Scaling Laws for LLMs

Haw-Shiuan Chang

Some slides from Mohit Iyyer

Overview

- What Are Scaling Laws?
- Data or Size?
- Power law vs linear-log?
- Where does Scaling Law Come from?
- Why does the Scaling Laws Matter?

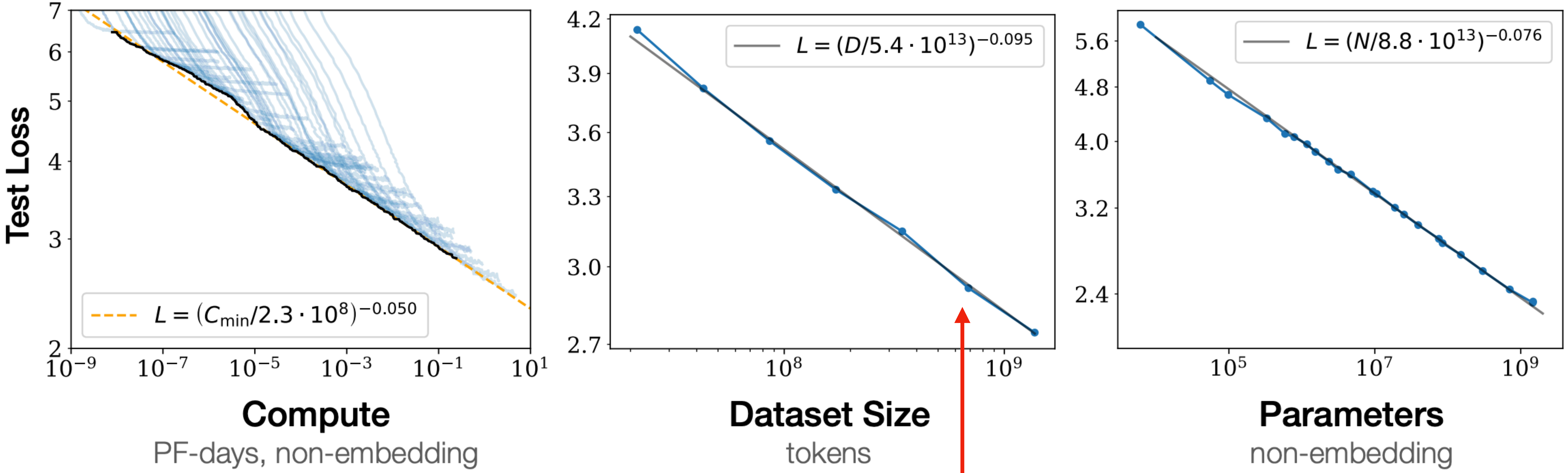
LLM Size

- Why does LLM or LRM need to be large?
  - Store more high-quality responses for SFT or reinforcement learning
  - Closer training and testing distribution

Can we quantify the effect of increasing the size of LLM?

What are Scaling Laws?

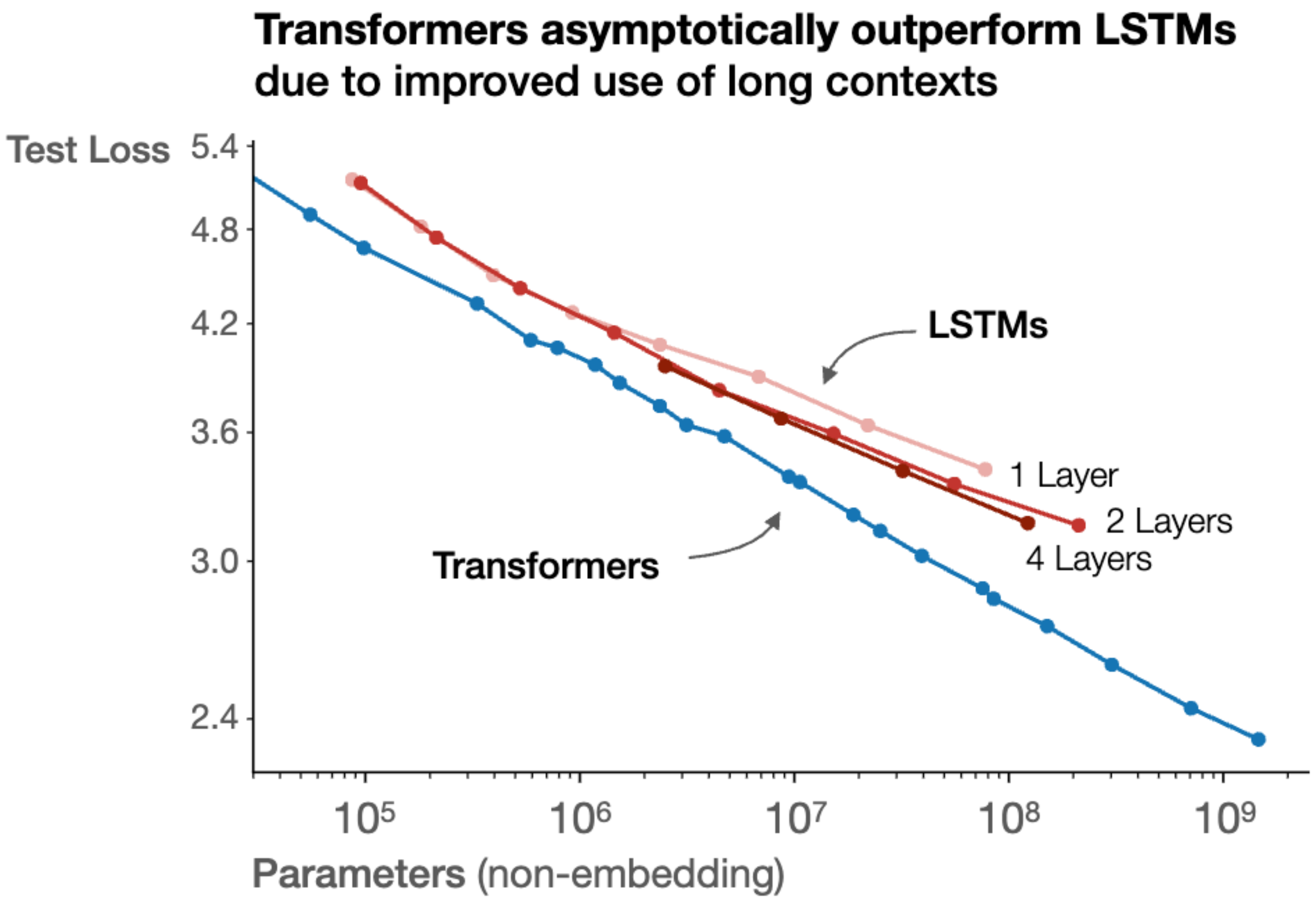
$$PP(W) = \exp\Big(-\frac{1}{N}\sum_{i}^{N} \log p(w_i \mid w_{<i})\Big) \qquad L = -\frac{1}{N}\sum_{i}^{N} \log p(w_i \mid w_{<i})$$

