Natural Language Processing

Scaling laws

Marco Kuhlmann
Department of Computer and Information Science

LINKÖPING UNIVERSITY

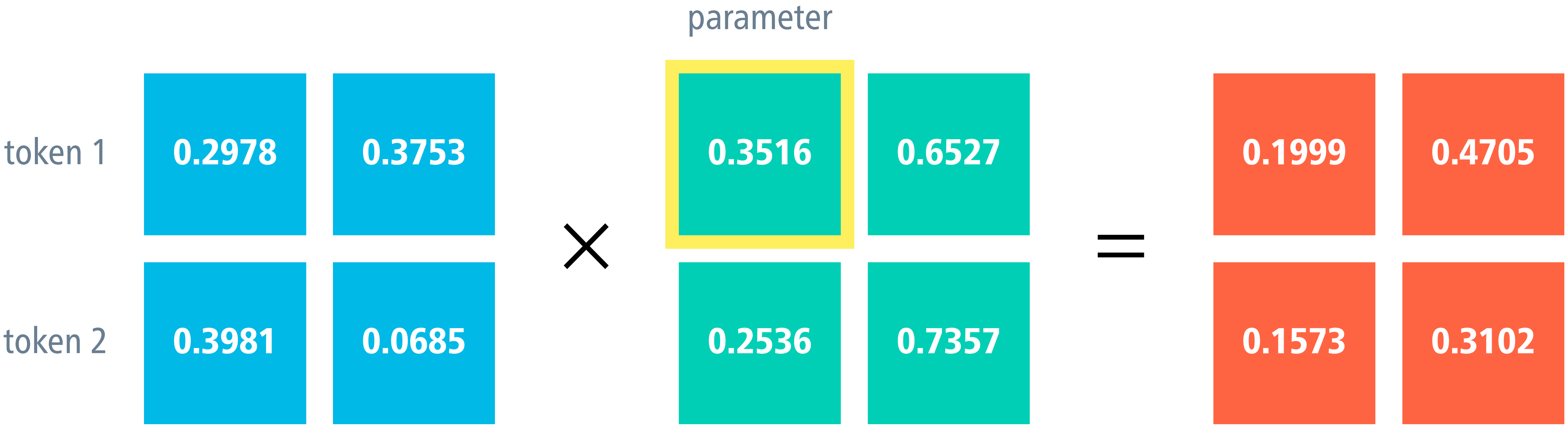
Scaling laws in language modelling

- **Scaling laws** describe how model performance improves as we increase key factors such as model size and training data size.
- Empirical results suggest that performance improvements obey a power law: performance increases, but at a diminishing rate.

