Scaling Data-Constrained Language Models

Niklas Muennighoff

Twitter: @Muennighoff

arxiv.org/abs/2305.16264

Niklas Muennighoff, Alexander M. Rush, Boaz Barak,
Teven Le Scao, Aleksandra Piktus, Nouamane Tazi,
Sampo Pyysalo, Thomas Wolf & Colin Raffel

Outline

(1) Scaling language models
Background on what, why &
how of scaling

(2) Data-constrained scaling
Scaling with repeated data
Mixing modalities & revising filtering

Please interrupt with questions / thoughts anytime!

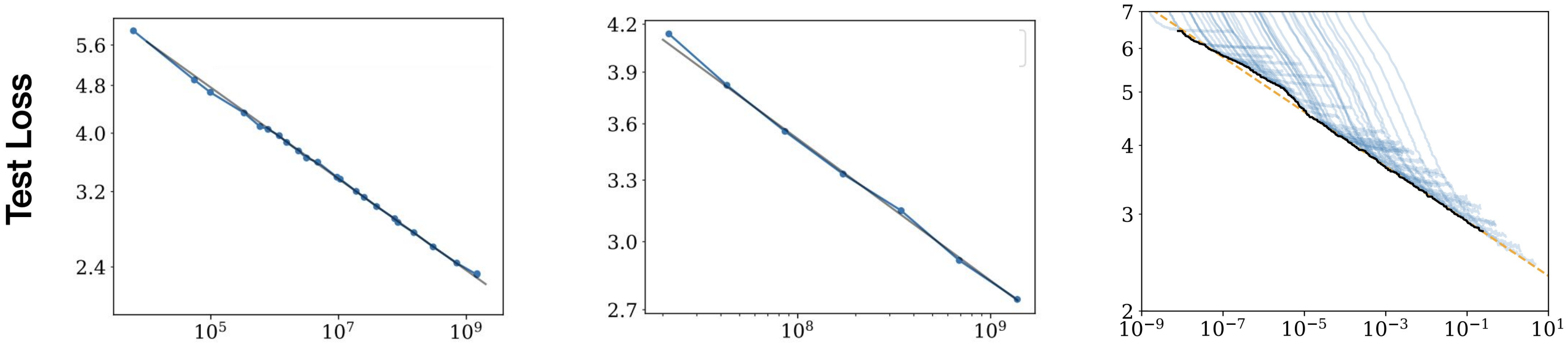
What is scaling?

α x Model size x Training data = Training compute

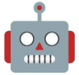
	Model size (# parameters)	Training data (# tokens)	Training compute (FLOPs)	Resources
BERT-base (2018)	109M	250B	1.6e20	64 TPU v2 for 4 days (16 V100 GPU for 33 hrs)
GPT-3 (2020)	175B	300B	3.1e23	~1,000x BERT-base
PaLM (2022)	540B	780B	2.5e24	6k TPU v4 for 2 months

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (2018)
Language Models are Few-Shot Learners (2020)
PaLM: Scaling Language Modeling with Pathways (2022)

How important is scaling? **(Return)**



Model size **Training data** **Compute**

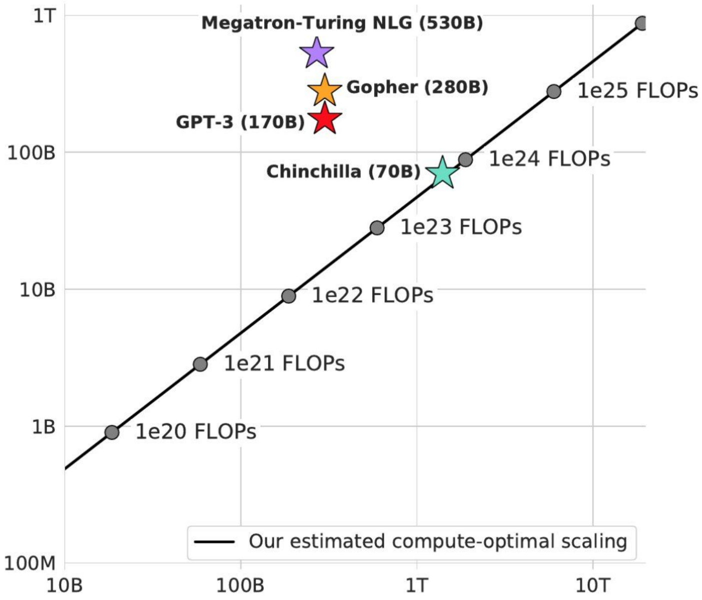
  

*Language models improve as a **power-law** with model size, training data, and amount of compute used for training.*

Scaling laws for neural language models (2020)

How to scale? **(Allocation)**

Model size (# parameters)



Training data (# tokens)

Optimal compute allocation is scaling model size & training data ***equally*** *(Chinchilla)*.

[Training compute-optimal large language models (2022)]

Predictive formula

We can estimate loss (L) given **model size (N),** training data (D), and learned constants:

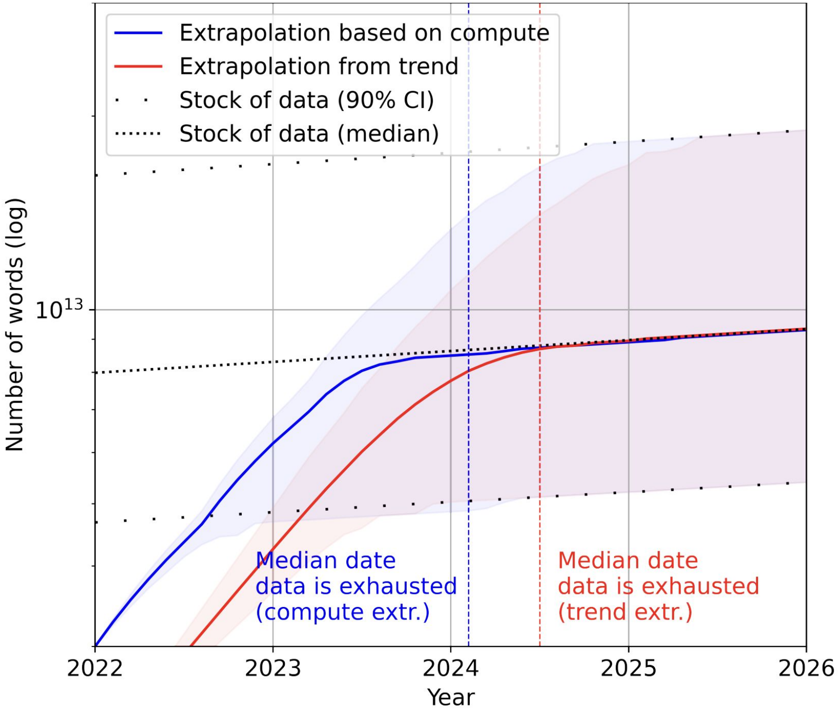
$$L(N, D) = \frac{A}{N^{\alpha}} + \frac{B}{D^{\beta}} + E$$

Fitting the constants, yields: $\alpha \approx \beta$

i.e. equal scaling of **N** and **D**.

