StableLM-3B-4E1T Technical Report

Jonathan Tow* Marco Bellagente Dakota Mahan Carlos Riquelme Ruiz

Abstract

We introduce StableLM-3B-4E1T, a 3 billion parameter language model pre-trained under the multi-epoch regime. We explore the impact of repeated tokens on downstream performance, training on 1 trillion tokens for 4 epochs. Our findings contribute to the ongoing research in scaling data-constrained language models.

1 Introduction

StableLM-3B-4E1T [1] is a 3 billion (3B) parameter language model pre-trained under the multi-epoch regime to study the impact of repeated tokens on downstream performance. Given the prior success in this area (Taylor et al., 2022 and Yi Tay* and Metzler, 2022), we ***train on 1 trillion (1T) tokens for 4 epochs (4E)*** following the observations of Muennighoff et al. (2023) in which they find "training with up to 4 epochs of repeated data yields negligible changes to loss compared to having unique data." Further inspiration for the token count is taken from (De Vries, 2023), which suggests a 2.96B model trained for 2.85 trillion tokens achieves a similar loss to a Chinchilla compute-optimal 9.87B language model ($k_n = 0.3$).

2 Model Architecture

The model is a decoder-only transformer similar to the LLaMA (Touvron et al., 2023) architecture with the following modifications:

- **Position Embeddings:** Rotary Position Embeddings (Su et al., 2023) applied to the first 25% of head embedding dimensions for improved throughput following Black et al. (2022).
- **Normalization:** LayerNorm (Ba et al., 2016) with learned bias terms as opposed to RMSNorm (Zhang and Sennrich, 2019).
- **Tokenizer:** GPT-NeoX (Black et al., 2022).

Parameter	Value
Parameters	2,795,443,200
Hidden Size	2560
Layers	32
Heads	32
Sequence Length	4096

Table 1: Model Architecture Details

*Corresponding author: jonathantow1@gmail.com

[1] https://huggingface.co/stabilityai/stablelm-3b-4e1t

This document is a static version of the original report published to Weights & Biases: https://stability.wandb.io/stability-llm/stable-lm/reports/StableLM-3B-4E1T--VmlldzoyMjU4

3 Training Data

The dataset is comprised of a filtered mixture of open-source large-scale datasets available on the HuggingFace Hub: Falcon RefinedWeb extract (Penedo et al., 2023), RedPajama-Data (Computer, 2023), and The Pile (Gao et al., 2020), both without the *Books3* subset, and StarCoder (Li et al., 2023). The complete list is provided in Table 1.

