PMLR, 2023.

Together AI. Redpajama: Open pretraining data for llms. `https://github.com/togethercomputer/RedPajama-Data`, 2023.

Ibrahim M Alabdulmohsin, Behnam Neyshabur, and Xiaohua Zhai. Revisiting neural scaling laws in language and vision. *Advances in Neural Information Processing Systems*, 35:22300–22312, 2022.

Newsha Ardalani, Carole-Jean Wu, Zeliang Chen, Bhargav Bhushanam, and Adnan Aziz. Understanding scaling laws for recommendation models. *arXiv*, 2208.08489, 2022. URL `https://arxiv.org/abs/2208.08489`.

Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *Proceedings of the National Academy of Sciences*, 121(27):e2311878121, 2024.

Yamini Bansal, Behrooz Ghorbani, Ankush Garg, Biao Zhang, Colin Cherry, Behnam Neyshabur, and Orhan Firat. Data scaling laws in nmt: The effect of noise and architecture. In *International Conference on Machine Learning*, pp. 1466–1482. PMLR, 2022.

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

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024.

Matthew A Branch, Thomas F Coleman, and Yuying Li. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. *SIAM Journal on Scientific Computing*, 21(1):1–23, 1999. doi: 10.1137/S1064827595289108.

Ari Brill. Neural scaling laws rooted in the data distribution. *arXiv preprint arXiv:2412.07942*, 2024.

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. *arXiv preprint arXiv:2410.03083*, 2024.

Łukasz Dębowski. A simplistic model of neural scaling laws: Multiperiodic santa fe processes. *arXiv preprint arXiv:2302.09049*, 2023.

Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. Strong model collapse. *arXiv preprint arXiv:2410.04840*, 2024a.

Elvis Dohmatob, Yunzhen Feng, Pu Yang, Francois Charton, and Julia Kempe. A tale of tails: Model collapse as a change of scaling laws. *arXiv preprint arXiv:2402.07043*, 2024b.

Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7382–7392, 2024.

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. *arXiv preprint arXiv:2101.00027*, 2020.

Behrooz Ghorbani, Orhan Firat, Markus Freitag, Ankur Bapna, Maxim Krikun, Xavier Garcia, Ciprian Chelba, and Colin Cherry. Scaling laws for neural machine translation. *arXiv preprint arXiv:2109.07740*, 2021.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22702–22711, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Le Quan Ha, Philip Hanna, Ji Ming, and Francis Jack Smith. Extending zipf's law to n-grams for large corpora. *Artificial Intelligence Review*, 32:101–113, 2009.

Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Joel Hestness, Newsha Ardalani, and Gregory Diamos. Beyond human-level accuracy: Computational challenges in deep learning. In *Proceedings of the 24th symposium on principles and practice of parallel programming*, pp. 1–14, 2019.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Derek Hoiem, Tanmay Gupta, Zhizhong Li, and Michal Shlapentokh-Rothman. Learning curves for analysis of deep networks. In *International conference on machine learning*, pp. 4287–4296. PMLR, 2021.

Marcus Hutter. Learning curve theory. *arXiv preprint arXiv:2102.04074*, 2021.

Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. Scaling laws for downstream task performance of large language models. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024.

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 1(3):3, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL `https://arxiv.org/abs/2310.06825`.

Mark Johnson and Dat Quoc Nguyen. How much data is enough? predicting how accuracy varies with training data size, 2017. `https://web.science.mq.edu.au/~mjohnson/papers/Johnson17Power-talk.pdf`.

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. *arXiv preprint arXiv:2001.08361*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. *arXiv preprint arXiv:2107.06499*, 2021.

Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive data sets*. Cambridge university press, 2020.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick 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 Kamal Mohamed Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Joshua P Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah M Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi 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 T 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 *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL `https://openreview.net/forum?id=CNWdWn47IE`.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

Rafid Mahmood, James Lucas, David Acuna, Daiqing Li, Jonah Philion, Jose M Alvarez, Zhiding Yu, Sanja Fidler, and Marc T Law. How much more data do i need? estimating requirements for downstream tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 275–284, 2022.

Pratyush Maini, Skyler Seto, He Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the web: A recipe for compute and data-efficient language modeling. *arXiv preprint arXiv:2401.16380*, 2024.

Eric Michaud, Ziming Liu, Uzay Girit, and Max Tegmark. The quantization model of neural scaling. *Advances in Neural Information Processing Systems*, 36:28699–28722, 2023.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376, 2023.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford infolab, 1999.

Tim Pearce and Jinyeop Song. Reconciling kaplan and chinchilla scaling laws. *arXiv preprint arXiv:2406.12907*, 2024.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023. URL `https://arxiv.org/abs/2306.01116`.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024.

Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. *Advances in Neural Information Processing Systems*, 37:100535–100570, 2024.

Zeyu Qin, Qingxiu Dong, Xingxing Zhang, Li Dong, Xiaolong Huang, Ziyi Yang, Mahmoud Khademi, Dongdong Zhang, Hany Hassan Awadalla, Yi R Fung, et al. Scaling laws of synthetic data for language models. *arXiv preprint arXiv:2503.19551*, 2025.

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. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL `http://jmlr.org/papers/v21/20-074.html`.

Jonathan S Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. *arXiv preprint arXiv:1909.12673*, 2019.

Utkarsh Sharma and Jared Kaplan. Scaling laws from the data manifold dimension. *Journal of Machine Learning Research*, 23(9):1–34, 2022.

Mustafa Shukor, Louis Bethune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. Scaling laws for optimal data mixtures. *arXiv preprint arXiv:2507.09404*, 2025.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. 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. *arXiv preprint*, 2024. URL `https://arxiv.org/abs/2402.00159`.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q Tran, Dani Yogatama, and Donald Metzler. Scaling laws vs model architectures: How does inductive bias influence scaling? *arXiv preprint arXiv:2207.10551*, 2022.

Mathurin Videau, Badr Youbi Idrissi, Daniel Haziza, Luca Wehrstedt, Jade Copet, Olivier Teytaud, and David Lopez-Paz. Meta Lingua: A minimal PyTorch LLM training library, 2024. URL `https://github.com/facebookresearch/lingua`.

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J van der Walt, Matthew Brett, Joshua Wilson, K Jarrod Millman, Nikolay Mayorov, Andrew R J Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, Ilhan Polat, Yu Feng, Eric W Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E A Quintero, Charles R Harris, Anne Archibald, Antônio H Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Maurice Weber, Daniel Y. Fu, Quentin Anthony, Yonatan Oren, Shane Adams, Anton Alexandrov, Xiaozhong Lyu, Huu Nguyen, Xiaozhe Yao, Virginia Adams, Ben Athiwaratkun, Rahul Chalamala, Kezhen Chen, Max Ryabinin, Tri Dao, Percy Liang, Christopher Ré, Irina Rish, and Ce Zhang. Redpajama: an open dataset for training large language models. *NeurIPS Datasets and Benchmarks Track*, 2024.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

Matei Zaharia, Mosharaf Chowdhury, Michael J Franklin, Scott Shenker, and Ion Stoica. Spark: Cluster computing with working sets. In *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, pp. 10–10. USENIX Association, 2010.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. *arXiv preprint arXiv:2402.17193*, 2024.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*, 2023.

Appendices

A Training Dataset

A.1 Baseline Dataset Choice

We build QualityPajama on top of CommonCrawl dataset assembled by **RedPajama-v2** AI (2023), which includes 84 Common Crawl snapshots from 2014 to 2023. RedPajama shares raw Common-Crawl dataset along with quality signals for each document but does not filter out any data from the mix. There is roughly 0.5 TB or 100 B tokens per snapshot per partition. We focus on the English subset from 34 snapshots and head partition, totaling approximately 15 TB of data (or 3 T tokens). This choice is motivated by three key considerations:

- **Minimal Pre-processing:** To be able to evaluate the impact of data quality interventions, we require a dataset that is minimally processed. RedPajama's CommonCrawl is preserving much of its original form while offering a clean interface.
- **Scale:** A dataset of substantial size is necessary to support scaling law analyses across multiple orders of magnitude—even after aggressive filtering. RedPajama-v2, is well suited in terms of both volume and temporal coverage. Given that our final dataset is $\approx 1\%$ of the original dataset, to enable an equal scaling range for all datasets, say upto 30B tokens, the original dataset should be in 3T tokens/15TB range.
- **URL Availability:** The presence of a URL for each document allows us to explore PageRank-based filtering techniques. This is particularly useful given that crawling algorithms like Hyper Centrality (used by Common Crawl) already introduce implicit biases that we can now systematically study.

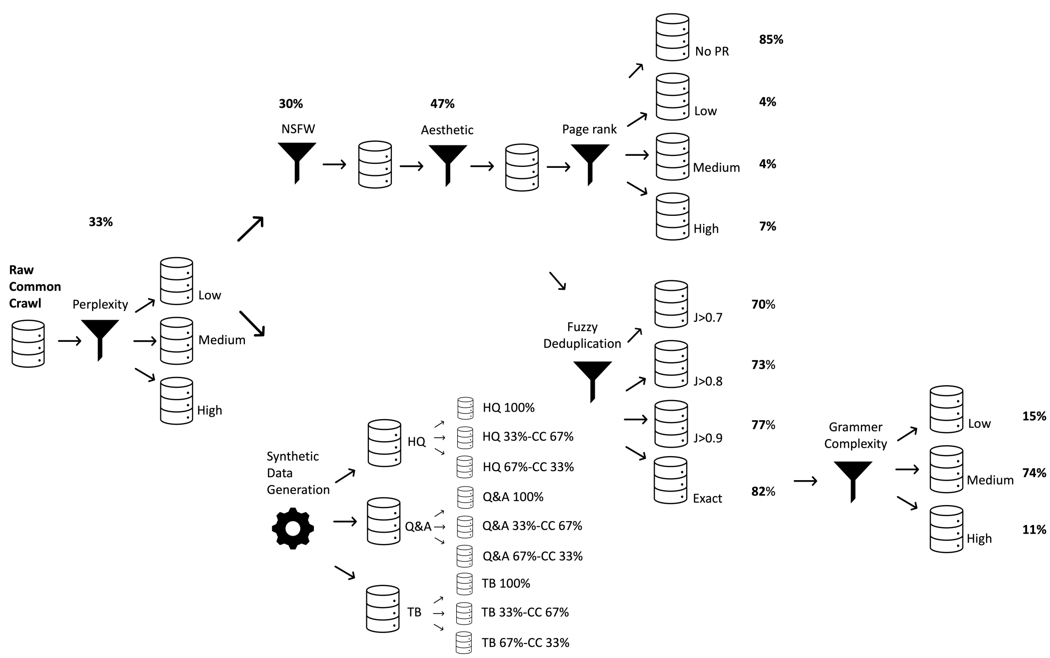


Figure 6: QualityPajama data pipeline. We show the filtering rate next to each filter. Given that our final dataset is $\approx 1\%$ of the original dataset in volume, to enable an equal scaling range for all datasets, say upto 30B tokens, the original dataset should be in 3T tokens/15TB range.

