Loss-to-Loss Prediction: Language model scaling laws across datasets

Anonymous Author(s)
Affiliation
Address
email

Abstract

While scaling laws provide a reliable methodology for predicting train loss across compute scales for a single data distribution, less is known about how to predict losses across distributions. In this paper, we derive a strategy for predicting one loss from another and apply it to predict across different pre-training datasets and from pre-training data to downstream task data. Our predictions extrapolate well even at 20x the FLOP budget used to fit the curves. More precisely, we find that there are simple shifted power law relationships between (1) the train losses of two models trained on two separate datasets when the models are paired by training compute, and (2) the train loss and the test loss on any downstream distribution for a single model. The results hold up for pre-training datasets that differ substantially (some are entirely code and others have no code at all) and across a variety of downstream tasks. Finally, we find that in some settings the shifted power law relationships can yield substantially more accurate predictions than extrapolating single-dataset scaling laws.

1 Introduction

Scaling laws [Kaplan et al., 2020, Hoffmann et al., 2022] have become a reliable tool for extrapolating model performance (as measured through, e.g., cross-entropy loss on held-out data), as well as a way to determine optimal model size given a FLOP budget [Llama 3 Team, 2024]. However, relatively little is known about how losses relate across different pretraining distributions, and from training data to downstream data. For example, how can a practitioner who fit a scaling law for a model trained on FineWeb estimate the model's performance on a different pretraining corpus, such as SmolLM or on a downstream task such as MMLU? And how would changing the pre-training dataset to FineWeb-edu instead change the results?

In this paper, we take a first pass at answering these questions. In particular, we observe two types consistent loss-to-loss relationships. First, when models that are trained on different training datasets are paired by training compute there is a shifted power law that relates the two losses. This has implications for the functional form of the scaling law as well as how this scaling law varies across datasets. Namely, the exponents and constants vary together in a structured way. Second, we consider train-to-test transfer where a model trained on one dataset is evaluated on a different dataset. Again, we find that a shifted power law is predictive (although with a slightly different shift). This is true both when evaluating transfer to validation loss on different pre-training datasets and when evaluating cross entropy loss on downstream tasks. These results have implications for data selection and understanding the predictable trends that underlie emergent behavior. Finally, for all these loss-to-loss relationships we find strongly predictive extrapolation to 20x the FLOP budget than was used to fit the curves.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

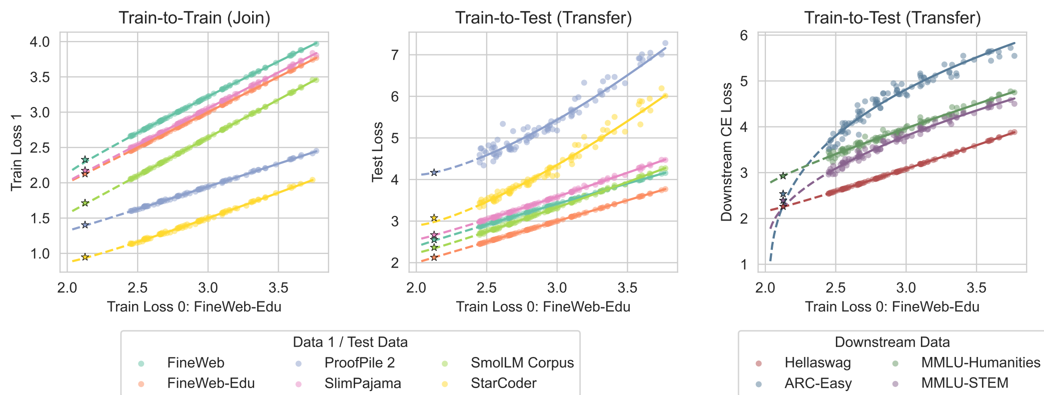

Figure 1: (Left) Train-to-train prediction from FineWeb-edu to all 6 training sets. Each datapoint represents a pair of models that are "joined" on model size N and dataset size D. Dashed lines represent extrapolation and stars represent 3.3B models trained with 20x compute of the largest dot that are *not* used to fit the curves. (Center) Train-to-test prediction from FineWeb-edu to all 6 validation sets. Each datapoint represents a single model and its "transfer" performance on the val data. (Right) Train-to-test prediction from FineWeb-edu to four downstream tasks. Loss on downstream tasks is the cross entropy loss of the correct answer to the multiple choice problem when phrased as a cloze task.

2 Related work

2.1 Scaling laws

Standard approaches to scaling laws attempt to fit a curve to the optimal number of model parameters N and training tokens D to minimize the *pre-training loss* under a given budget of FLOPs [Hestness et al., 2017, Kaplan et al., 2020, Hoffmann et al., 2022, Porian et al., 2024, Abnar et al., 2021, Maloney et al., 2022, Bordelon et al., 2024].

To fit these curves, it is useful to specify a parametric form of the loss in terms of N and D. Hoffmann et al. [2022] assumes this curve takes the following form:

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

This formula is inspired by classical upper bounds on a loss decomposition that attributes error to Bayes risk (entropy), approximation error (from having finite parameters), and estimation error (from having finite data) [Bottou and Bousquet, 2007].

On the other hand Kaplan et al. [2020] instead assumes that:

$$L(N, D) = \left(\left(\frac{A}{N} \right)^{\alpha/\beta} + \frac{B}{D} \right)^\beta \tag{2}$$

Below, we will advocate for a slightly different functional form that blends the two of these.

Regardless of the functional form, scaling laws have been an integral part of the success of modern neural language models. Our work builds on the ideas originated in this line of work and extends them to consider how to translate scaling laws across data distributions.

2.2 Scaling laws for transfer and downstream tasks

Scaling laws for pre-training loss are useful as a proxy to guide pre-training, but we ultimately care about downstream task performance. Prior work attempting to tackle this issue has found that directly computing hard metrics like accuracy can lead to the appearance of emergent behaviors and suggests using softer metrics like cross entropy loss instead [Schaeffer et al., 2024a,b]. This is corroborated by Du et al. [2024] which notes that while downstream accuracy can vary smoothly with training loss at some points in the curve, the hardness of the accuracy metric means that no progress in accuracy above random chance will be observed until some "emergent" loss level.