[Training compute-optimal large language models (2022)]

Scaling is data-constrained



High-quality language data
Papers: ~1T tokens
Books: ~1.6T tokens
+ Other sources (Wikipedia etc)

Code data
GitHub: ~14T tokens

Low-resource languages
Finnish 🇫🇮 (6M speakers): 38B tokens
(across public and closed sources incl. libraries, social media, web crawls etc.)

Will we run out of data? An analysis of the limits of scaling datasets in Machine Learning (2022)
chinchilla's wild implications (2022)
FinGPT: Large Generative Models for a Small Language (2023)

Outline

(1) Scaling language models
Background on what, why & how of scaling

(2) Data-constrained scaling
Scaling with repeated data
Mixing modalities & revising filtering

Repeating data considered harmful for LLMs

GPT-3: "Data are sampled **without replacement** during training..."

PaLM: "We train all three models on exactly one epoch of the data ... and choose the mixing proportions **to avoid repeating data in any subcomponent**."

Is repeating 📚data really so bad?

[Language Models are Few-Shot Learners (2020)]
[PaLM: Scaling Language Modeling with Pathways (2022)]

Experimental setup

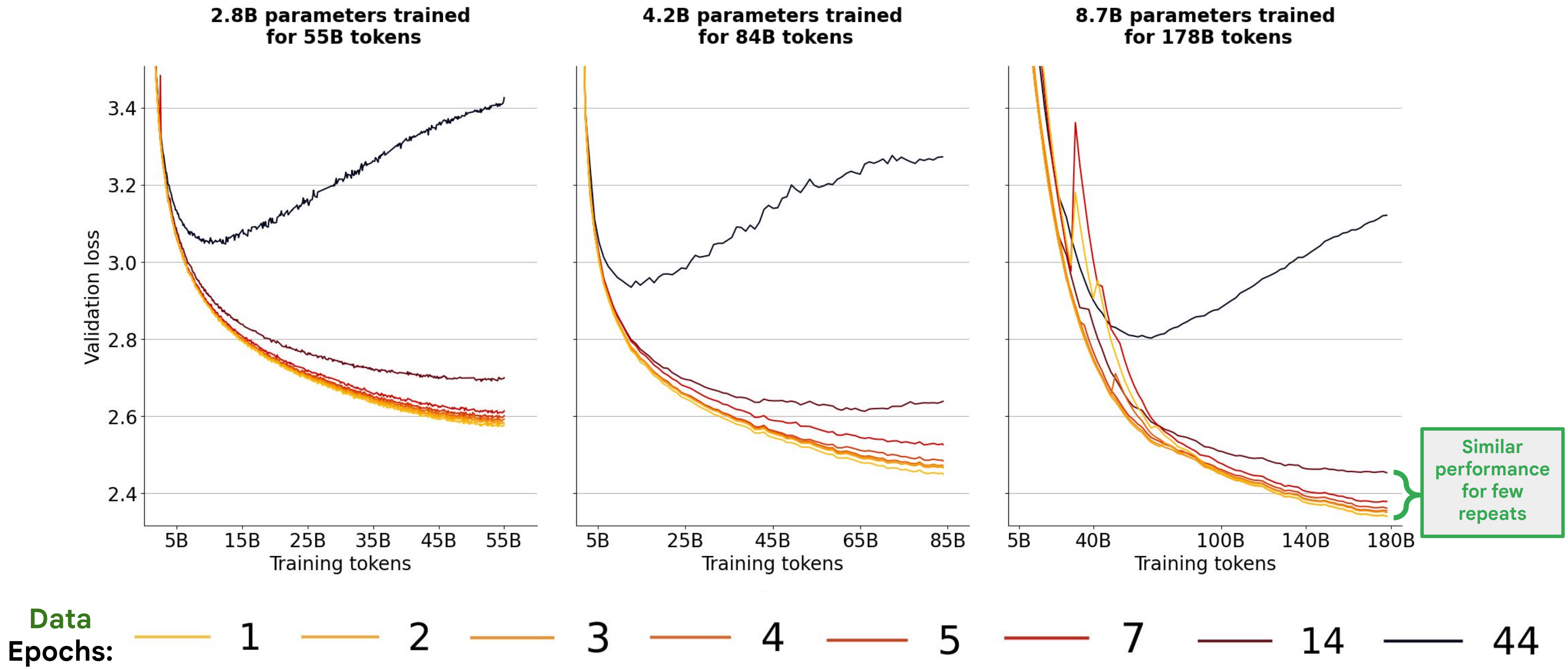
Training compute (FLOPs)	Model size (# parameters)	Training data (# tokens)
9.3e20	2.8B	55B
2.1e21	4.2B	84B
9.3e21	8.7B	178B

For each setup, train 8 models with different amounts of unique training data that is repeated

\+ ~300 miscellaneous runs

Use common large language modeling presets:
- architecture (GPT-2 transformer)
- hyperparameters (Chinchilla)
- datasets (web crawls like C4)

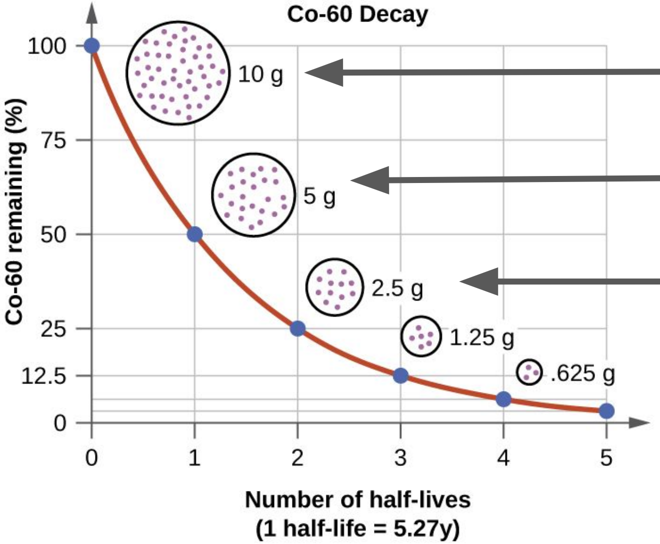
Repeating data (**Return**)



Hypothesis: Data repeating as exponential decay

Intuitively, each time unique data is repeated it loses a fraction (δ) of its original value.