A.2 Derivative Datasets

Figure 6 illustrates the pipeline used to construct the QualityPajama benchmark suite. To support this, we developed a scalable Spark-based Zaharia et al. (2010) data processing framework—*PajamaKit*—that enables rapid experimentation with filtering, deduplication, and other data curation strategies.

A.2.1 Heuristic-based Data Quality Filters

We carefully hand-pick a set of filters that are deemed to improve quality to the extent that they are included in many data recipes used for curating well-known datasets such as C4 Raffel et al. (2020), Dolma Soldaini et al. (2024), RedPajama Weber et al. (2024), RefinedWeb Penedo et al. (2023) and FineWeb Penedo et al. (2024). These include NSFW filtering, format-based filtering, grammar-based filtering, deduplication, etc. We also include some less explored filters, like PageRanking score to study their effectiveness. We apply these filters sequentially and extract intermediate datasets after each stage. Heuristic filters usually are accompanied with some knobs to control their filtering degree. For instance, deduplication has a similarity threshold for deeming two samples duplicate and we are curious to understand: how does this knob controls quality? Where applicable, we experiment with multiple thresholds and retain only the "best" filtered dataset for downstream filtering. The filtering pipeline includes:

- **NSFW Filtering:** We remove all pages containing inappropriate or offensive language.
- **Aesthetic Filters:** We exclude documents containing undesirable patterns such as "lorem ipsum," inline code (e.g., "{", "javascript"), and those with a high alphanumeric character ratio (above 0.8).
- **PageRank Filtering:** We partition documents into four groups—low, medium, high, and not-found—based on their PageRank scores Page et al. (1999). Since Common Crawl sampling is biased towards high Hyper Centrality (correlated with PageRank), our analysis exposes implicit biases in many web-derived corpora. The thresholds are chosen to split the PageRank score distribution in our reference table into three equal parts.
- **Deduplication:** We apply deduplication at page granularity within each snapshot. For *fuzzy deduplication*, we use MinHashLSH Leskovec et al. (2020) at different Jaccard similarity thresholds (0.7, 0.8, 0.9, 1.0). We build MinHash signatures on top of pre-processed lower-cased bi-grams with 256 permutations. We use signature to build a similarity graph, from which connected components (clusters of near-duplicates) are identified. Within each cluster, the document with lowest perplexity score is retained.
- **Grammar Complexity:** We use *average sentence length* as a simple first-order proxy for syntactic complexity. Using NLTK for sentence and token segmentation, we bin documents into categories of short, medium, long, and very long sentences.

A.2.2 Synthetic Curation Techniques

While the literature on synthetic data generation is very rich, only a few have been proposed and deployed for pretraining large language models Li et al. (2023); Javaheripi et al. (2023); Abdin et al. (2024); Maini et al. (2024). Our goal here is not to generate new content but to clean up the existing content through careful prompting. We use three techniques proposed in the literature. All synthetic data was generated using a Mistral-Instruct-7b-v0.1 model with the following sampling parameters:

- Temperature: 0.7
- Top-p (nucleus sampling): 0.95

These parameters were chosen to balance creativity and coherence in the generated text.

We implemented distinct pipelines that represent leading methodologies in synthetic data generation. We modified the prompts from the original work (if available) to promote better format-following and encourage longer, high-quality text. Generation procedures are detailed below with full prompts provided in boxes A.2.2.1-A.2.2.4.

- **High Quality Rephrasing (HQ)** Inspired by WRAP Maini et al. (2024), we prompt LLM to rewrite source documents into clear, coherent, and well-structured text.
- **Question Answering Rephrasing (QA)** Inspired by WRAP Maini et al. (2024), we prompt LLM to convert source document into a conversational QA format.
- **Textbook-style Rephrasing (TB)** Inspired by family of Phi models Li et al. (2023); Javaheripi et al. (2023); Abdin et al. (2024), we first convert text into book chapter titles and

then prompt the LLM to generate new content for each chapter, with variations in prompts for different target audiences (grade school, college, expert, general).

Light heuristic post-filtering was applied to all generated synthetic datar, removing documents that were excessively short (e.g., less than 50 tokens) or excessively long relative to the target length for that generation type, if such outputs occurred despite prompt length guidance. The goal of this light filtering was to remove egregious generation errors without overly sanitizing the data or significantly altering its distribution.

A.2.2.1 Prompt Template HQ Rephrasing

- **System Prompt:** Provide direct and detailed response to the instructions without adding additional notes.
- **[USER]:** For the following document, regardless of its original content or formatting, write a full article of the same content in high quality English language as in texts on Wikipedia: [xxxx]. Provide the rephrased article without any additional notes. Long article with full length and complete details. Rephrased article:

A.2.2.2 Prompt Template QA Rephrasing

- **System Prompt:** Provide direct and detailed response to the instructions without adding additional notes.
- **[USER]:** For the following document, regardless of its original content or formatting, convert it into a comprehensive list of question-answer pairs with multiple tags of "Question:" followed by "Answer:", where questions and answers cover complete information of the original document. Document: [xxxx]. Provide the converted question-answer pairs without any additional notes. Question-answer pairs with corresponding tags ("Question:", "Answer:"):

A.2.2.3 Prompt Template for Generating Textbook-style Synthetic Data: Step 1, Outline Generation

- **Step 1: generate an outline based on input text.**
- **System Prompt:** Provide direct and detailed response to the instructions without adding additional notes.
- **[USER] <4 versions>:** Imagine you are a prolific author tasked with writing a textbook. You are working on writing a textbook involving the knowledge and information of the following text. Text: [xxxx]\n Your task is to write an outline for the textbook. Your target audiences are <grade school students/college students/field experts/general public>. The textbook has 10 chapters in total plus title, introduction, and appendices. Textbook outline:

A.2.2.4 Prompt Template for Generating Textbook-style Synthetic Data: Step 2, Chapter Generation

- **Step 2: generate each section based on outline.**
- **System Prompt:** Provide a direct and detailed response to the instructions without adding additional notes.
- **[USER]:** Imagine you are a prolific author tasked with writing a textbook. You are working on writing a textbook with the following outline.\n Outline: [xxxx] \n Your task is to write Chapter x of the textbook. Your target audiences are grade school students. Include exercises at the end of the chapter to test the reader's knowledge of the chapter and then provide reference answers to each question.

B EVALUATION DATASET

Unlike prior scaling law works that report training loss Hestness et al. (2017); Hoffmann et al. (2022) or test loss on a held-out validation set Kaplan et al. (2020) from training distribution, we measure upstream loss on a held-out test set from original CC as well as a diverse set of 16 non-code/math

Table 3: Model configuration parameters for different scale sizes.

Model	Hidden Dim	#Layers	#Heads	Batch Size	Grad Acc	DP	TP	#Params
100m	576	7	9	4	8	1	1	175,628,736
200m	832	10	13	4	8	1	1	298,632,256
500m	1280	16	20	4	8	1	1	653,436,160
1b	1792	22	28	4	8	1	1	1,317,616,384
2b	2240	28	35	4	8	1	1	2,292,740,800
3b	2624	32	41	2	8	2	1	3,360,234,048
4b	2816	34	44	1	8	4	1	4,103,539,968
6b	3200	40	50	1	1	32	1	5,801,833,600
8b	3648	45	57	1	1	32	1	8,122,355,904
11b	4096	51	64	2	1	16	2	11,372,228,608

English text domains from The Pile (Gao et al., 2020). Because we use scaling laws for comparative analysis across data interventions, it is critical to assess model performance on external validation sets to enable fair and meaningful comparisons across different training datasets.

C Model Design

C.1 List of Trained Models

We adopt a standard transformer-based model architecture based on LLaMA3 Grattafiori et al. (2024) for all of our scaling analysis. In Table 3 we list the model size and configuration of all models used in this study.

C.1.1 Training and Evaluation Hyperparameters

We trained all models from scratch using the *Meta Lingua* library (Videau et al., 2024) across one or multiple nodes depending on model size. We use AdamW (Kingma & Ba, 2014) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.1, paired with a cosine schedule and 10% linear warmup. All runs used a 4096-token context length, a 1M-token effective batch size, and the Llama 3 TikToken tokenizer (128k vocab) (Grattafiori et al., 2024). Table 4 and Table 5 list hyperparameters for training and evaluation. Table 3 lists local batch size, gradient accumulation, data parallelism (DP) and tensor parallelism (TP) employed for each model size. These parameters are chosen such that global batch size remains at 1M token across all experiments.