On the other hand, [Gadre et al., 2024] claims that downstream accuracy can be predicted as a function of training loss with a similar exponential curve to the one we propose for predicting downstream loss. However, they only claim this is predictable when averaging over many tasks and carefully selecting which tasks to use. In this paper when considering downstream tasks we focus on single downstream tasks and find loss to be a more stable downstream metric than accuracy. A more detailed discussion of loss versus accuracy is in Appendix B.

Another related line of work comes from the distributional robustness literature on "accuracy on the line" [Miller et al., 2021, Tripuraneni et al., 2021, Awadalla et al., 2022]. This phenomena focuses on the relationship between the accuracy of a single model across two closely related tasks, like different versions of imagenet, and finds that accuracy on one will predict accuracy on the other. We consider loss rather than accuracy, language modeling rather than vision, and find non-linear fits.

Note, in this work we focus on zero shot transfer where there is no finetuning on the target task. Prior work on "transfer scaling laws" focuses instead on a finetuning setting [Hernandez et al., 2021, Abnar et al., 2021, Isik et al., 2024], which is interesting, but beyond the scope of this work.

3 Setting

3.1 Notation

We are interested in studying transfer across different training distributions. To formalize this, we will two distributions: P_0 and P_1. We will consider P_0 as the "source" and P_1 as the target. The goal is to use a function of the loss on P_0 to predict the loss on P_1. As an example, P_0 could be FineWeb and P_1 could be Starcoder or Hellaswag. We use L_i to indicate the loss calculated on distribution P_i (averaged per-token). If P_1 represents a multiple choice task, we will let L_1 be the loss of correct answer when the question is phrased as a cloze task (following [Schaeffer et al., 2024b, Madaan et al., 2024]) and let Err_1 be the multiple choice error (i.e. 1 - accuracy).

Given a pre-training distribution P_i, we let $\hat{f}_i^{N,D}$ denote an N parameter model trained on D tokens sampled from P_i. Our results present comparisons across losses L_0, L_1 for models $\hat{f}_0^{N,D}$, $\hat{f}_1^{N,D}$ when sweeping across different choices of P_0, P_1, as well as N, D.

When we refer to a scaling law fit from Equation (3) on distribution P_i, we will append a subscript to the corresponding parameters. For example, the irreducible entropy of the scaling law fit on P_0 is denoted by E_0.

3.2 Experimental methodology

To facilitate our analysis, we pre-train models of varying size with varying flop budgets on 6 pre-training datasets: FineWeb [Penedo et al., 2024], FineWeb-edu [Penedo et al., 2024], Proof Pile 2 [Azerbayev et al., 2023, Computer, 2023, Paster et al., 2023], SlimPajama [Soboleva et al., 2023], SmolLM Corpus [Ben Allal et al., 2024], and Starcoder v1 [Li et al., 2023]. We train all models using OLMo [Groeneveld et al., 2024] and generally follow hyperparameter settings from Wortsman et al. [2023], Zhao et al. [2024]. Full hyperparameters can be found in Appendix F. Importantly, we use a linear warmup and cosine decay schedule for every run and only report the final performance [Porian et al., 2024].

FLOP budgets for our sweep range from 2e17 to 4.84e19 and model sizes range from 20M to 1.7B. The optimal model at the largest FLOP budget is roughly 750M (it varies per dataset). The total grid contains 528 models, or 88 models per dataset. For our extrapolation experiments, we train 6 larger models (one for each dataset) at a FLOP budget of 1e21 each of size 3.3B. Full scaling law fits illustrating all runs can be found in Appendix D and Appendix E.

4 Predicting loss across datasets

In this section, after a brief discussion of the functional form of scaling laws, we present the two main loss-to-loss relationships that we observe in this paper: train-to-train and train-to-test.

4.1 Functional form of the scaling law

There are two key differences between Equation (1) and Equation (2):

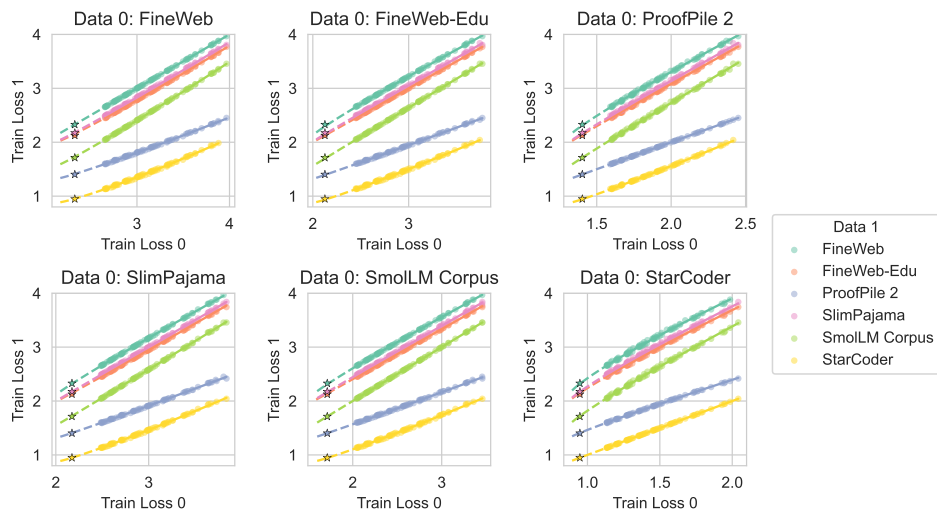

Figure 2: Train-to-train fits. Each point on the plot represents the final loss of two models: $\hat{f}_0^{N,D}$ which is trained on dataset 0 and $\hat{f}_1^{N,D}$ which is trained on dataset 1. The models are paired when they use the same number of parameters N and tokens D.

1. Equation (1) includes the irreducible entropy of the training distribution
2. Equation (2) potentially includes cross terms that depend on both N and D.

In this work, we will incorporate both of these differences to create a third, slightly different functional form. This gives us the following form:

$$L(N, D) = E + \left(\left(\frac{A}{N} \right)^{\alpha/\beta} + \frac{B}{D} \right)^{\beta} \tag{3}$$

Full fits of these scaling laws can be found in Appendix D and they generally fit the data well.