Radioactive decay is an example of exponential decay:



1

1 * 0.5

1 * 0.5 * 0.5

- δ = 0.5

Sum up the value at each 📚data repeat

D' = value of total data, U = unique data, R_D = number of repetitions

$$D' = U + (1-\delta)U + (1-\delta)^2 U + \cdots + (1-\delta)^{R_D} U$$

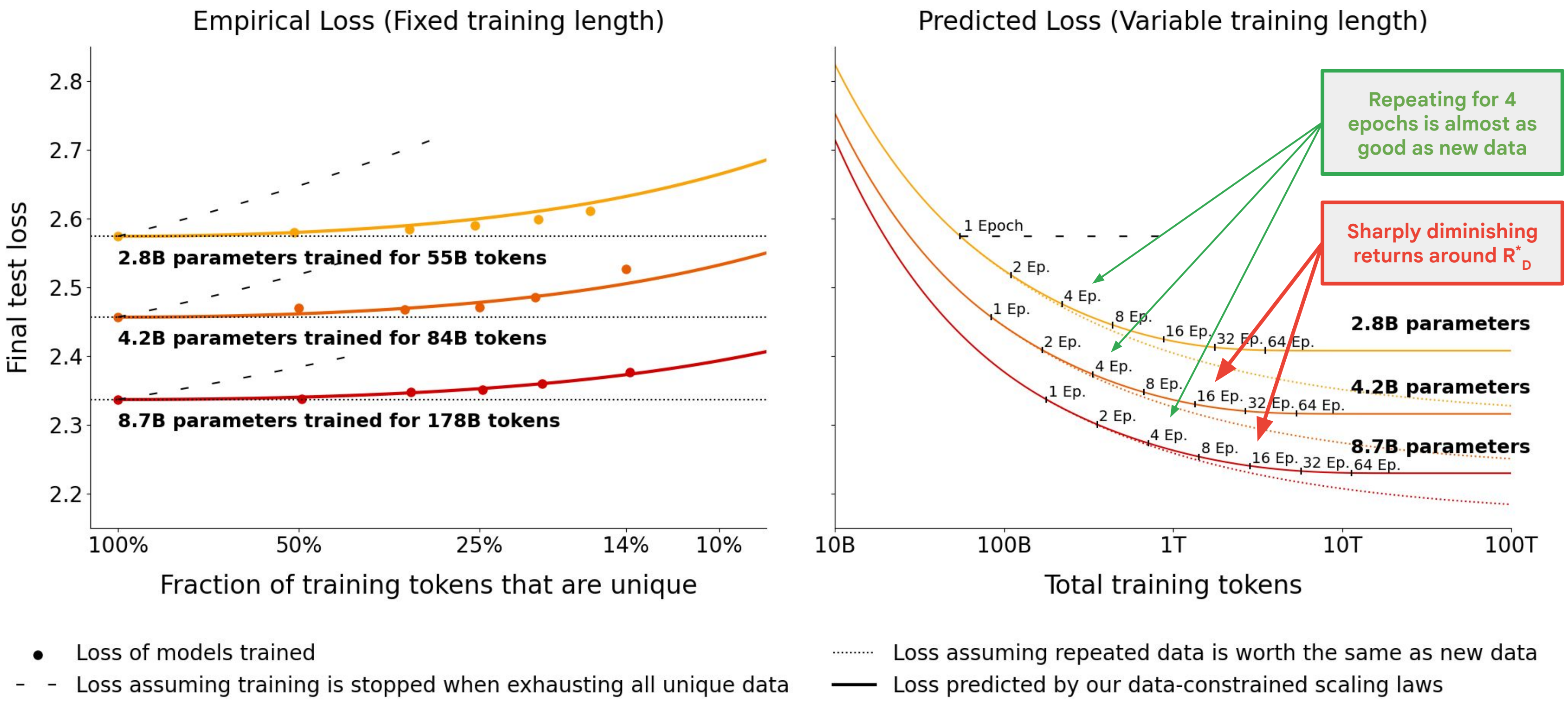
- If $\delta = 1$: repeated data is worth nothing (only first U counts)
- If $\delta = 0$: repeated data is as good as new data
- If $\delta = 0.5$: repeated data retains 50% of its prior value at each repeat

Approximation: $$D' = U + U \cdot R_D^* \cdot (1 - e^{-R_D/R_D^*})$$

R^*_D= learned parameter, number of times you can repeat before **sharply diminishing returns**

- If $R^*_D = 0$: repeated data is worth nothing
- If R^*_D = infinity: repeated data is as good as new data

Predicting loss (**Return**)



Estimate loss given parameters and repeated data

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

$$N' = U_N + U_N R_N^* \left(1 - e^{\frac{-R_N}{R_N^*}}\right)$$

$$U_N = \min\{N_{opt}, N\}$$

$$D' = U_D + U_D R_D^* \left(1 - e^{\frac{-R_D}{R_D^*}}\right)$$

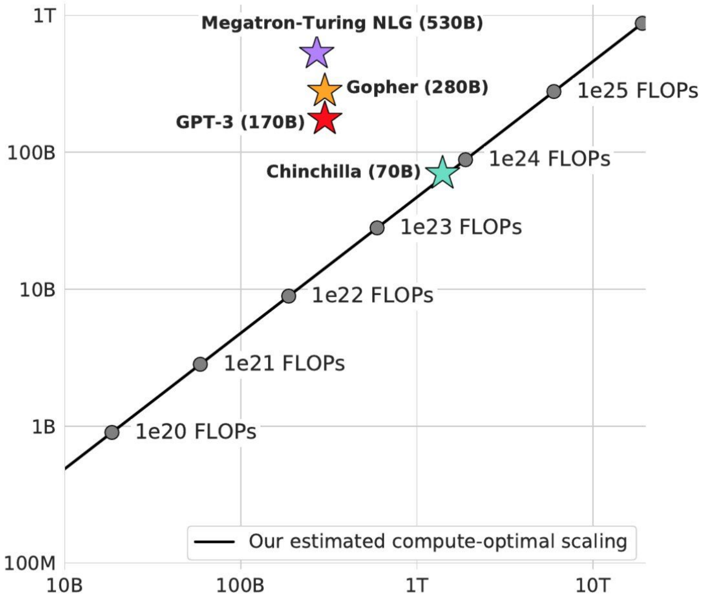
Fit on data from ~200 training runs to learn R^*_D and R^*_N