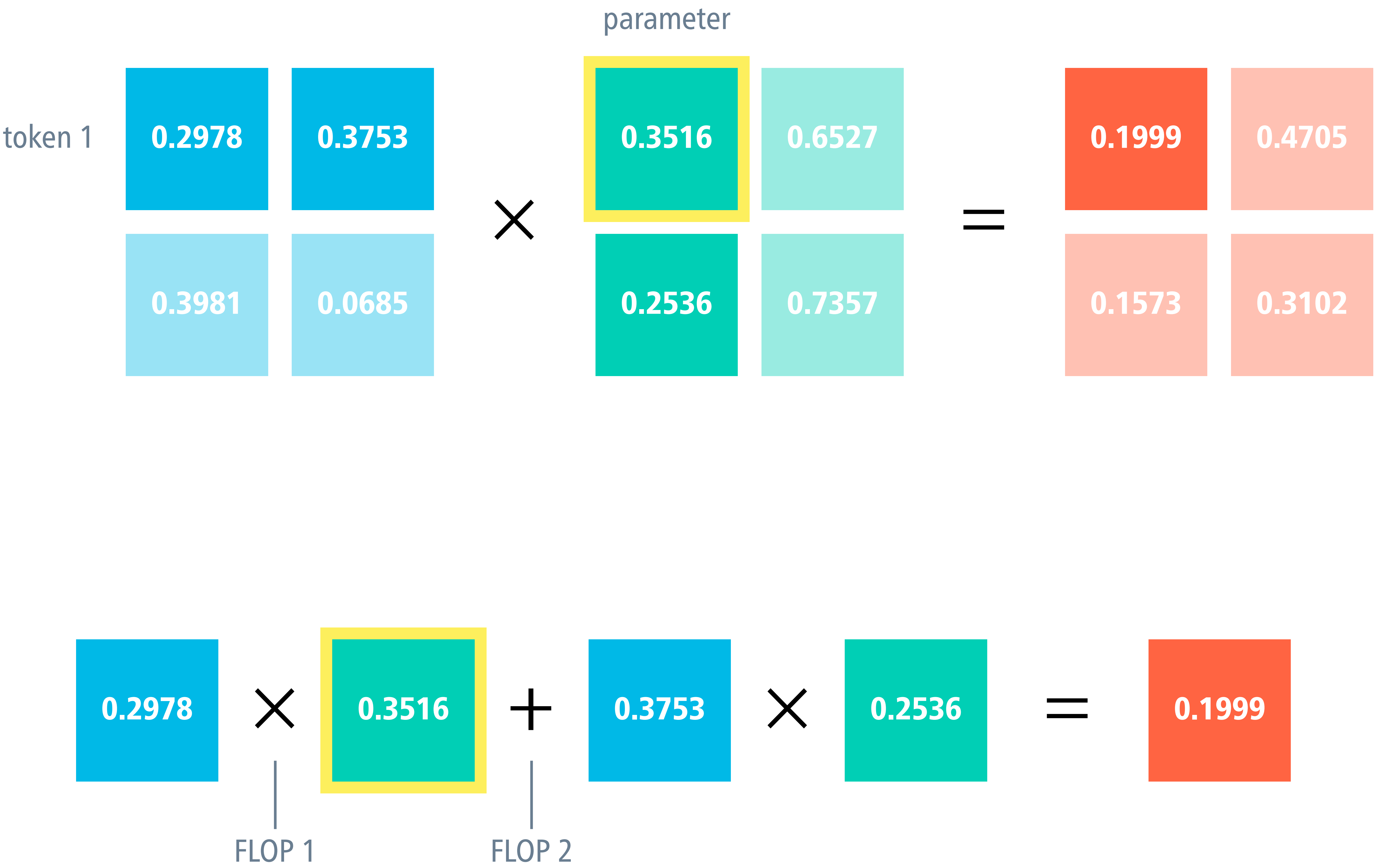
  cf. Heap's law
- Scaling laws can help developers answer many practically relevant questions about resource allocation.

Computational cost

- The cost of language model training is a function of the number of model parameters, P, and the number of training tokens, T.
- The standard unit for measuring computational cost is the number of **floating point operations (FLOPs)**.
- For the Transformer architecture, a useful approximation for the computational cost C is $C \cong 6PT$.

