Training Compute-Optimal Large Language Models

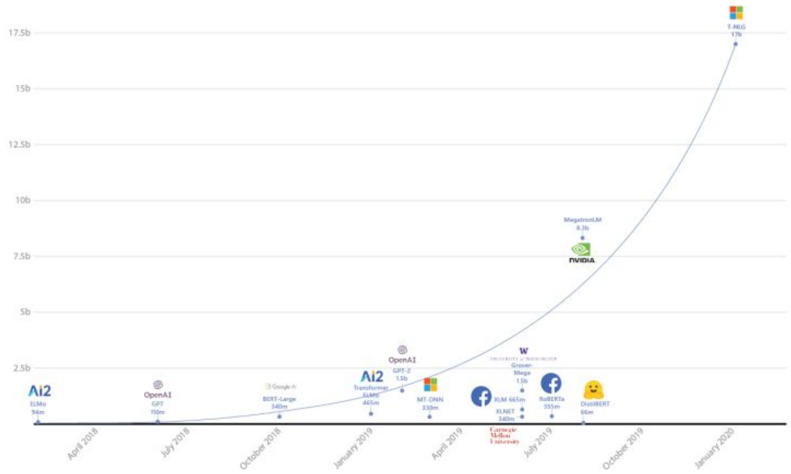
By Anika Maskara and Simon Park

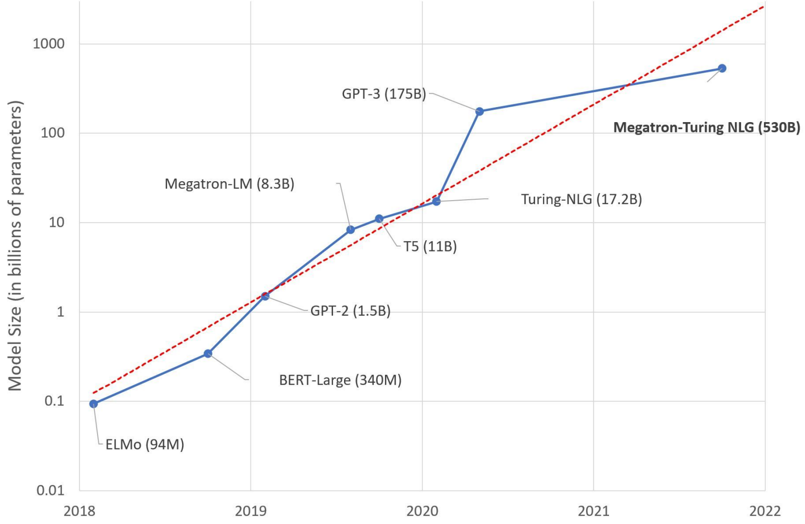
10/24/2022

Outline

1. **Introduction**
2. Initial Scaling Law (Kaplan et al., 2020)
3. Modified Scaling Law (Hoffman et al., 2022)
4. Chinchilla (Hoffman et al., 2022)
5. Beyond Scaling Law

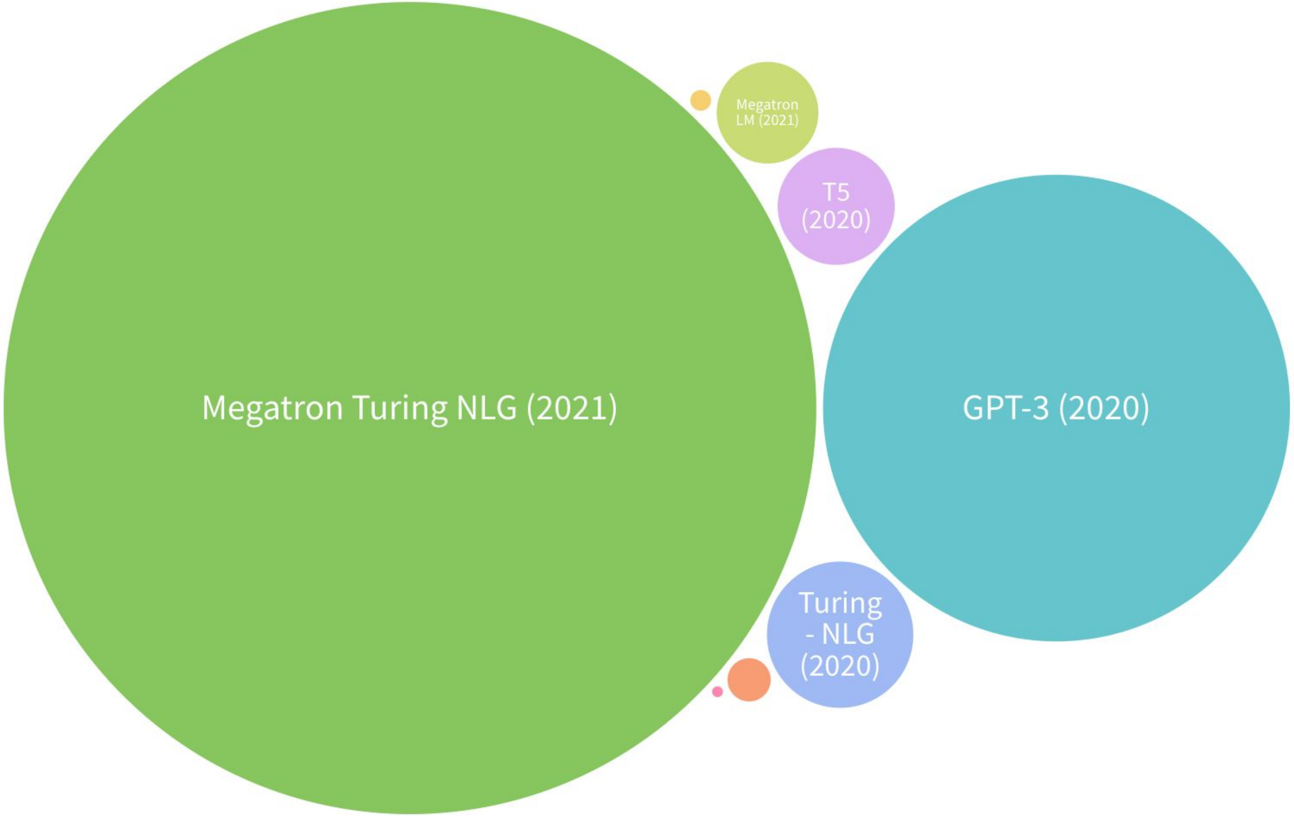
Language Models have been Getting Bigger…





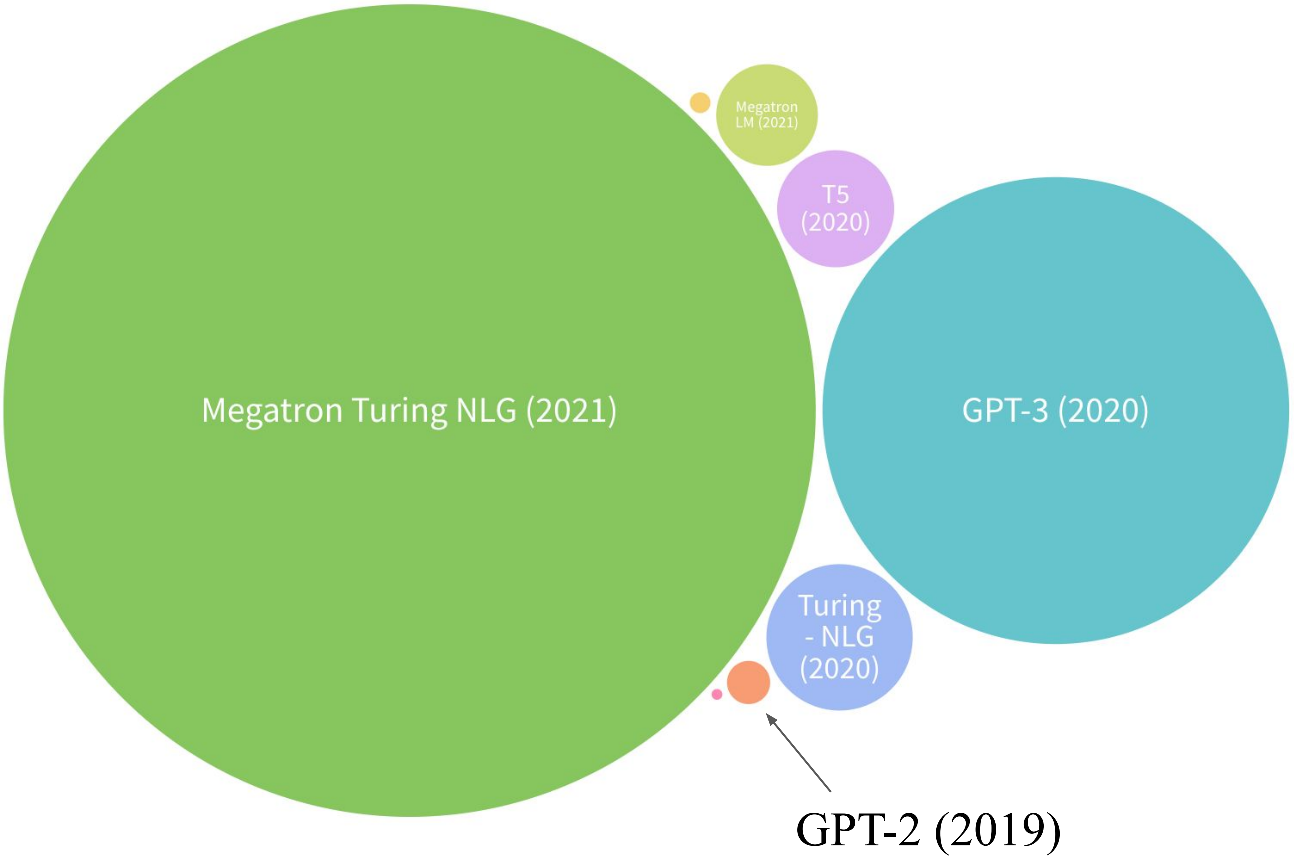
[Image Source] [Image Source]

…..a lot bigger

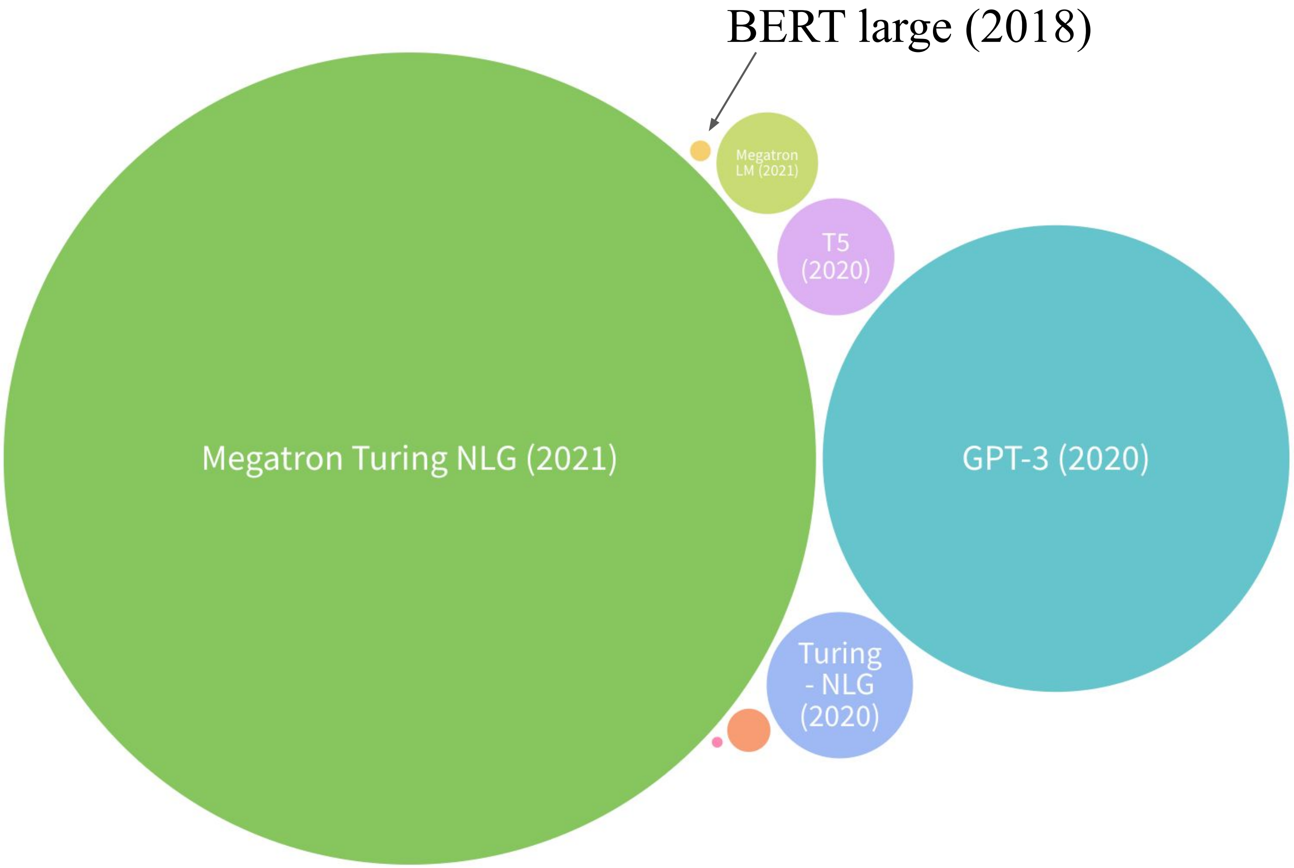


Megatron LM (2021)
T5 (2020)
Megatron Turing NLG (2021)
GPT-3 (2020)
Turing - NLG (2020)