Table 4: Training Hyperparameters

Hyperparameter	Value
Optimizer	AdamW
Peak Learning Rate	$3e-4$
Min. LR Ratio	$1e-6$
Warmup Steps	10%
Gradient Clipping	1.0
Sequence Length	4096
Effective Batch Size	1M tokens
Prefetch Size	1024
Add BOS token	True
Add EOS token	True
Model Data Type	bf16
Epochs	1
GPU Hardware	NVIDIA A100 80GB

Table 5: Hyperparameters for Perplexity Evaluation

Hyperparameter	Value
Max Tokens to Generate	1024
Generator Data Type (dtype)	bf16

D Details on Scaling Analysis Setup

We perform a **joint scaling law fit** using the following parametric form:

$$L(N, D) = A \cdot N^{-\alpha} + B \cdot D^{-\beta} + E$$

We empirically estimate the parameters by fitting this function to the validation loss of over 100 models, ranging from 100M to 8B (3B) parameters and trained on 100M to 40B (200B) tokens for filtering (synthetic) interventions. We use the `scipy.optimize.curve_fit` Virtanen et al. (2020) function in Python, specifically the Trust Region Reflective (`'trf'`) optimizer Branch et al. (1999), which supports bounded, nonlinear least squares. Each datapoint is visited only once during training, consistent with standard scaling law methodology. This avoids confounding effects from data repetition and ensures fair comparison across datasets.

Curve-fitting and Initialization: The initial conditions are drawn from previous work Besiroglu et al. (2024) that challenged the assumptions used in the original Chinchilla paper Hoffmann et al. (2022). Specifically, we initialize the parameters as:

$$[A, B, \alpha, \beta, E] = [482,\ 2085,\ 0.3478,\ 0.3658,\ 1.8]$$

Parameter Count Definition: There exists inconsistency in prior work regarding whether to include embedding parameters in the total parameter count N. OpenAI's scaling law analysis Kaplan et al. (2020) excludes embedding parameters, while the Chinchilla analysis Hoffmann et al. (2022) includes them. We examined both conventions and found that the qualitative trends and conclusions remain consistent. For consistency, here we report the results using the total parameter count *including* embeddings.

E Scaling Law Fit Statistics

Table 6 reports the relative uncertainty of each scaling law parameter, computed as the ratio of the standard error to the estimated value (`std/mean`) using `scipy.optimize.curve_fit`. This metric reflects how confidently each parameter is identified by the fit: lower values indicate more stable and well-constrained estimates. Across datasets, most parameters exhibit reasonable uncertainty—typically below 0.5—suggesting that the scaling law fits are generally robust.

F Scaling Law Component Analysis

In Section 4, Figure 1, we showed the impact of a handful of data quality interventions on components of scaling law. Here we show the impact of all 23 datasets from QualityPajama benchmark suite. We group the results based on the type of interventions. We also compare the best from each group.

F.1 Heuristic Filters

To study the effect of heuristic-based data quality interventions on scaling behavior, we apply a sequence of commonly used filters, including NSFW removal, aesthetic filtering, PageRank-based filtering, deduplication at varying similarity thresholds, and grammar-based filtering via average sentence length. These filters are chosen based on their frequent use in high-quality dataset pipelines such as C4 Raffel et al. (2020), Dolma Soldaini et al. (2024), and FineWeb Penedo et al. (2024). For each filter, we evaluate its impact on the scaling law parameters by comparing the fitted values

Table 6: Normalized variability (std/mean) of scaling law components across different data interventions. Values for α and β are shown in absolute terms.

Dataset	α **(std/mean)**	β **(std/mean)**	**A (std/mean)**	**B (std/mean)**
Heuristic Filters				
orig	0.29	0.14	1.60	0.51
deduped_0.7	0.37	0.19	2.10	0.56
deduped_0.9	0.33	0.16	1.71	0.53
deduped_1.0	0.38	0.21	1.99	0.59
high_pr	0.57	0.62	2.81	1.29
long_text	0.46	0.54	2.97	1.22
low_pr	0.33	0.52	1.92	0.94
med_pr	0.38	0.49	2.45	1.02
medium_text	0.42	0.52	2.52	1.19
no_pr	0.41	0.47	2.46	1.06
nsfw	0.41	0.43	2.99	1.14
short_text	0.40	0.45	2.31	1.09
aesthetic	0.48	0.58	3.18	1.33
High-Quality Synthetic Variants (HQ / QA / CC)				
CC100	0.63	0.14	4.51	0.80
HQ100	0.63	0.16	4.93	0.79
HQ33-CC67	0.65	0.13	4.60	0.78
HQ67-CC33	0.60	0.14	4.65	0.77
QA-100	0.52	0.16	4.58	0.81
QA-33-CC67	0.58	0.12	4.54	0.77
QA-67-CC33	0.51	0.12	4.52	0.75
Textbook-style Synthetic Variants (TB)				
TB100	0.24	0.09	1.85	0.33
TB33-CC67	0.28	0.08	2.40	0.41
TB67-CC33	0.26	0.08	2.15	0.36

before and after its application, as well as across different configurations (e.g., similarity thresholds for deduplication or percentile cutoffs for PageRank). Detailed analyses are shown in Figures 7, 8, and 9.

F.2 Synthetic Data Generation

To evaluate the impact of synthetic data interventions on scaling behavior, we curate datasets using three prompting strategies: high-quality rephrasing (HQ), question-answer transformation (QA), and textbook-style rewriting (TB). These methods draw inspiration from prior work on synthetic pretraining data Maini et al. (2024); Li et al. (2023); Javaheripi et al. (2023); Abdin et al. (2024), and are applied using the Mistral-Instruct-7B model Jiang et al. (2023). We mix synthetic data with natural data at different ratios (e.g., 33% synthetic, 67% original). We fit scaling laws on these synthetic variants to analyze how text rewriting influences parameter stability and scaling behavior. Detailed results are shown in Figure 10, 11, 12, and 13.

G Limitation of Zipfian Distribution Theory

While prior work has suggested that token distribution characteristics—such as Zipfian structure could explain power law exponent's behavior, our empirical findings show that this theory may not be sufficient to explain the variation in power-law exponents. We analyzed token frequency distributions across our filtered datasets (Figure 14) and found that the Zipfian exponents are weakly negatively correlated with the model size exponent α (correlation = -0.37), and show little to no correlation with the data size exponent β (correlation = -0.005). Table 7 shows scaling exponents across datasets. In several cases, datasets with nearly identical token distributions exhibit substantially different scaling behavior. This suggests that simple distributional statistics, such as Zipfian

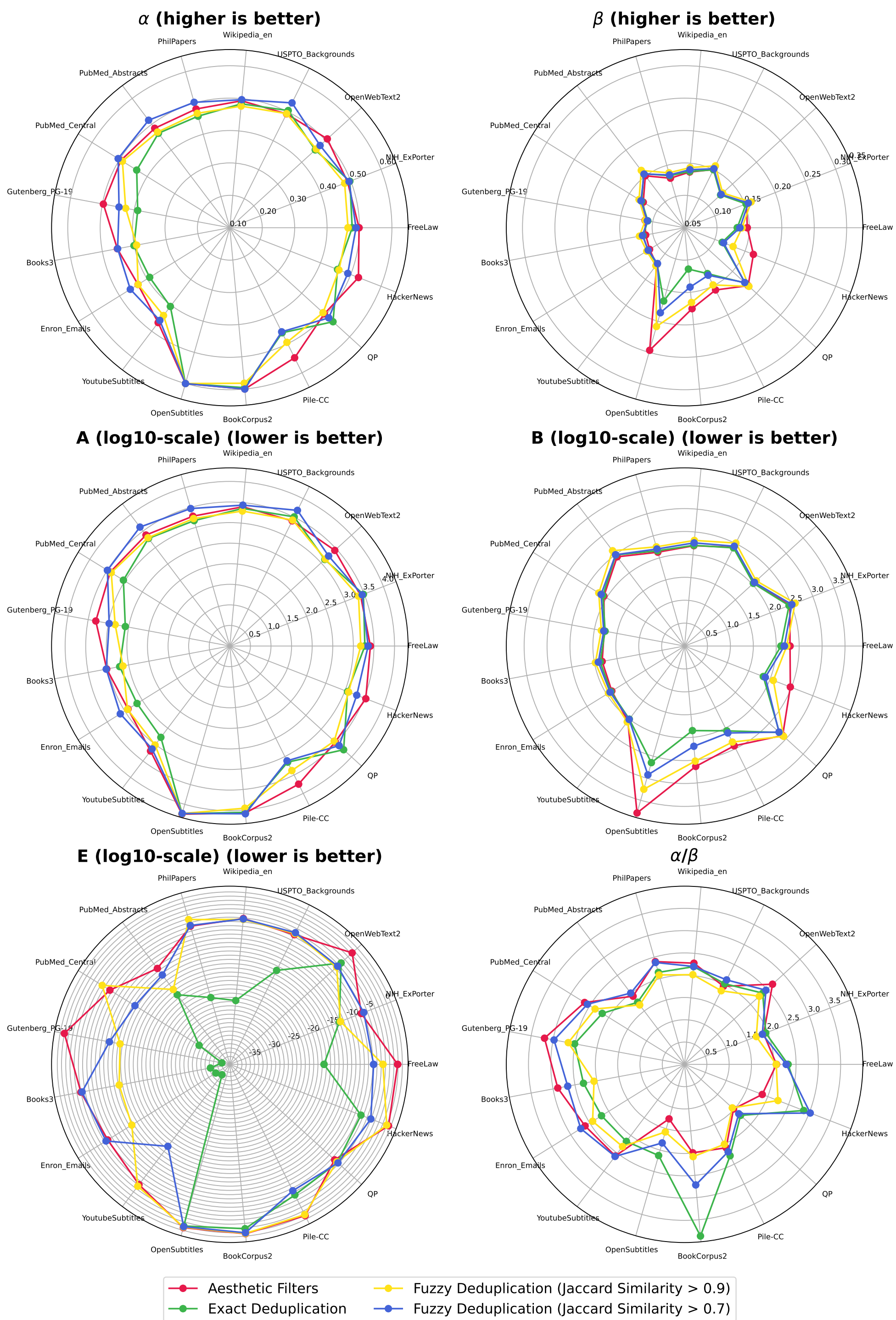


Figure 7: **How does deduplication affect scaling law components?** The red line marks the dataset before any deduplication is applied. Other lines represent deduplicated variants using different similarity thresholds.