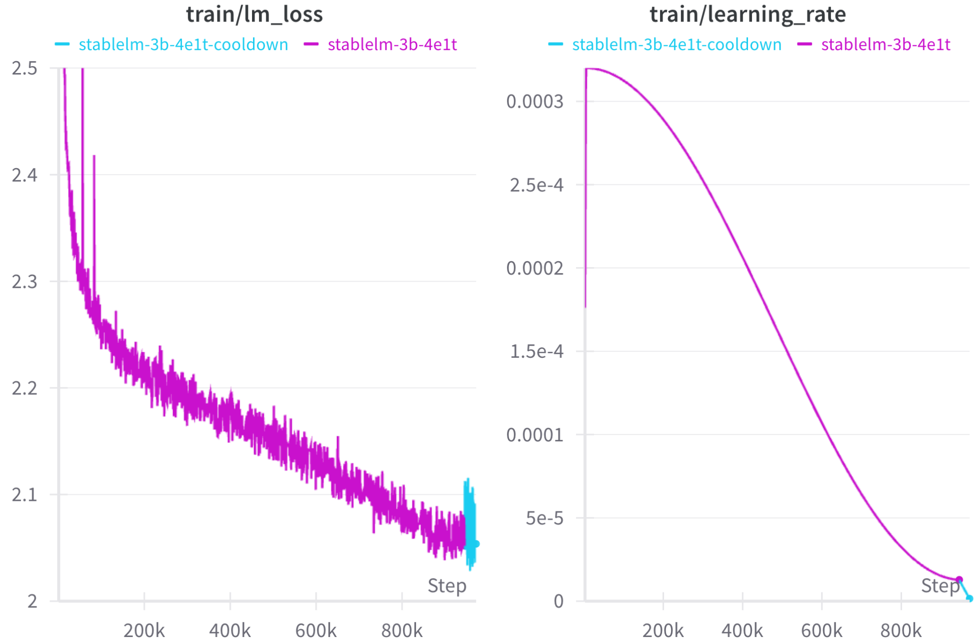
Dataset	Subset	Num Tokens (NeoX)	Num Docs	Category
The Pile	ArXiv	19,769,458,882	1,441,920	Academic
The Pile	PubMed	22,378,915,742	2,964,625	Academic
S2ORC		60,552,319,208	11,592,936	Academic
The Pile	PhilPapers	644,077,299	33,881	Academic
S2ORC	peS2o	57,200,107,871	38,811,179	Academic
The Pile	PG-19	4,719,327,141	50,579	Books
RefinedWeb		580,957,303,522	967,989,228	Web
RedPajama	Common Crawl (2023)	188,371,605,706	111,402,716	Web
RedPajama	C4	174,769,707,653	364,868,892	Web
The Pile	OpenWebText2	8,947,174,650	8,012,025	Web
RedPajama	StackExchange	20,544,276,837	29,825,086	Social
The Pile	UbuntuIRC	1,871,044,039	2,807	Social
The Pile	HackerNews(2006-2020)	2,031,470,476	1,571,968	Social
The Pile	EuroParl	1,562,068,114	69,814	Law
The Pile	FreeLaw	13,805,827,414	4,542,840	Law
Pile Of Law		16,377,540,899	3,096,719	Law
DM Math		3,728,203,638	972,502	Math
AMPS		324,711,403	2,635,350	Math
RedPajama	GitHub	58,930,922,707	28,793,312	Code
StarCoder	C	7,197,443,940	204,250	Code
StarCoder	CPP	8,944,383,599	221,536	Code
StarCoder	Java	11,801,463,022	388,908	Code
StarCoder	JavaScript	8,451,649,925	354,224	Code
StarCoder	Python	12,073,208,678	475,750	Code
RedPajama	Wiki	24,839,086,595	29,834,171	Wiki
	Total	1,310,793,298,960		

Figure 1: Open-source datasets used for multi-epoch training. Note that the total token count does not account for the reduced size after downsampling C4, Common Crawl (2023), and GitHub to obtain 1T tokens.

The data mixture is primarily based on the reported DoReMi (Xie et al., 2023) optimal mixture for The Pile domains. Given the extensive web data, we recommend fine-tuning base StableLM-3B-4E1T for your downstream tasks.

4 Training Procedure

The model is trained for 972k steps in bfloat16 precision with a global context length of 4096 instead of the multi-stage ramp-up from 2048-to-4096 as done for StableLM-Alpha v2. The batch size is set to 1024 (4,194,304 tokens). We optimize with AdamW (Loshchilov and Hutter, 2017) and use linear warmup for the first 4.8k steps, followed by a cosine decay schedule to 4% of the peak learning rate. Early instabilities are attributed to extended periods in high learning rate regions. We do not incorporate dropout (Srivastava et al., 2014) due to the model's relatively small size. Detailed hyperparameters are provided in the model configuration in the `StableLM` repository.



(a) Training Loss (b) Learning Rate Schedule

Figure 2: Training Dynamics

During training, we evaluate natural language benchmarks and observe steady improvements throughout training until the tail end of the learning rate decay schedule. For this reason, we decided to linearly **cool down** the learning rate towards 0, similar to Zhai et al. (2022), in hopes of squeezing out performance. We plan to explore alternative schedules in future work.