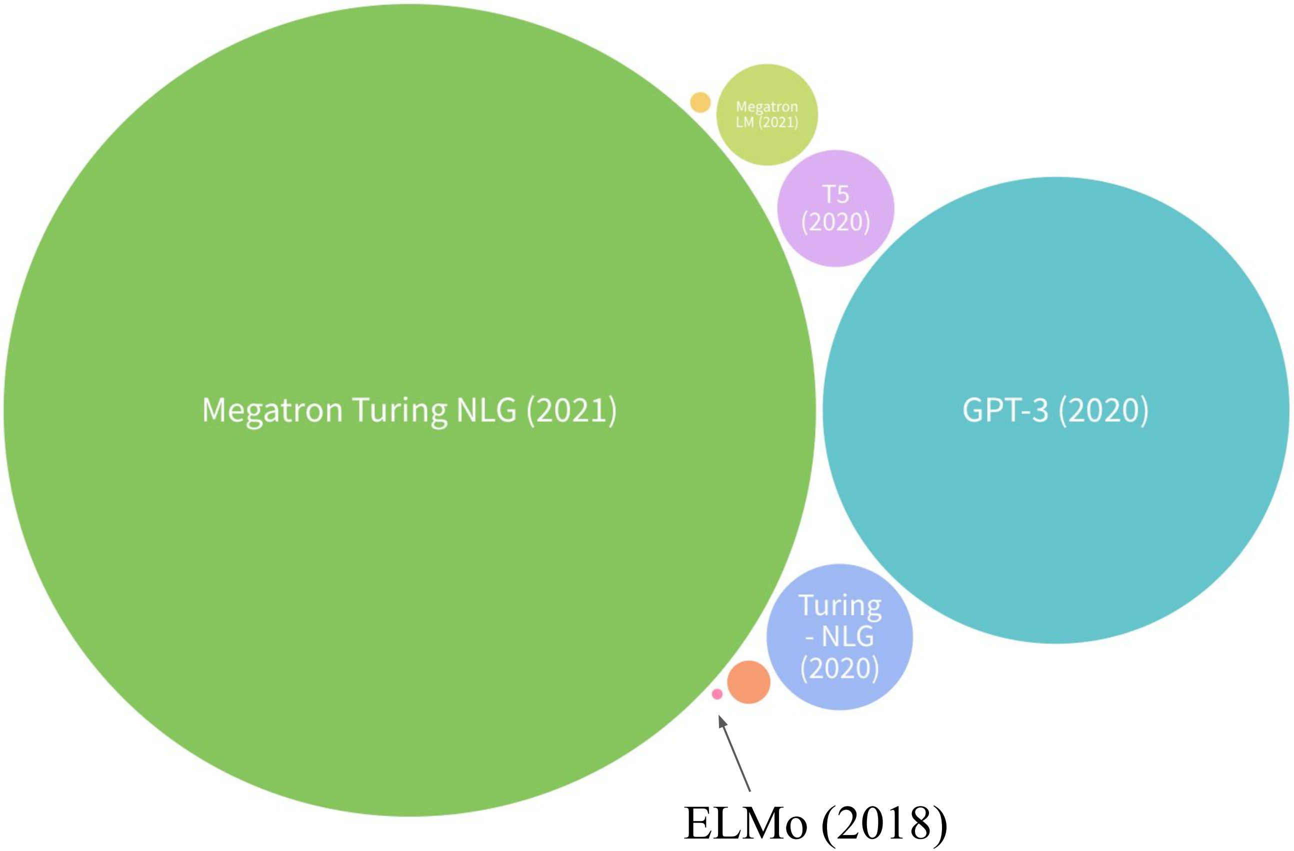
…..a lot bigger


Megatron LM (2021)
T5 (2020)
Megatron Turing NLG (2021)
GPT-3 (2020)
Turing - NLG (2020)
GPT-2 (2019)

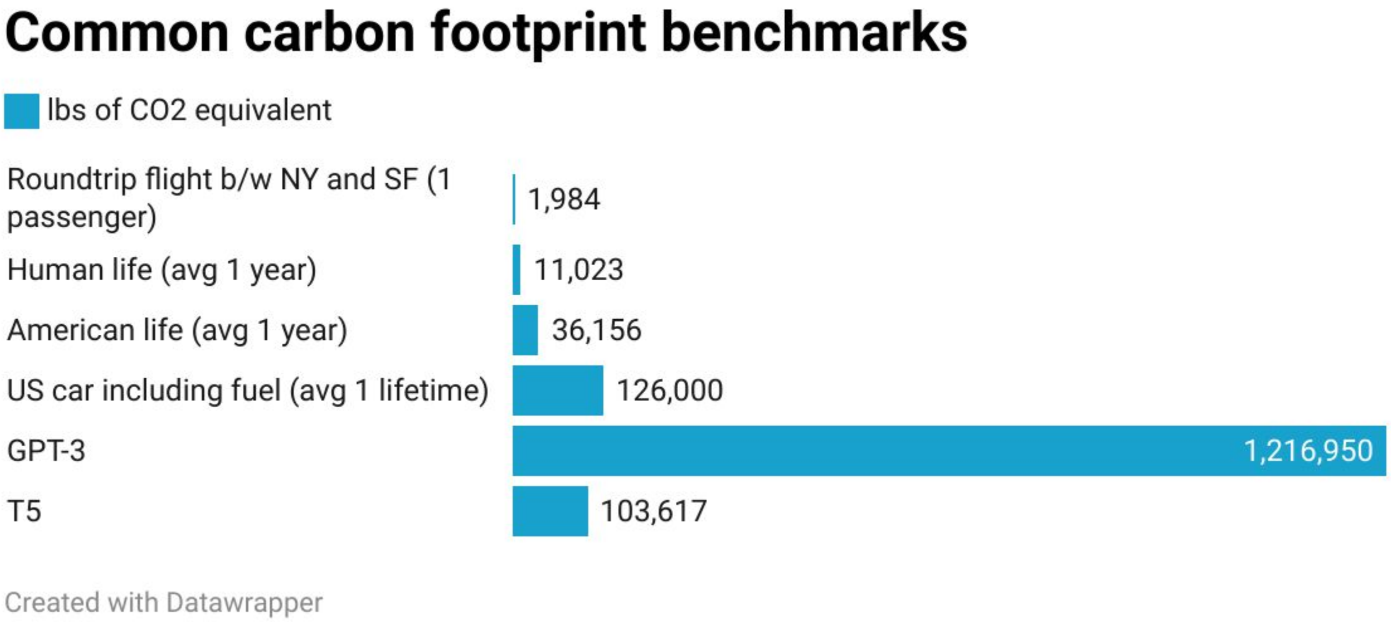
…..a lot bigger


BERT large (2018)
Megatron LM (2021)
T5 (2020)
Megatron Turing NLG (2021)
GPT-3 (2020)
Turing - NLG (2020)

…..a lot bigger


Megatron Turing NLG (2021)
Megatron LM (2021)
T5 (2020)
GPT-3 (2020)
Turing - NLG (2020)
ELMo (2018)

Q1: Why do we care about studying scaling law of LLMs?



[Data Source: (Strubell et al., 2019)] [Data Source: (Patterson et al. 2021)]

Big Models Require Big Pockets, and not just at training

Sources estimate that **training GPT-3** required at least **$4,600,000**

That’s a lot, but at least few-shot means the model only has to be trained once?

Big Models Require Big Pockets, and not just at training

Sources estimate that **training GPT-3** required at least **$4,600,000**

That’s a lot, but at least few-shot means the model only has to be trained once?

Yes, but **inference is still expensive**

One recent estimate pegged the cost of **running GPT-3** on a single AWS web server to cost **$87,000 a year** at minimum

Our assumption

bigger models $\rightarrow$ better performance

*This may be true, but is increasing model size the most **efficient** way of improving performance?*

Understanding FLOPs
(floating point operations)

$$C \sim 6ND$$

C = number of FLOPs (computations)
N = number of model parameters
D = amount of training data

Understanding FLOPs — Forward Pass

Matrix multiplication (e.g., attention QKV projection) requires **2 * size of matrix** (1 for multiplication, 1 for addition)

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} A_{11}x_1 + A_{12}x_2 + \cdots + A_{1n}x_n \\ A_{21}x_1 + A_{22}x_2 + \cdots + A_{2n}x_n \\ \vdots \\ A_{m1}x_1 + A_{m2}x_2 + \cdots + A_{mn}x_n \end{bmatrix}$$

Understanding FLOPs — Forward Pass

N is roughly the **sum of size of all matrices**

FLOPs for **forward** pass on a **single token** is roughly **2N**

FLOPs for **forward** pass for the **entire dataset** is roughly **2ND**

Understanding FLOPs — Backward Pass

Backward pass needs to calculate the derivative of loss with respect to **each hidden state** and for **each parameter**

FLOPs for **backward** pass is roughly **twice** of **forward** pass

FLOPs for **backward** pass for the **entire dataset** is roughly **4ND**

Understanding FLOPs

$$\mathbf{C \sim 6ND}$$

If we had a **computational budget** on C,
Increasing model size **N** = **Decreasing** dataset size **D**

But we also expect **more data → better performance**

Key Question

Increase N → better performance

Increase D → better performance

But we have a **budget** on **C ~ 6ND**

Key Question

*To **maximize** model performance,*

*how should we **allocate C** to **N** and **D**?*

Key Question

*To **maximize** model performance,*

*how should we **allocate C** to **N** and **D**?*

$$N_{opt}(C), D_{opt}(C) = \underset{N,D \text{ s.t. } \mathrm{FLOPs}(N,D)=C}{\operatorname{argmin}} L(N, D)$$

[Equation Source: (Hoffman et al., 2022)]

Key Question (rephrased)

*What is the **relationship** between **loss** and **N, D**?*

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

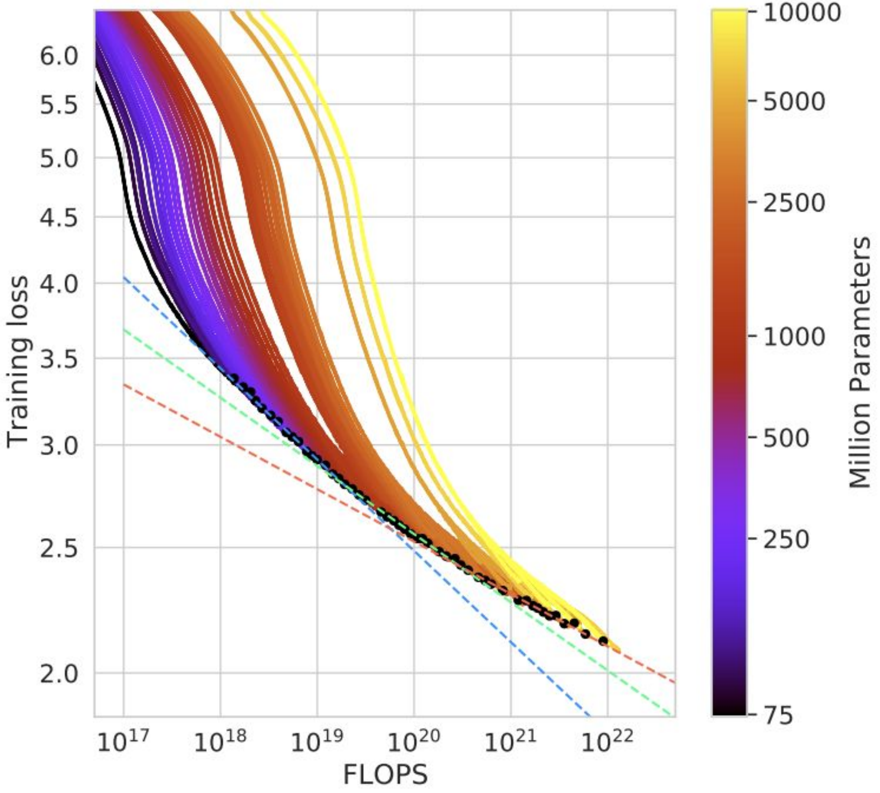
[Equation Source: (Hoffman et al., 2022)]

Is Power-Law the best fit?

Based on **empirical observation**

No theoretical background

(Hoffman et al.) also observe concavity in their model at high compute budgets, suggesting the **need for a more detailed model**



[[Figure Source: (Hoffman et al., 2022)]]

Outline

1. Introduction
2. **Initial Scaling Law (Kaplan et al., 2020)**
3. Modified Scaling Law (Hoffman et al., 2022)
4. Chinchilla (Hoffman et al., 2022)
5. Beyond Scaling Law

Kaplan et al., 2020

Scaling Laws for Neural Language Models

Jared Kaplan *
Johns Hopkins University, OpenAI
jaredk@jhu.edu

Sam McCandlish*
OpenAI
sam@openai.com

Tom Henighan
OpenAI
henighan@openai.com

Tom B. Brown
OpenAI
tom@openai.com

Benjamin Chess
OpenAI
bchess@openai.com

Rewon Child
OpenAI
rewon@openai.com

Scott Gray
OpenAI
scott@openai.com

Alec Radford
OpenAI
alec@openai.com

Jeffrey Wu
OpenAI
jeffwu@openai.com

Dario Amodei
OpenAI
damodei@openai.com

Training Details

Model: **Decoder-only Transformer (N = 0.7K ~ 1.5B params)**

Dataset: WebText2 **(D = 22B tokens)**

Batch Size (B): 0.5M

Step Size (S): 0.25M

Optimizer: Adam (+ Adafactor)

Learning rate: 3000 warmup steps, max LR = 2e-3, **cosine decay to 0**

Loss: **autoregressive cross-entropy loss** over 1024-token context

Main Results

- Performance scales with **model size (N)** and **dataset size (D)**
- If assuming **fixed batch size**,
  D should increase by **1.7x** when **N** increases by **2x**
- If assuming **optimal batch size**,
  D should increase by **1.3x** when **N** increases by **2x**

Main Results

- Performance scales with **model size (N)** and **dataset size (D)**
- If assuming **fixed batch size**,
  D should increase by **1.7x** when **N** increases by **2x**
- If assuming **optimal batch size**,
  D should increase by **1.3x** when **N** increases by **2x**

Commonly Cited Result

Outline

2. Initial Scaling Law (Kaplan et al., 2020)
   a. **Fixed Batch Size Case**
   b. Optimal Batch Size Case
   c. Limitations

Experiment 1 : Change D

Fix N = 1.5B

Fix B = 0.5M

Vary D = 21M ~ 22B (fixed subsets of WebText2)

Early stop whenever loss ceased to decrease

Experiment 2 : Change N

Fix D = 22B

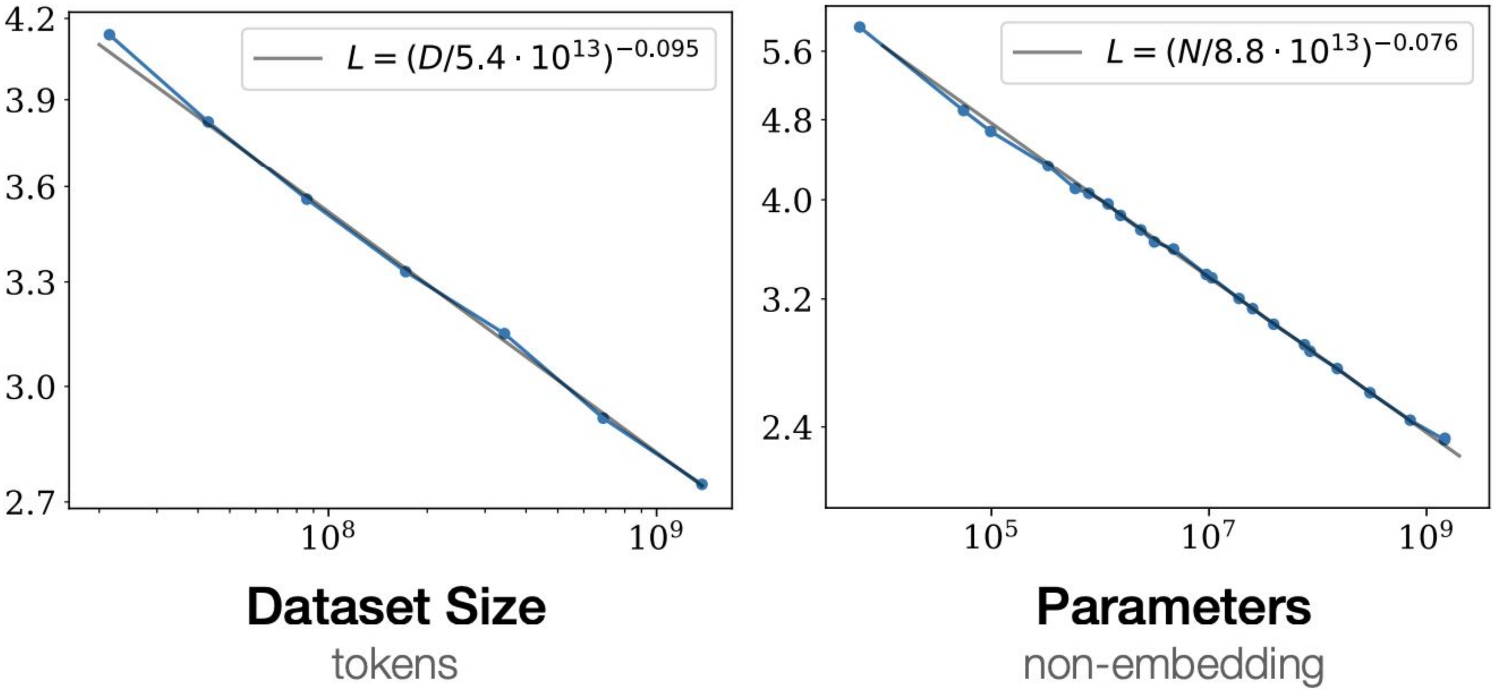
Fix B = 0.5M

Fix S = 0.25M

Vary N = 0.7K ~ 1.5B

Train until convergence

Results of Experiment 1, 2



[Figure Source: (Kaplan et al., 2020)]