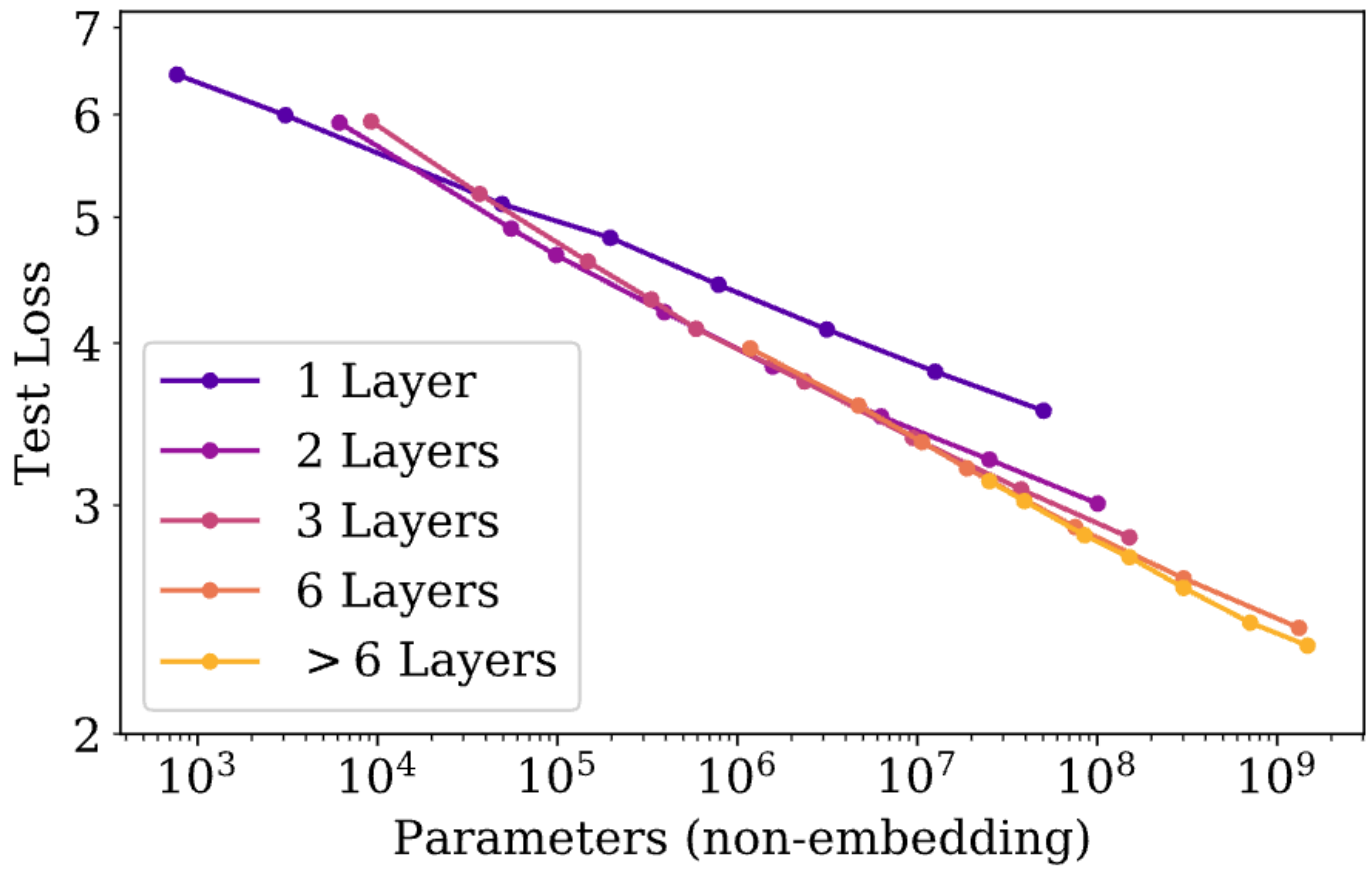


Linearly decay as the size exponentially increases -> a linear-log function

Kaplan et al., 2020

Architectures and Scaling Laws





Observations from Kaplan et al., 2020

- Performance depends strongly on scale (model params, data size, and compute used for training), weakly on model shape (e.g., depth, width)

- Perf vs scale can be modeled with power laws

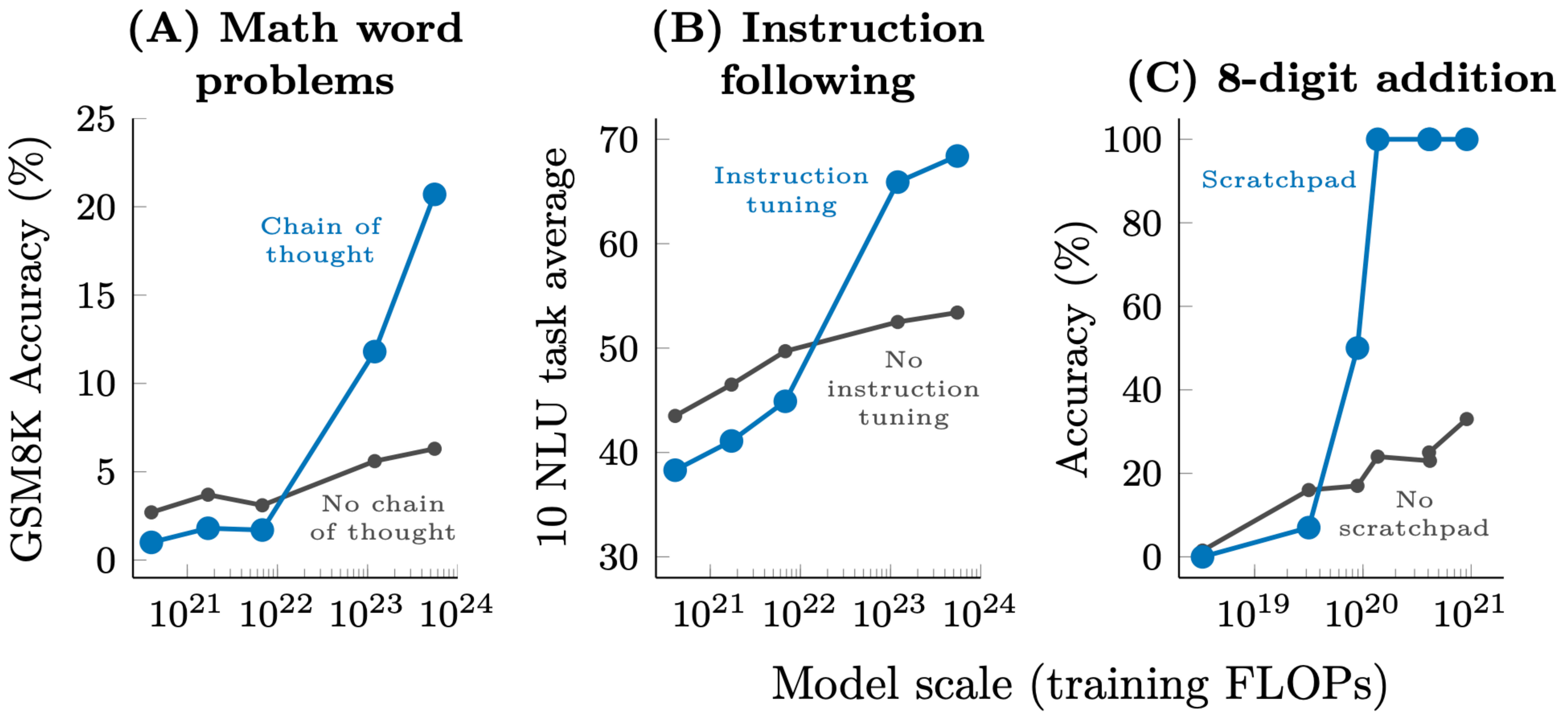
$$L(N) = (N_c/N)^{\alpha_N}$$

Not linear-log. Will explain later

- Perf improves most if model size and dataset size are scaled up together. Increasing one while keeping the other fixed leads to diminishing returns