4.2 Train-to-train prediction

Our first main result is to observe a consistent scaling relationship between train losses across datasets. Explicitly, we find that by fitting just two parameters K and κ we can capture and extrapolate the scaling relationship between pairs of training losses as follows:

$$L_1(\hat{f}_1^{N,D}) \approx K \cdot \left(L_0(\hat{f}_0^{N,D}) - E_0 \right)^{\kappa} + E_1 \tag{4}$$

Note, this is comparing *different* losses and *different* models, but the models are pairs since they each have N parameters trained on D tokens. Also, recall that E_0, E_1 are the irreducible errors from *independent* scaling law fits on P_0 and P_1 respectively. Finally, note that since we are only fitting a slope and exponent, each curve is linear on a shifted log-log scale. However, since we are plotting 6 curves in one plot, each with different E_1, we cannot display them all consistently log-log plot and opt for a linear scale. Results for fitting these curves can be seen in Figure 2.

Implications. Train-to-train prediction mainly has implications into how the scaling laws relate to each other across pre-training datasets when we use the same model family and learning algorithms. For example:

- When we change data distributions, β changes, but the ratio of exponents α/β remains constant. Moreover, the numerator constants A, B vary together as we change the distribution.
- Equation (3) is the only formulation of the underlying scaling law that is compatible with the train-to-train fit given by Equation (4).

We should also note that the exponents κ

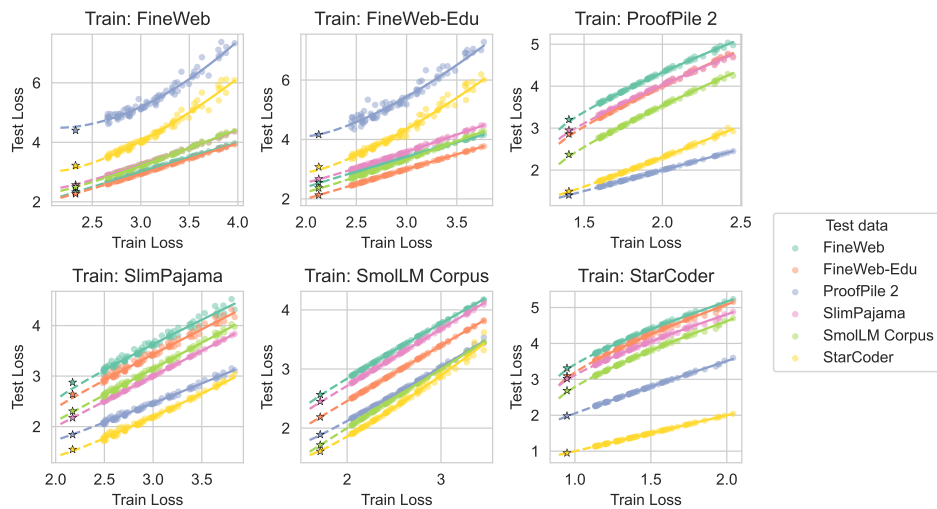

Figure 3: Train-to-test fits. Each datapoint represents a single model trained on the dataset in the subplot title and then evaluated on a different dataset as indicated by the color.

4.3 Train-to-test prediction

Next, we want to go beyond the train loss and consider translating the train loss to a test loss for the same model under a different distribution. The

$$L_1(\hat{f}_0^{N,D}) \approx K \cdot \Big(L_0(\hat{f}_0^{N,D}) - E_0\Big)^{\kappa} + E_{1|0} \tag{5}$$

Note, this is comparing *different* losses, but the *same* model. Further, note that we define $E_{1|0}$ to be the irreducible error of L_1 for the optimal function on P_0 with infinite model and data sizes:

$$E_{1|0} := L_1(f_0^*) \tag{6}$$

We can estimate this quantity by fitting a scaling law to L_1 under data from P_0.

Results in Figure 3 show prediction across validation sets from the pre-training distributions. Results in Figure 4 translate from train-to-downstream where we use downstream multiple choice questions. Following [Schaeffer et al., 2024b, Madaan et al., 2024], we evaluate the downstream tasks by the cross entropy loss on the correct answer when the question is phrased as a cloze task. Here we show results for Hellaswag [Zellers et al., 2019], ARC-Easy [Clark et al., 2018], and a subset of MMLU [Hendrycks et al., 2020], further results for ARC-Challenge, Openbook QA [Mihaylov et al., 2018], PIQA [Bisk et al., 2020], SciQ [Welbl et al., 2017], Winogrande [Sakaguchi et al., 2021], and the rest of MMLU are in Appendix C.

Note that Kaplan et al. [2020] points out a similar trend to Figure 3 in Figure their Section 3.2.2, but they only consider transfer to wikipedia and books and assume the relationship to be linear. By considering a broader array of datasets, we are able to see a more nuanced picture of transfer.

Implications. Train-to-test prediction has several implications:

- The predictions across pre-training datasets indicate the importance of data selection. Even if we extrapolate the curves to their ends (where they reach the irreducible error), the loss on transfer datasets do not reach close to the actual irredible error for the task, i.e. $E_{1|0}$ does not approach E_0.
- Downstream loss is predictable and does not illustrate any sort of emergent properties. Tracking this downstream loss gives a smooth proxy to extrapolate performance on tasks of interest.
- Some tasks have convex relationships ($\kappa > 1$) with pre-training loss where decreases in pre-training loss have diminishing returns, while others have concave relationships ($\kappa < 1$) where decreases in pre-training loss actually have increasing returns to transfer. Downstream tasks typically have concave relationships.

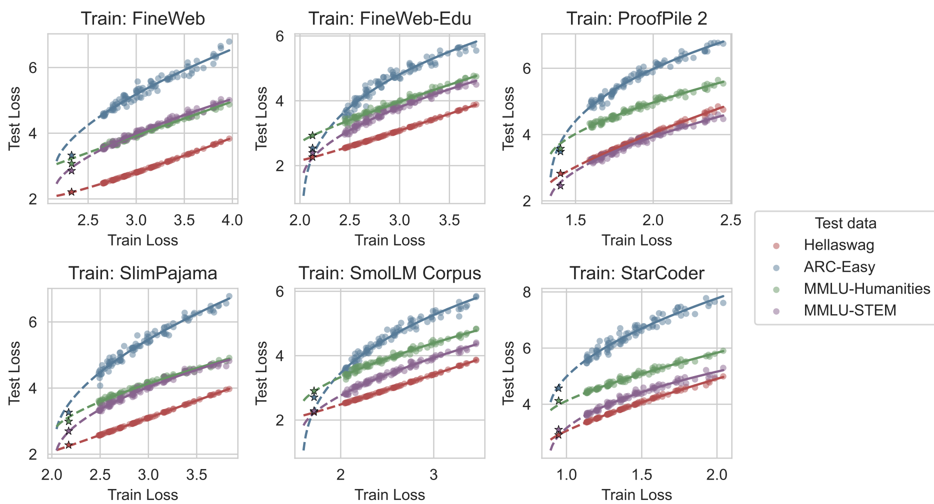

Figure 4: Train-to-test transfer for downstream tasks. On the test set we evaluate the CE loss of the correct multiple choice answer as a cloze task.