Experiment 3 : Change both D and N

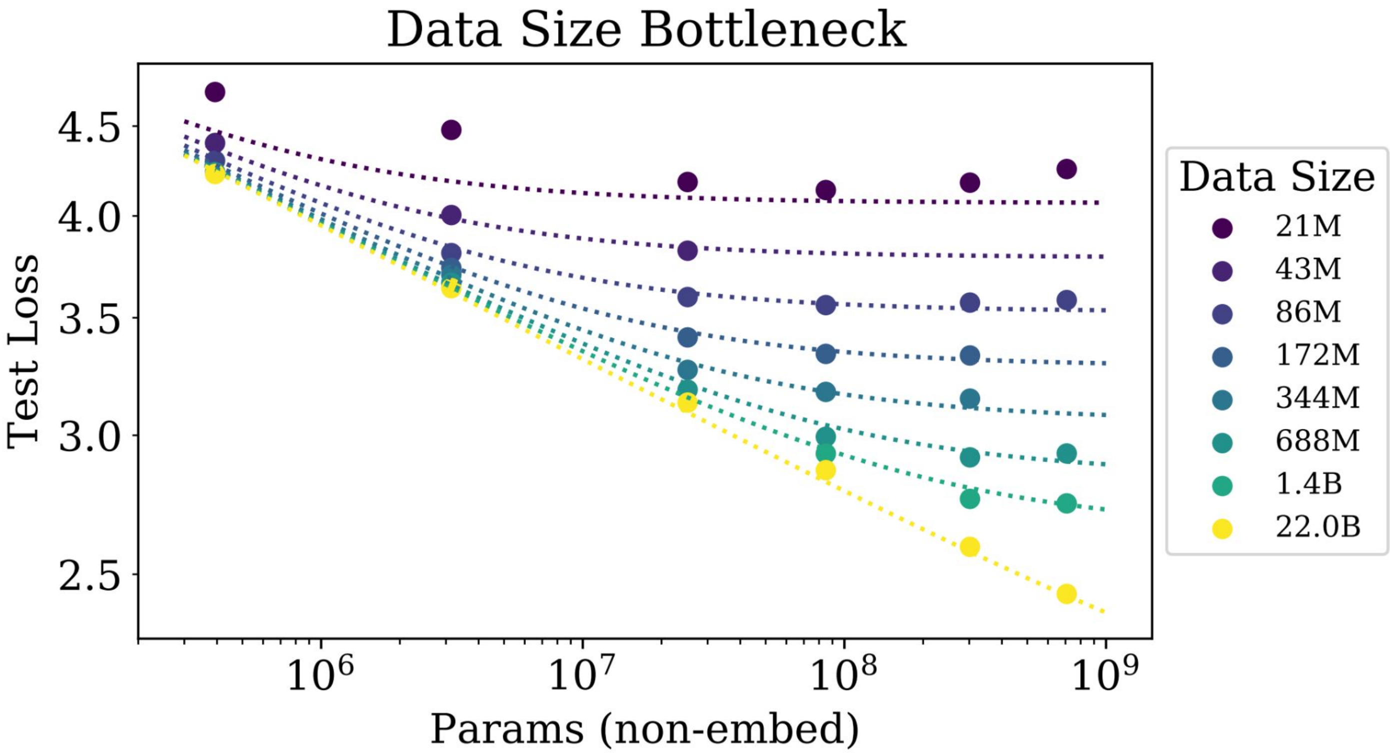
Fix B = 0.5M

Vary N = 0.4M ~ 0.7B

Vary D = 21M ~ 22B

Early stop whenever loss ceased to decrease

Result of Experiment 3



[Figure Source: (Kaplan et al., 2020)]

Conclusion

D should increase by **1.7x** when **N** increases by **2x**

Conclusion

D should increase by **1.7x** when **N** increases by **2x**

But what if we have a compute budget?

Outline

2. Initial Scaling Law (Kaplan et al., 2020)
   a. Fixed Batch Size Case
   b. **Optimal Batch Size Case**
   c. Limitations

Compute-Optimal Batch Size

Critical Batch Size **dependent on the loss** (not N, D) (McCandlish et al., 2018)

Compute-Optimal Batch Size

Critical Batch Size **dependent on the loss** (not N, D) (McCandlish et al., 2018)

(e.g., ~1M at the end of training for the best models in Experiments 1~3)

Compute-Optimal Batch Size

Critical Batch Size **dependent on the loss** (not N, D) (McCandlish et al., 2018)

(e.g., ~1M at the end of training for the best models in Experiments 1~3)

B << Critical Batch Size: **FLOP** minimized

Compute-Optimal Batch Size

Critical Batch Size **dependent on the loss** (not N, D) (McCandlish et al., 2018)

(e.g., ~1M at the end of training for the best models in Experiments 1~3)

B << Critical Batch Size: **FLOP** minimized

B >> Critical Batch Size: **Training Time (i.e., step size)** minimized

Compute-Optimal Batch Size

Critical Batch Size **dependent on the loss** (not N, D) (McCandlish et al., 2018)

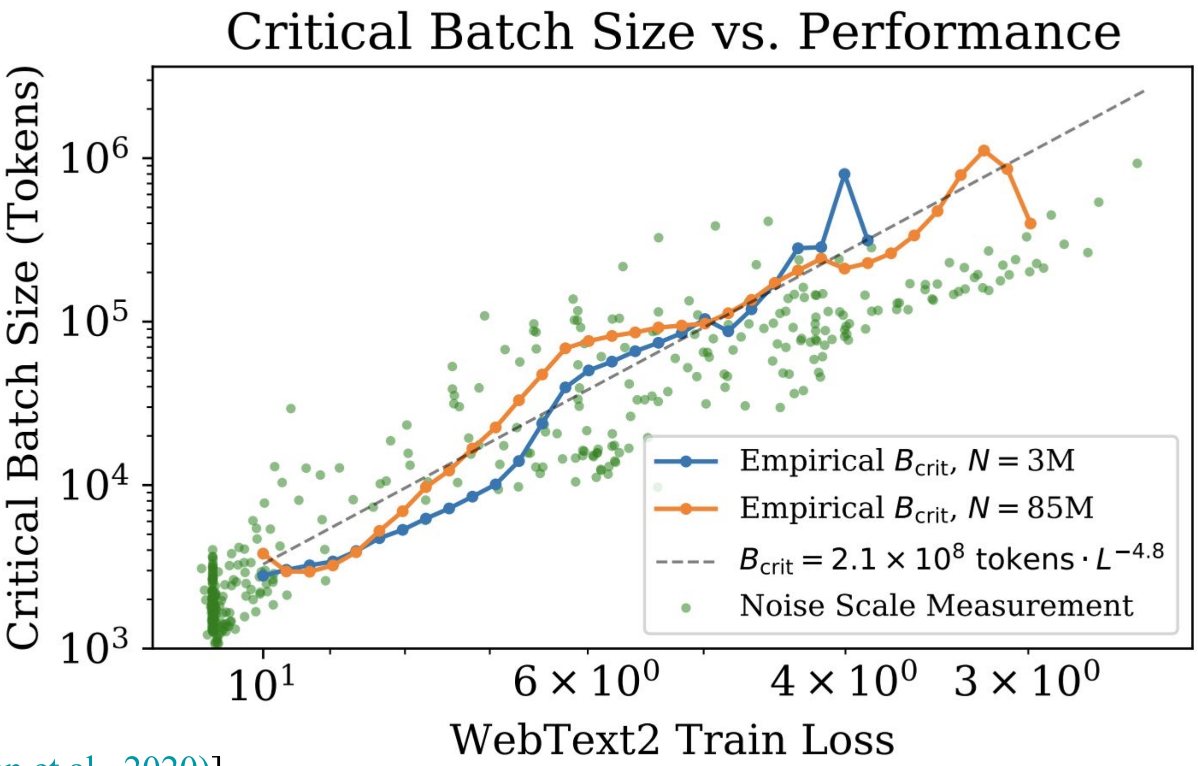
(e.g., ~1M at the end of training for the best models in Experiments 1~3)

B << Critical Batch Size: **FLOP** minimized

B >> Critical Batch Size: **Training Time (i.e., step size)** minimized

B == Critical Batch Size: **Trade-off**

Compute-Optimal Batch Size



[Figure Source: (Kaplan et al., 2020)]

Revisiting Experiment 3

Assuming we ran Experiment 3 again with **B << Critical Batch Size**,

It is possible to estimate the **minimum FLOP (C_min) to reach the same loss**

$$C_{\mathrm{min}}(C) \equiv \frac{C}{1 + B/B_{\mathrm{crit}}(L)}$$

Revisiting Experiment 3

Assuming we ran Experiment 3 again with **B << Critical Batch Size**,

It is possible to estimate the **minimum FLOP (C_min) to reach the same loss**

And the **optimal** model size **N** for the **target C_min**

Conclusion

$$N \propto {C_{min}}^{0.73} \qquad D \propto {C_{min}}^{0.27}$$

Conclusion

$$N \propto C_{min}^{0.73} \qquad D \propto C_{min}^{0.27}$$

D should increase by **1.3x** when **N** increases by **2x**

Outline

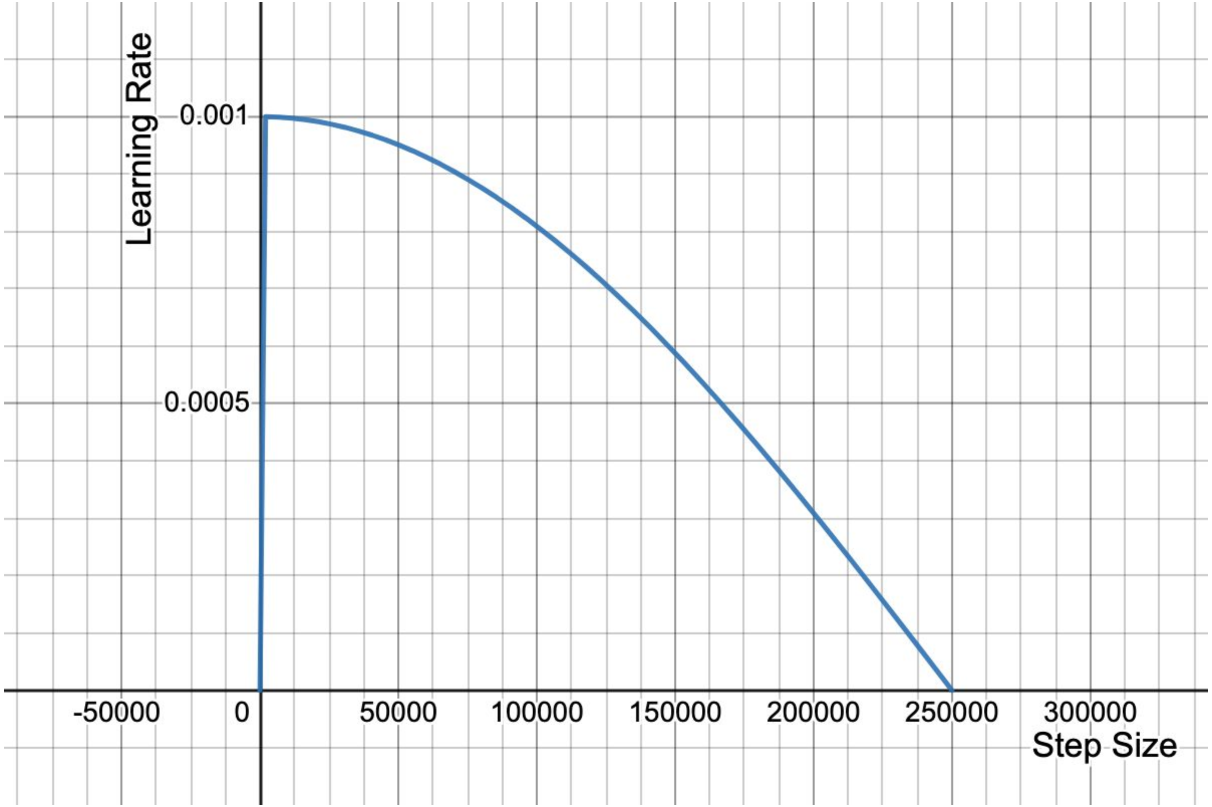
2. Initial Scaling Law (Kaplan et al., 2020)
    a. Fixed Batch Size Case
    b. Optimal Batch Size Case
    c. Limitations