Furthermore, our initial pre-training stage relies on the flash-attention API (Dao, 2023) with its out-of-the-box triangular causal masking support. This forces the model to attend similarly to different documents in a packed sequence. In the cool-down stage, we instead reset position IDs and attention masks at EOD tokens for all packed sequences after empirically observing improved sample quality (read: less repetition) in a concurrent experiment. We hypothesize that this late adjustment leads to the notable degradation in byte-length normalized accuracies of ARC-Easy (Clark et al., 2018) and SciQ (Johannes Welbl, 2017).

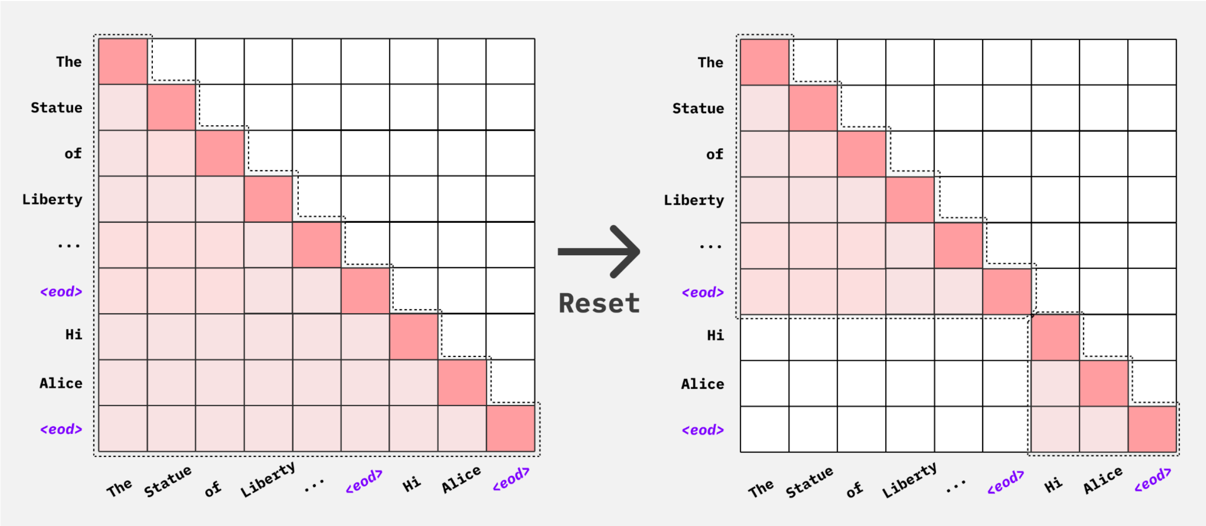
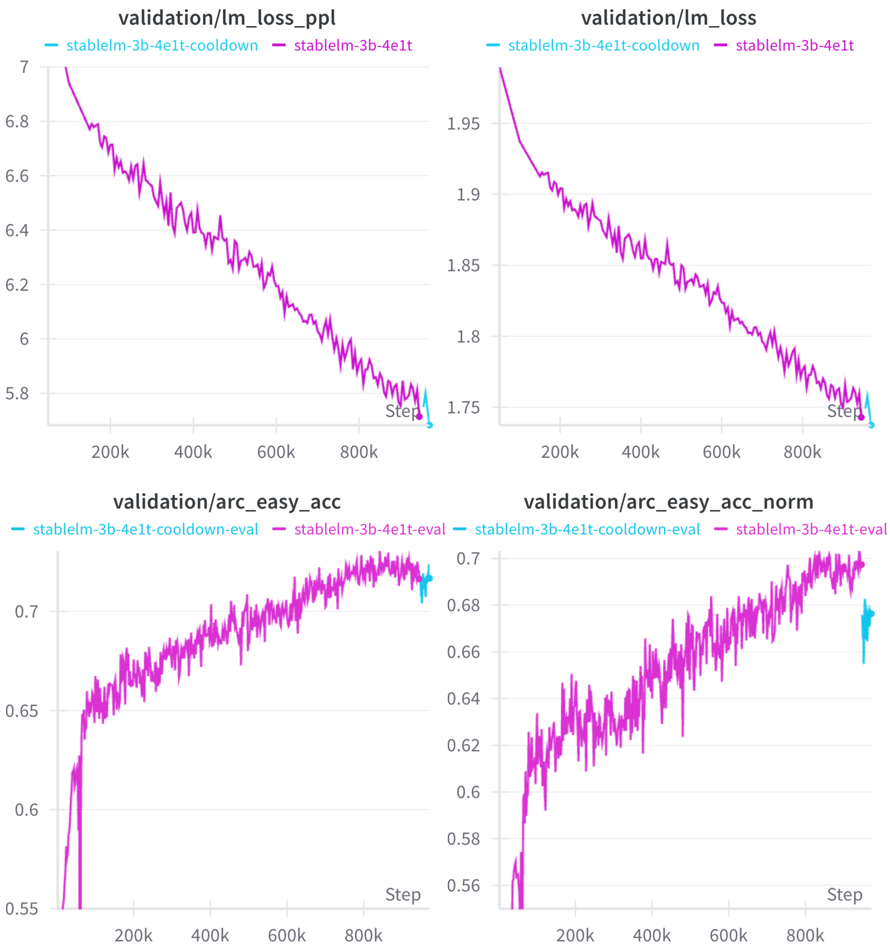