5 Loss-to-loss prediction can outperform independent scaling laws

Consider the following situation that a practitioner could encounter: after having fit a scaling law and performed a large run on one dataset, they want to know how training on a different dataset may yield different results. One could fit an independent scaling law on the new dataset, but that would not be leveraging the computation that has already been done. Instead, we can use loss-to-loss prediction.

Explicitly, consider two pre-training distributions P_0 and P_1. Assume that we have fit a set of small models on each distribution, and have just trained $\hat{f}_0^{\bar{N},\bar{D}}$ with large $\bar{N}, \bar{D}$. Then we compare the following procedures for estimating what would happen for $\hat{f}_1^{\bar{N},\bar{D}}$ on some loss L:

- Independent scaling. We do not use any information from P_0. We take the small models trained on P_1 and their losses under L and fit a scaling law to extrapolate to $\bar{N}, \bar{D}$.
- Loss-to-loss. We fit a translation between P_0 and P_1, this allows us to predict the pre-training loss of $\hat{f}_1^{\bar{N},\bar{D}}$. Then if we want to predict a specific test loss L if we were to pre-train on P_1, we compose this prediction with another translation from pre-training loss to test loss that is fit on the small models from P_1.

Results comparing relative error for predicting performance of the 3.3B models are presented in Table 1. We see clear gains of translation over independent scaling laws. For predicting pre-training loss these gains can be 8x reductions in error. When composing two separate translations to predict test loss, the gains are still 2x to 3x.

Setting	Independent error	Loss-to-loss error (ours)
Train-to-train	5.00%	0.61%
Train-to-test	3.64%	1.17%
Train-to-downstrem	9.53%	5.02%

Table 1: Relative error of training loss predictions by extrapolating scaling laws versus translating scaling laws. We average across all pairs of distinct pre-training datasets and all test and downstream tasks. We observe substantial reductions in error from translation as compared to independent scaling.

Note: translation is using more data as input than independent scaling, since it has access to the large model pre-trained on P_0. The benefit here is that there is no existing method to leverage this extra data as standard scaling laws cannot leverage information from training runs on different datasets.

A full discussion of the paper is deferred to Appendix A due to space constraints.

References

Samira Abnar, Mostafa Dehghani, Behnam Neyshabur, and Hanie Sedghi. Exploring the limits of large scale pre-training. *arXiv preprint arXiv:2110.02095*, 2021.

Anas Awadalla, Mitchell Wortsman, Gabriel Ilharco, Sewon Min, Ian Magnusson, Hannaneh Hajishirzi, and Ludwig Schmidt. Exploring the landscape of distributional robustness for question answering models. *arXiv preprint arXiv:2210.12517*, 2022.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. Llemma: An open language model for mathematics, 2023.

Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. Smollm-corpus, 2024. URL `https://huggingface.co/datasets/HuggingFaceTB/smollm-corpus`.

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

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

Blake Bordelon, Alexander Atanasov, and Cengiz Pehlevan. A dynamical model of neural scaling laws. *arXiv preprint arXiv:2402.01092*, 2024.

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. *Advances in neural information processing systems*, 20, 2007.

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. *arXiv preprint arXiv:1803.05457*, 2018.

Together Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023. URL `https://github.com/togethercomputer/RedPajama-Data`.

Zhengxiao Du, Aohan Zeng, Yuxiao Dong, and Jie Tang. Understanding emergent abilities of language models from the loss perspective. *arXiv preprint arXiv:2403.15796*, 2024.

Samir Yitzhak Gadre, Georgios Smyrnis, Vaishaal Shankar, Suchin Gururangan, Mitchell Wortsman, Rulin Shao, Jean Mercat, Alex Fang, Jeffrey Li, Sedrick Keh, et al. Language models scale reliably with over-training and on downstream tasks. *arXiv preprint arXiv:2403.08540*, 2024.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

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.

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.

Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. Scaling laws for downstream task performance of large language models. *arXiv preprint arXiv:2402.04177*, 2024.

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.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, et al. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*, 2023.

Llama 3 Team. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Lovish Madaan, Aaditya K Singh, Rylan Schaeffer, Andrew Poulton, Sanmi Koyejo, Pontus Stenetorp, Sharan Narang, and Dieuwke Hupkes. Quantifying variance in evaluation benchmarks. *arXiv preprint arXiv:2406.10229*, 2024.

Alexander Maloney, Daniel A Roberts, and James Sully. A solvable model of neural scaling laws. *arXiv preprint arXiv:2210.16859*, 2022.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International conference on machine learning*, pages 7721–7735. PMLR, 2021.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text, 2023.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. *arXiv preprint arXiv:2406.19146*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? *Advances in Neural Information Processing Systems*, 36, 2024a.

Rylan Schaeffer, Hailey Schoelkopf, Brando Miranda, Gabriel Mukobi, Varun Madan, Adam Ibrahim, Herbie Bradley, Stella Biderman, and Sanmi Koyejo. Why has predicting downstream capabilities of frontier ai models with scale remained elusive? *arXiv preprint arXiv:2406.04391*, 2024b.

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://www.cerebras.net/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B.

Nilesh Tripuraneni, Ben Adlam, and Jeffrey Pennington. Covariate shift in high-dimensional random feature regression. *arXiv preprint arXiv:2111.08234*, 2021.

Johannes Welbl, Nelson F Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*, 2017.

Mitchell Wortsman, Peter J Liu, Lechao Xiao, Katie Everett, Alex Alemi, Ben Adlam, John D Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, et al. Small-scale proxies for large-scale transformer training instabilities. *arXiv preprint arXiv:2309.14322*, 2023.

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

Rosie Zhao, Depen Morwani, David Brandfonbrener, Nikhil Vyas, and Sham Kakade. Deconstructing what makes a good optimizer for language models. *arXiv preprint arXiv:2407.07972*, 2024.