$R^*_D = 15.4$ ($\delta \approx 0.06$)

$R^*_N = 5.3$ ($\delta \approx 0.19$)

Reminder: Equal scaling when **not repeating data**

Model size
(# parameters)

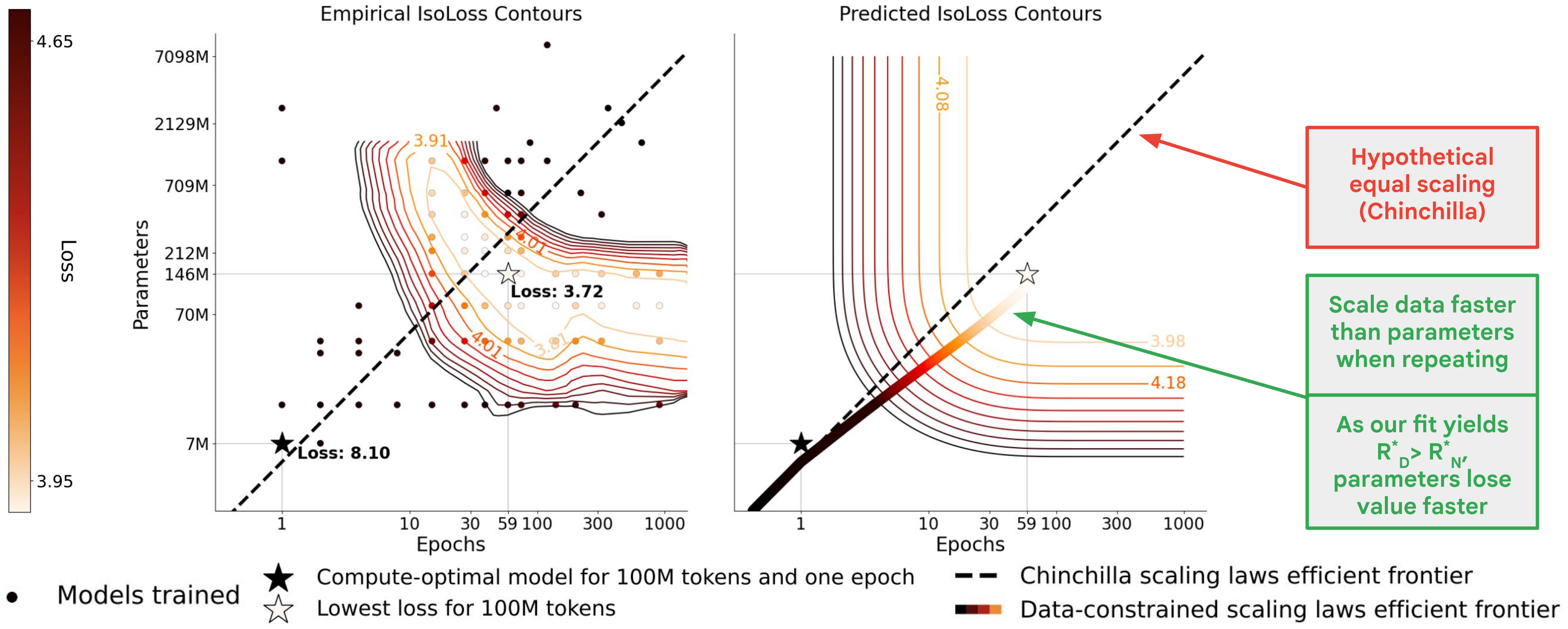


Training data (# tokens)

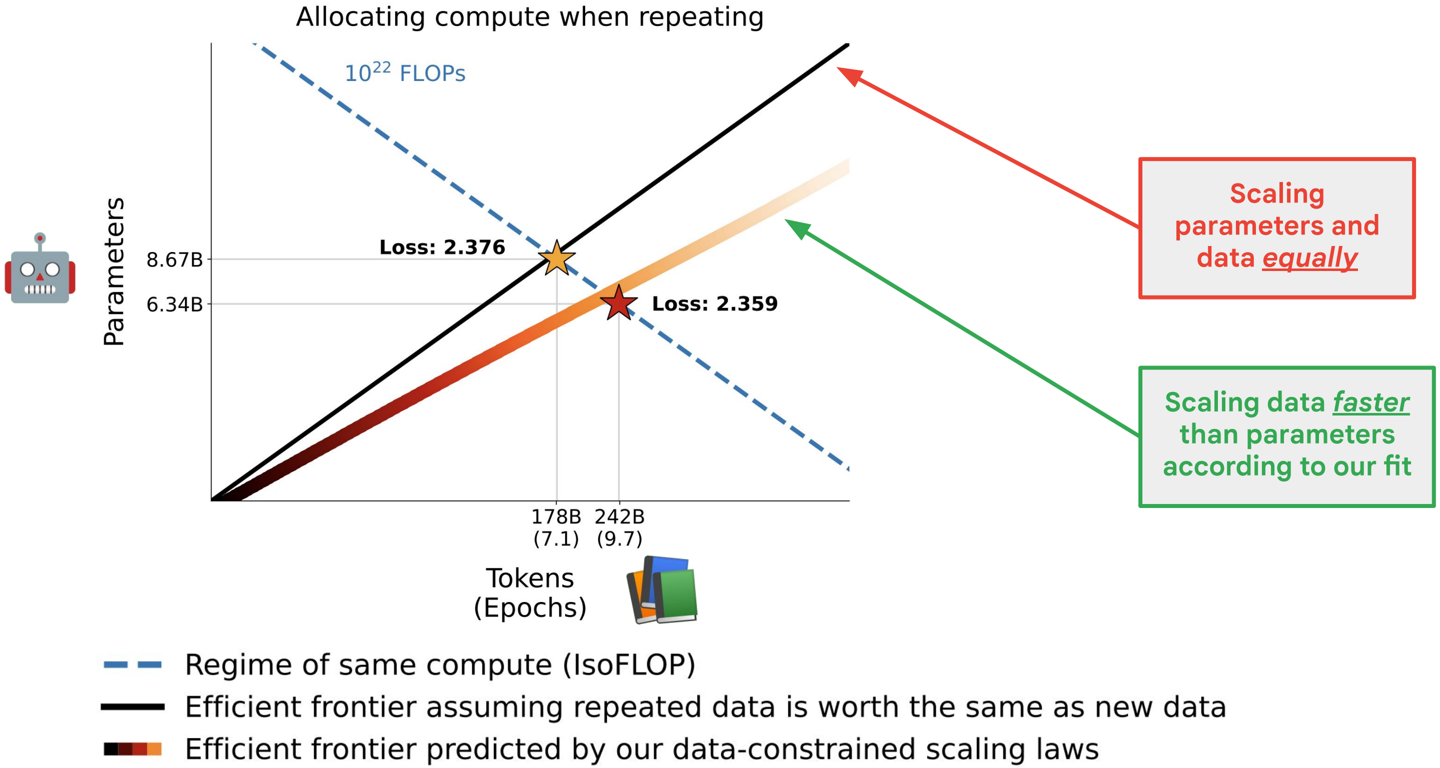
[Training compute-optimal large language models (2022)]()