Limitations

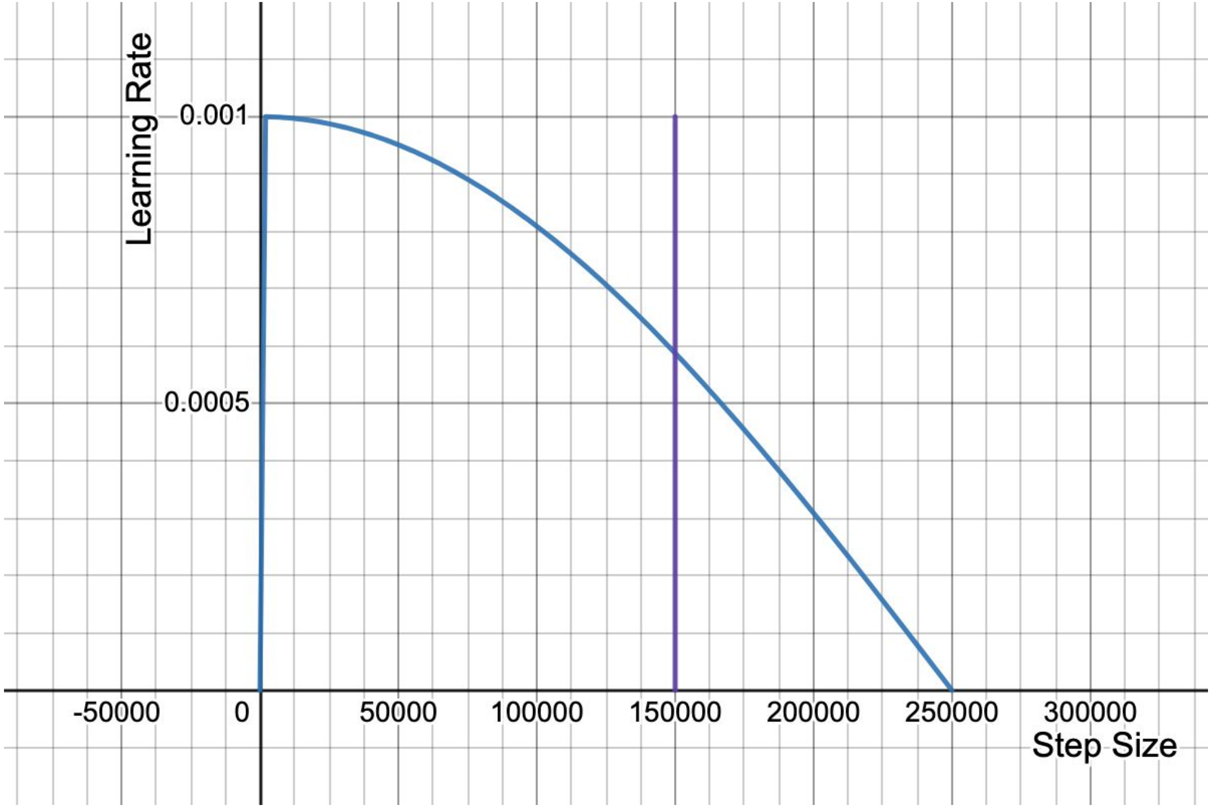
1. Needs to **adjust batch size** during training

2. Results were based on **early stop**, while **learning rate schedule** was calculated for the full 250K steps

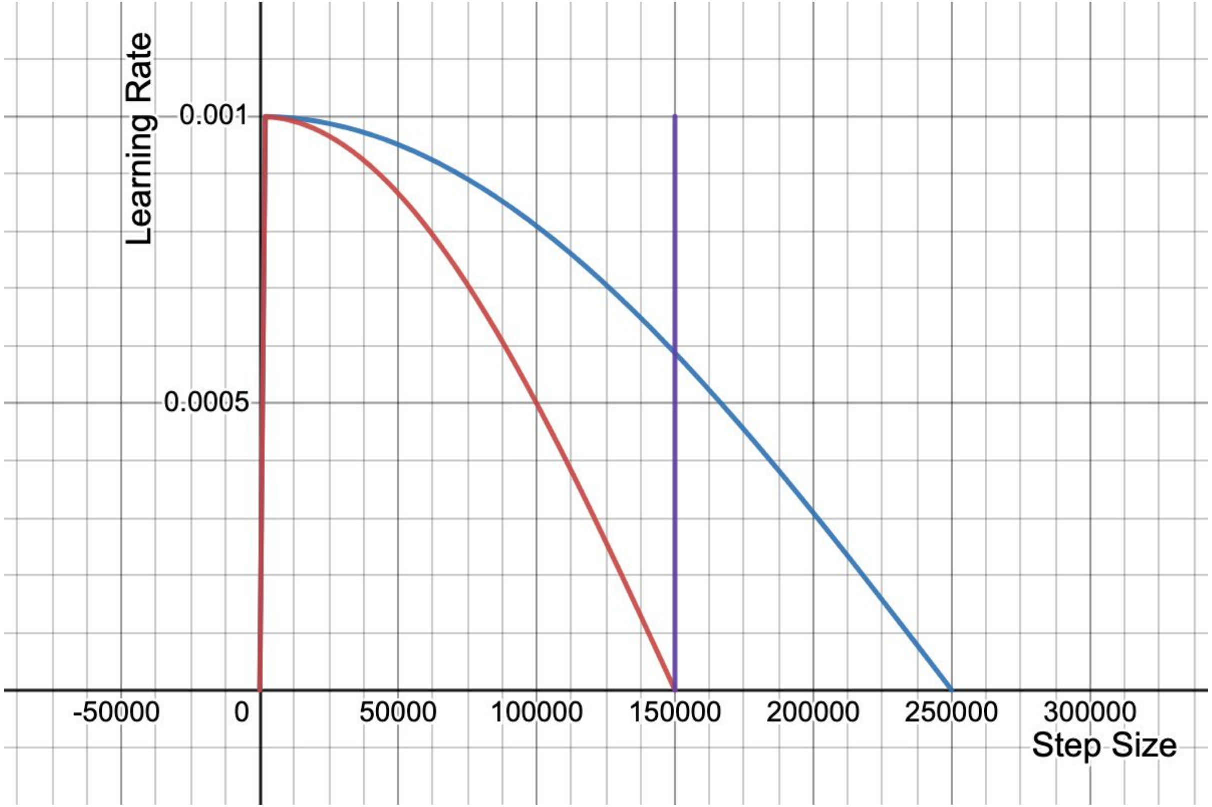
Learning Rate Schedule and Early Stop



Learning Rate Schedule and Early Stop



Learning Rate Schedule and Early Stop



Outline

1. Introduction
2. Initial Scaling Law (Kaplan et al., 2020)
3. **Modified Scaling Law (Hoffman et al., 2022)**
4. Chinchilla (Hoffman et al., 2022)
5. Beyond Scaling Law

Training Compute-Optimal Large Language Models

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*

*Equal contributions

Given a particular FLOPs (Floating Point Operation) budget, how should one trade-off model size and training data?

$$N_{opt}(C), D_{opt}(C) = \underset{N,D \text{ s.t. } \mathrm{FLOPs}(N,D)=C}{\operatorname{argmin}} L(N, D)$$

C = number of FLOPs (computations)
N = number of model parameters
D = amount of training data

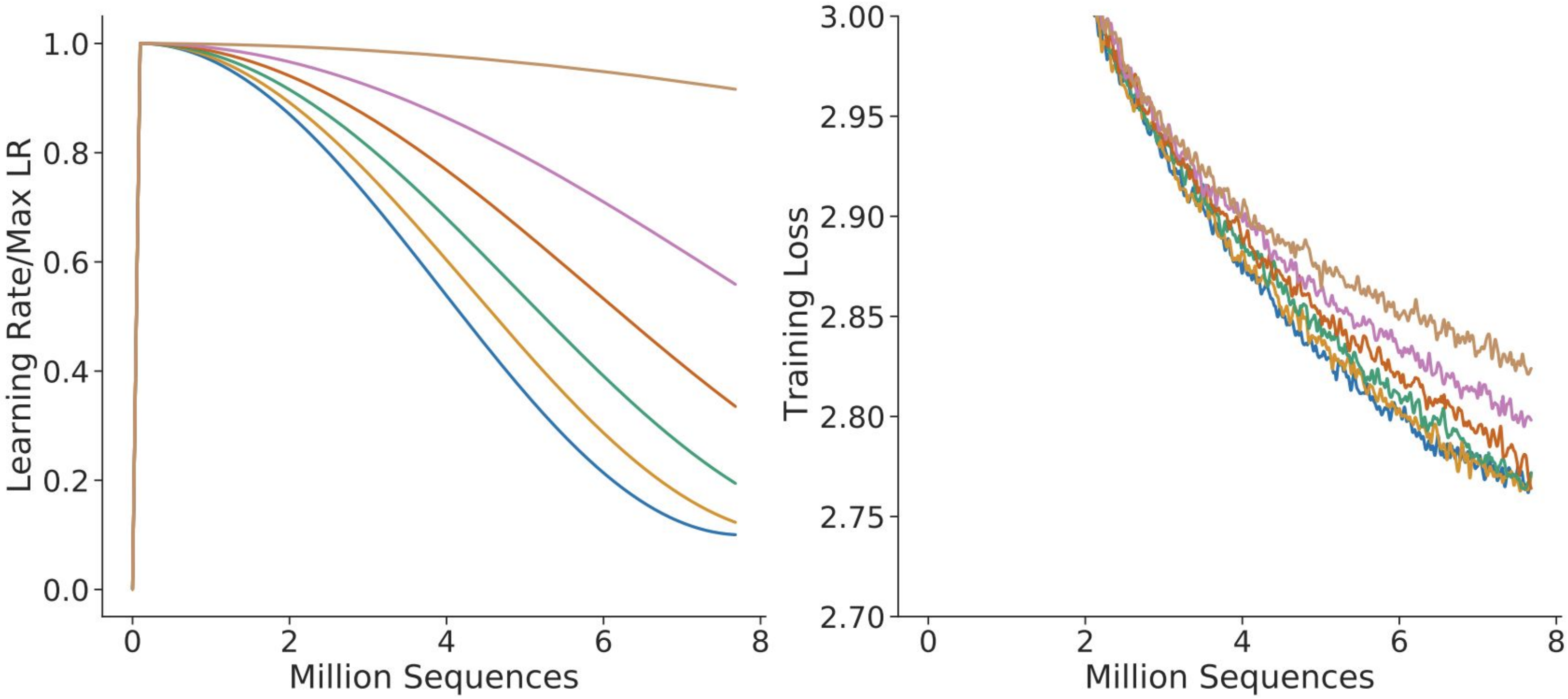
N, **D** should scale at **same rate**

Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	0.50 (0.488, 0.502)	0.50 (0.501, 0.512)
2. IsoFLOP profiles	0.49 (0.462, 0.534)	0.51 (0.483, 0.529)
3. Parametric modelling of the loss	0.46 (0.454, 0.455)	0.54 (0.542, 0.543)
Kaplan et al. (2020)	0.73	0.27

[Table Source: (Hoffman et al., 2022)]

Q2: How do the conclusions of (Kaplan et al.) and (Hoffman et al.) differ? What caused the differences?

Early Stopping leads to Underperformance



[Figure Source: (Hoffman et al., 2022)]

(Kaplan et al.) vs (Hoffman et al.)

(Kaplan et al.)

Learning rate - based on **250K** steps

Batch Size - based on **B <= critical batch size**

(Hoffman et al.)

Learning rate - based on **actual step size**

Batch Size - **fixed**

Outline

3. Modified Scaling Law (Hoffman et al., 2022)
    a. **Approach 1**
    b. Approach 2
    c. Approach 3
    d. Results

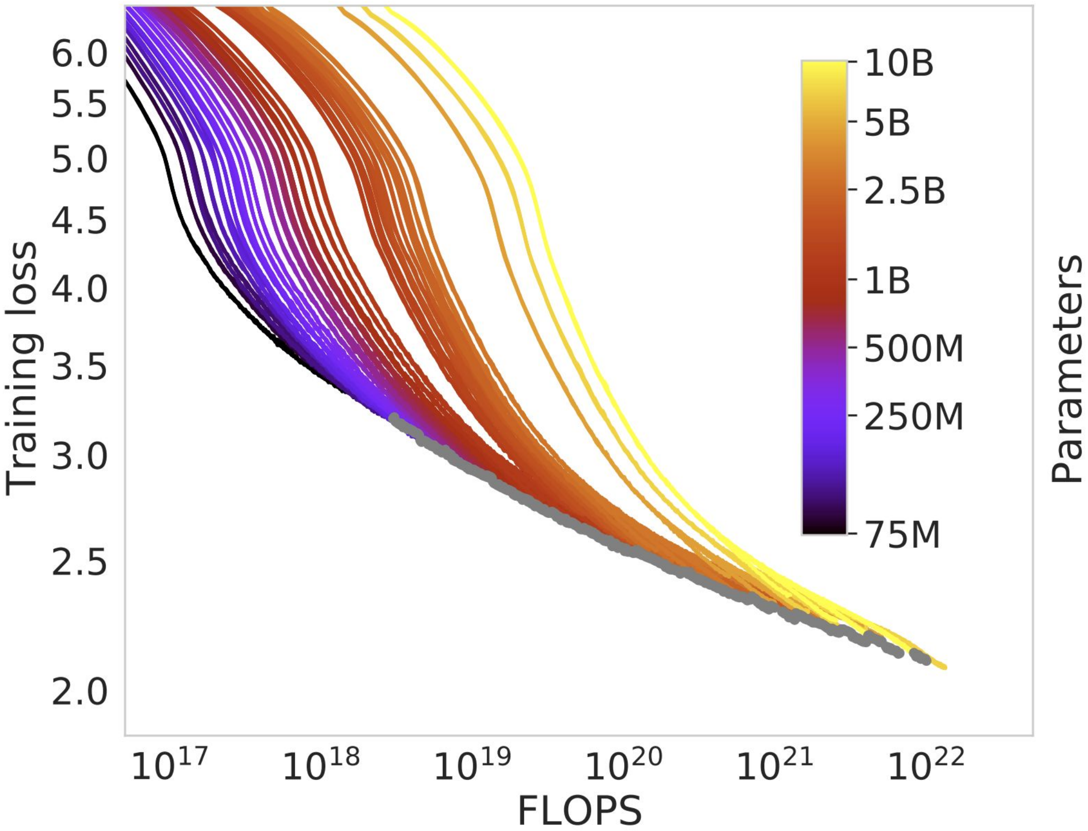
Approach 1: Fix N and vary D

For each N, train 4 different models with **different D**

Interpolate these curves to get a continuous mapping

For each FLOPs, pick the model with the lowest training loss

C = number of FLOPs (computations)
N = number of model parameters
D = amount of training data



[Figure Source: (Hoffman et al., 2022)]