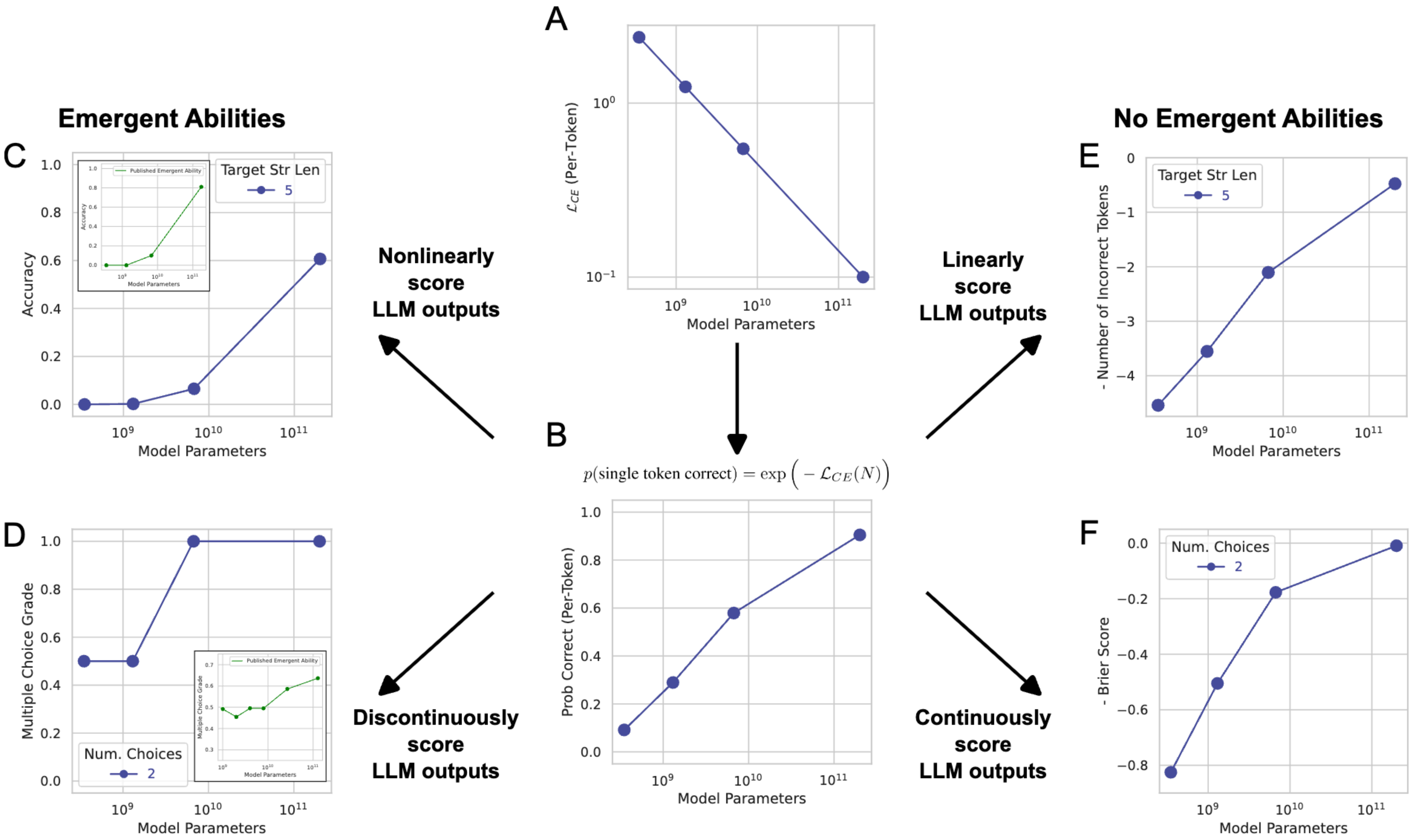
- Larger models are more sample efficient than smaller models, take fewer steps / data points to reach same loss

Emerging Behavior



Emergent Abilities of Large Language Models, Wei et al., TMLR 2022

In Many Cases, still Linear



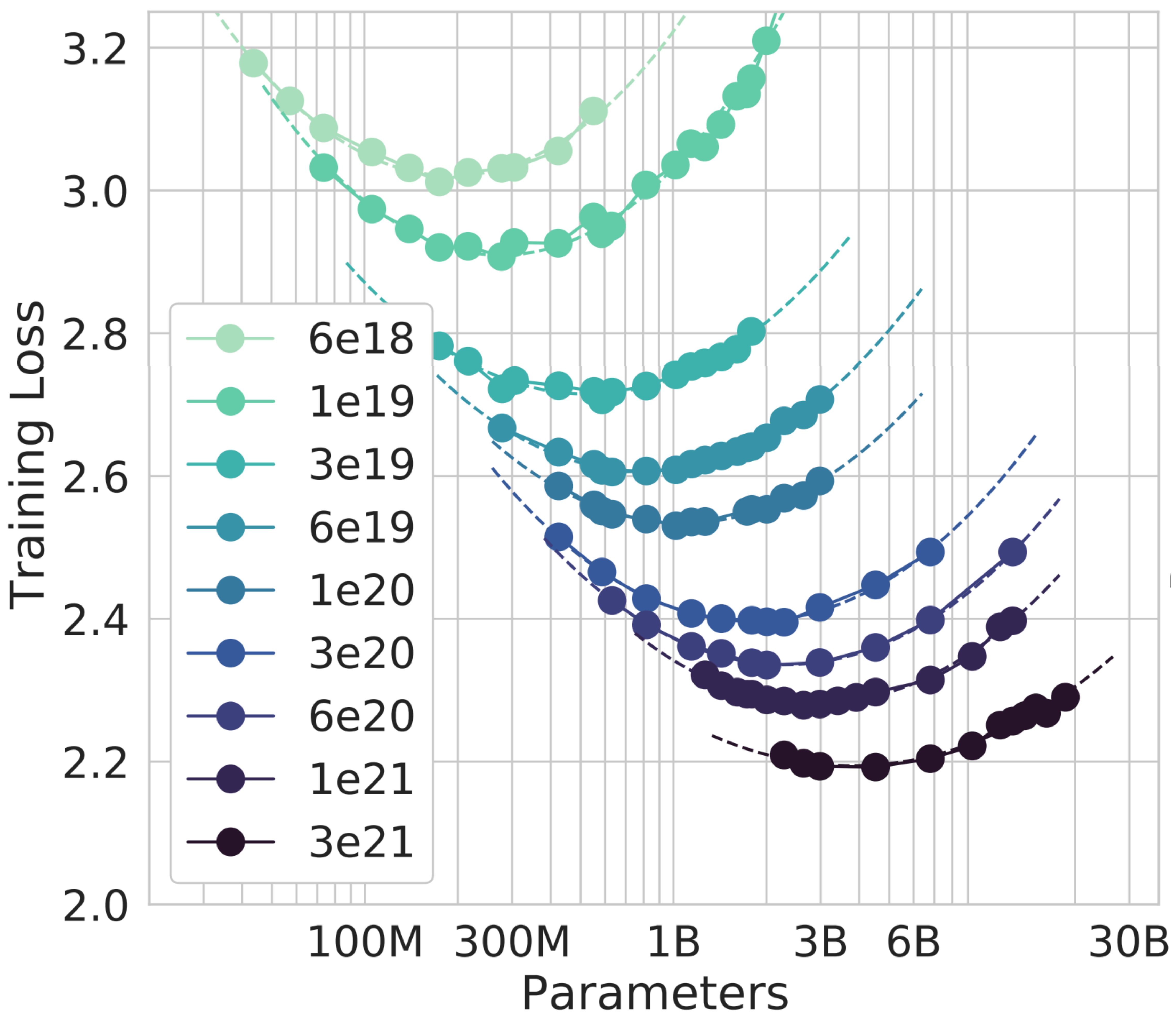
Are Emergent Abilities of Large Language Models a Mirage? (https://arxiv.org/pdf/2304.15004)

Data or Size

Let’s say you can use one GPU for one day

- Would you train a 5 million parameter LM on 100 books?
- What about a 500 million parameter LM on one book?
- Or a 100k parameter LM on 5k books?