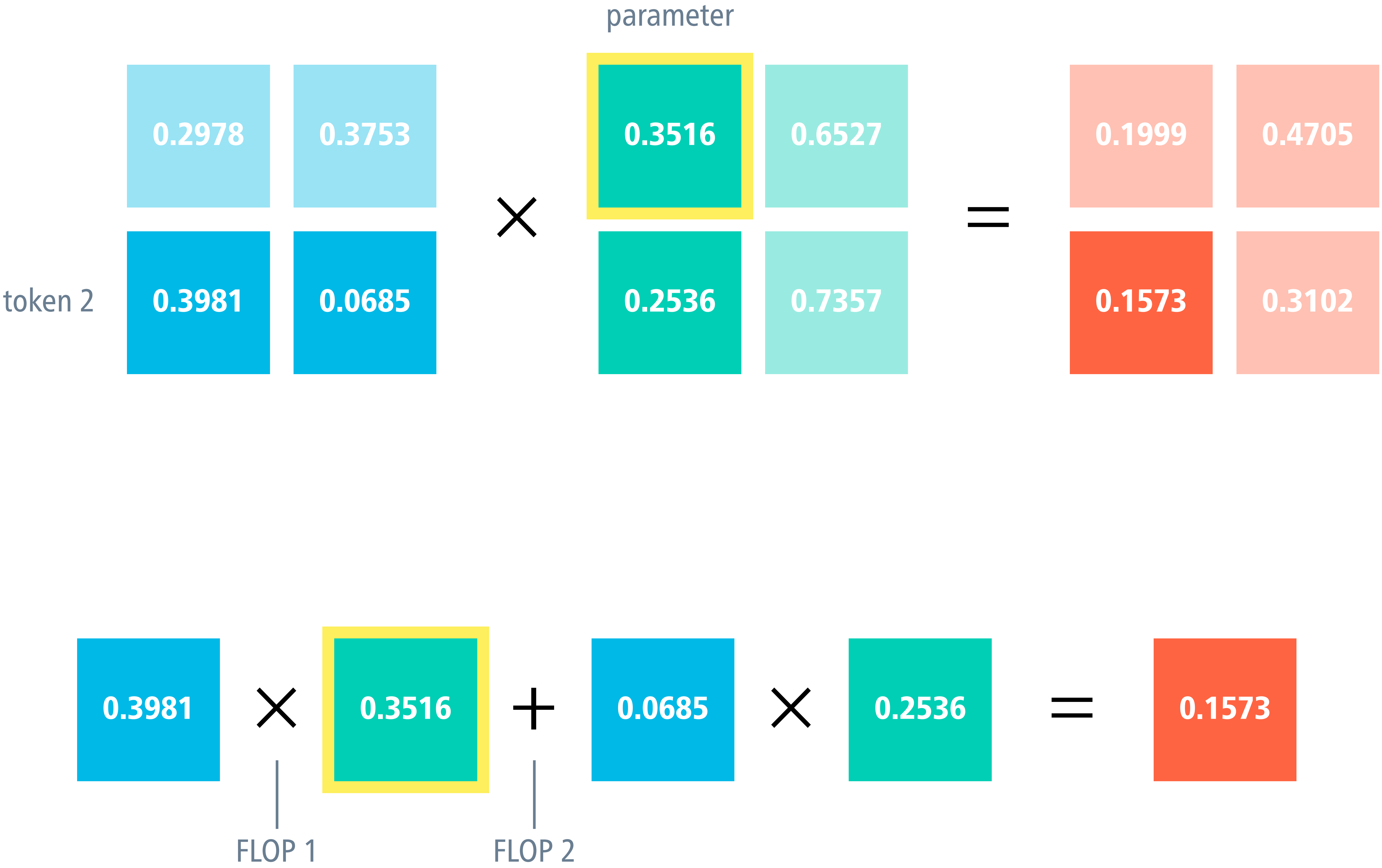


$$\begin{array}{lcc} \text{token 1} & 0.2978 & 0.3753 \\ \text{token 2} & 0.3981 & 0.0685 \end{array} \times \begin{array}{cc} \overset{\text{parameter}}{\boxed{0.3516}} & 0.6527 \\ 0.2536 & 0.7357 \end{array} = \begin{array}{cc} 0.1999 & 0.4705 \\ 0.1573 & 0.3102 \end{array}$$



$$\begin{array}{ll}\text{token 1} & 0.2978 \quad 0.3753\\ & 0.3981 \quad 0.0685\end{array} \times \begin{bmatrix}\overset{\text{parameter}}{\boxed{0.3516}} & 0.6527\\ 0.2536 & 0.7357\end{bmatrix} = \begin{bmatrix}0.1999 & 0.4705\\ 0.1573 & 0.3102\end{bmatrix}$$