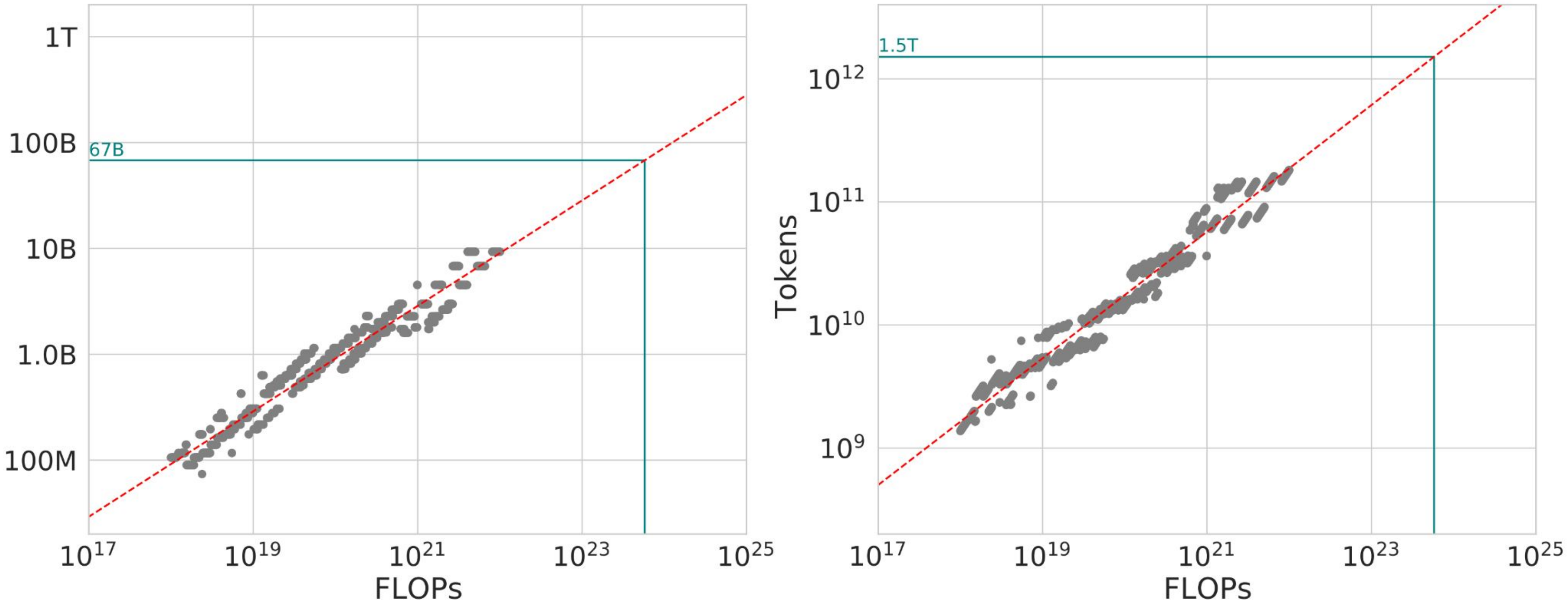
Approach 1: Fix N and Vary D

For each N, train 4 different models with **different D**

Interpolate these curves to get a continuous mapping

For each FLOPs, pick the model with the lowest training loss

Fit a power law relationship between C and N, D



[Figure Source: (Hoffman et al., 2022)]

Results of Approach 1

Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	$0.50\ (0.488, 0.502)$	$0.50\ (0.501, 0.512)$
2. IsoFLOP profiles	$0.49\ (0.462, 0.534)$	$0.51\ (0.483, 0.529)$
3. Parametric modelling of the loss	$0.46\ (0.454, 0.455)$	$0.54\ (0.542, 0.543)$
Kaplan et al. (2020)	0.73	0.27

[Table Source: (Hoffman et al., 2022)]

Outline

3. Modified Scaling Law (Hoffman et al., 2022)
    a. Approach 1
    b. **Approach 2**
    c. Approach 3
    d. Results

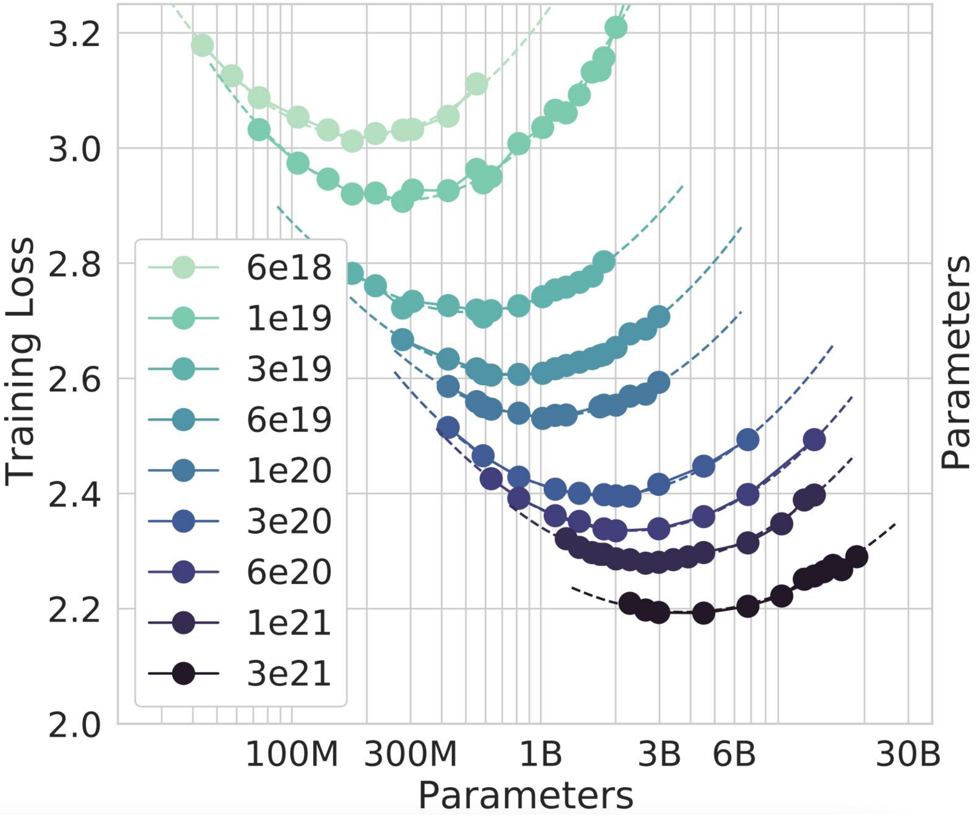
Approach 2: IsoFLOP Profiles

For each FLOPs budget C, train models of **different size N**

For each model, **choose the appropriate D** such that C ~ 6ND

E.g., bigger models are trained on less data to meet FLOPs constraint

C = number of FLOPs (computations)
N = number of model parameters
D = amount of training data



[[Figure Source: (Hoffman et al., 2022)]]

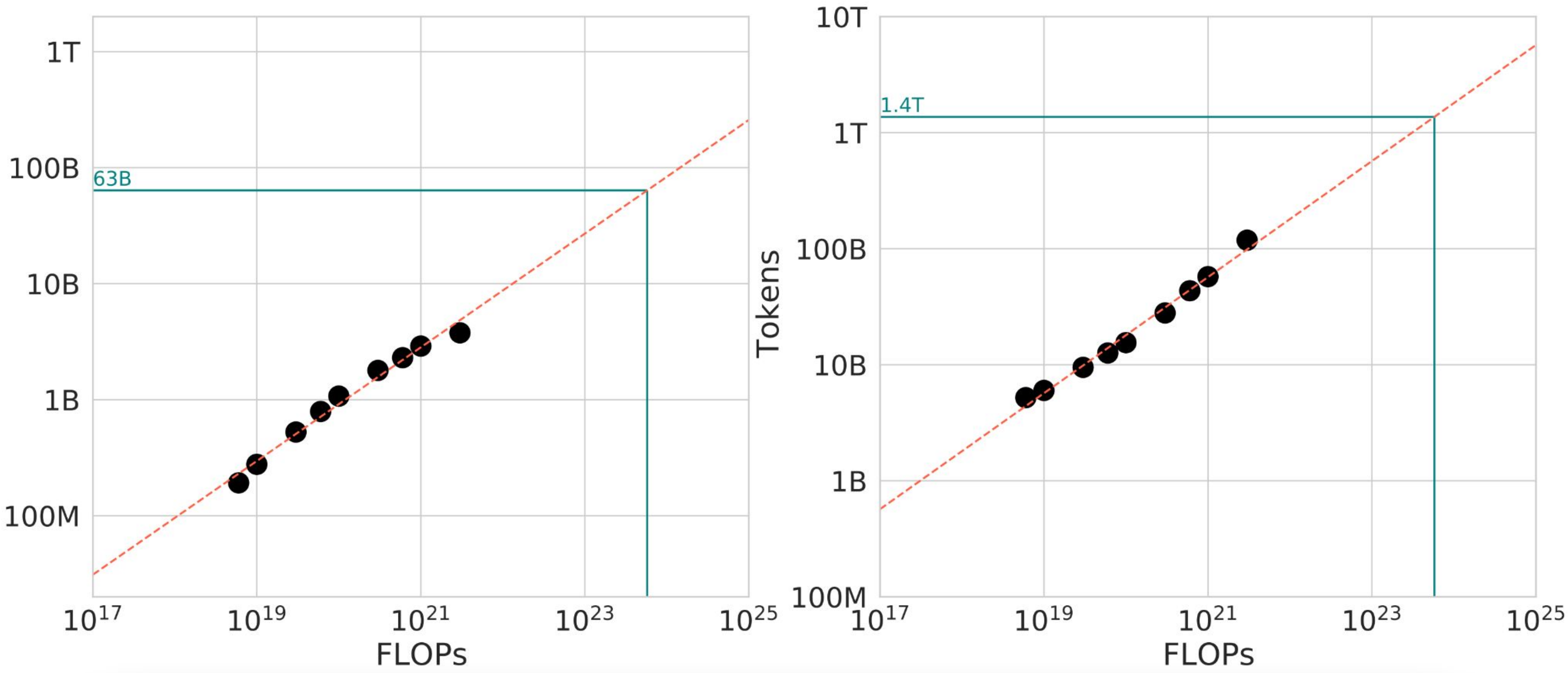
Approach 2: IsoFLOP Profiles

For each FLOPs budget C, train models of **different size N**

For each model, **choose the appropriate D** such that C ~ 6ND

E.g., bigger models are trained on less data to meet FLOPs constraint

Fit a power law relationship between C and N, D



[Figure Source: (Hoffman et al., 2022)]

Results of Approach 2

Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	0.50 (0.488, 0.502)	0.50 (0.501, 0.512)
2. IsoFLOP profiles	0.49 (0.462, 0.534)	0.51 (0.483, 0.529)
3. Parametric modelling of the loss	0.46 (0.454, 0.455)	0.54 (0.542, 0.543)
Kaplan et al. (2020)	0.73	0.27

[Table Source: (Hoffman et al., 2022)]

Outline

3. Modified Scaling Law (Hoffman et al., 2022)
    a. Approach 1
    b. Approach 2
    c. **Approach 3**
    d. Results

Approach 3: Parametric Loss Function

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

1. **E**: loss of ideal generative model (entropy of natural language)
2. **N**: larger model → better performance
3. **D**: larger dataset → better performance

Determining Coefficients

1. **Choose initial values of E, A, B, α, β** from a grid of values
2. Find the **Huber loss** based on the predicted log loss of the model on (N, D) and observed log loss (data from Approach 1, 2)
3. Iteratively, run the L-BFGS algorithm (some variant of **Gradient Descent**)

Results of Approach 3

Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	0.50 (0.488, 0.502)	0.50 (0.501, 0.512)
2. IsoFLOP profiles	0.49 (0.462, 0.534)	0.51 (0.483, 0.529)
3. Parametric modelling of the loss	0.46 (0.454, 0.455)	0.54 (0.542, 0.543)
Kaplan et al. (2020)	0.73	0.27

[Table Source: (Hoffman et al., 2022)]

Outline

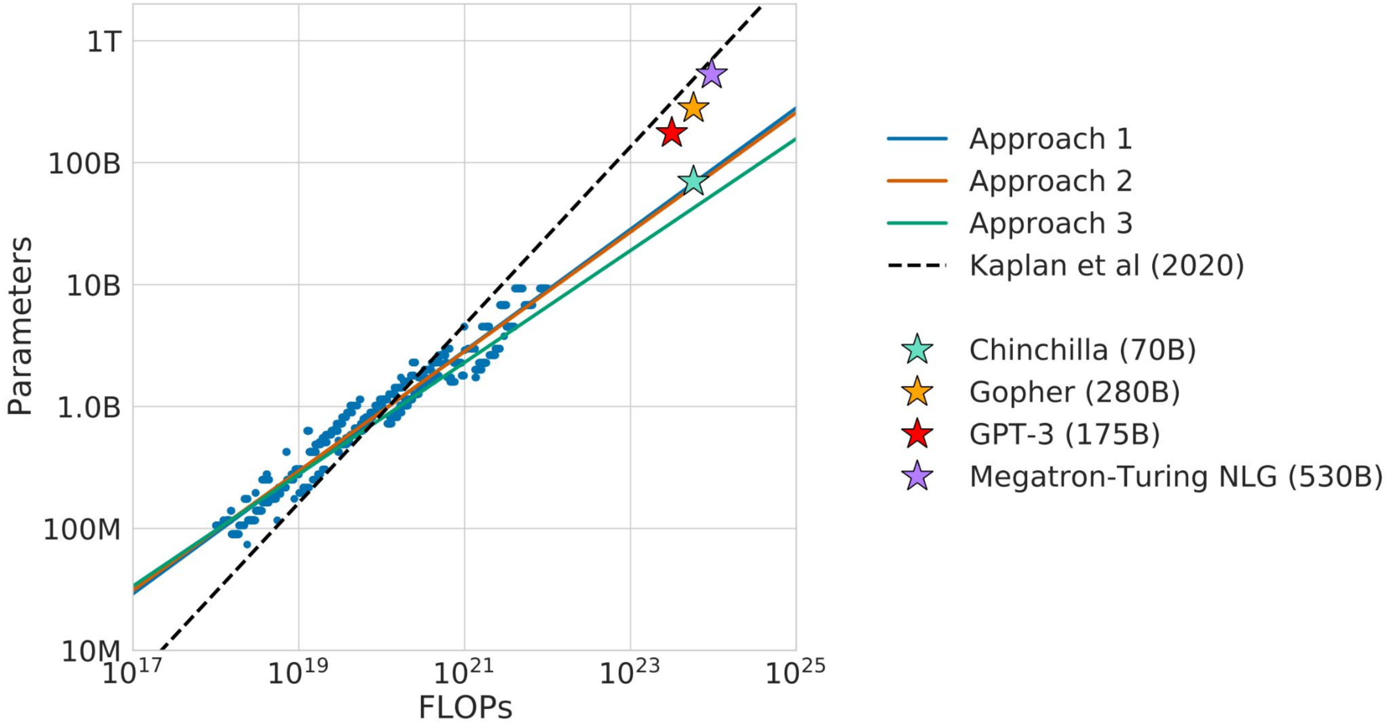
3. Modified Scaling Law (Hoffman et al., 2022)
   a. Approach 1
   b. Approach 2
   c. Approach 3
   d. **Results**

Results of Approach 1 ~ 3

Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	0.50 (0.488, 0.502)	0.50 (0.501, 0.512)
2. IsoFLOP profiles	0.49 (0.462, 0.534)	0.51 (0.483, 0.529)
3. Parametric modelling of the loss	0.46 (0.454, 0.455)	0.54 (0.542, 0.543)
Kaplan et al. (2020)	0.73	0.27

[Table Source: (Hoffman et al., 2022)]

Today's models are **overparameterized** and **undertrained**



[Figure Source: (Hoffman et al., 2022)]

Outline

1. Introduction
2. Initial Scaling Law (Kaplan et al., 2020)
3. Modified Scaling Law (Hoffman et al., 2022)
4. **Chinchilla (Hoffman et al., 2022)**
5. Beyond Scaling Law

Given Gopher's compute budget, can we train a more **computationally efficient** model?