How to scale when repeating? (**Allocation**)

Training on 100M tokens of unique data with varying model size and data repetitions



Testing our predictions at scale (**Allocation**)



Testing our predictions at scale - Downstream (**Allocation**)

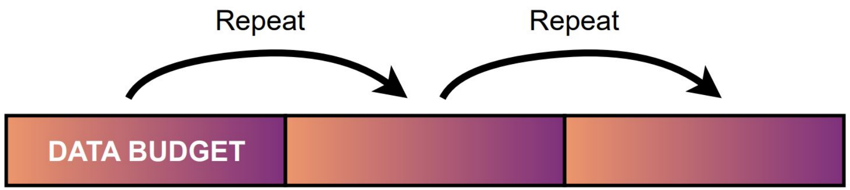
Task	Chinchilla: 8.7B parameters & 7 epochs	Data-Constrained: 6.3B parameters & 10 epochs
HellaSwag*	37.5	38.1
StoryCloze*	66.8	68.4
XSum*	3.0	3.8
...16 other NLP tasks...		
Average	**23.5**	**<u>25.9</u>**

*Average across 0-5 fewshots & rescaled

Complementary strategies to solve data constraints

Thus far:

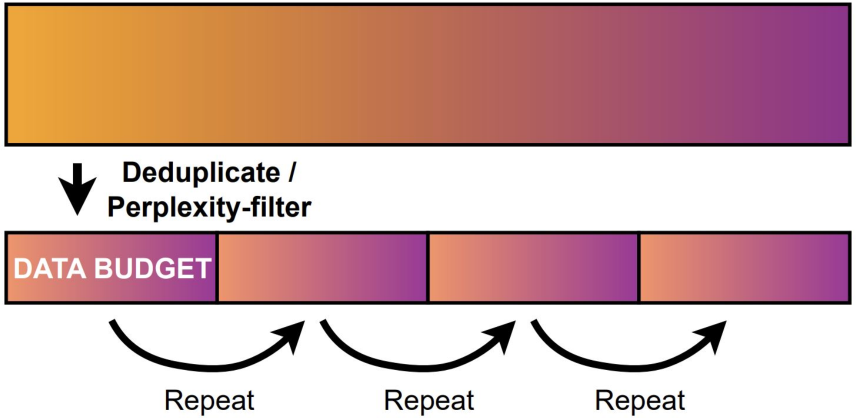
Repeating



Other strategies:

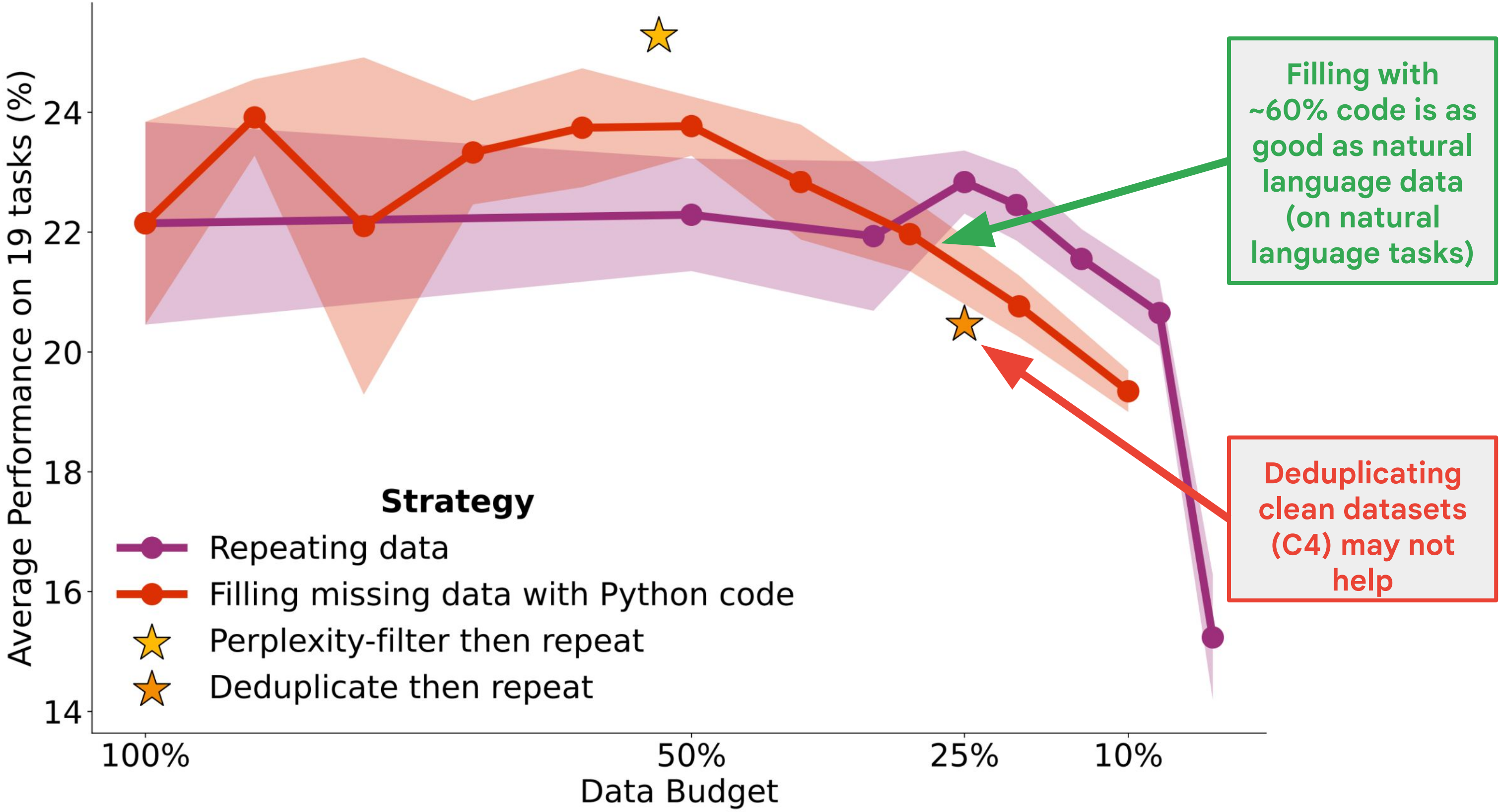
Filling with code

Revise filtering



Complementary strategies to solve data constraints

97 models
trained for 2.1e21
FLOPs each



Takeaway #1

Repeating LLM data ~4x is fine.

Takeaway #2

50% code data is fine.

Takeaway #3

Quality-filtering + repeating can be a good strategy

Scaling Data-Constrained Language Models - Impact

NEURAL INFORMATION PROCESSING SYSTEMS

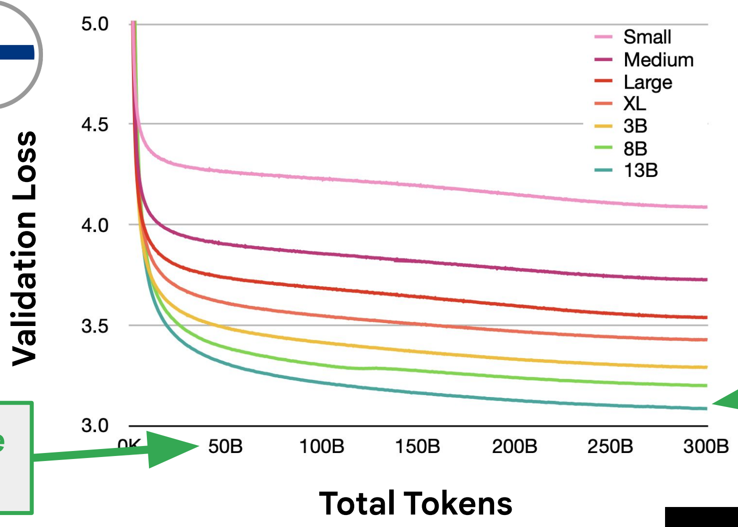
Outstanding Main Track Runner-Ups

Scaling Data-Constrained Language Models

Authors: Niklas Muennighoff · Alexander Rush · Boaz Barak · Teven Le Scao · Aleksandra Piktus · Sampo Pyysalo · Thomas Wolf · Colin Raffel

Poster session 2: Tue 12 Dec 5:15 p.m. — 7:15 p.m. CST, **#813**

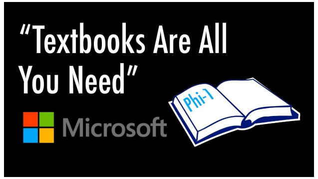
Oral: Tue 12 Dec 3:40 p.m. — 4:40 p.m. CST, Hall C2 (level 1)



38B unique tokens

8 epochs of data

FinGPT: Large Generative Models for a Small Language



PORO

MISTRAL AI_ ?

天工

S.

SILO Language Models

AI2 OLMo
Open Language Model

? Gemini

Thanks!

Twitter: @Muennighoff

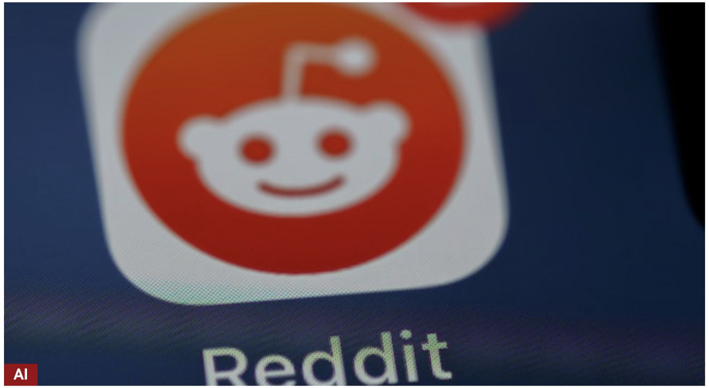
Niklas Muennighoff, Alexander M. Rush, Boaz Barak,
Teven Le Scao, Aleksandra Piktus, Nouamane Tazi,
Sampo Pyysalo, Thomas Wolf, Colin Raffel

Acknowledgements: This work was co-funded by the European Union under grant agreement No 101070350. The authors wish to acknowledge CSC – IT Center for Science, Finland, for generous computational resources on the LUMI supercomputer. We are thankful for the immense support from teams at LUMI and AMD, especially Samuel Antao. Hugging Face provided storage and additional compute instances. This work was supported by a Simons Investigator Fellowship, NSF grant DMS-2134157, DARPA grant W911NF2010021, and DOE grant DE-SC0022199. We are grateful to Harm de Vries, Woojeong Kim, Mengzhou Xia and the EleutherAI community for exceptional feedback. We thank Loubna Ben Allal for help with the Python data and Big Code members for insightful discussions on scaling laws. We thank Thomas Wang, Helen Ngo and TurkuNLP members for support on early experiments.

Appendix

Scaling is **data**-constrained

UPDATED 22:38 EDT / APRIL 18 2023



Reddit to charge for access to its API to counter free data scraping by AI companies

Google Books

Pages 362 to 556 are not shown in this preview.