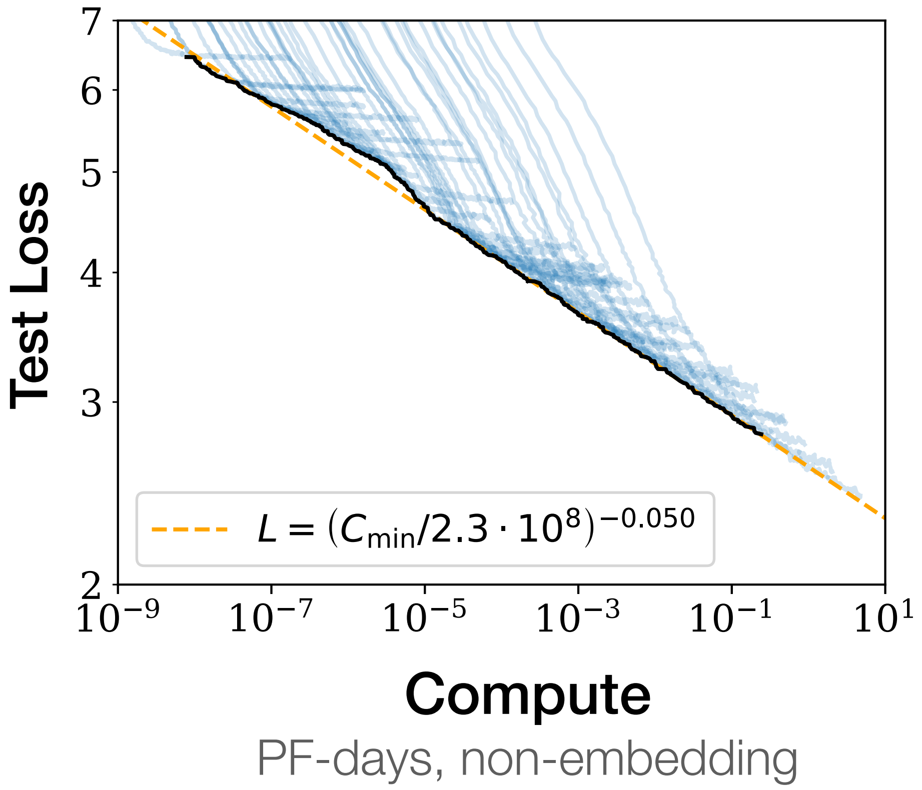
$$0.2978 \underset{\text{FLOP 1}}{\times} \boxed{0.3516} \underset{\text{FLOP 2}}{+} 0.3753 \times 0.2536 = 0.1999$$