Chinchilla

VS

Gopher

[Image Source] [Image Source]

Chinchilla is small(er)

Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

[Image Source] [Table Source: (Hoffman et al., 2022)]

Comparison with Gopher

N smaller by 4x, **D larger** by 4x

Less compute for inference and fine-tuning

But also **stronger performance**

Performance of Chinchilla

VS

[Image Source] [Image Source]

Evaluations Tasks for Chinchilla

- Language Modelling
- MMLU
- Reading Comprehension
- BIG-bench
- Common Sense
- Closed Book QA
- Gender Bias and Toxicity

Evaluations Tasks for Chinchilla

- **Language Modelling**
- MMLU
- Reading Comprehension
- BIG-bench
- Common Sense
- Closed Book QA
- Gender Bias and Toxicity

Language Modelling

Measure test perplexity (in bits-per-byte) of 20 datasets from the Pile (Gao et al., 2021)

Chinchilla outperforms Gopher on all 20 datasets

Note: because of large training data, there is an **increased risk of train/test leak**

Analysis Per Dataset

Subset	*Chinchilla* (70B)	*Gopher* (280B)
pile_cc	**0.667**	0.691
pubmed_abstracts	**0.559**	0.578
stackexchange	**0.614**	0.641
github	**0.337**	0.377
openwebtext2	**0.647**	0.677
arxiv	**0.627**	0.662
uspto_backgrounds	**0.526**	0.546
freelaw	**0.476**	0.513
pubmed_central	**0.504**	0.525
dm_mathematics	1.111	1.142
hackernews	**0.859**	0.890
nih_exporter	**0.572**	0.590
opensubtitles	**0.871**	0.900
europarl	**0.833**	0.938
books3	**0.675**	0.712
philpapers	**0.656**	0.695
gutenberg_pg_19	**0.548**	0.656
bookcorpus2	**0.714**	0.741
ubuntu_irc	1.026	1.090

[Table Source: (Hoffman et al., 2022)]

Evaluations Tasks for Chinchilla

- Language Modelling
- **MMLU**
- Reading Comprehension
- BIG-bench
- Common Sense
- Closed Book QA
- Gender Bias and Toxicity

MMLU — Massive Multitask Language Understanding

Answer exam-like **multiple choice questions** on **57 subjects** (Hendrycks et al., 2020)

E.g., college mathematics, high school physics, professional law

Example Data from MMLU

An observational study in diabetics assesses the role of an increased plasma fibrinogen level on the risk of cardiac events. 130 diabetic patients are followed for 5 years to assess the development of acute coronary syndrome. In the group of 60 patients with a normal baseline plasma fibrinogen level, 20 develop acute coronary syndrome and 40 do not. In the group of 70 patients with a high baseline plasma fibrinogen level, 40 develop acute coronary syndrome and 30 do not. Which of the following is the best estimate of relative risk in patients with a high baseline plasma fibrinogen level compared to patients with a normal baseline plasma fibrinogen level?
(A) (40/30)/(20/40)
(B) (40*40)/(20*30)
(C) (40*70)/(20*60)
(D) (40/70)/(20/60)

Figure 69: A Virology example.

[Figure Source: (Hendrycks et al., 2020)]

Chinchilla Outperforms Gopher on Average

	Random	25.0%
	Average human rater	34.5%
175B	GPT-3 5-shot	43.9%
280B	*Gopher* 5-shot	60.0%
70B	***Chinchilla* 5-shot**	**67.6%**
	Average human expert performance	*89.8%*

[Table Source: (Hoffman et al., 2022)]

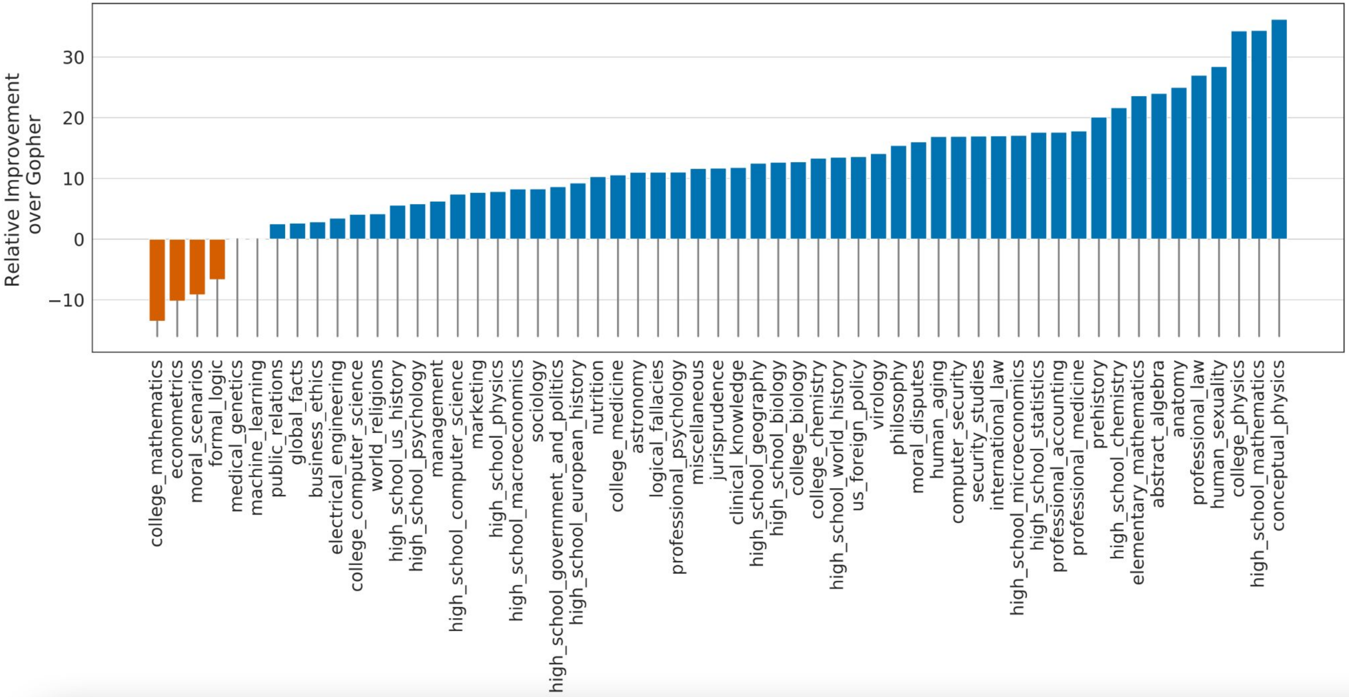
Analysis Per Task

Chinchilla **outperforms** Gopher on **51 tasks**

Achieves a **similar performance** on **2 tasks**

Underperforms Gopher on **4 tasks** (college mathematics, econometrics, moral scenarios, formal logic)

Analysis Per Task



[Figure Source: (Hoffman et al., 2022)]

Analysis Per Task

Chinchilla achieves **> 90% accuracy** on **4 tasks**

High school government and politics, international law, sociology, US foreign policy

First model to achieve 90% accuracy on a particular subject

Evaluations Tasks for Chinchilla

- Language Modelling
- MMLU
- **Reading Comprehension**
- BIG-bench
- Common Sense
- Closed Book QA
- Gender Bias and Toxicity

Reading Comprehension

Answer a fill-in-the-blank question on a passage

LAMBADA (Paperno et al., 2016): novel excerpt

RACE-M, RACE-H (Lai et al., 2017): middle-, high-school exam questions

Example Data from LAMBADA

Context: The battery on Logan's radio must have been on the way out. So he told himself. There was no other explanation beyond Cygan and the staff at the White House having been overrun. Lizzie opened her eyes with a flutter. They had been on the icy road for an hour without incident.
Target sentence: Jack was happy to do all of the ______.
Target word: driving

[Figure Source: (Paperno et al., 2016)]

Example Data from RACE-M, RACE-H

Evidence: "The park is open from 8 am to 5 pm."

Question: The park is open for __ hours a day.

Options: A.eight B.nine C.ten D.eleven

[Figure Source: (Lai et al., 2017)]

Chinchilla Outperforms Gopher

	70B	280B	175B	530B
	Chinchilla	*Gopher*	GPT-3	MT-NLG 530B
LAMBADA Zero-Shot	**77.4**	74.5	76.2	76.6
RACE-m Few-Shot	**86.8**	75.1	58.1	-
RACE-h Few-Shot	**82.3**	71.6	46.8	47.9

[Table Source: (Hoffman et al., 2022)]

Evaluations Tasks for Chinchilla

- Language Modelling
- MMLU
- Reading Comprehension
- **BIG-bench**
- Common Sense
- Closed Book QA
- Gender Bias and Toxicity

BIG-bench

Collection of **'difficult' tasks** for current models (Srivastava et al., 2022)

Currently has **204 tasks** and is growing with Github pull requests

(Hoffman et al., 2022) used **62 tasks**

Example Data from BIG-bench

```
Which of the following sentences makes more sense?
choice: It started raining because the driver turned the
  wipers on.
choice: The driver turned the wipers on because it started
  raining.
```

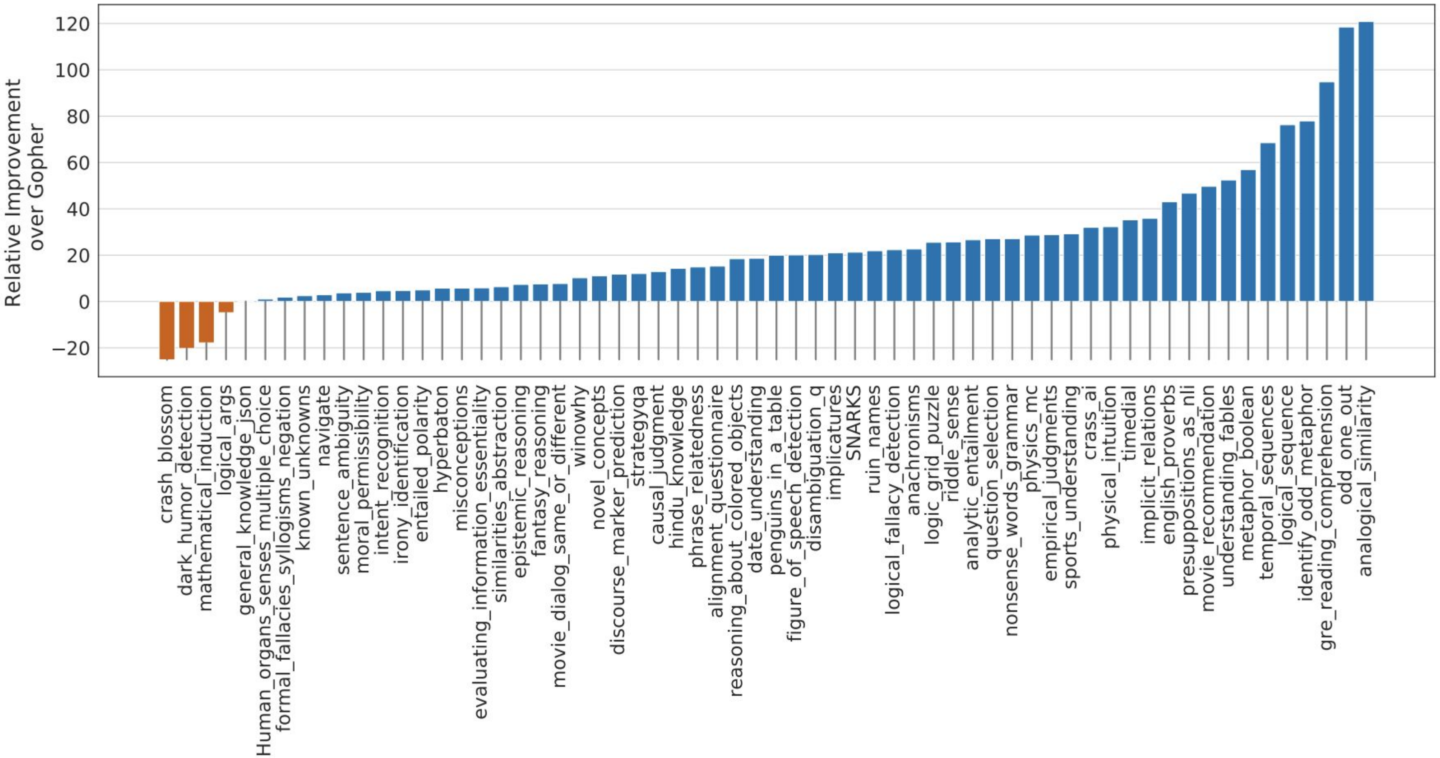
[Figure Source: (Srivastava et al., 2022)]

Analysis Per Task

Chinchilla **outperforms** Gopher on **58 tasks**

Underperforms Gopher on **4 tasks**

Analysis Per Task



[Figure Source: (Hoffman et al., 2022)]

Evaluations Tasks for Chinchilla

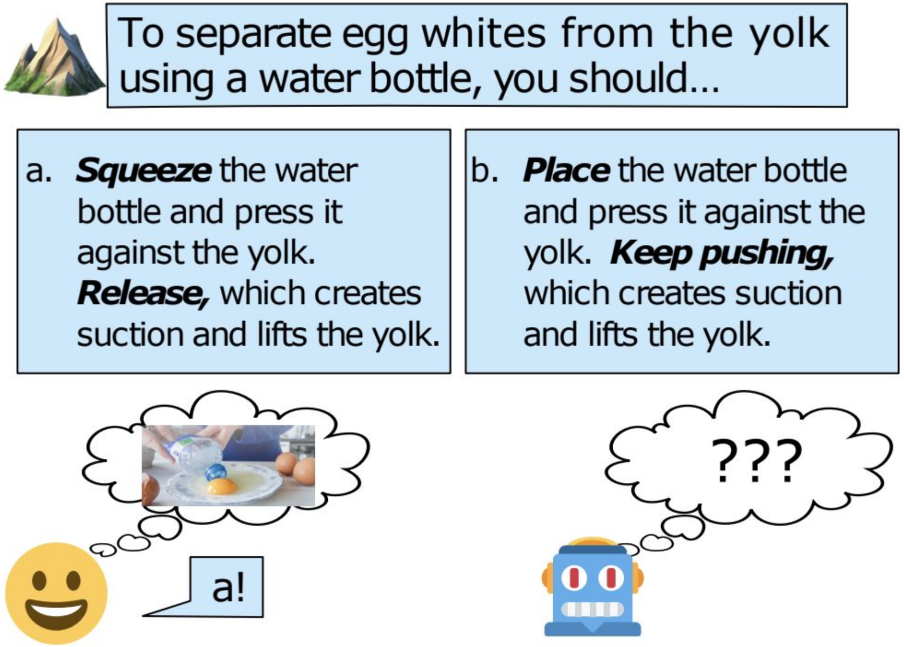
- Language Modelling
- MMLU
- Reading Comprehension
- BIG-bench
- **Common Sense**
- Closed Book QA
- Gender Bias and Toxicity