Chinchilla (Hoffmann et al., 2022)



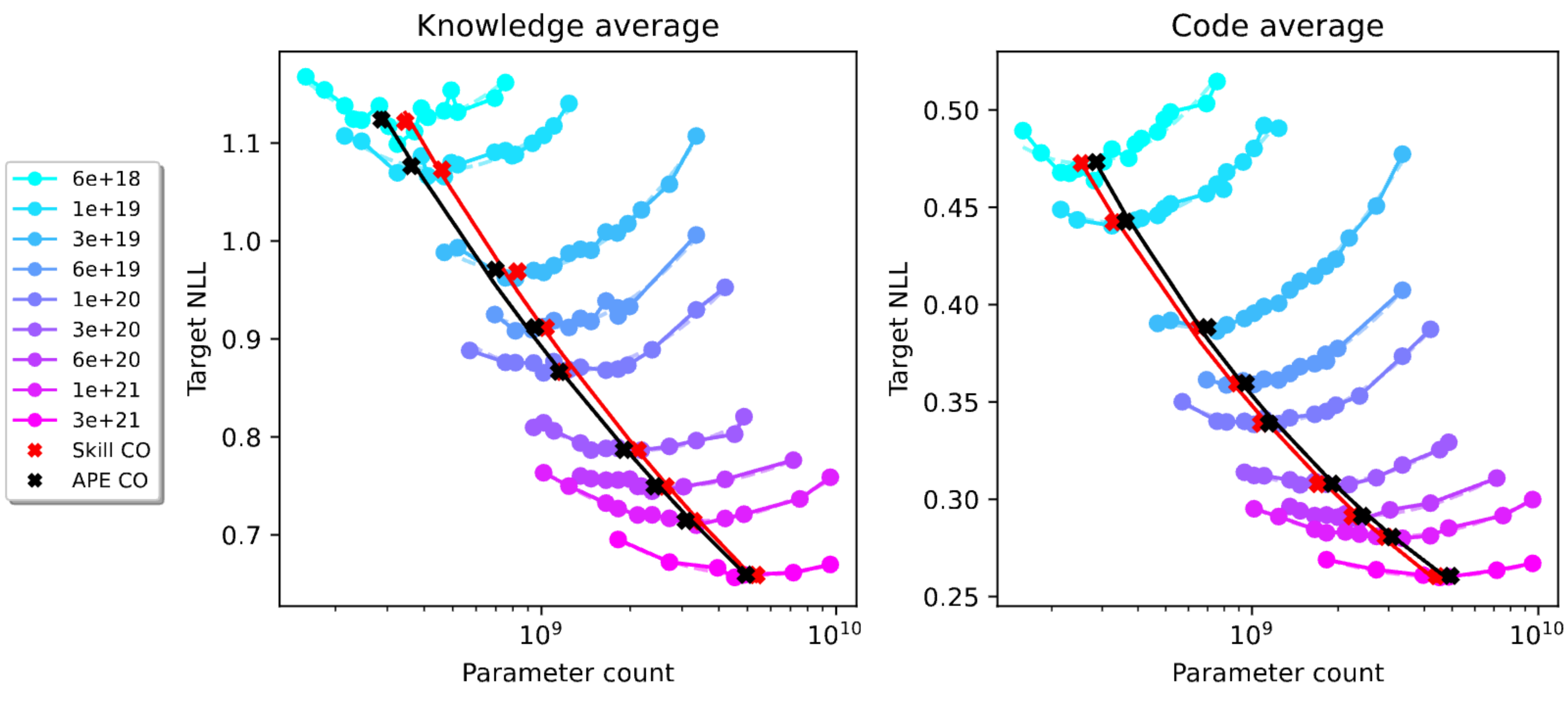
Quick takeaways

- **Kaplan et al., 2020**: if you're able to increase your compute budget, you should prioritize increasing model size over data size
  - With a 10x compute increase, you should increase model size by 5x and data size by 2x
  - With a 100x compute increase, model size 25x and data 4x
- **Hoffmann et al., 2022**: you should increase model and data size at the same rate
  - With a 10x compute increase, you should increase both model size and data size by 3.1x
  - With a 100x compute increase, both model and data size 10x

Conclusion

- To leverage the GPU resources optimally, when the model size becomes 10x larger, the training corpus size needs to be **at least** 10x larger.
- Intuitively, it is because 10x larger models could memorize 10x more things
- In practice, the training corpus is often much larger than this Chinchilla recommendation.
    - Why?

Some Tasks need Larger Models



(a) Knowledge QA.

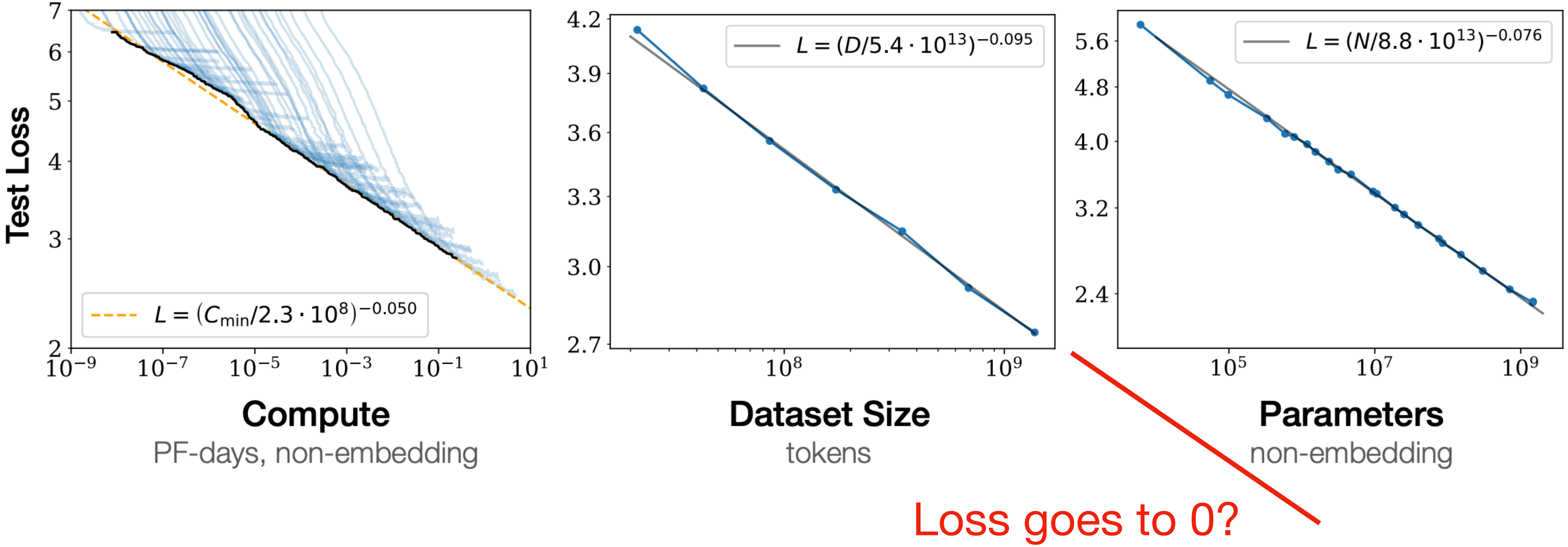
(b) Code.