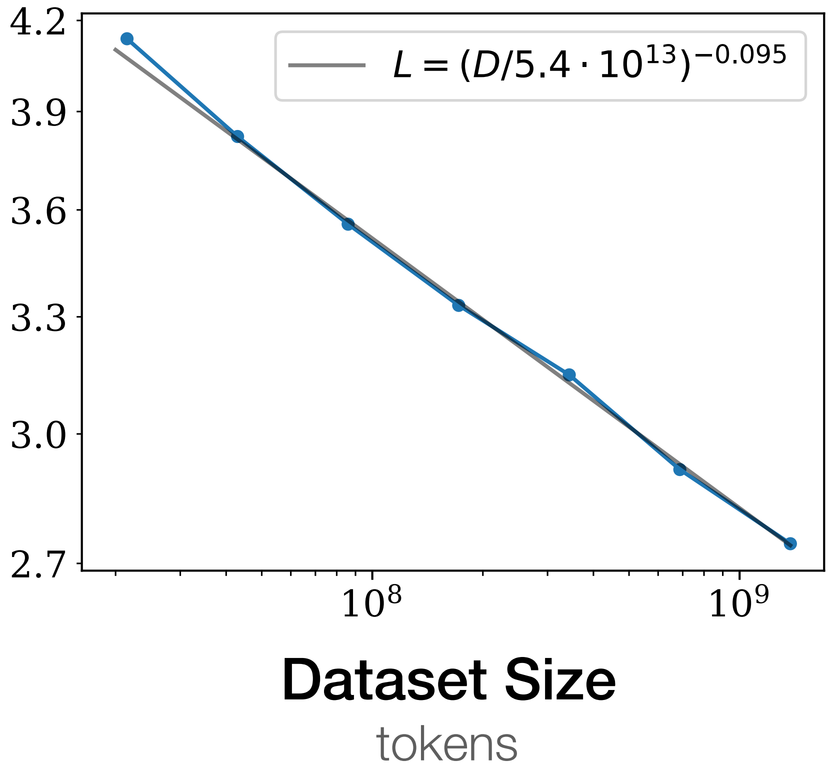


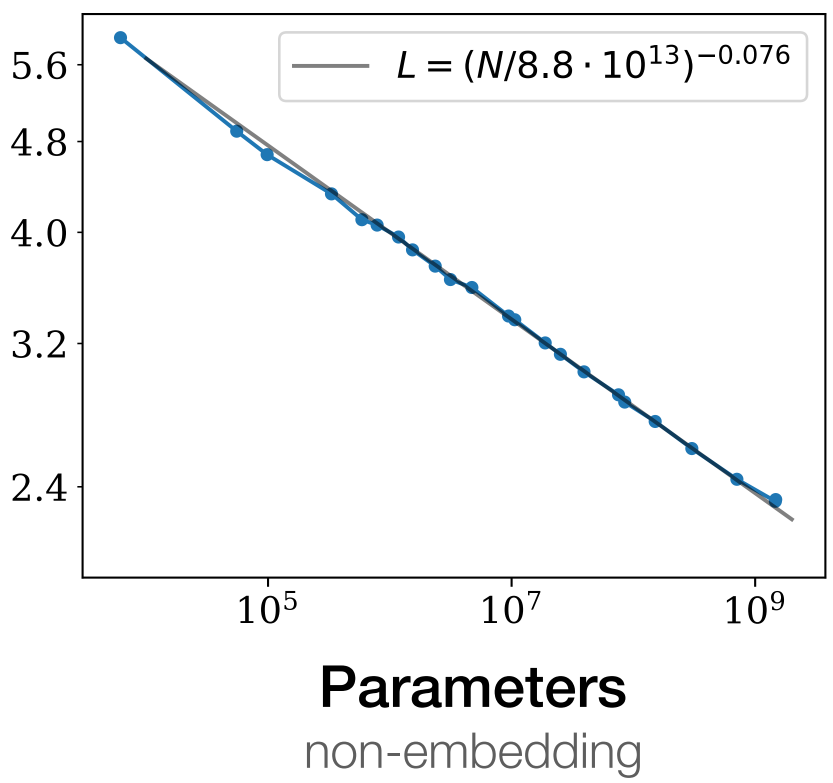
$$\begin{bmatrix} 0.2978 & 0.3753 \\ 0.3981 & 0.0685 \end{bmatrix} \times \begin{bmatrix} 0.3516 & 0.6527 \\ 0.2536 & 0.7357 \end{bmatrix} = \begin{bmatrix} 0.1999 & 0.4705 \\ 0.1573 & 0.3102 \end{bmatrix}$$