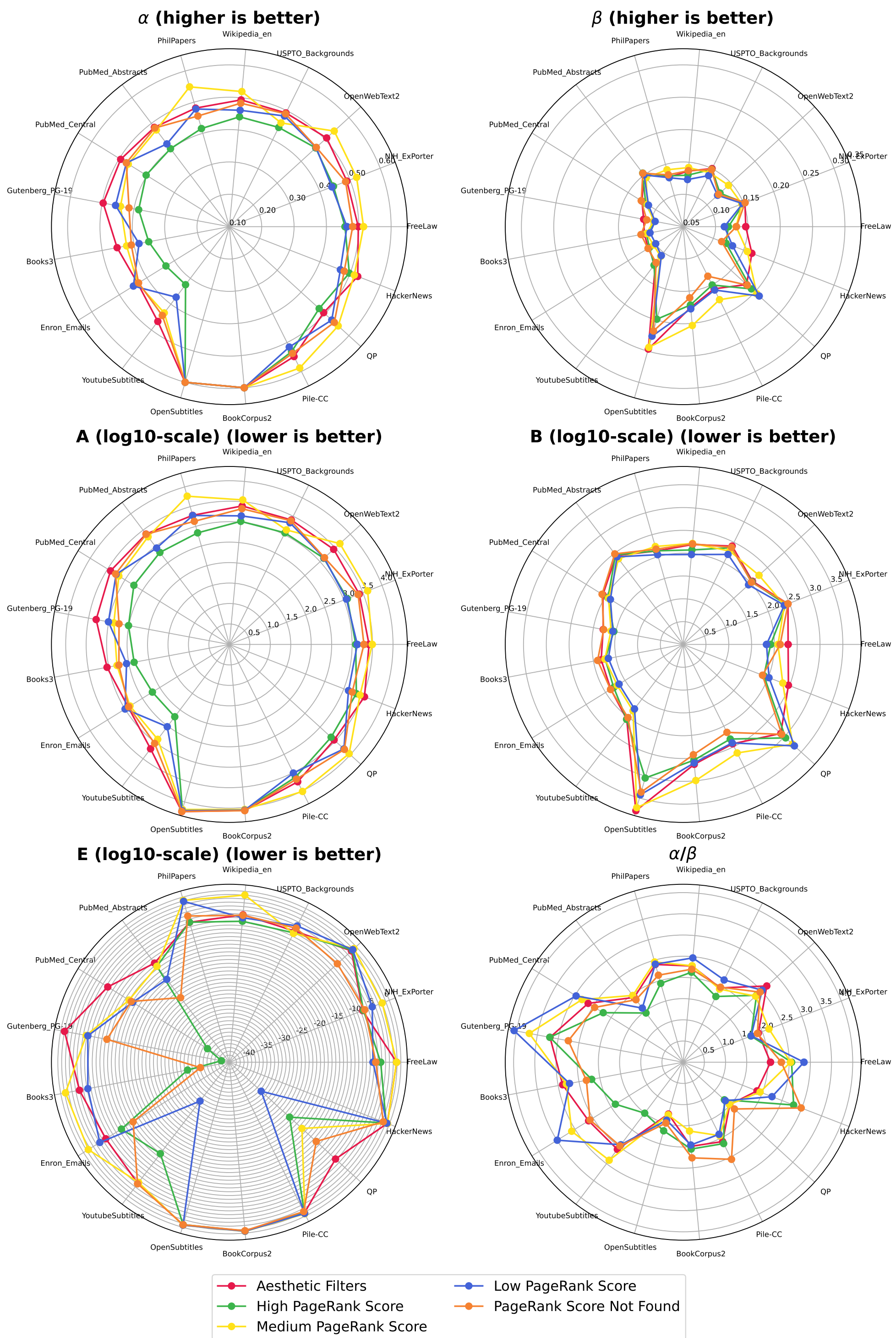


Figure 8: **How does PageRank-based filtering affect scaling law components?** The red line denotes the dataset before applying any PageRank filters. Other lines correspond to thresholds applied to the PageRank score. `Low PageRank` retains pages with scores below X, `High PageRank` retains those above Y, and `Medium PageRank` keeps pages between X and Y. `PageRank Not Found` includes pages missing from the reference PageRank table. Thresholds X and Y are set to the 33^{rd} and 67^{th} percentiles of the score distribution of pages in the PageRank table.

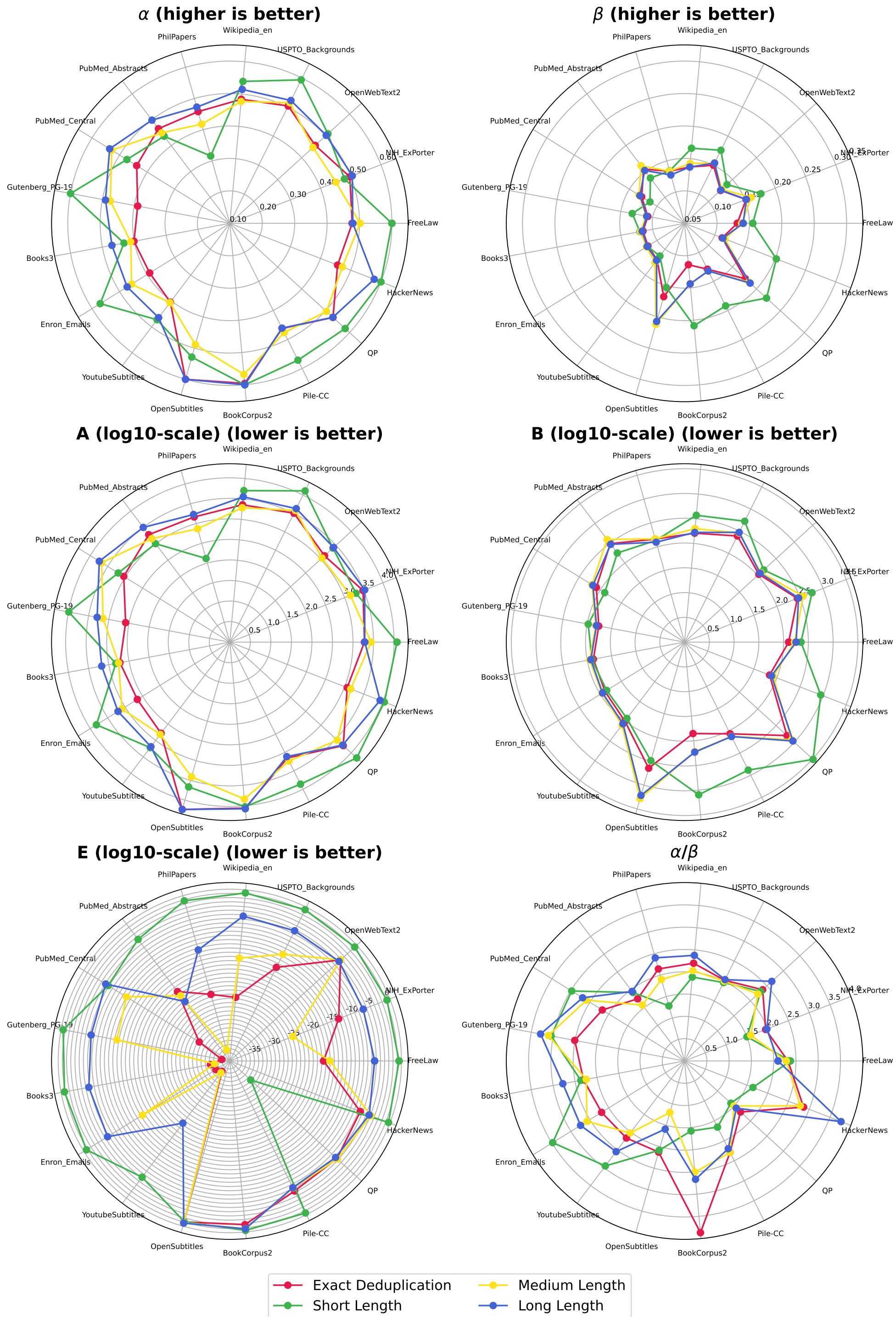


Figure 9: **How does grammar complexity (average sentence length) affect scaling law components?** The red line indicates the dataset before applying any sentence length filters. Datasets are filtered based on average sentence length, with thresholds set at 10 tokens for `short text` and 25 tokens for `medium text`.