Common Sense

Answer various **common sense questions**

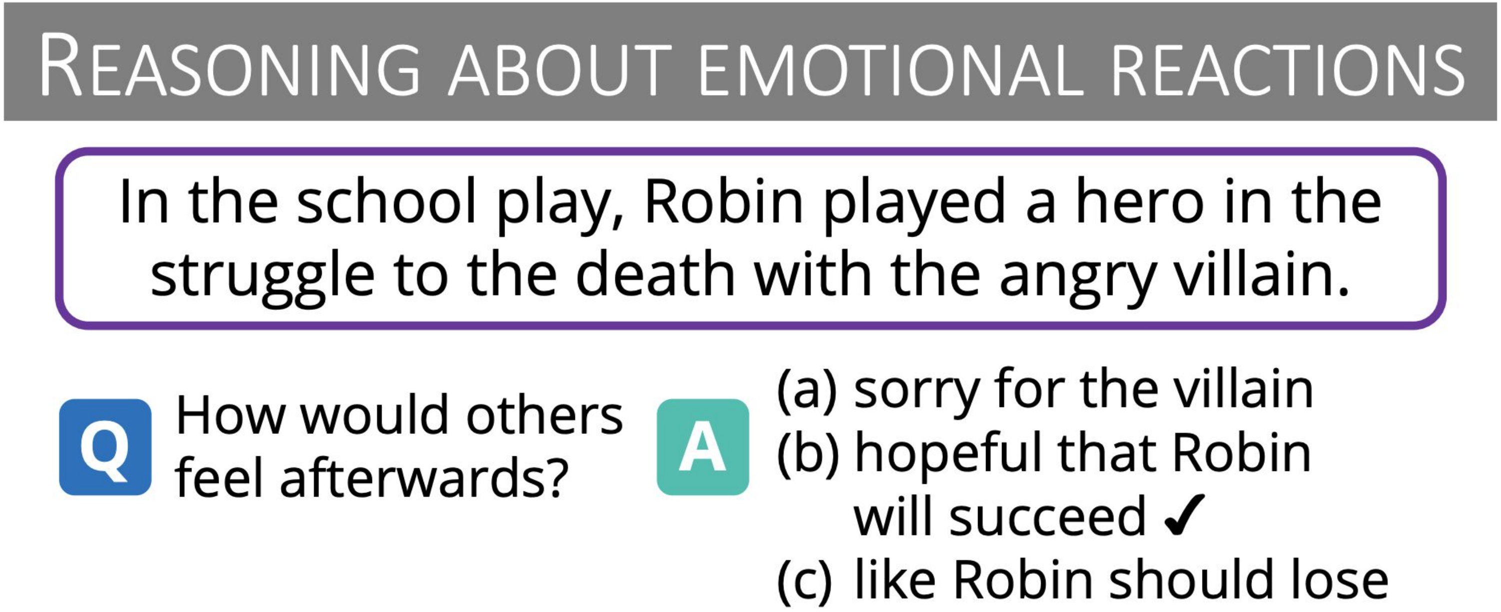
E.g., reasoning about the physical world, pronoun resolution, emotion inferrance

Example Data from PIQA



[Figure Source: (Bisk et al., 2019)]

Example Data from SIQA



[Figure Source: (Sap et al., 2019)]

Chinchilla Outperforms Gopher

	70B	280B	175B	530B	
	Chinchilla	*Gopher*	GPT-3	MT-NLG 530B	Supervised SOTA
HellaSWAG	**80.8%**	79.2%	78.9%	80.2%	93.9%
PIQA	81.8%	81.8%	81.0%	**82.0%**	90.1%
Winogrande	**74.9%**	70.1%	70.2%	73.0%	91.3%
SIQA	**51.3%**	50.6%	-	-	83.2%
BoolQ	**83.7%**	79.3%	60.5%	78.2%	91.4%

[Table Source: (Hoffman et al., 2022)]

Evaluations Tasks for Chinchilla

- Language Modelling
- MMLU
- Reading Comprehension
- BIG-bench
- Common Sense
- **Closed Book QA**
- Gender Bias and Toxicity

Closed Book QA

Answer short-answer questions without external sources

Question: what color was john wilkes booth's hair
Wikipedia Page: John_Wilkes_Booth
Long answer: Some critics called Booth "the handsomest man in America" and a "natural genius", and noted his having an "astonishing memory"; others were mixed in their estimation of his acting. He stood 5 feet 8 inches (1.73 m) tall, had jet-black hair , and was lean and athletic. Noted Civil War reporter George Alfred Townsend described him as a "muscular, perfect man" with "curling hair, like a Corinthian capital".

Short answer: jet-black

[Source: (Kwiatkowski et al., 2019)]

Chinchilla Outperforms Gopher

70B 280B 175B

	Method	*Chinchilla*	*Gopher*	GPT-3	SOTA (open book)
Natural Questions (dev)	0-shot	16.6%	10.1%	14.6%	54.4%
	5-shot	31.5%	24.5%	-	
	64-shot	35.5%	28.2%	29.9%	
TriviaQA (unfiltered, test)	0-shot	67.0%	52.8%	64.3 %	-
	5-shot	73.2%	63.6%	-	
	64-shot	72.3%	61.3%	71.2%	
TriviaQA (filtered, dev)	0-shot	55.4%	43.5%	-	72.5%
	5-shot	64.1%	57.0%	-	
	64-shot	64.6%	57.2%	-	

[Table Source: (Hoffman et al., 2022)]

Outline

1. Introduction
2. Initial Scaling Law (Kaplan et al., 2020)
3. Modified Scaling Law (Hoffman et al., 2022)
4. Chinchilla (Hoffman et al., 2022)
5. **Beyond Scaling Law**

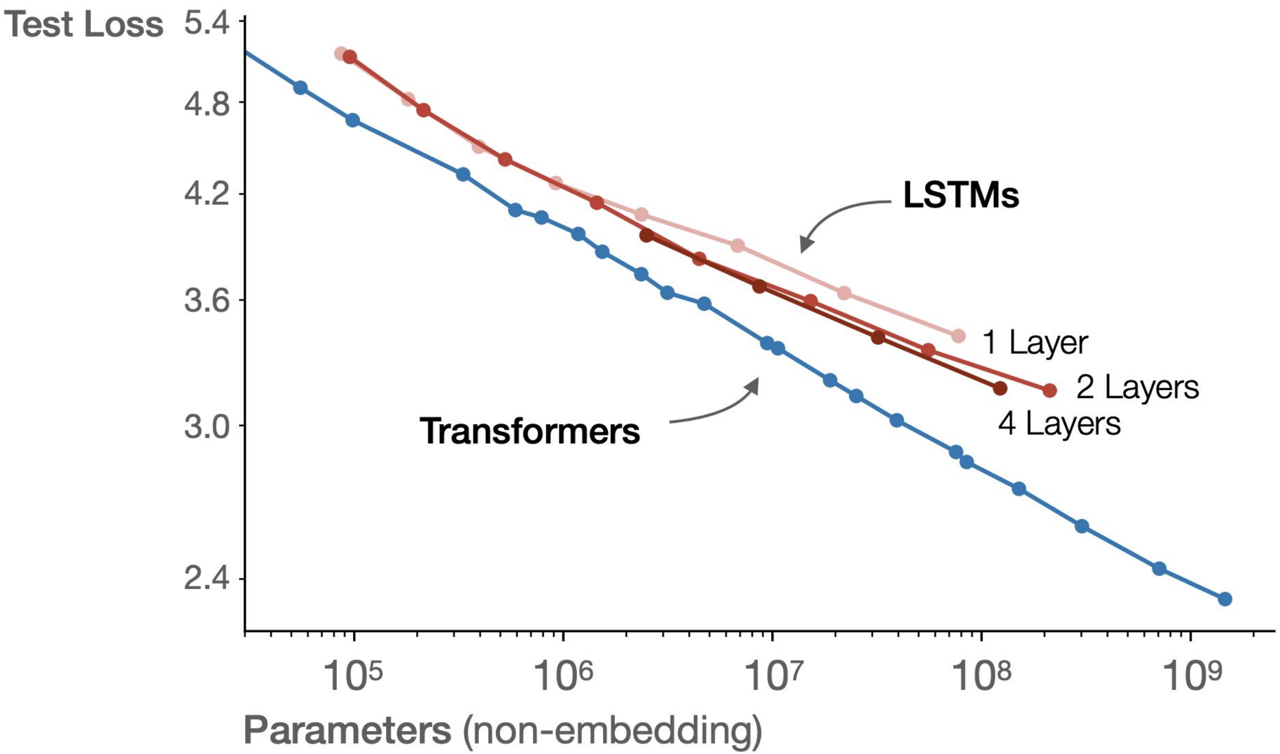
Generalization of the Scaling Law

Other **architecture** — (Kaplan et al., 2020) tests the scaling law on LSTM and Universal Transformers (encoder-decoder model)

Other **dataset** — (Hoffman et al., 2022) tests the scaling law on different datasets (e.g., C4, Github)

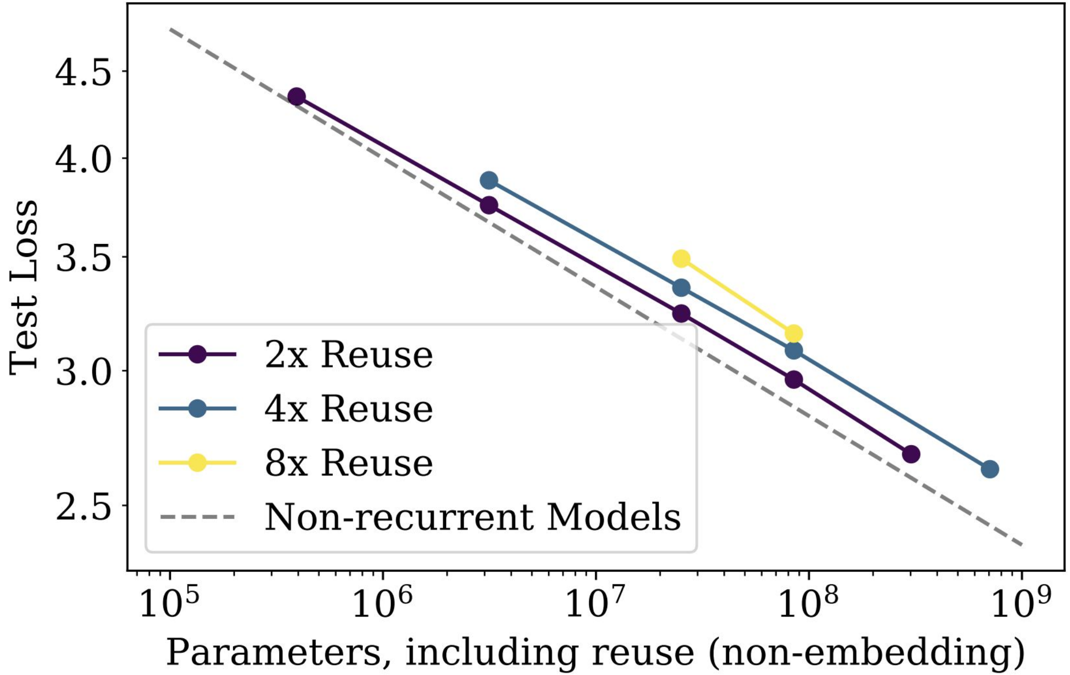
Other **domain** — (Henighan et al., 2020) test the scaling law on different domains (e.g., image, video)

Generalization to LSTM



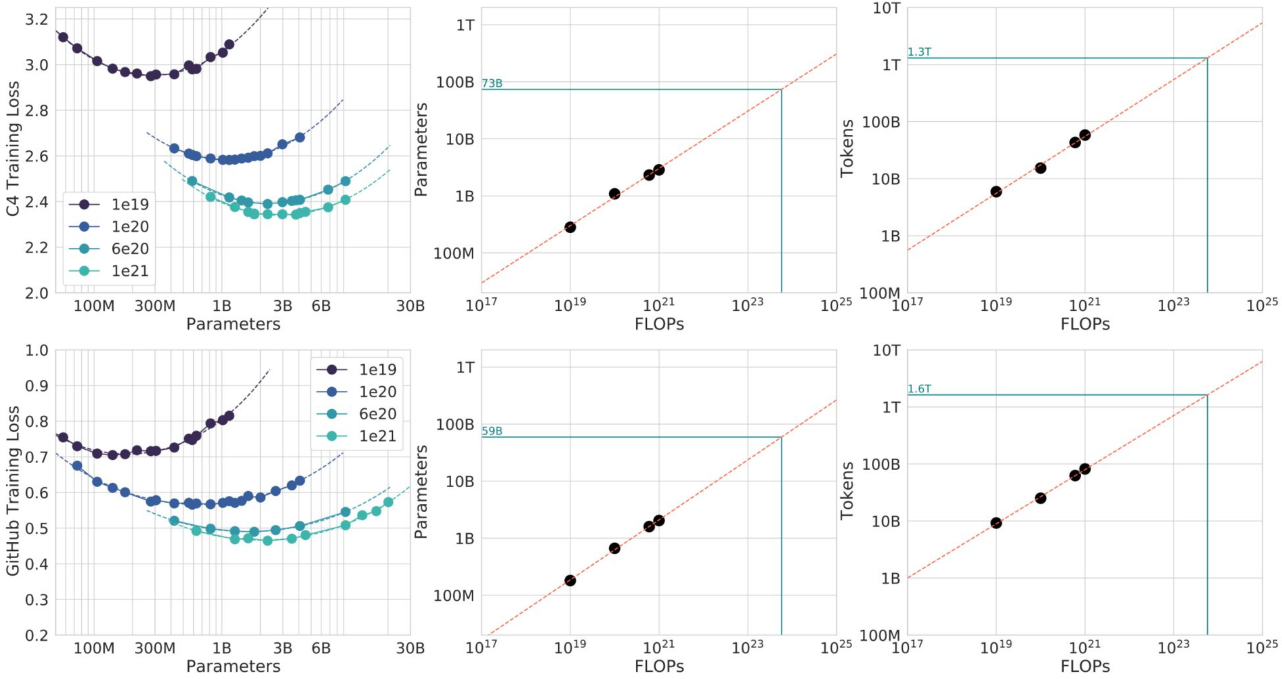
[Figure Source: (Kaplan et al., 2020)]

Generalization to Universal Transformers



[Figure Source: (Kaplan et al., 2020)]