$$0.3981 \times 0.3516 + 0.0685 \times 0.2536 = 0.1573$$

Performance improves smoothly with scale







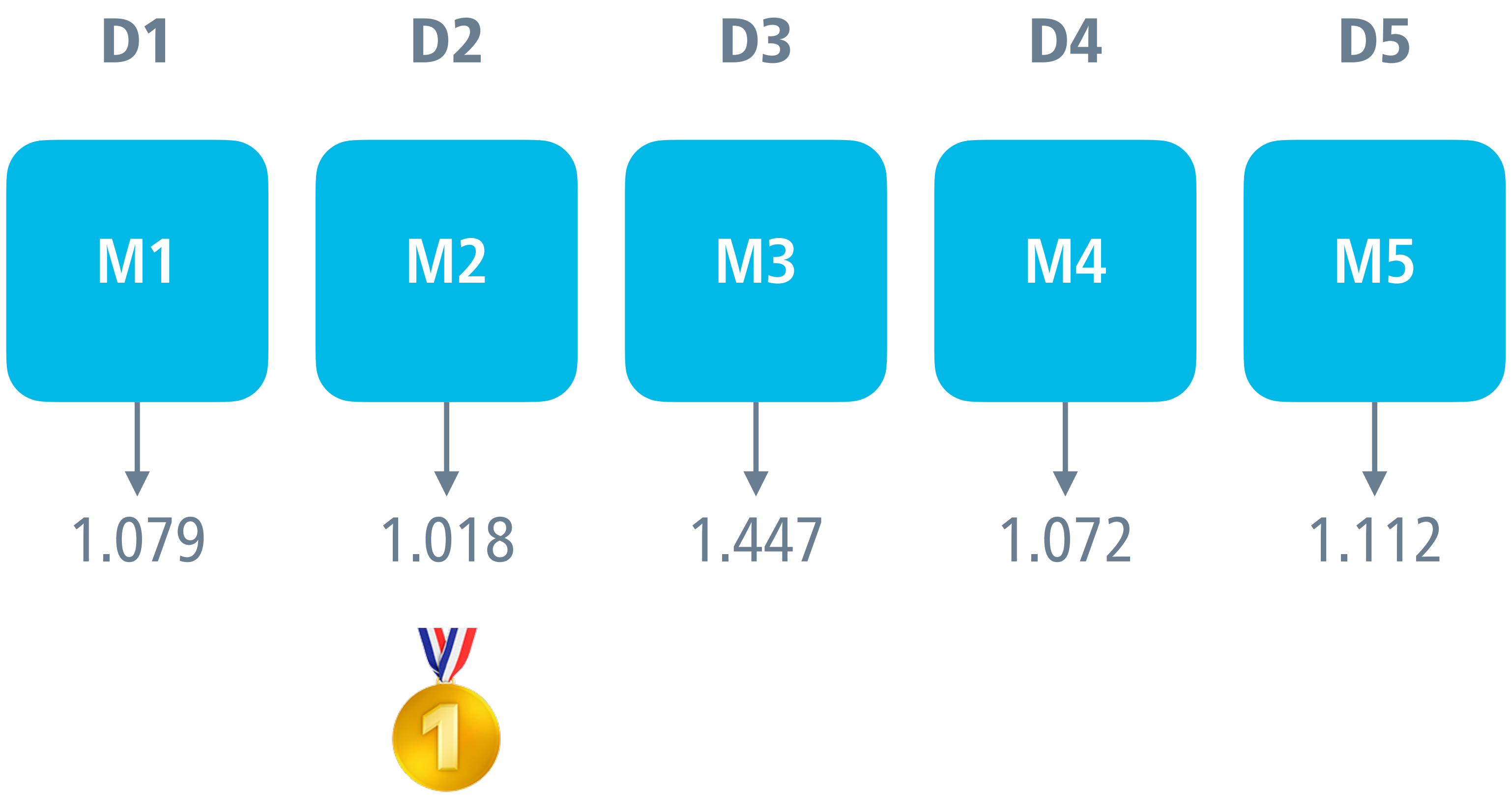
Performance improves smoothly as we increase **compute**

Performance improves smoothly as we increase **dataset size**

Performance improves smoothly as we increase **model size**

Kaplan et al. (2020)