Compute Optimal Scaling of Skills: Knowledge vs Reasoning (https://arxiv.org/pdf/2503.10061)

Power Law vs Linear Log

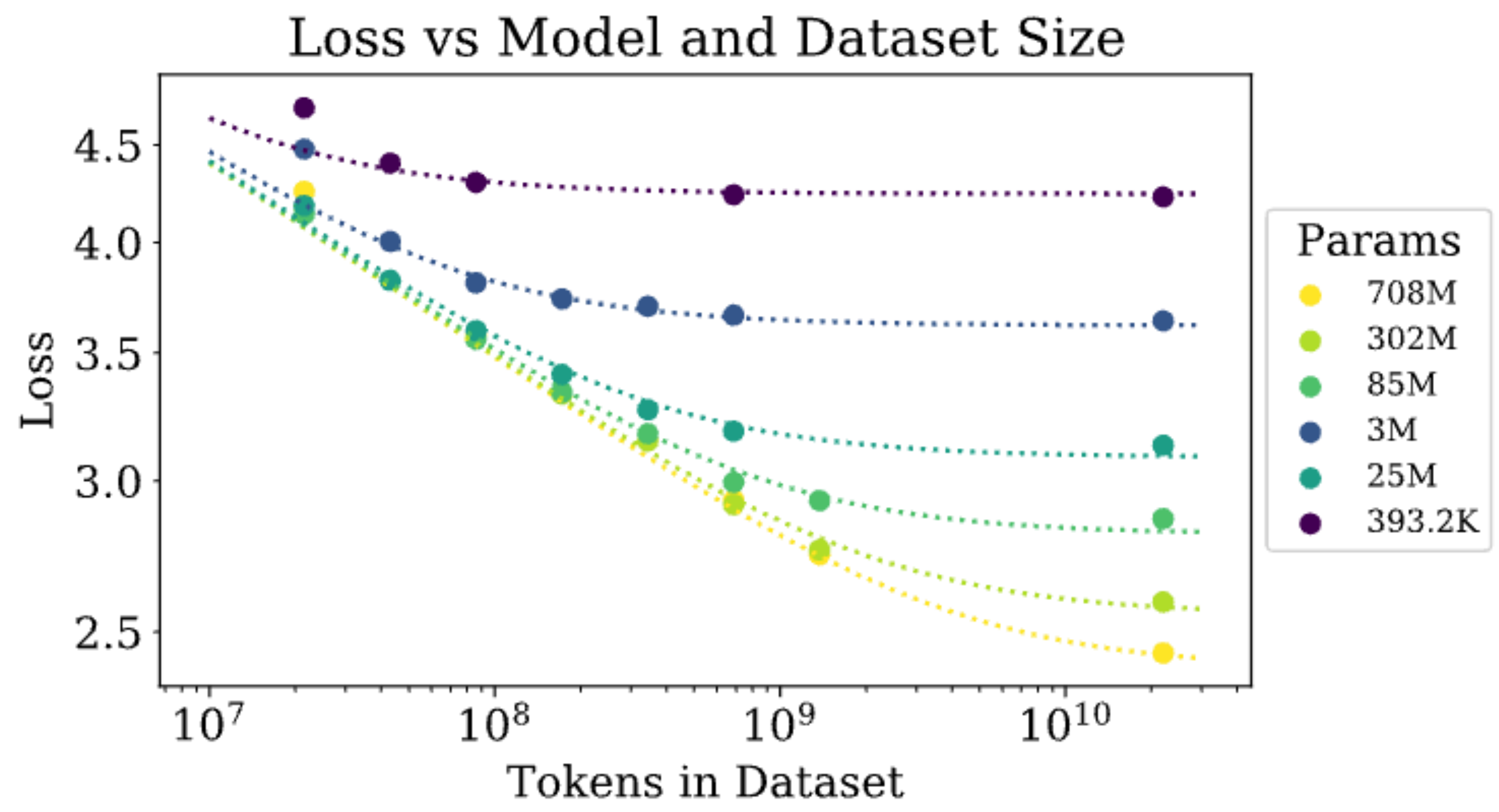
Scaling Laws?

$$PP(W) = \exp\left(-\frac{1}{N}\sum_{i}^{N}\log p(w_i \mid w_{<i})\right) \qquad L = -\frac{1}{N}\sum_{i}^{N}\log p(w_i \mid w_{<i})$$



Kaplan et al., 2020

Power Law



Locally linear-log function

Optimal loss

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

Model size

Dataset Size

Derivation

Power law

$$L(N) = (N_c/N)^{\alpha_N}$$

Example

$$y = x^{-0.01}$$

$$x^{-0.01} = e^{-0.01\ln(x)}$$

Taylor Expansion

$$e^a \approx 1 + a + \frac{a^2}{2!} + \cdots$$

Ignore (0.01ln(x))^2

$$e^{-0.01\ln(x)} \approx 1 - 0.01\ln(x)$$

Linear-log function

Where does the Scaling Law Come from?

The midterm won’t cover this

Why does the ability increase linearly as the size increases exponentially?