Generalization to C4 and Github code



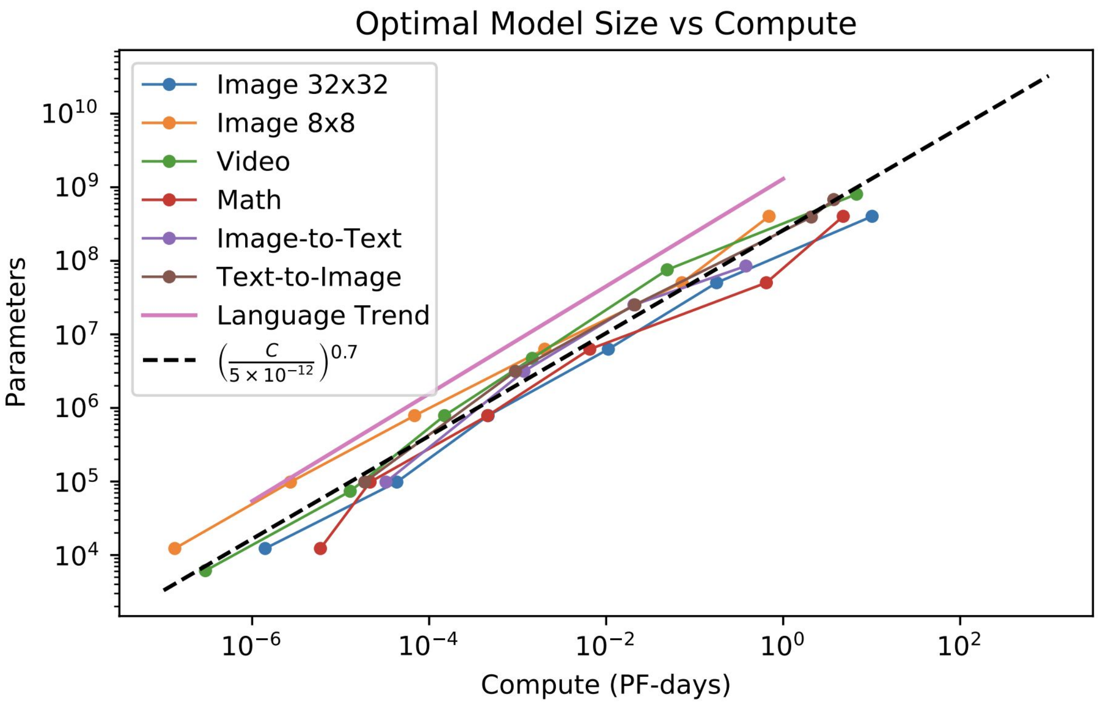
[Figure Source: (Hoffman et al., 2022)]

Generalization to C4 and Github code

Approach	Coef. a where $N_{opt} \propto C^a$	Coef. b where $D_{opt} \propto C^b$
C4	0.50	0.50
GitHub	0.53	0.47
Kaplan et al. (2020)	0.73	0.27

[Figure Source: (Hoffman et al., 2022)]

Generalization to Image, Video, etc.

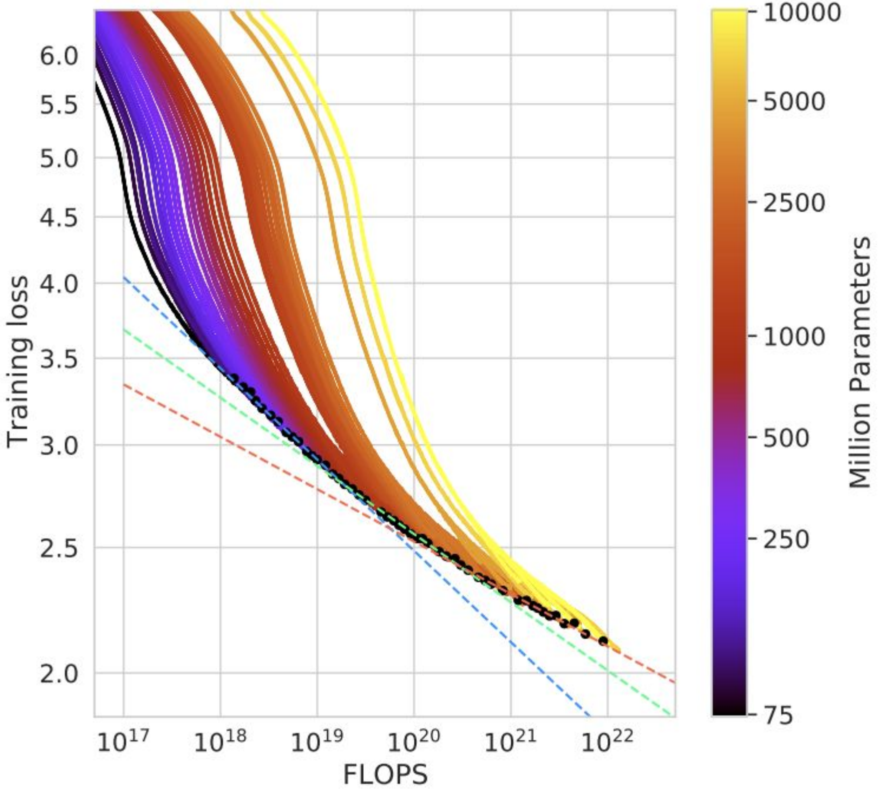


[Figure Source: (Henighan et al., 2020)]

Is Power-Law the best fit?

(Hoffman et al.) observe **concavity in their model at high compute budgets**

The importance of **dataset** might **increase** for **high compute budgets.**



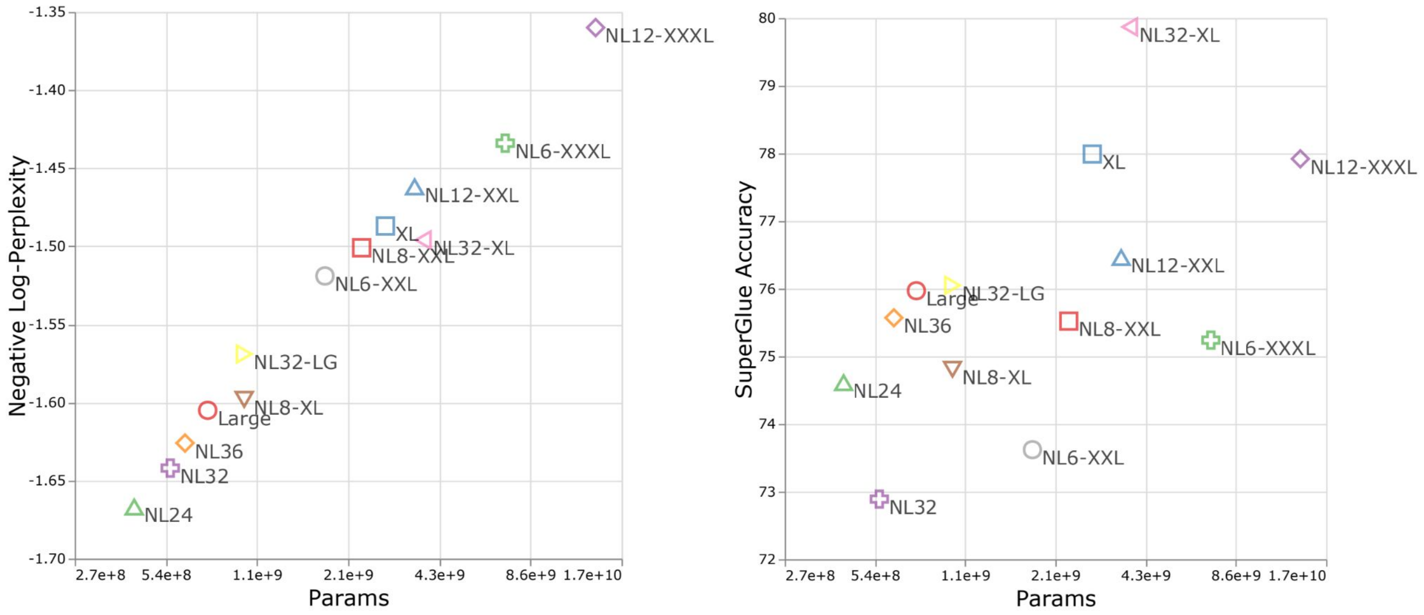
[Figure Source: (Hoffman et al., 2022)]

Scaling Law For Fine-Tuning (Tay et al., 2021)

Downstream performance **after fine-tuning** does not scale with model size

Downstream performance does **scale with depth**, but not necessarily with dimension

Downstream Performance Does Not Depend on N

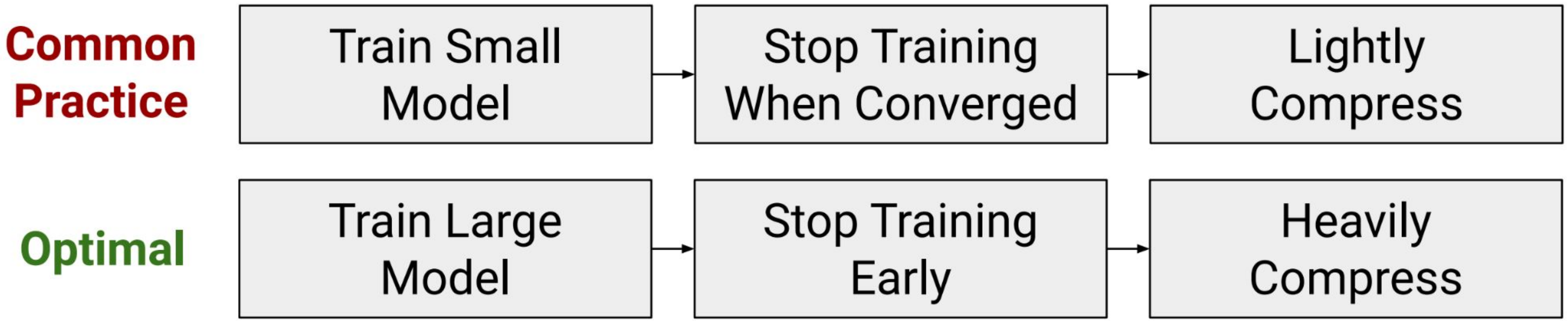


(a) Pre-training scaling

(b) Fine-tuning scaling

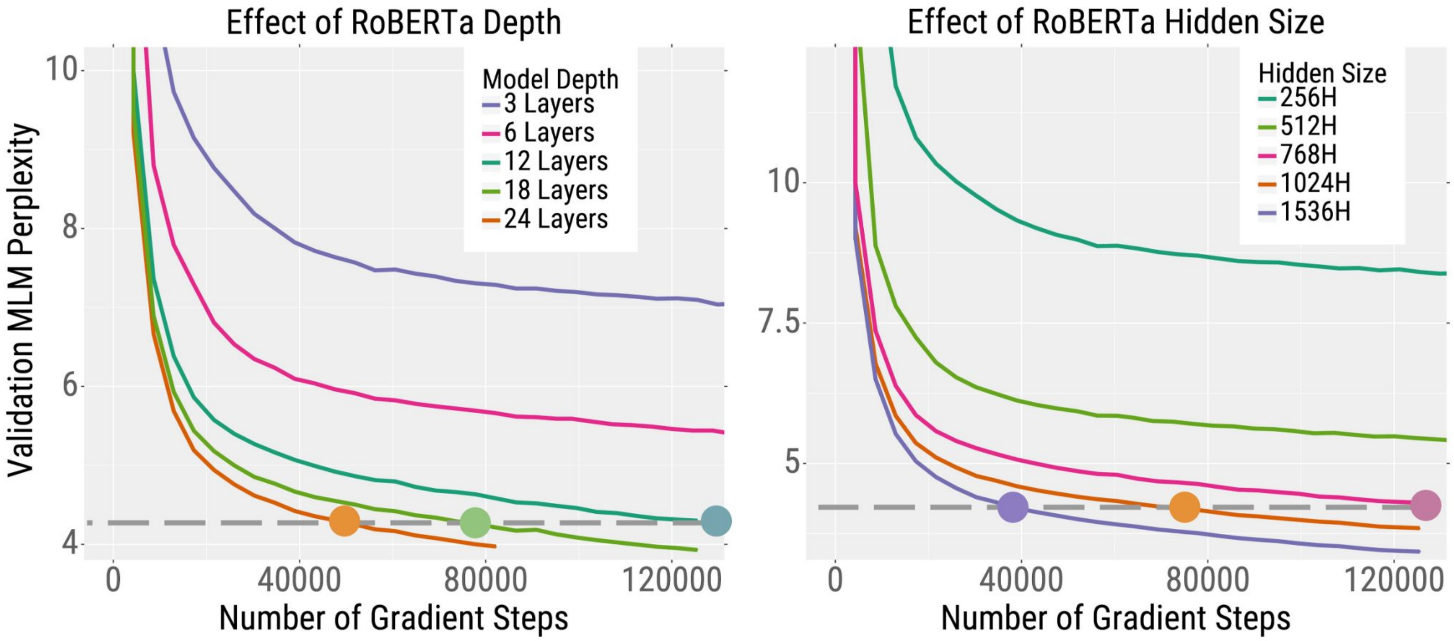
[[Figure Source: (Tay et al., 2021)]]

Train Large, Then Compress (Li et al., 2020)



[Figure Source: (Li et al., 2020)]

Deeper and Wider Models Converge in Fewer Steps



[Figure Source: Slides by (Li et al., 2020)]

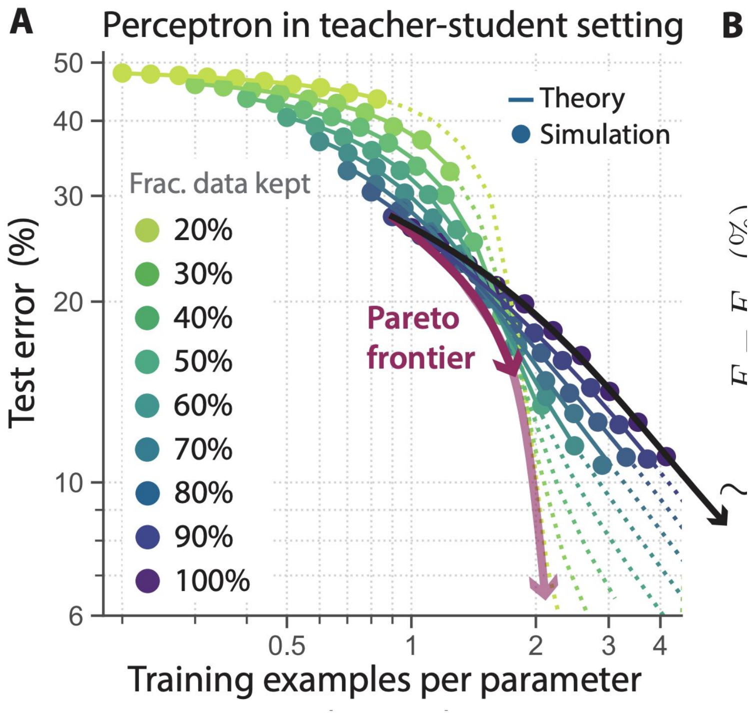
Data Pruning (Sorcher et al., 2022)

Develop a metric to measure the **quality of data**

Prune the data to include only high quality data

Importance of dataset size decreases significantly

The More Data We Prune, The Less Data Matters



[Figure Source: (Sorcher et al., 2022)]

Q3: (a) Do you think we can extend this study of LLMs to other types such as encoder-decoder models? Can you make your guess of the scaling law?

(b) These studies simply consider # of tokens as a proxy for training corpus. Do you think it is possible to take the quality/redundancy of the training data into account?