Figure 3: Toy demonstration of attention mask resetting.

Data composition was modified during the cool-down. Specifically, we remove Ubuntu IRC, OpenWebText, HackerNews, and FreeLaw for quality control and further NSFW filtering while upsampling C4. The distribution shift is likely responsible for the increased loss (+0.02 nats) from the initial stage.

See the plots below for validation dynamics across our hold-out set and common NLP benchmarks.



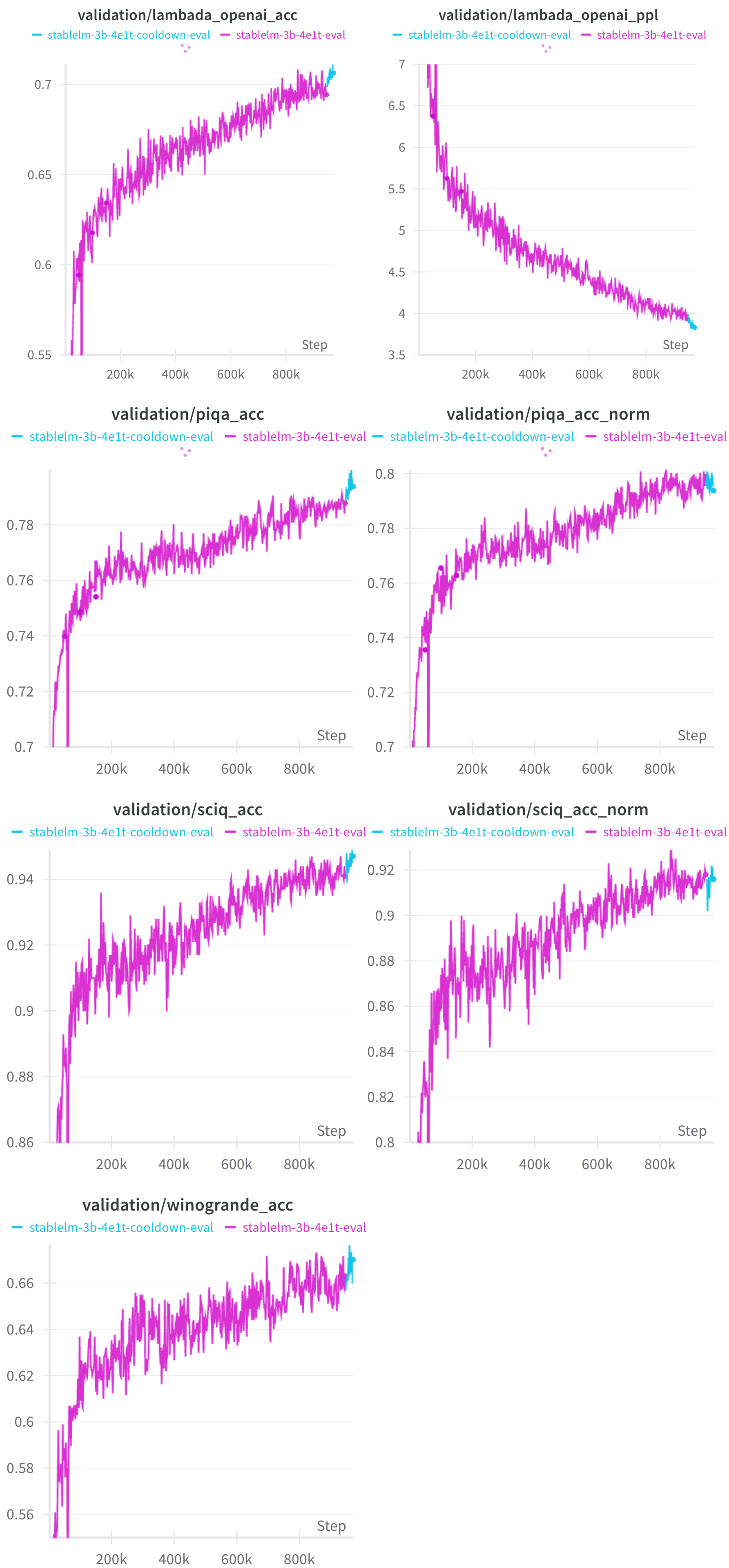


Figure 4: Validation Dynamics

Note: The released checkpoint is taken from step 970k according to validation loss and average downstream performance.

5 Downstream Results

The following zero-shot evaluations are performed with EleutherAI's `lm-evaluation-harness` (Gao et al., 2021) using the `lm-bench` branch of Stability AI's fork.

Model	Avg	ARC-C	ARC-E	BoolQ	HSwag	LAMB	OBQA	PIQA	SciQ	Wino
LLaMA 2 7B	68.75	43.00	76.26	77.74	75.94	73.47	31.40	77.75	93.60	69.61
Qwen-7B	67.91	45.39	67.38	74.56	88.85[a]	69.67	32.20	73.99	93.20	65.98
Falcon-7B	67.83	40.27	74.41	73.55	76.35	74.56	30.60	79.49	94.00	67.25
MPT-7B	67.36	40.53	74.92	73.94	76.17	68.64	31.40	78.89	93.70	68.03