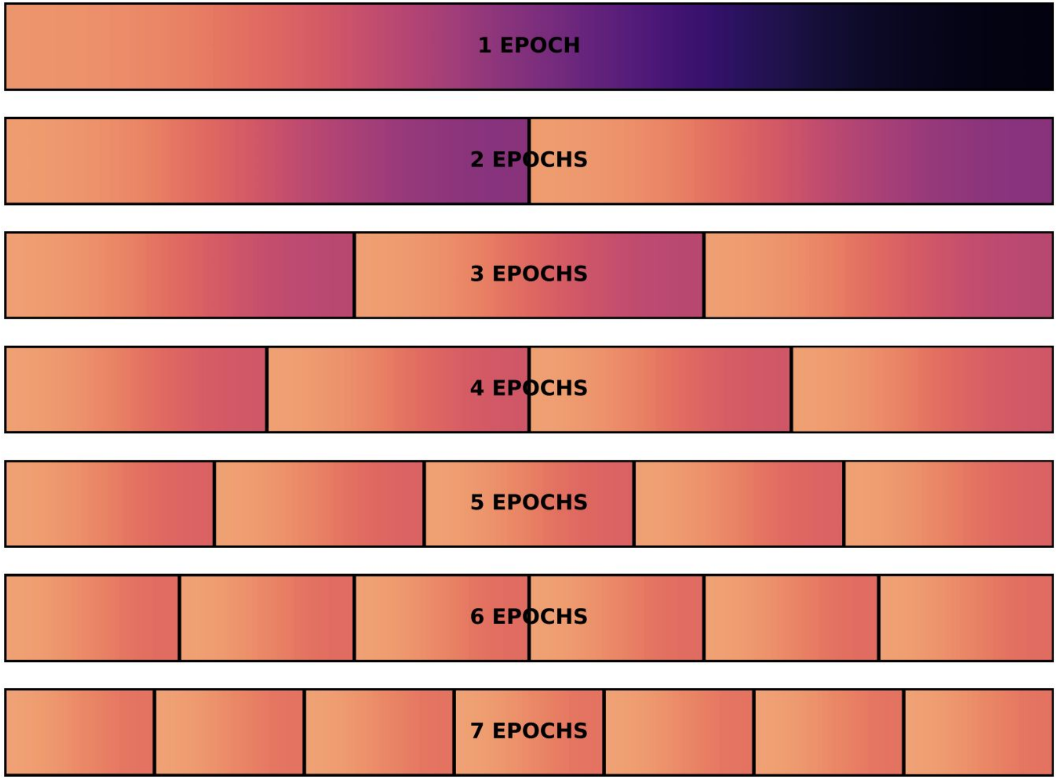
Elon Musk
@elonmusk
Subscribe

To address extreme levels of data scraping & system manipulation, we've applied the following temporary limits:

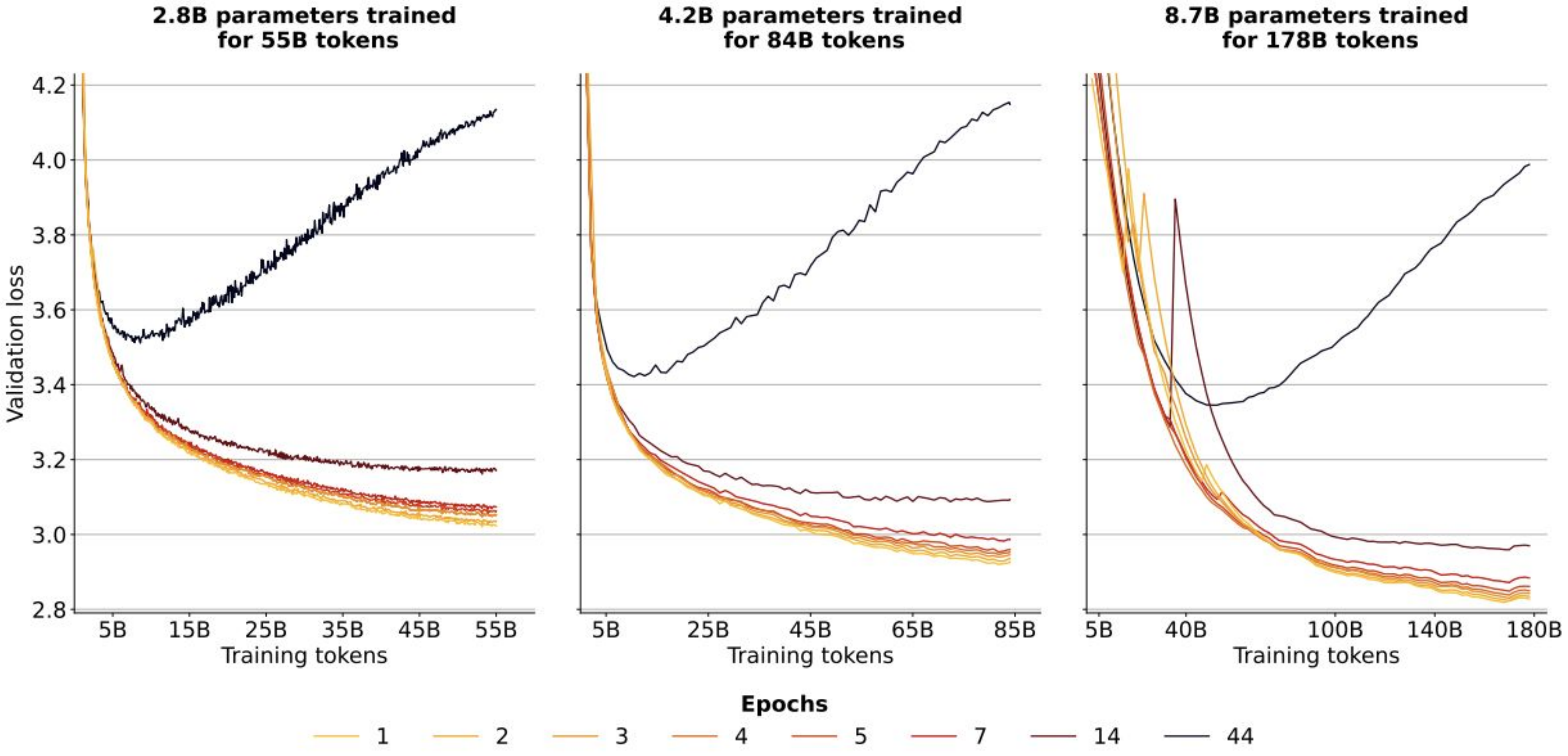
- Verified accounts are limited to reading 6000 posts/day
- Unverified accounts to 600 posts/day
- New unverified accounts to 300/day