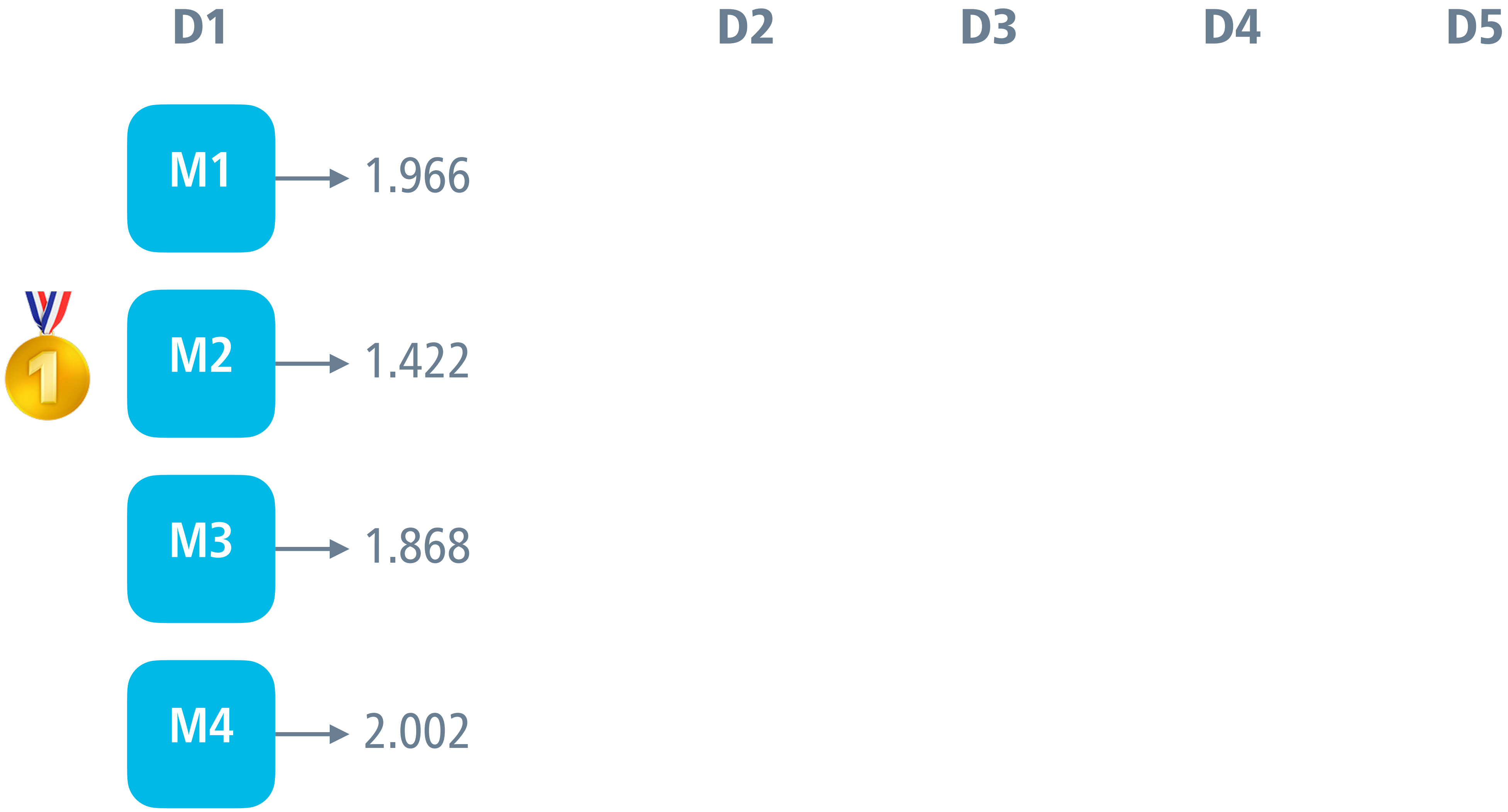
Putting scaling laws into practice



D1	D2	D3	D4	D5
M1	M2	M3	M4	M5
1.079	1.018	1.447	1.072	1.112
	🥇			

old paradigm: train a few models, select the best one

