

Training Large Language Models

Scaling Laws and Chinchilla

Learning goals

- Understand Chinchilla
- Understand the various scaling laws

SCALING LAWS

► Kaplan et al. (2020)

- Performance depends strongly on scale, weakly on model shape
 - Scale means: parameters N , data D , and compute C
 - Shape means: depth and width
- Smooth power laws
 - Performance has power-law relation with each factor N , D , C
 - When not bottlenecked by the other two
 - Trend spanning more than six orders of magnitude
- Universality of overfitting
 - Performance enters regime of diminishing returns if N or D held fixed while the other increases

SCALING LAWS

- Universality of training
 - Training curves follow predictable power-laws
 - Their parameters are roughly independent of model size
 - It is possible to predict by extrapolating the early part of the training curve
- Transfer improves with test performance
 - When evaluating on text with different distribution from training text, results are strongly correlated to those on the validation set
 - Transfer to different distribution incurs a constant penalty but improves in line with performance on training set
- Sample efficiency
 - Large models are more sample-efficient than small models
 - They reach same performance with fewer optimization steps

SCALING LAWS

- Convergence is inefficient
 - When C is fixed but N and D are not, optimal performance is achieved by training very large models and stopping significantly short of convergence
- Optimal batch size
 - Ideal size is a power of the loss only
 - It is ~ 1 -2 million tokens for the largest models we can train

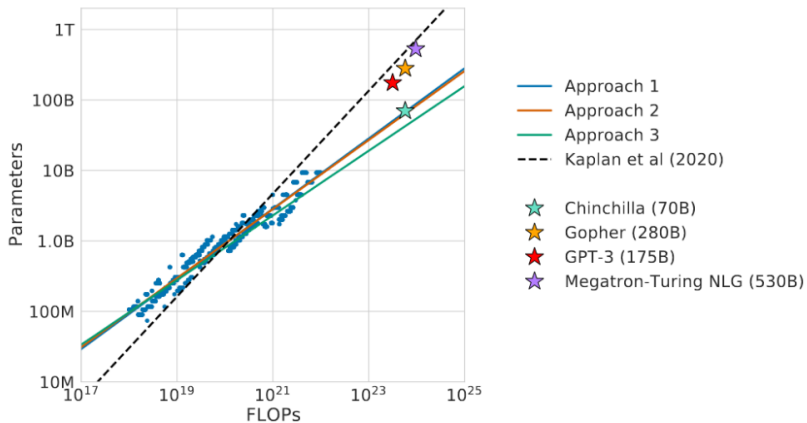
Larger language models will perform better and be more sample efficient than current models.

COMPUTE-OPTIMAL LLMs

Given a fixed FLOPs budget, how should we trade-off model size and text size to optimize performance? ► Hoffmann et al., 2022

- Find N and D so that $FLOPs(N, D) = C$ and $L(N, D)$ is minimal
- Empirically estimated N and D based on 400 models.
 - Ranging from 70 M to 16 B parameters
 - Trained on 5 B to 400 B tokens
- Results verified using Chinchilla
 - Chinchilla has 70 B parameters and is trained on 1.4 T tokens
 - 4x less parameters and 4x more tokens than Gopher
 - Chinchilla outruns Gopher and has reduced memory footprint and inference cost

COMPUTE-OPTIMAL LLMs



► Source: Hoffmann et al., 2022

CHINCHILLA AND THE OTHER LLMs

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
<i>Gopher</i> (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
<i>Chinchilla</i>	70 Billion	1.4 Trillion

► Source: Hoffmann et al., 2022

Model	Layers	Number Heads	Key/Value Size	d_{model}	Max LR	Batch Size
<i>Gopher</i> 280B	80	128	128	16,384	4×10^{-5}	3M \rightarrow 6M
<i>Chinchilla</i> 70B	80	64	128	8,192	1×10^{-4}	1.5M \rightarrow 3M

► Source: Hoffmann et al., 2022

CHINCHILLA ON MMLU

Random	25.0%
Average human rater	34.5%
GPT-3 5-shot	43.9%
<i>Gopher</i> 5-shot	60.0%
<i>Chinchilla</i> 5-shot	67.6%
Average human expert performance	89.8%
June 2022 Forecast	57.1%
June 2023 Forecast	63.4%

► Source: Hoffmann et al., 2022

CHINCHILLA ON QA

	Method	<i>Chinchilla</i>	<i>Gopher</i>	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%	-	

► Source: Hoffmann et al., 2022