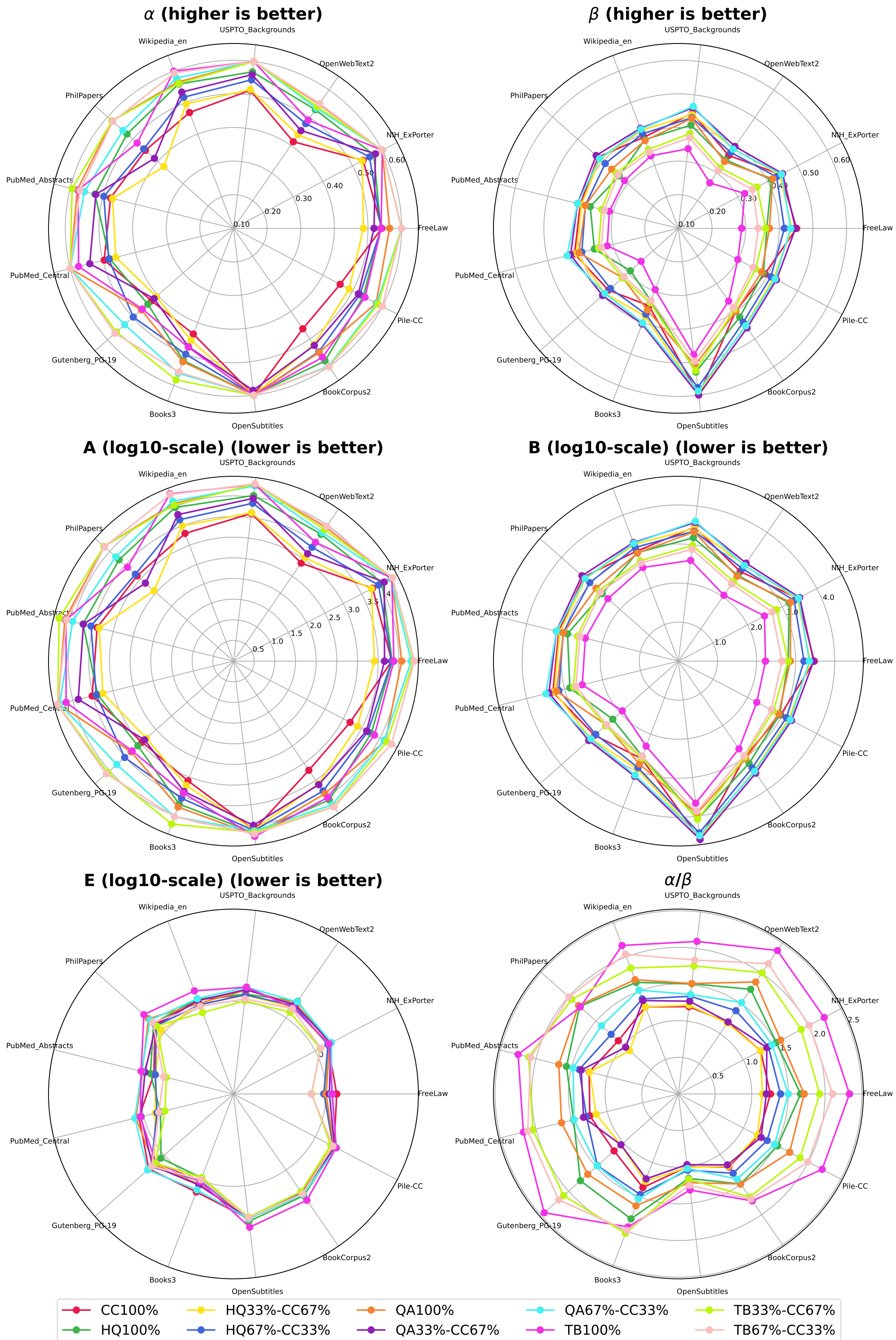


Figure 10: **How does synthetic data influence scaling law components?** Different lines show different synthetic data generation techniques and mixing ratio, and along the radial axis we have the validation set.

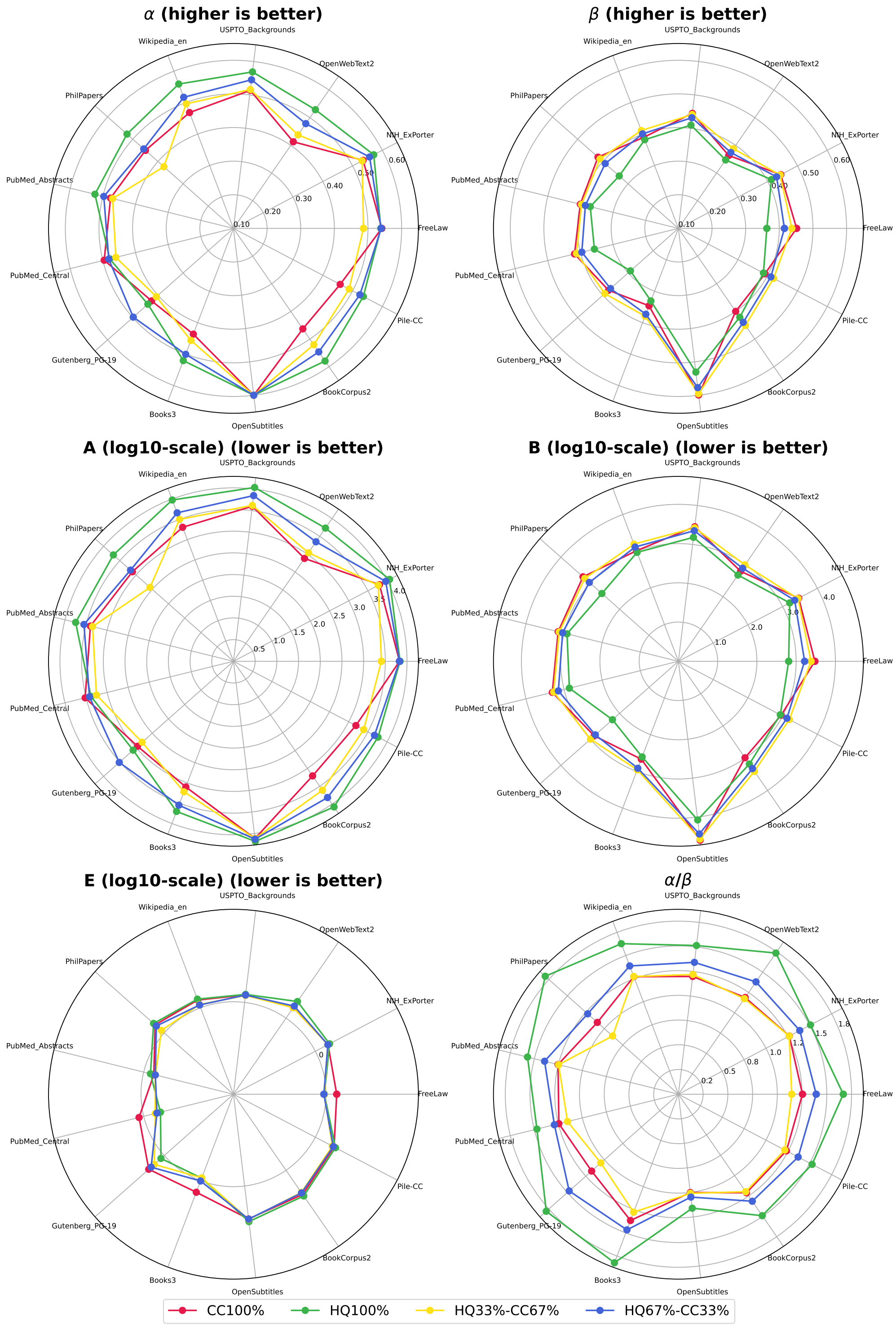


Figure 11: **How does HQ synthetic data generation influence data quality?** HQ refers to high-quality rephrasing, and CC refers to the raw natural Common Crawl dataset. HQ[N]-CC[M] refers to the mixture of synthetic and natural and N and M captures the percentage.

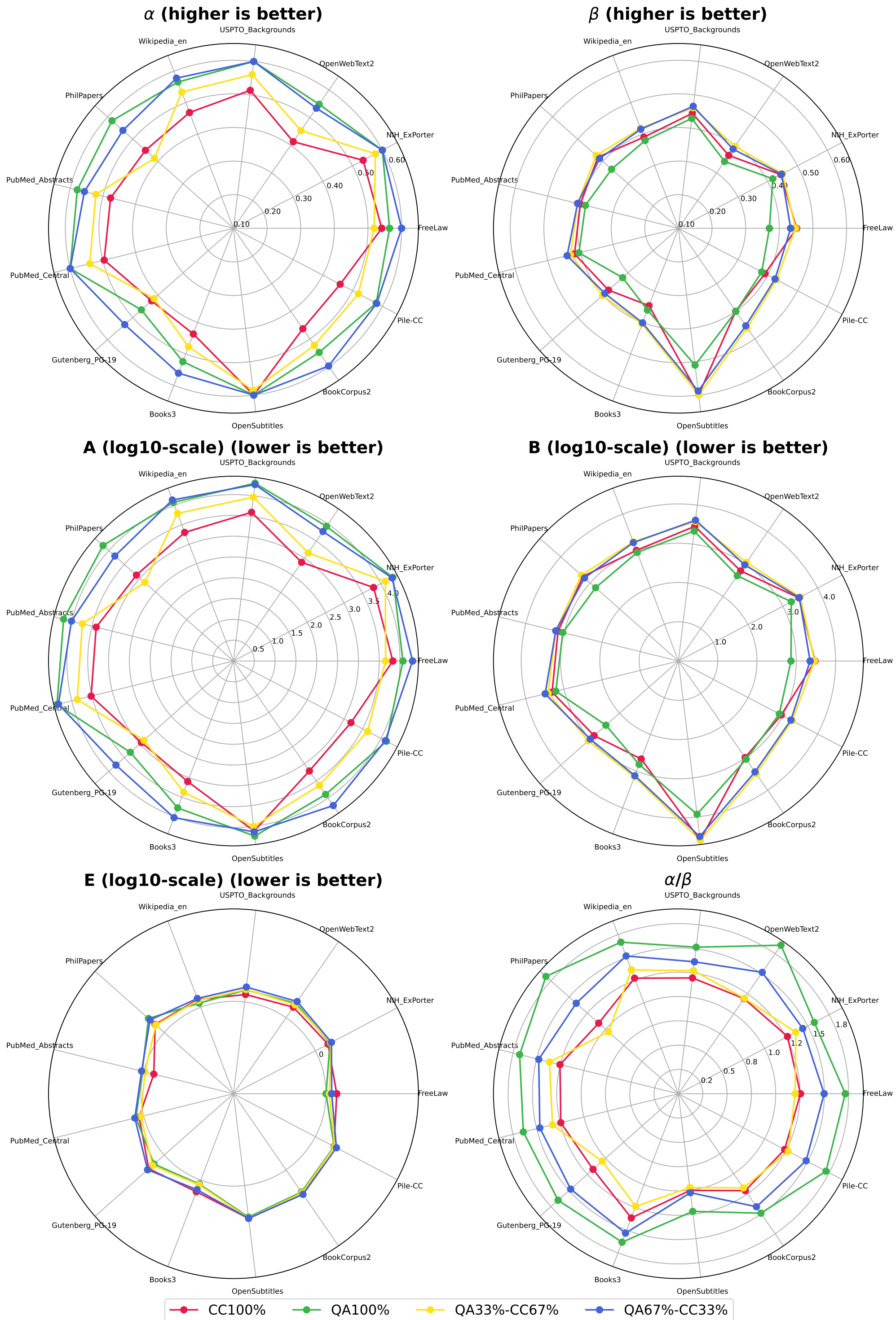


Figure 12: **How does QA synthetic data generation influence data quality?** QA refers to Question-Answering rephrasing, and CC refers to the raw natural Common Crawl dataset. QA[N]-CC[M] refers to the mixture of synthetic and natural and N and M captures the percentage.