A Discussion

Here we discuss the implications of our findings, some limitations, and directions for future work.

Implications

- The train-to-train results imply a slightly modified functional form for scaling laws so that they remain valid scaling laws after passing through a shifted power law.
- The train-to-train results illustrate the similarities in scaling laws across data and in how the laws vary in a structured manner.
- The train-to-test results can inform dataset selection by providing clear predictions across a variety of downstream tasks.
- Loss-to-loss translations provide a mechanism for using data from scaling law runs that are computed on different training distributions to yield better predictions.

Limitations and disclaimers

- Our fits rely on estimating the asymptotic entropy of various scaling laws. This is a fundamentally difficult quantity to estimate and we hypothesize that where our fits fail it is often due to poor estimates of this quantity.
- Note that many of the train-to-test transfer cases seem to have noisier trends at high losses. It is not totally clear if this is pure noise or may be indicative that the power law trend does not hold as globally as we hypothesize.
- We only test on a relatively small set of downstream tasks compared to all possible choices. We also focus on multiple choice tasks instead of generative tasks since they have been more extensively studied in prior work and have easier to compute proxy loss metrics.
- Our results hold for our specific choices of hyperparameters and may not hold under some other choices. In particular, we would be interested in checking robustness to pre-training hyperparameters like sequence len, batch size, and learning rate.

Future work

- One exciting direction is to take the implications of the loss-to-loss relationships further so as to directly inform data mixing and filtering. Once we have reliable predictions, we can use those to inform choices about which data to train on.
- We hope to gain a tighter theoretical understanding as to why the loss-to-loss relationships are so clean by studying simplified models.
- Our results connect surprisingly disparate datasets. We are able to predict performance on code data from data that contains no code and visa-versa. It would be nice to have a better mechanistic understanding of how this works. It is possible that all the models converge to "features" that share some high level distributional properties (e.g. similar eigenvalue decay of the covariance). Or at a different level of granularity, it is possible that there the data is more similar than we think and there is a large enough amount of English in code and visa versa that losses are predictive. Or perhaps there are particular shared structures that emerge, e.g. in context learning.

B From loss to accuracy

We focus on loss-to-loss prediction, but it of course would be interesting to be able to predict accuracy. Prior work [Schaeffer et al., 2024a,b, Du et al., 2024] indicates that predicting accuracy from loss can be difficult, and we generally agree. However, other work [Gadre et al., 2024] claims that downstream accuracy can be predictable in some cases and we want to consider here whether accuracy is predictable in our data with methods similar to those presented in the main text.

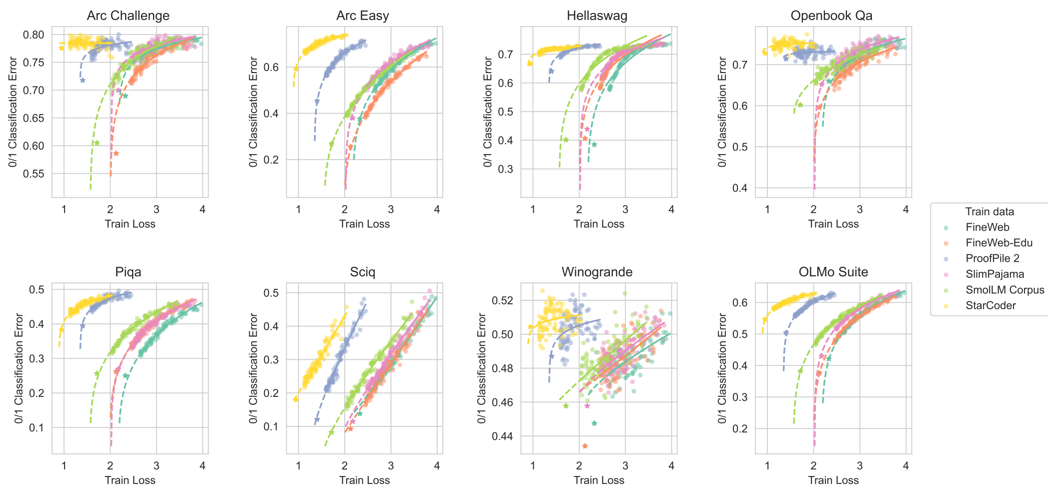

Figure 5: Fitting training loss to accuracy on the OLMo tasks individually (first 7 subplots), and then in aggregate (bottom right).

In particular, [Gadre et al., 2024] specifically finds that when they select a subset of 17 particularly easy benchmarks (where performance is better than chance for small models), then they can get good predictions for the average accuracy by fitting shifted power laws with a methodology similar to the one that we use for loss-to-loss prediction (but where $E_{1|0}$ is treated as a free parameter). We are able to reproduce a similar result on our suite of 7 tasks from OLMo, see Figure 5. The fits are fairly good for the aggregate, but it is clear that some of the fits (e.g. Hellaswag and ARC challenge) are systematically wrong. They end up overestimating the error because power law fits fundamentally cannot handle the fact that bad models will perform at random chance. The asymptotics of a power law mean that as $L \to \infty$ we get $Err \to \infty$, which is not possible. This is fundamentally related to the loss perspective on emergence [Du et al., 2024] where for multiple choice tasks there is some value of loss where the models start performing better than random chance. This is also perhaps even more clear for MMLU in Figure 6. In general, we would not expect this technique to work on individual tasks and especially not on more challenging tasks.

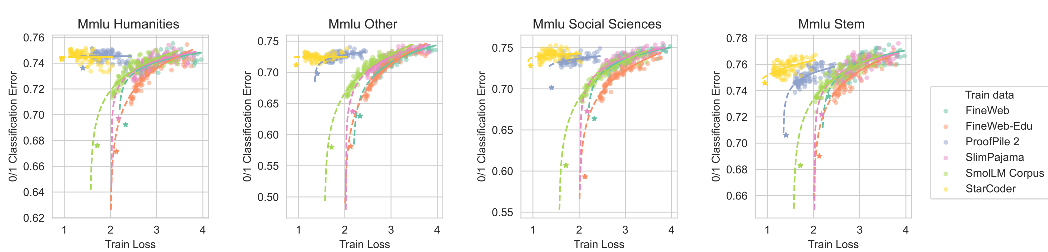

Figure 6: Fitting training loss to accuracy on MMLU splits.