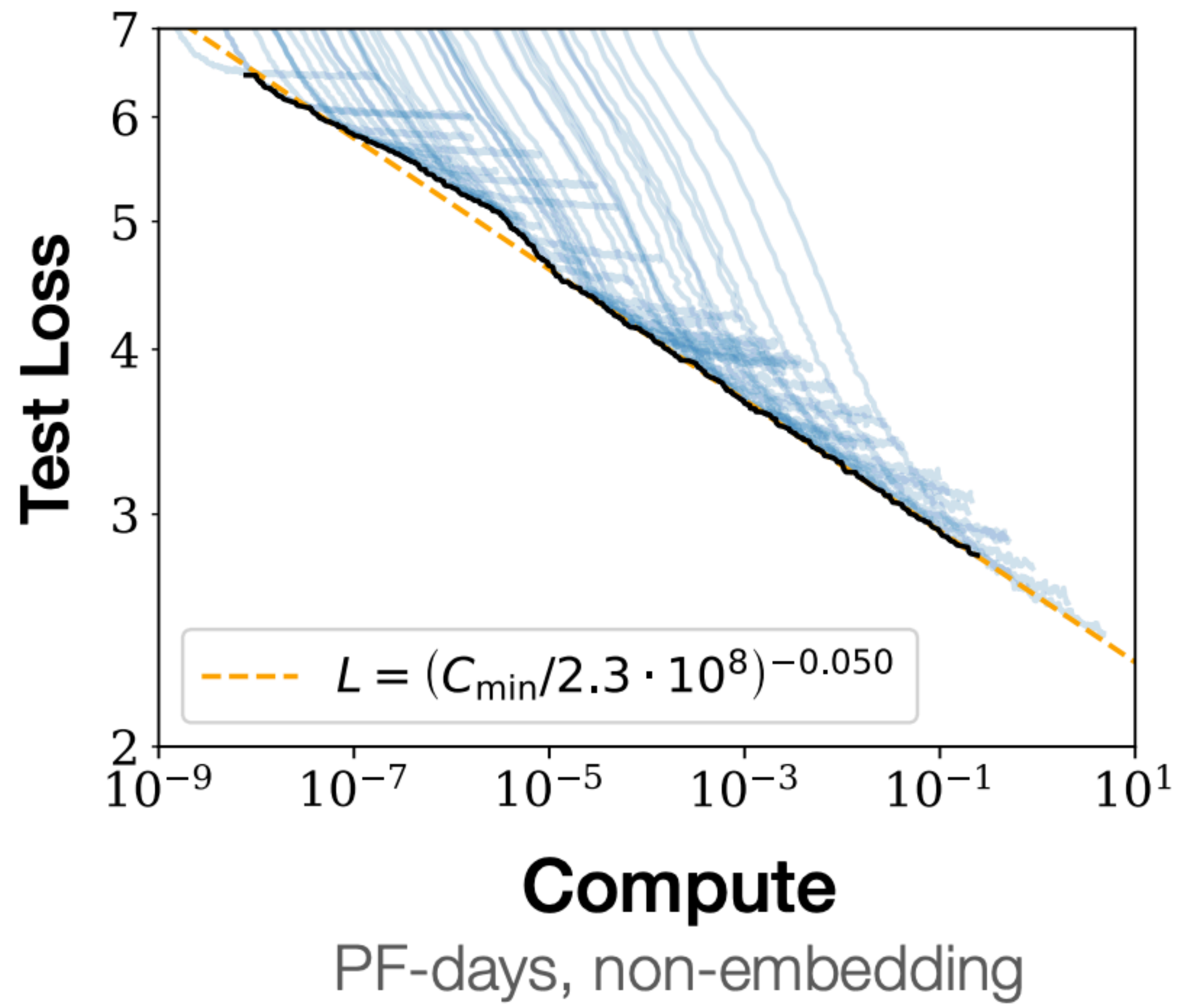


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

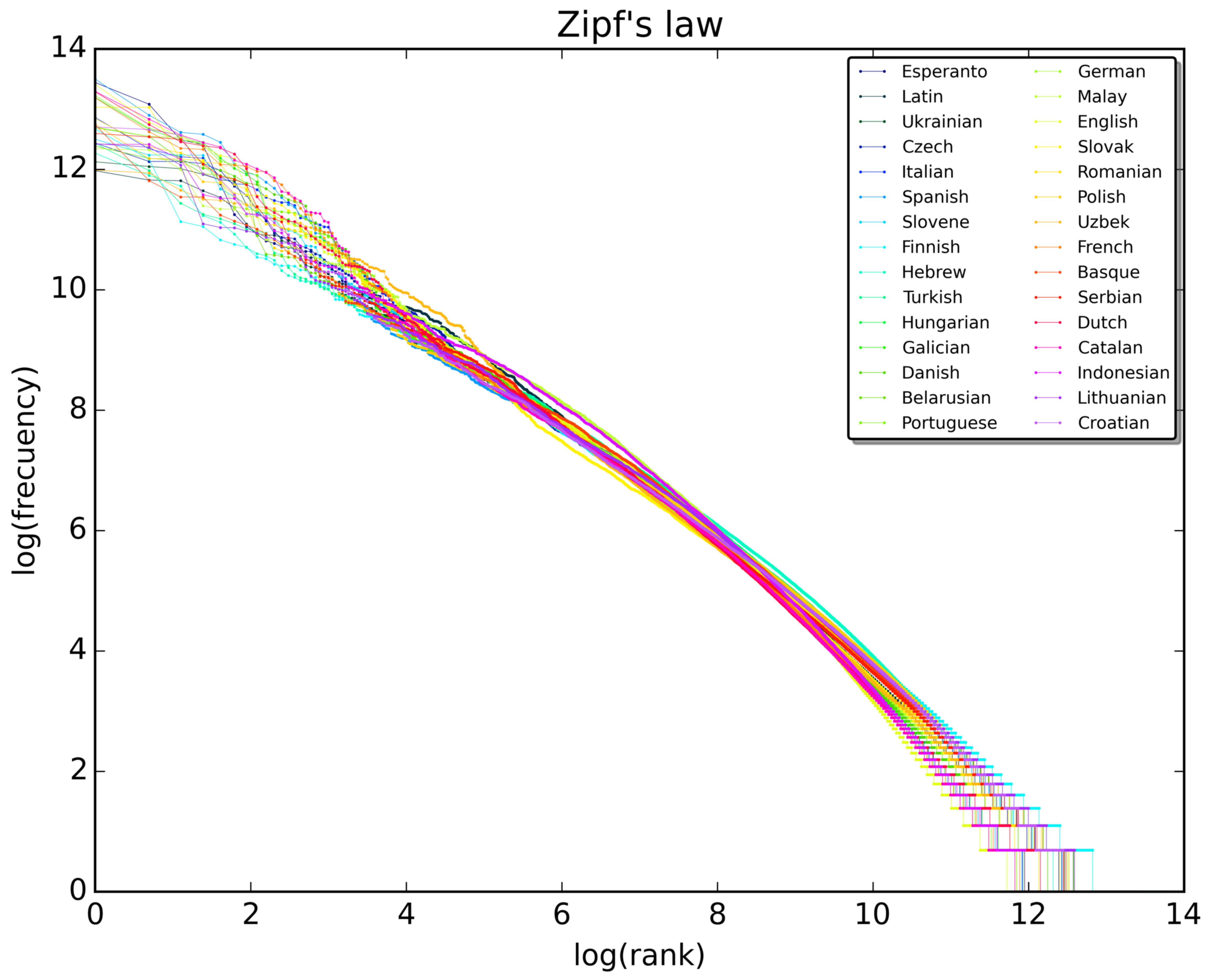
Kaplan et al., 2020

There are many different theories and I want just to use an example to give you some intuitions

Zipf Law

Prob of observations	Num words	Total Prob
0.01	10 words	0.1
0.001	100 words	0.1
0.0001	1000 words	0.1

...



https://en.wikipedia.org/wiki/Zipf%27s_law

Different Corpus Sizes

Prob of observations	Num words	Total Prob	Freq(w) for Size 1000	Freq(w) for Size 10000	Freq(w) for Size 100000
0.01	10 words	0.1	10	100	1000
0.001	100 words	0.1	1	10	100
0.0001	1000 words	0.1	0.1	1	10
			…		
			Success rate = 10%	Success rate = 20%	Success rate = 30%

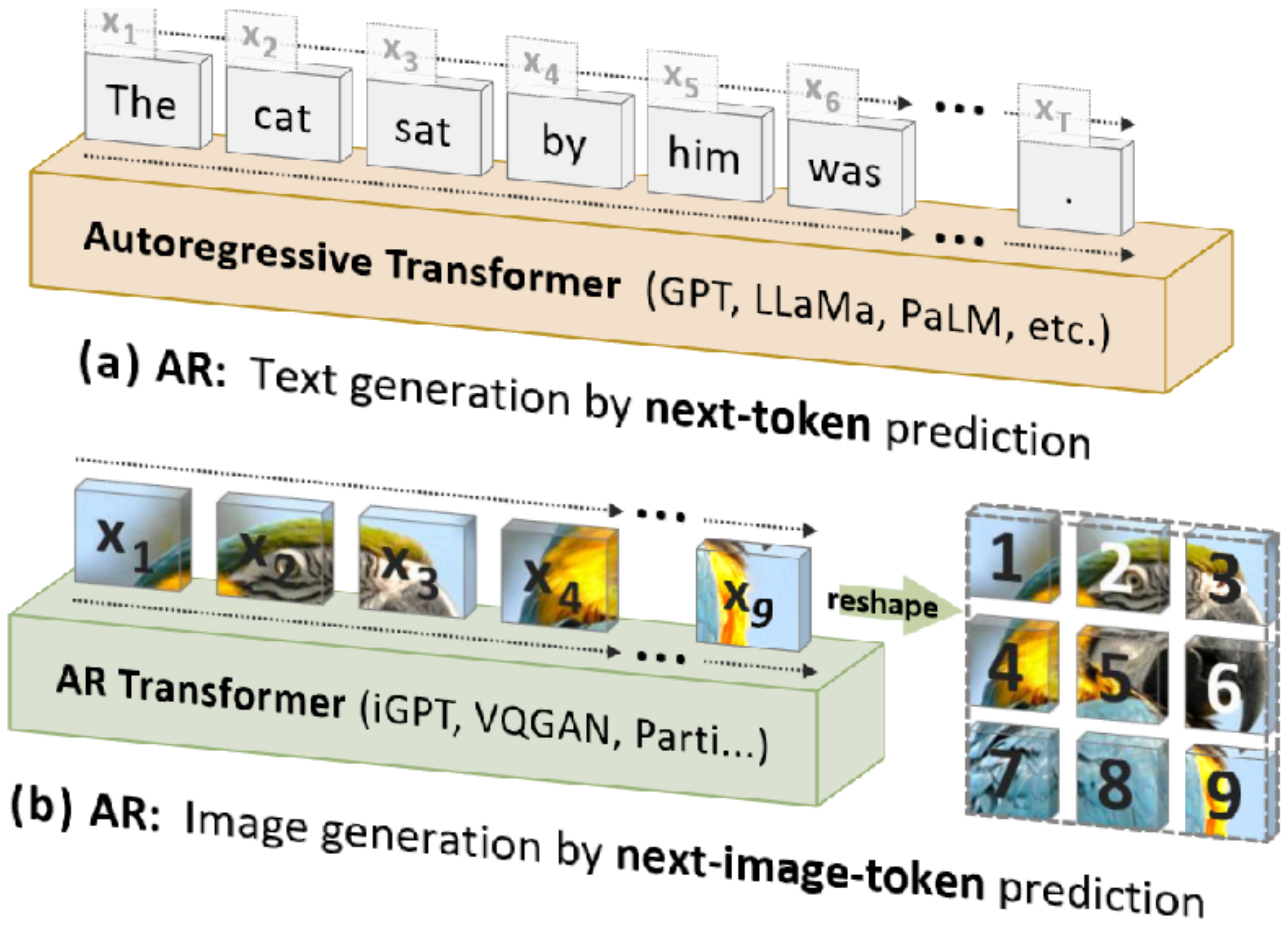
Maybe the reasoning ability is rare, so it might "emerge"