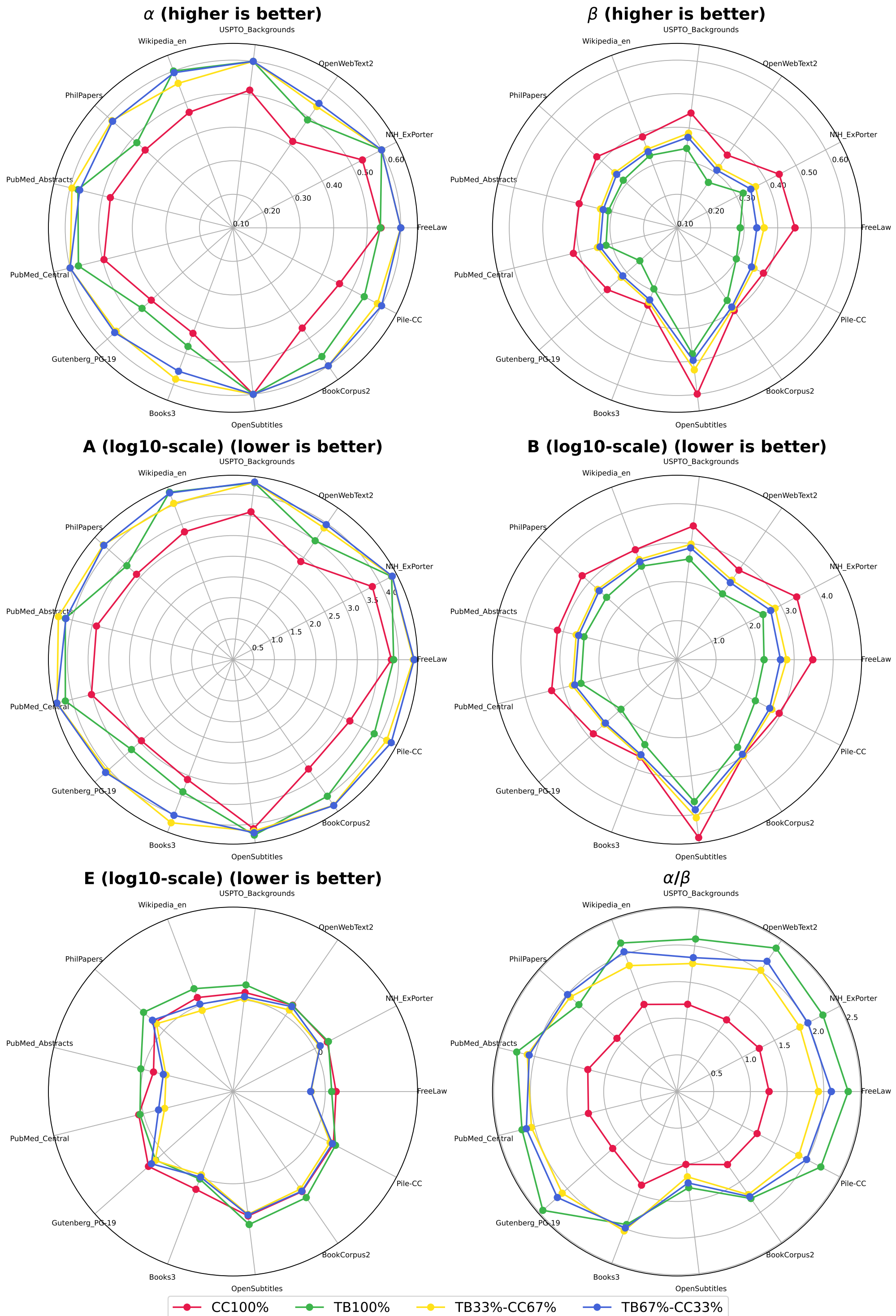


Figure 13: **How does TB synthetic data generation influence data quality?** TB refers to Textbook-style rephrasing, and CC refers to the raw natural Common Crawl dataset. TB[N]-CC[M] refers to the mixture of synthetic and natural and N and M captures the percentage.

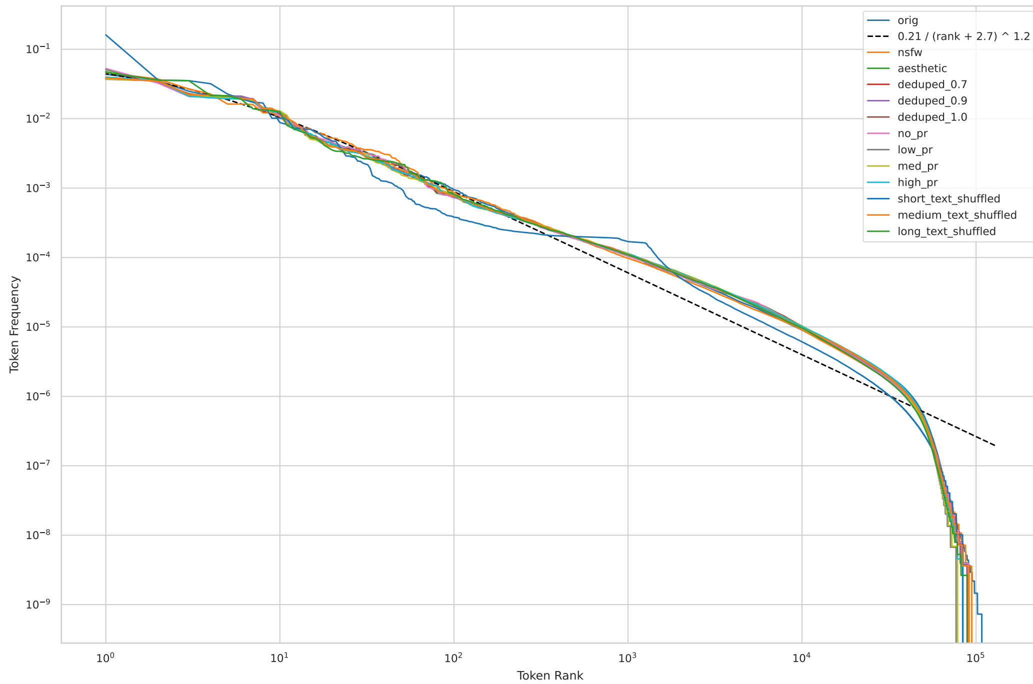


Figure 14: Token distribution across different QualityPajama datasets

exponents, fail to capture the deeper structural or semantic properties that influence scaling dynamics. It is possible that higher-order n-gram patterns or conceptual structures provide a more explanatory signal.

Filter	**Zipf Exponent (z)**	**Scaling Exponent (α)**	**Scaling Exponent (β)**
high_perplexity	1.1820	0.3509	0.2536
nsfw	1.0950	0.4341	0.1982
aesthetic	1.0907	0.4097	0.1946
deduped_0.7	1.3898	0.3633	0.2173
deduped_0.9	1.3938	0.3392	0.2393
deduped_1.0	1.1638	0.3499	0.2093
no_pr	1.0599	0.3884	0.1855
low_pr	1.3943	0.3772	0.1805
med_pr	1.3744	0.4157	0.1982
high_pr	1.2625	0.3499	0.1826
medium_text	1.3541	0.3740	0.1885
long_text	1.2944	0.4106	0.1833

Table 7: Zipf exponent z and scaling law exponents α and β across different data filters.

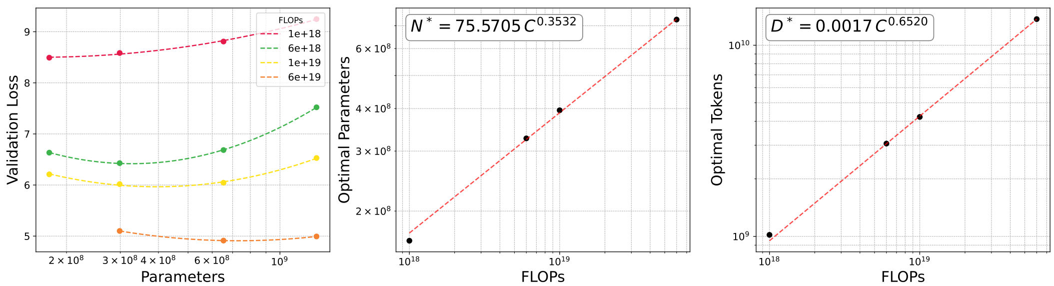
H Validating Scaling Law Fits

Hoffmann et al. Hoffmann et al. (2022) propose three distinct approaches for estimating scaling law components. The first approach holds model size constant while varying the number of training tokens. The second approach uses isoFLOP curves to identify the compute-optimal design point—that is, the configuration within each isoFLOP family that minimizes loss. The third approach involves fitting a parametric loss function to observed data.