StableLM 3B 4E1T	66.93	37.80	72.47	75.63	73.90	70.64	31.40	79.22	94.80	66.54
Baichuan2-7B Base	66.93	42.24	75.00	73.09	72.29	70.99	30.40	76.17	94.60	67.56
StableLM-Base-Alpha-7B v2	66.89	38.48	73.19	70.31	74.27	74.19	30.40	78.45	93.90	68.82
Open LLaMA 7B v2	66.32	38.82	71.93	71.41	74.65	71.05	30.20	79.16	93.80	65.82
Phi-1.5	65.57	44.45	76.14	74.53	62.62	52.75	37.60	76.33	93.20	72.53
GPT-NeoX-20B	65.57	37.88	72.90	69.48	71.43	71.98	29.80	77.42	93.10	66.14
BTLM-3B-8K-Base[b]	63.59	34.90	70.45	69.63	69.78	66.23	27.60	75.84	92.90	64.96
Pythia 12B	62.69	31.83	70.20	67.31	67.38	70.64	26.40	76.28	90.20	64.01
Open LLaMA 3B v2	62.43	33.87	67.59	65.69	69.99	66.74	26.00	76.66	92.40	62.90
GPT-J 6B	62.34	33.96	66.96	65.44	66.24	68.23	29.00	75.57	91.50	64.17
Pythia-6.9B	60.58	31.83	67.21	64.01	63.88	67.01	25.80	75.08	89.80	60.62
Pythia 2.8B	58.52	30.12	63.47	64.13	59.44	65.15	23.80	74.10	88.20	58.25

[a] Outlier results
[b] Previous 3B Pre-Trained SOTA

Table 2: Zero-shot performance across popular language modeling and common sense reasoning benchmarks. `lm-eval` results JSONs can be found in the `evals` directory of the `StableLM` repository.