For similar reasons, we also found it difficult to fit loss-to-error maps from the downstream CE loss to the classification error. For completeness, these results for the OLMo suite are included in Figure 7. One interesting thing about these curves is that now there is convergence across pre-training distributions where irrespective of the pre-training distribution there is a consistent relationship between downstream CE loss and classification error. This does suggest that the CE error is a useful proxy since it mediates the pre-training-specific effects from the test accuracy.

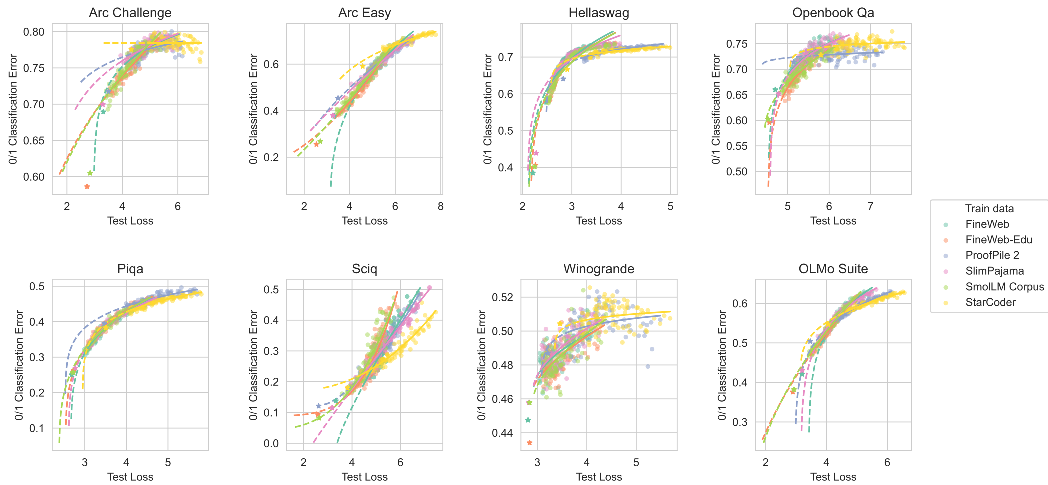

Figure 7: Prediction from test loss to error also struggles, but does show unified trends across pre-training distributions.

C Full downstream loss relationships

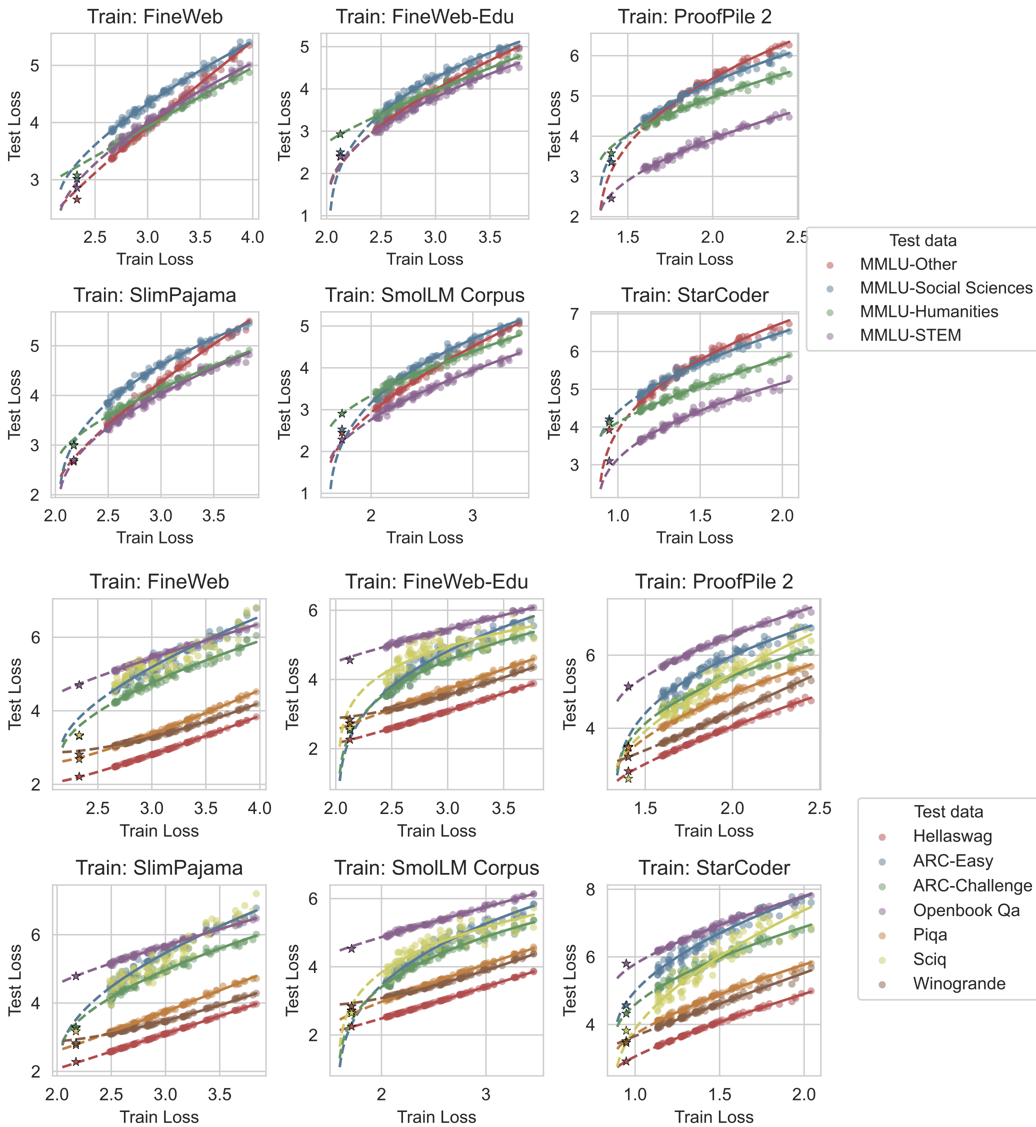

Figure 8: Train-to-test predictions across all individual downstream tasks.

D Scaling law fits

We follow the methodology from Hoffmann et al. [2022], Besiroglu et al. [2024] for fitting scaling law curves and illustrate fits for both Equation (3) and Equation (1).