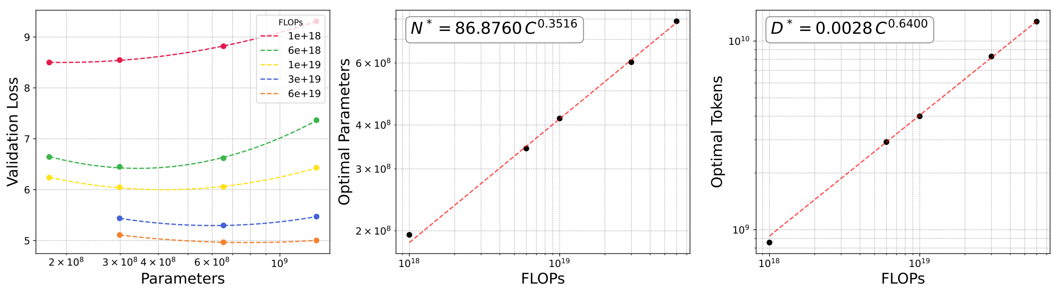
In this work, we primarily use the parametric loss function throughout our analysis. However, we also conduct a limited set of experiments to generate isoFLOP curves for a subset of our datasets, enabling a comparative evaluation. To validate our parametric fits, we compare them against the predictions obtained from the isoFLOP profiles.

The isoFLOP approach predicts scaling exponents a and b, where $N^* = A \cdot C^a$ and $D^* = B \cdot C^b$, and connects to the parametric loss function components via the relationships $a = \frac{\beta}{\alpha+\beta}$ and $b = \frac{\alpha}{\alpha+\beta}$.

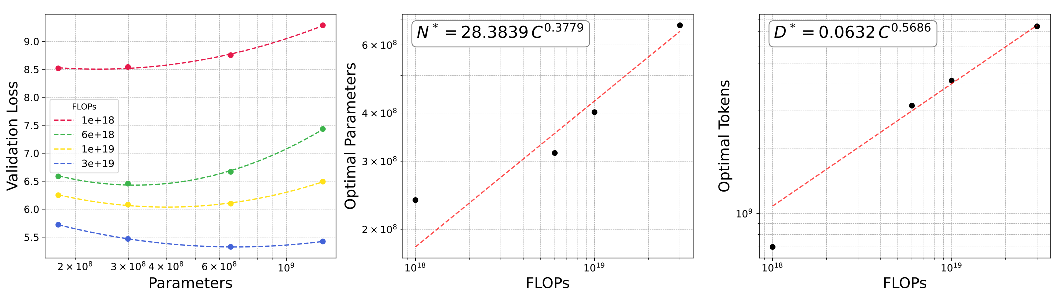
Figures 15–17 show isoFLOP curve fittings against validation loss for several validation sets, each corresponding to a different dataset. In each figure, the parametric form estimates of the scaling law components are reported in the caption for comparison. The average absolute relative error between isoFLOP curve vs. parametric fit estimation of a and b is 0.21 and 0.24.



(a) deduped_0.7



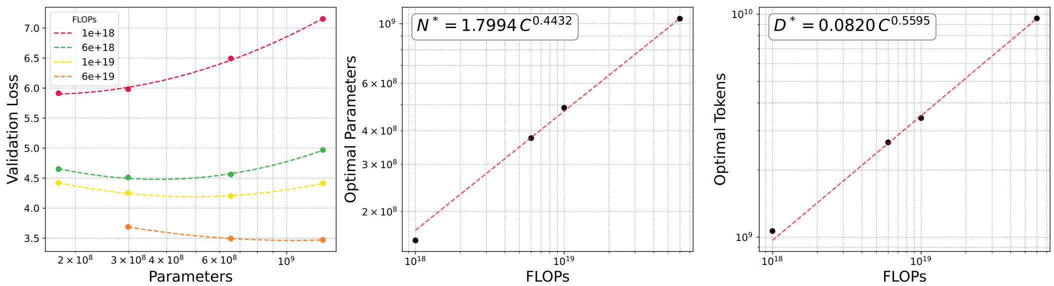
(b) deduped_0.9



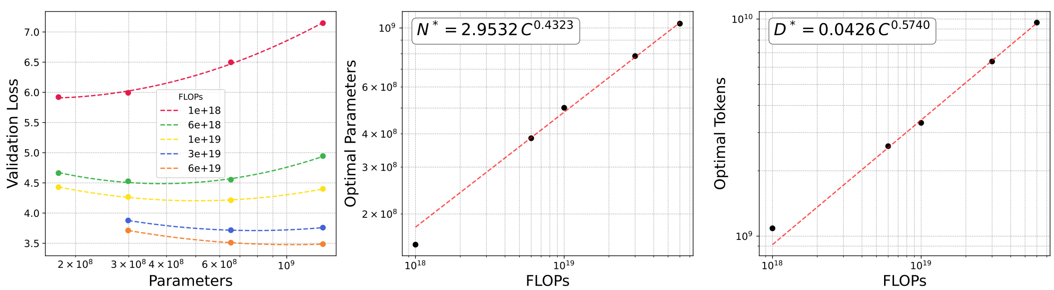
(c) deduped_1.0

Figure 15: **IsoFLOP Curve Approach** applied across different training sets. Validation loss is evaluated on the *ArXiv* subset from the Pile dataset. The parametric (vs. isoFLOP) estimates of the exponent a are 0.3322 (vs. 0.3532), 0.3738 (vs. 0.3516), and 0.3561 (vs. 0.3779) for (a), (b), and (c), respectively.

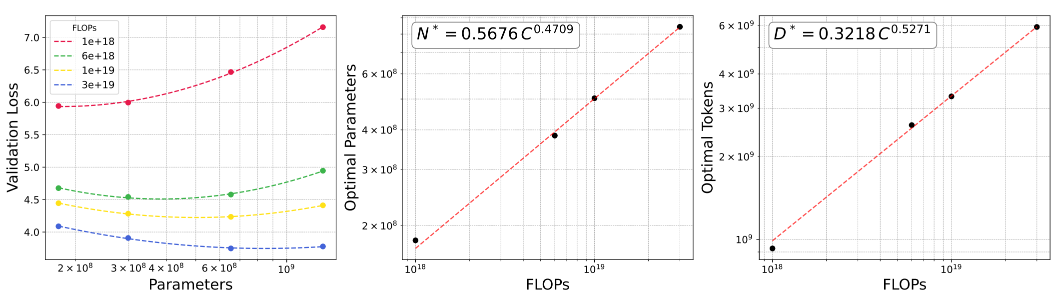
.



(a) deduped_0.7



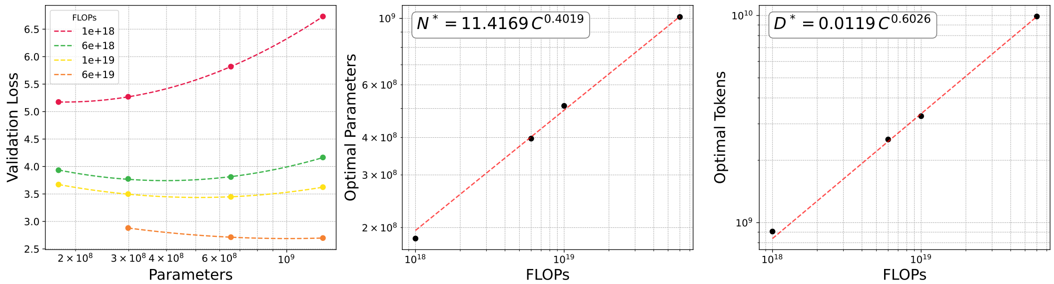
(b) deduped_0.9



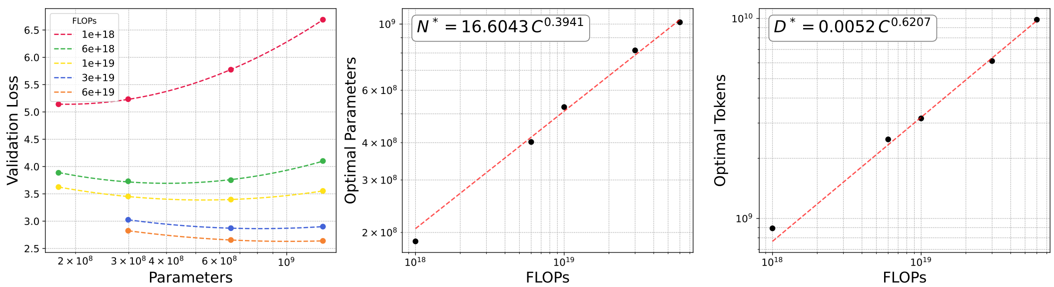
(c) deduped_1.0

Figure 16: **IsoFLOP Curve Approach** applied across different training sets. Validation loss is evaluated on the *FreeLaw* subset from the Pile dataset. The parametric (vs. isoFLOP) estimates of the exponent a are 0.3213 (vs. 0.4432), 0.3623 (vs. 0.4323), and 0.3175 (vs. 0.4709) for (a), (b), and (c), respectively.

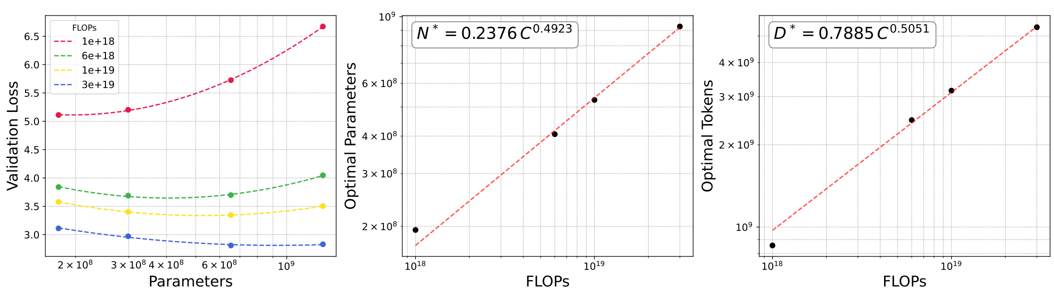
.