StableLM-3B-4E1T achieves state-of-the-art performance (September 2023) at the 3B parameter scale for open-source models and is competitive with many of the popular contemporary 7B models, even outperforming our most recent 7B StableLM-Base-Alpha-v2.

6 System Details

Hardware: StableLM-3B-4E1T was trained on the Stability AI cluster across 256 NVIDIA A100 40GB GPUs (AWS P4d instances). Training began on August 23, 2023, and took approximately 30 days to complete.

Software: We use an internal fork of GPT-NeoX (Andonian et al., 2023), train under 2D parallelism (Data and Tensor Parallel) with ZeRO-1 (Rajbhandari et al., 2020), and rely on flash-attention as well as SwiGLU and Rotary Embedding kernels from FlashAttention-2 (Dao, 2023).

Figure 5: Throughput Logging. TFLOPS are estimated using GPT-NeoX's get_flops function.

7 Conclusion

In this technical report, we present StableLM-3B-4E1T, a 3 billion parameter language model trained on 1 trillion tokens for 4 epochs. Our results provide further evidence for the claims in Muennighoff et al. (2023) at the trillion token scale, suggesting multi-epoch training as a valid approach to improving downstream performance when working under data constraints.

8 Acknowledgements

We thank the MLOps team, Richard Vencu and Sami Kama, for 30 days of uninterrupted pre-training; Reshinth Adithyan, James Baicoianu, Nathan Cooper, Christian Laforte, Nikhil Pinnaparaju, and Enrico Shippole, for helpful discussion and guidance.

References

Alex Andonian, Quentin Anthony, Stella Biderman, Sid Black, Preetham Gali, Leo Gao, Eric Hallahan, Josh Levy-Kramer, Connor Leahy, Lucas Nestler, Kip Parker, Michael Pieler, Jason Phang, Shivanshu Purohit, Hailey Schoelkopf, Dashiell Stander, Tri Songz, Curt Tigges, Benjamin Thérien, Phil Wang, and Samuel Weinbach. GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch, 9 2023. URL https://www.github.com/eleutherai/gpt-neox.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022.

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:1803.05457v1*, 2018.

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

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning, 2023.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, 2022.

Harm De Vries. Go smol or go home, 2023. URL https://www.harmdevries.com/post/model-size-vs-compute-overhead/.

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

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021. URL https://doi.org/10.5281/zenodo.5371628.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. Openwebtext corpus, 2019.

Peter Henderson, Mark Krass, Lucia Zheng, Neel Guha, Christopher D Manning, Dan Jurafsky, and Daniel Ho. Pile of law: Learning responsible data filtering from the law and a 256gb open-source legal dataset. *Advances in Neural Information Processing Systems*, 35:29217–29234, 2022.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022.

Matt Gardner Johannes Welbl, Nelson F. Liu. Crowdsourcing multiple choice science questions. 2017.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, João Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Zebaze, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy, Jason Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvashi Bhattacharyya, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailey Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, Brendan Dolan-Gavitt, Danish Contractor, Siva Reddy, Daniel Fried, Dzmitry Bahdanau, Yacine Jernite, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. Starcoder: may the source be with you!, 2023.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Dan S Weld. S2orc: The semantic scholar open research corpus. *arXiv preprint arXiv:1911.02782*, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. URL https://api.semanticscholar.org/CorpusID:53592270.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models, 2023.

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

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, 2023.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE, 2020.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale, 2019.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. *arXiv preprint arXiv:1904.01557*, 2019.

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

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15 (1):1929–1958, January 2014. ISSN 1532-4435.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science, 2022. URL https://arxiv.org/abs/2211.09085.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining, 2023.

Vinh Q. Tran Xavier Garcia Dara Bahr Tal Schuster Huaixiu Steven Zheng Neil Houlsby Yi Tay*, Mostafa Dehghani* and Donald Metzler. Unifying language learning paradigms. 2022.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12104–12113, 2022.

Biao Zhang and Rico Sennrich. Root mean square layer normalization, 2019.