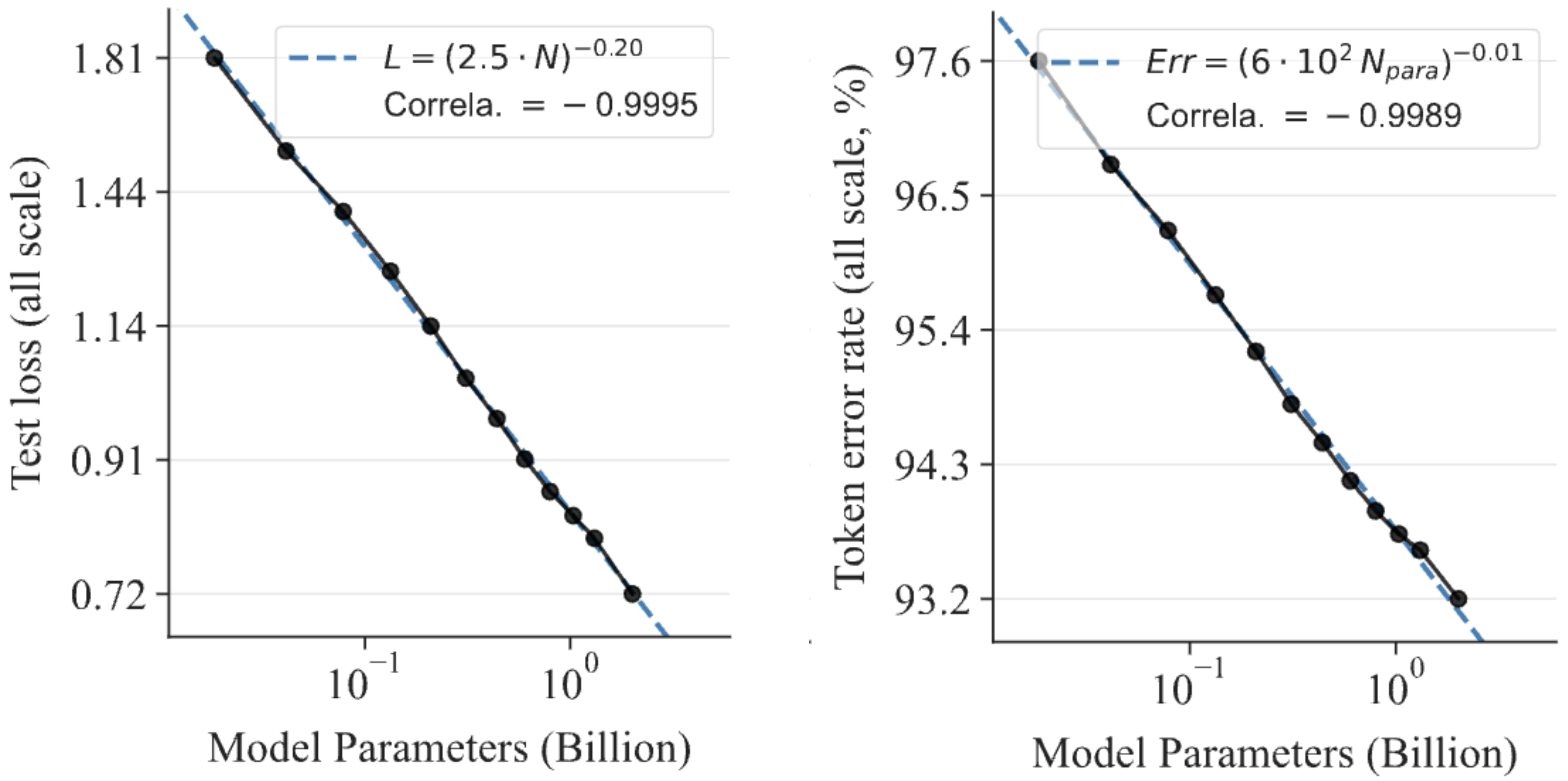
Instruction: Please explain a randomly sampled word w