(a) deduped_0.7



(b) deduped_0.9



(c) deduped_1.0

Figure 17: **IsoFLOP Curve Approach** applied across different training sets. Validation loss is evaluated on the *CC* held-out test set. The parametric (vs. isoFLOP) estimates of the exponent a are 0.429507 (vs. 0.401894), 0.471800 (vs. 0.394094), and 0.424924 (vs. 0.492310) for (a), (b), and (c), respectively.

.

I Beyond Chinchilla: Step-by-Step Exploration Towards a Cheap, Robust Scaling-Law Form

Classical neural scaling-law formulations implicitly assume that the training and evaluation data are drawn from the same underlying distribution. In practice, however, scaling laws are most often used in settings where the train and test distributions differ—sometimes greatly. As data curation pipelines evolve or filtering strategies become more aggressive, the training distribution can drift substantially away from the downstream evaluation distributions for which we ultimately make predictions. This mismatch violates the assumptions of standard scaling-law forms and directly impacts parameter identifiability.

A second major pain-point arises from the requirement of observing high-compute datapoints in order to accurately estimate the irreducible loss floor E. When the data does not extend far enough into the large-N/large-D regime, the asymptotic term becomes weakly identified. In these regimes, the optimizer often compensates by pushing E toward zero and allowing the amplitude terms A and B to absorb what should be the irreducible component.

In this report, we document our step-by-step journey toward discovering a more robust scaling-law form with limited datapoints, beginning with failure modes in the baseline Chinchilla formulation and describing the empirical insights that guided each modification.

Methodology: We evaluate scaling law forms across 299 train-validation pairs, formed by the Cartesian product of 23 training sets from QualityPajama and 13 validation sets from the Pile. For each train-validation pair, we have around ∼40 datapoints with ∼9×14 $N \times D$ coverage, with model sizes ranging from 20M–3B parameters (∼2 orders of magnitude) and data sizes ranging from 100M–38B tokens (2.6 orders of magnitude).

Measure of Success: We define a composite quality score (0–100 points) that evaluates each scaling-law model across four key dimensions: (1) **Parameter Stability** (40 points) measures the fraction of train-validation pairs where each parameter's coefficient of variation (CV) is below 1.0, ensuring reliable parameter estimates; for 7-parameter models, the 8 points normally assigned to A and B are split evenly between their amplitude (A_0, B_0) and exponent (γ) components. (2) **E-Collapse Avoidance** (20 points) rewards models whose irreducible error satisfies $E \geq 0.1$, penalizing the pathological $E \to 0$ solutions that affects standard chinchilla formulation. (3) R^2 **Performance** (20 points) scores out-of-bag predictive accuracy on a linear scale, with $R^2 = 1.0$ receiving full credit. (4) **RMSE Performance** (20 points) evaluates absolute prediction error, normalized between the best and worst RMSE observed across all models to reward lower error. This balanced rubric ensures that scaling-law models achieve stable parameter estimates, avoid degenerate solutions, and maintain strong predictive accuracy. Underperformance in any dimension substantially reduces the overall score.

Table 8: Step-by-step exploration toward a more robust scaling law. The table reports the catastrophic failure rate for each parameter under different fitting techniques. For all parameters except E, catastrophic failure is defined as the fraction of train-validation pairs whose coefficient of variation exceeds 1 (CV > 1). For E, catastrophic failure is defined as the fraction of pairs with $E < 0.1$. The blue column corresponds to the original Chinchilla-style approach, and the green column highlights our final winning recipe. * in-sample results for MCMC.

Parameter	MLE bootstrap (Baseline)	MCMC	MLE + E Regularization (ER)	MLE + ER + AB scale dependent	MLE + ER + AB Distance-Modulated)
A	14%	95%	66%	1%	3%
B	9%	41%	11%	0%	0.7%
α	7%	0%	0%	0%	0%
β	0%	0%	0%	0%	0%
γ_A	—	—	—	15%	16%
γ_B	—	—	—	3%	3%
E	59%	30%	3%	0%	0%
OOB R^2	0.86	−27*	0.90	0.97	0.97
OOB RMSE	0.37	5.3*	0.39	0.20	0.20
Quality Score	64	−482	71	98	97

STEP 1. CHINCHILLA BASELINE FAILURE: ASYMPTOTIC FLOOR COLLAPSE IN THE CHINCHILLA FORM

Using the standard 5-parameter Chinchilla model, we observe that 59% of train-validation pairs produce near-zero asymptotes (i.e., $E < 0.1$). This is incompatible with theory. We expect that when train and test distributions diverge, the asymptotic floor be strictly positive and approximated by $E = H_{\text{val}} + \text{KL}(\text{val}\|\text{train})$.

This collapse indicates that, without additional information, the model prefers to "explain away" the asymptote by reallocating mass into A and B. The failure is most severe when train and validation distributions diverge and we lack high-compute datapoints to anchor the asymptote.

STEP 1.1 EXPLORING ALTERNATIVE BASELINE: MCMC BAYESIAN INFERENCE

We also explored MCMC bayesian inference. While this approach eliminates the E collapse problem, we found it has markedly worse performance compared to the baseline MLE bootstrap approach and it is very unstable estimates for A, B and E:

- 98% of train-validation pairs have catastrophic A failures ($A_{\text{CV}} > 1$)
- 41% of train-validation pairs have catastrophic B failures ($B_{\text{CV}} > 1$)
- 30% of train-validation pairs have catastrophic E failures ($E_{\text{CV}} > 1$)

STEP 2. FIXING E: DISTANCE-DEPENDENT REGULARIZATION

To correct the asymptotic floor collapse, we introduce a distance-dependent regularization that anchors E toward the theoretical floor:

- When train and validation distributions are similar, we use weak regularization and trust the data to identify E
- When the distributions diverge, regularization strength increases smoothly toward the theoretical floor.

This removes the asymptotic collapse:

- E-collapse reduces from 59% $\rightarrow$ 3%

However, it introduces a new and unexpected failure:

- Catastrophic A failures (CV > 1) jump from 14% $\rightarrow$ 66% (4.5$\times$ worse)

This reveals a key insight: Forcing E to stay non-zero shifts the burden of explaining curvature onto A and B, causing them to deform unnaturally. This tells us the Chinchilla form lacks sufficient flexibility to balance the interaction between E, A, and B under distribution shift.

For brevity, we only describe the final winning approach here, though we explored several alternatives for regularizing E, including hard constraints, regularization around different targets (e.g., the E estimated via our MCMC fits), and variants with fixed regularization strength.

STEP 3. 1D SLICED FITS SHOW THAT A AND B ARE SCALE-DEPENDENT

Inspired by observations in prior work, we revisited 1D sliced scaling-law fit dynamics. For each train-validation pair:

- We fit $L(N)$ at fixed D to estimate A and α.
- We fit $L(D)$ at fixed N to estimate B and β.

These sliced fits revealed that the Chinchilla functional form is too rigid over the N-D plane. Within a single train-validation pair, where the model assumes that parameters are constant across all values of model size and data size, we observe systematic drift:

- A and α vary consistently as D changes ($R^2 \approx 0.5$).
- B and β vary consistently as N changes ($R^2 \approx 0.3$).

We also found clear correlations between train-validation distance and parameter instability:

- B_{CV} correlates with distance ($\rho = -0.34$, $p = 10^{-9}$), suggesting that reliability of B declines as distributions diverge.
- Similarly α_{CV} and β_{CV} correlate with distance ($\rho \approx -0.40$, $p < 10^{-12}$), suggestion that overall calibration worsens under domain shift.

These diagnostics suggested that the empirical loss surface contains scale-dependent curvature and the rigid Chinchilla structure is not sufficient once the distribution shift becomes moderate or large.

Guided by these observations, we explored two families of extensions:

1. Fully scale-dependent parameters:
   - $A(D)$, $\alpha(D)$
   - $B(N)$, $\beta(N)$
2. Distance-modulated scale dependence:
   - $A(D, s)$, $\alpha(D, s)$
   - $B(N, s)$, $\beta(N, s)$
   - with s derived from JS divergence

However, we found that allowing the exponents to become scale-dependent resulted in unstable fits: each train-validation slice contains only ∼30–50 points, which is insufficient to reliably estimate 9+ parameters. Moreover, our E-regularized approach already yields stable estimates for α and β. We also observed that incorporating distance into $A(D, s)$ and $B(N, s)$ provided no measurable benefit and only increased the complexity of the formulation. For these reasons, we revert to the simpler $A(D)$ and $B(N)$ parameterization.

In contrast, allowing only the amplitude terms to depend on scale is well supported by the data. The simplest and most stable choice is:

- $A(D)$ varies with D
- $B(N)$ varies with N
- α and β remain constant within each pair

This preserves identifiability and uses only 7 parameters.

The resulting model is:

$$L(N, D) = A_0 \cdot D^{\gamma_1} \cdot N^{-\alpha} + B_0 \cdot N^{\gamma_2} \cdot D^{-\beta} + E \tag{1}$$

where:

$$\begin{aligned} A(D) &= A_0 \cdot D^{\gamma_1} \quad \text{\# amplitude changes with } D \\ B(N) &= B_0 \cdot N^{\gamma_2} \quad \text{\# amplitude changes with } N \\ &\alpha \text{ and } \beta \text{ remain constant within each pair} \end{aligned}$$

Parameters (7 total): A_0, γ_1, α, B_0, γ_2, β, E

This minimal extension captures the scale-dependent behavior observed in the 1D sliced fits while staying within the parameter budget supported by the available datapoints. Results are listed in Table 8.