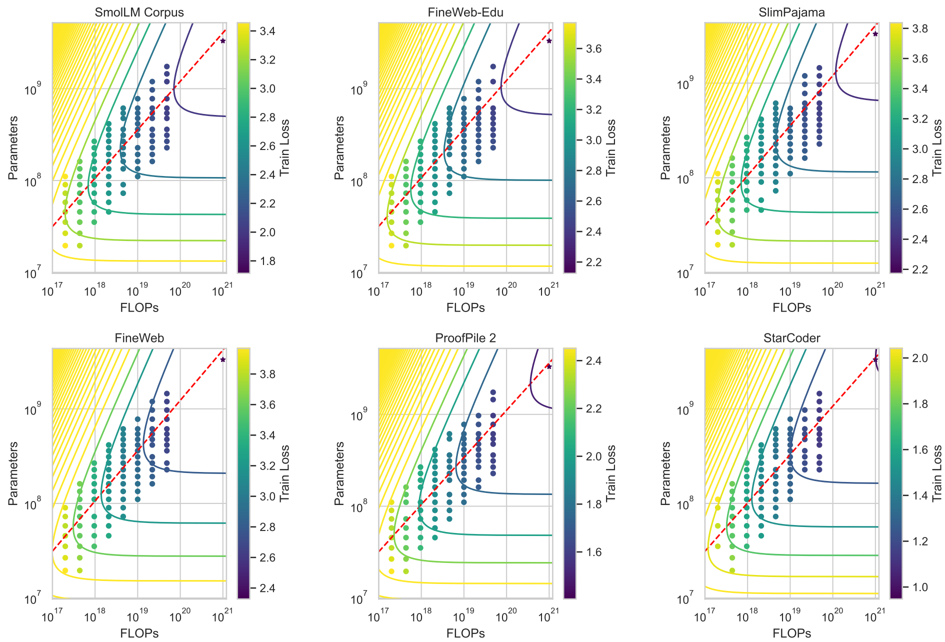

Figure 9: Contour plots for the curves fit with Equation (3) (our version of the scaling law parameterization). Red line indicates the optimal model size. The star point is not used for fitting the curves.

Data	A	B	E	α	β	a
SmolLM Corpus	7.00e+07	9.68e+08	1.57	0.43	0.48	0.53
FineWeb-Edu	5.96e+07	8.20e+08	2.00	0.43	0.48	0.53
SlimPajama	6.33e+07	8.97e+08	2.02	0.42	0.47	0.53
FineWeb	6.41e+07	8.84e+08	2.19	0.41	0.47	0.53
ProofPile 2	1.94e+07	3.09e+08	1.35	0.47	0.50	0.51
StarCoder	2.09e+07	3.49e+08	0.89	0.49	0.52	0.51

Table 2: Parameters for the curves fit with Equation (3) (our version of the scaling law parameterization). $a = \frac{\beta}{\alpha+\beta}$ is the exponent of the optimal model size relative to FLOPs.

Data	A	B	E	α	β	a
SmolLM Corpus	3.02e+03	1.19e+04	1.59	0.46	0.47	0.51
FineWeb-Edu	2.83e+03	1.17e+04	2.03	0.46	0.47	0.51
SlimPajama	2.34e+03	1.16e+04	2.04	0.45	0.47	0.51
FineWeb	1.83e+03	5.32e+03	2.17	0.43	0.43	0.50
ProofPile 2	3.52e+03	5.31e+03	1.33	0.50	0.45	0.48
StarCoder	8.38e+03	1.01e+04	0.89	0.55	0.48	0.47

Table 3: Parameters for the curves fit with Equation (1) (the chinchilla version of the scaling law parameterization). $a = \frac{\beta}{\alpha+\beta}$ is the exponent of the optimal model size relative to FLOPs.

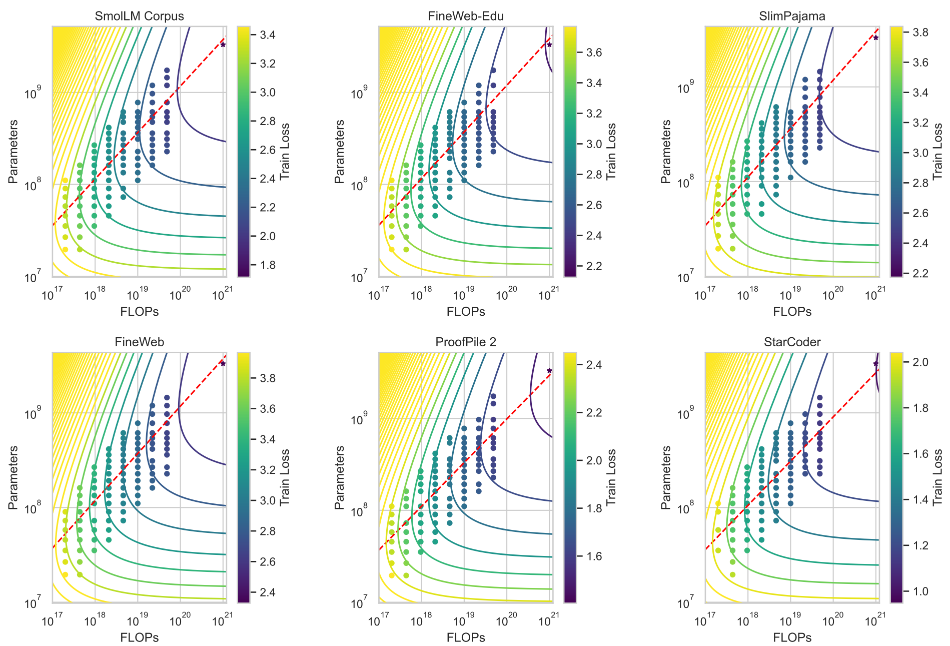

Figure 10: Contour plots for the curves fit with Equation (1) (the chinchilla version of the scaling law parameterization). Red line indicates the optimal model size. The star point is not used for fitting the curves.