1:01 AM · Jul 2, 2023 · **531.6M** Views

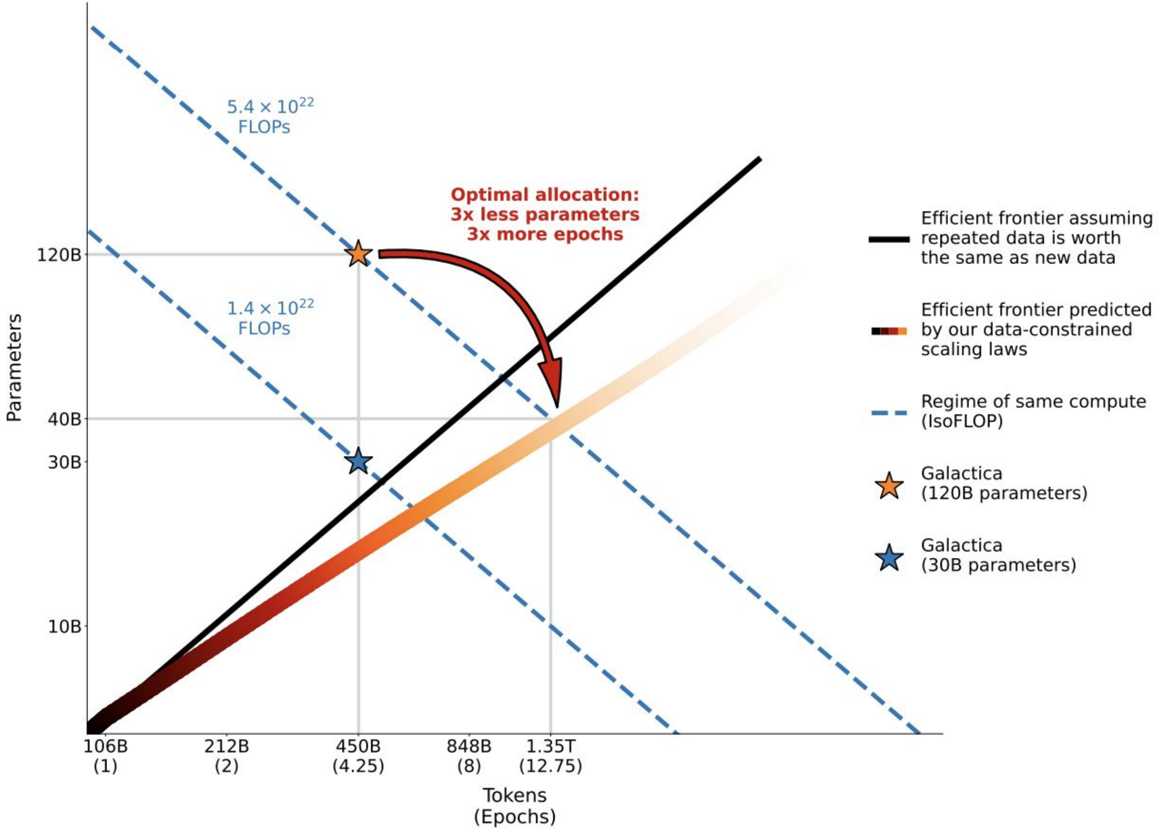
Dataset Setup



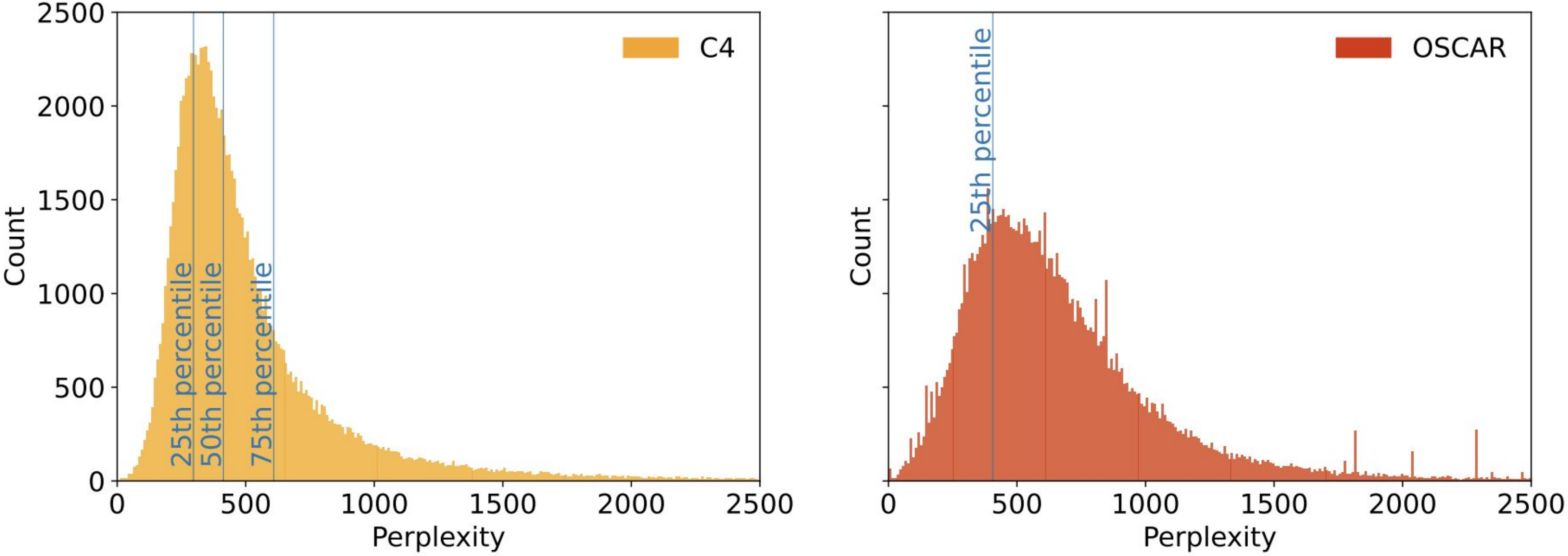
Repeating data on OSCAR (**Return**)



Case Study: Galactica



Perplexity filtering



Approximations

$$D' = U + (1-\delta)U + (1-\delta)^2U + \cdots + (1-\delta)^{R_D}U$$

$$= U + (1-\delta)U\frac{(1-(1-\delta)^{R_D})}{\delta}$$ ***(Geometric Series)***

Let $R_D^* = \frac{1-\delta}{\delta}$ & $(1-\delta) \approx e^{-\delta} \approx e^{-1/R_D^*}$

$$\Rightarrow \quad D' = U + U \cdot R_D^* \cdot (1 - e^{-R_D/R_D^*})$$