Assuming LLM can only do that after seeing at least 10 times

Scaling Laws is Everywhere

- NLP
- Vision
- IR/recommendation
- …





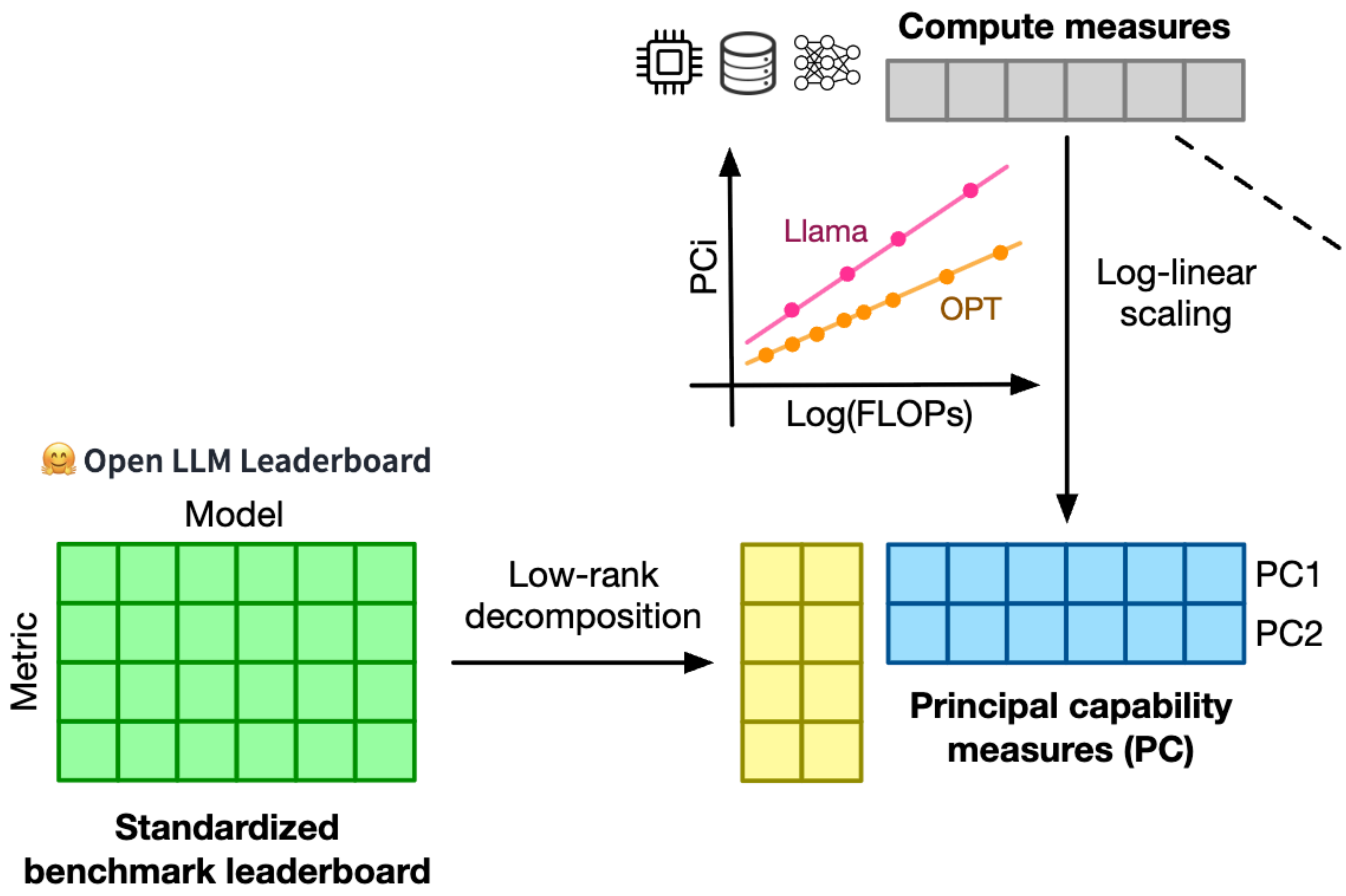
Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction (https://openreview.net/pdf?id=gojL67CfS8)

Why?

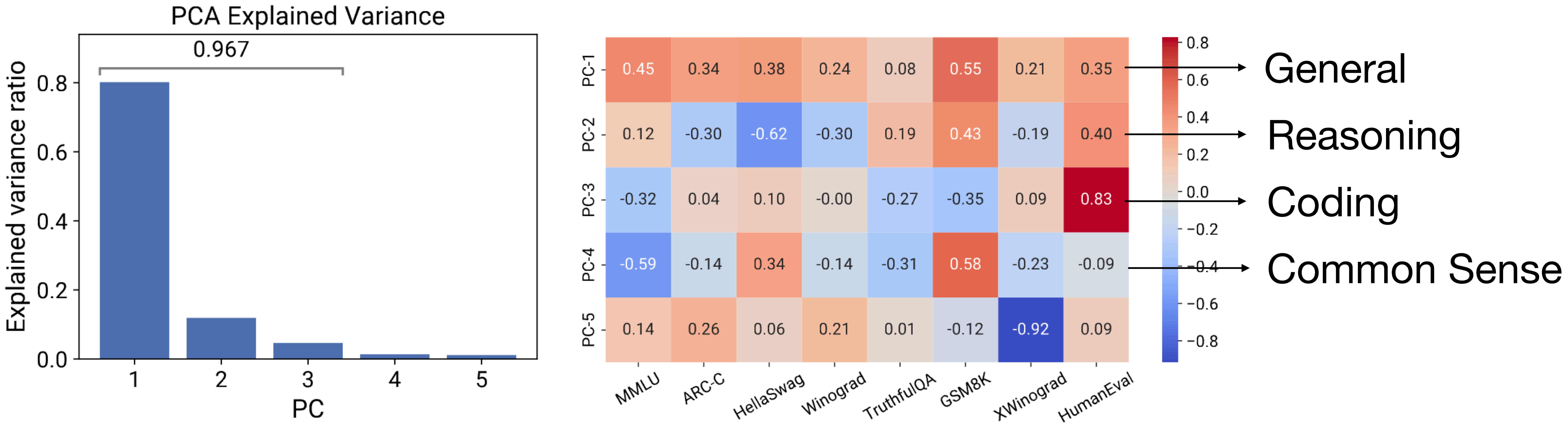
Still an Open Question

Why does Scaling Law Matter?

The midterm won’t cover this



Observational Scaling Laws and the Predictability of Language Model Performance (https://arxiv.org/pdf/2405.10938)



(a) PCA explained variance

(b) Principal component weights

Figure 2: Just a few capability dimensions explain most variability on a diverse range of standard LM benchmarks. We find that (a) the benchmark-model matrix is **low-dimensional** with the top 3 PCs explaining $\sim 97\%$ of the variance and (b) the PCs are **interpretable**: PC-1, PC-2, and PC-3 emphasize LMs' general, reasoning, programming capabilities, respectively.

Scaling Law for General Performance

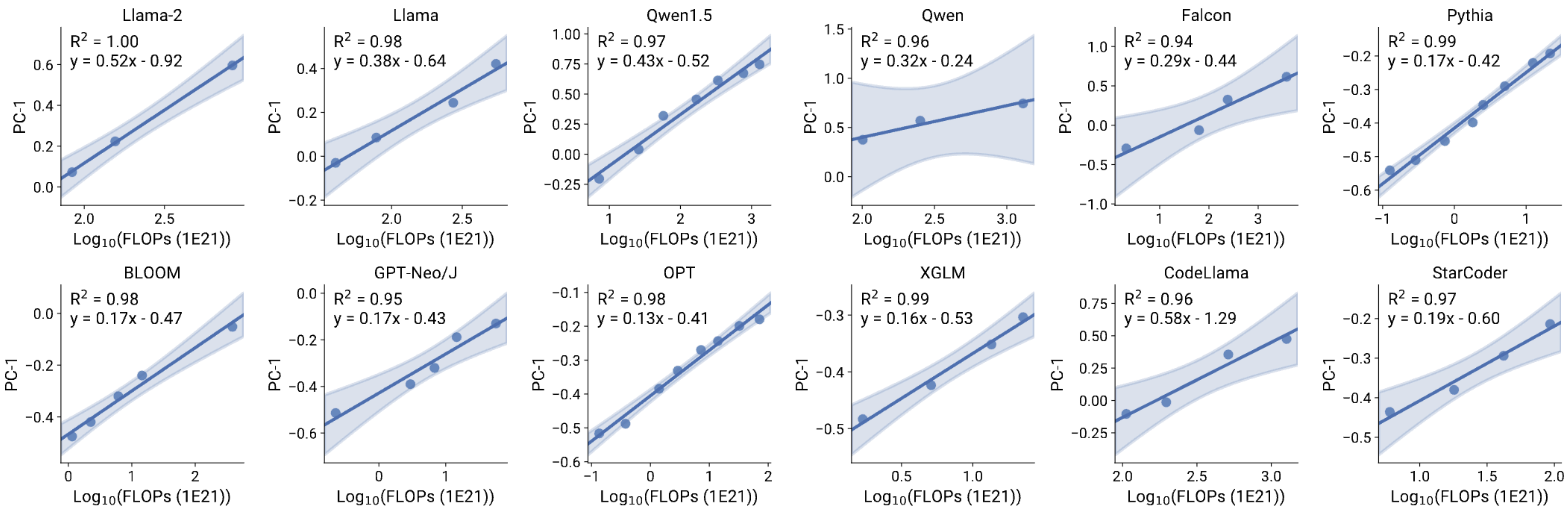


Figure 3: The extracted PC measures *linearly correlate* with log-compute within each model family. The linearity generally holds for various model families, and also for lower-ranked PCs (Fig. C.2).