E Iso-flop scaling laws

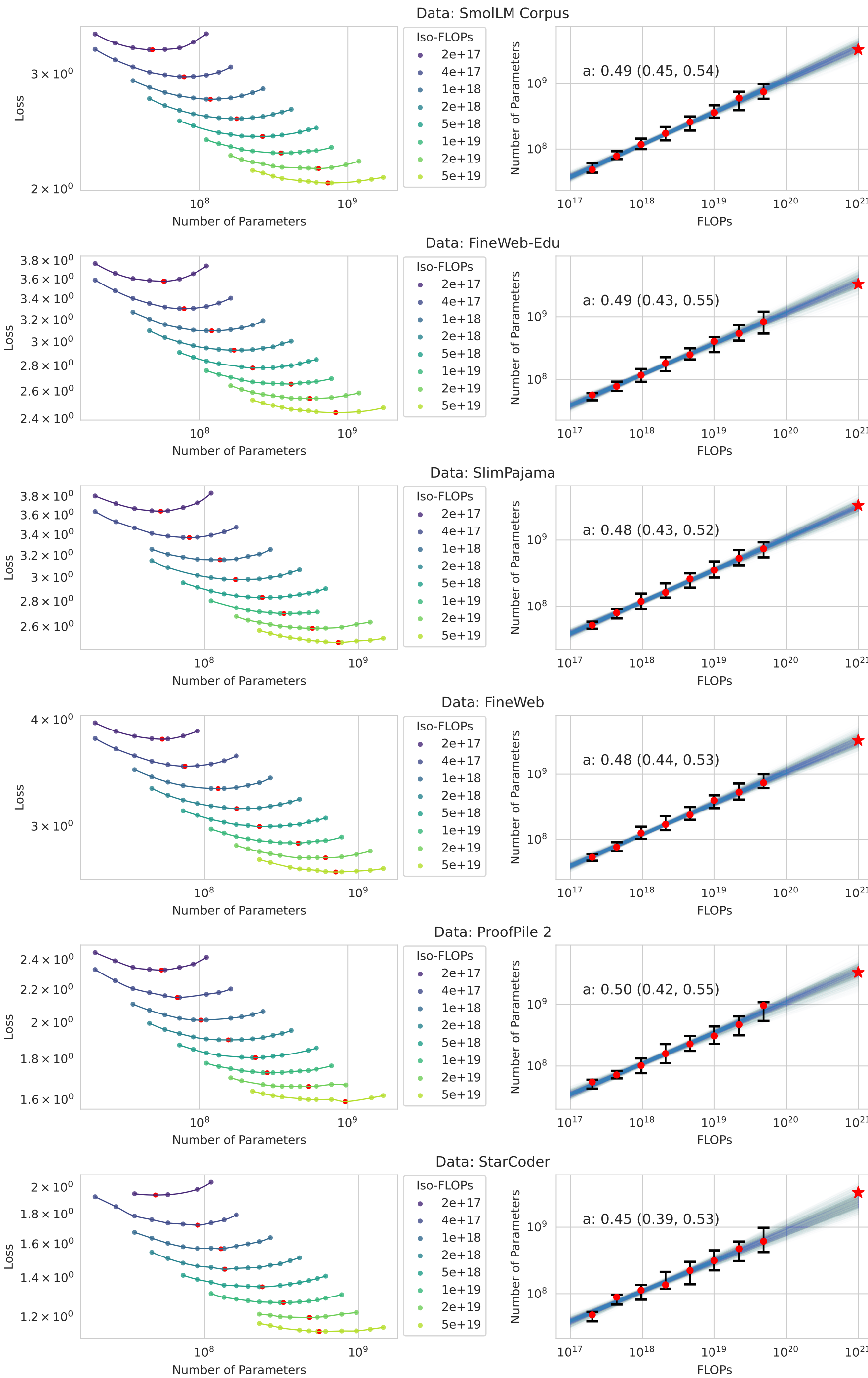

Figure 11: Iso-flop scaling laws across all pre-training datasets. optimal FLOP exponents are within the margin of error across all datasets. Error bars derived by the noise-and-interpolate method of Porian et al. [2024] with heuristic variance added to the loss values as in that paper.

349 ## F Hyperparameters

Table 4: Model parameters [Groeneveld et al., 2024, Wortsman et al., 2023, Zhao et al., 2024]

Parameter	Value
n	6-24 for small models, 40 for the 3.3B model
Number of heads	n
Head dimension	64
MLP hidden multiplier	4
Depth	n
Context length	512
Activation	GeLU
Positional encoding	RoPE
Biases	False
Normalization	PyTorch Layernorm
QK normalization	True
Precision	Mixed, bfloat16
Tokenizer	Llama2

Table 5: Training parameters [Groeneveld et al., 2024, Wortsman et al., 2023, Zhao et al., 2024]

Parameter	Value
Optimizer	Adam
Batch size	1024
Learning rate	1e-3
Schedule	Linear warmup, cosine decay
Warmup steps	20% of total steps
z-loss coefficient	1e-4
Weight decay	0.0
β_1	0.9
β_2	0.95
ϵ	1e-15

350 ## G Full loss-to-loss parameter fits from Figure 1

Table 6: Train-to-train fits

Data 0	Data 1	κ	K	E_0	E_1
FineWeb-Edu	FineWeb	0.93	1.08	2.03	2.17
FineWeb-Edu	FineWeb-Edu	1.00	1.00	2.03	2.03
FineWeb-Edu	ProofPile 2	1.02	0.63	2.03	1.33
FineWeb-Edu	SlimPajama	0.98	1.04	2.03	2.04
FineWeb-Edu	SmolLM Corpus	1.00	1.08	2.03	1.59
FineWeb-Edu	StarCoder	1.11	0.63	2.03	0.89

Table 7: Train-to-test fits

Train data	Test data	κ	K	E_0	$E_1\|0$
FineWeb-Edu	FineWeb	0.96	1.02	2.03	2.42
FineWeb-Edu	FineWeb-Edu	1.00	1.00	2.03	2.03
FineWeb-Edu	ProofPile 2	1.44	1.37	2.03	4.12
FineWeb-Edu	SlimPajama	1.08	1.05	2.03	2.57
FineWeb-Edu	SmolLM Corpus	1.08	1.11	2.03	2.25
FineWeb-Edu	StarCoder	1.32	1.49	2.03	2.91

Table 8: Train-to-downstream fits

Train data	Test data	κ	K	E_0	$E_1\|0$
FineWeb-Edu	Hellaswag	1.08	0.93	2.03	2.18
FineWeb-Edu	ARC-Easy	0.33	4.85	2.03	0.00
FineWeb-Edu	MMLU-Humanities	0.87	1.23	2.03	2.76
FineWeb-Edu	MMLU-STEM	0.54	2.24	2.03	1.59