Putting scaling laws into practice



D1 D2 D3 D4 D5

M1 → 1.966

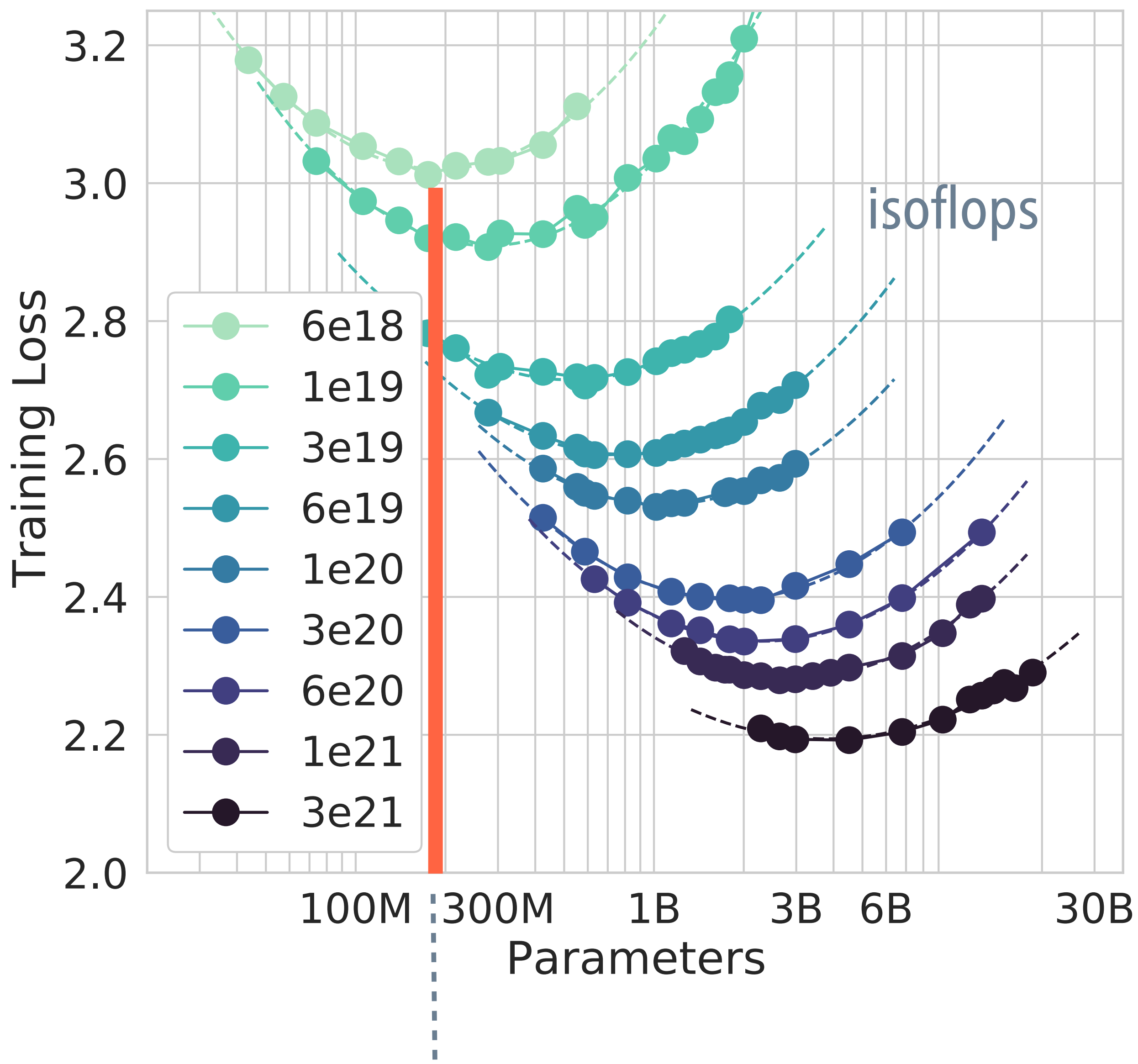
🥇 M2 → 1.422

M3 → 1.868

M4 → 2.002

new paradigm: train many small models, up-scale the best one

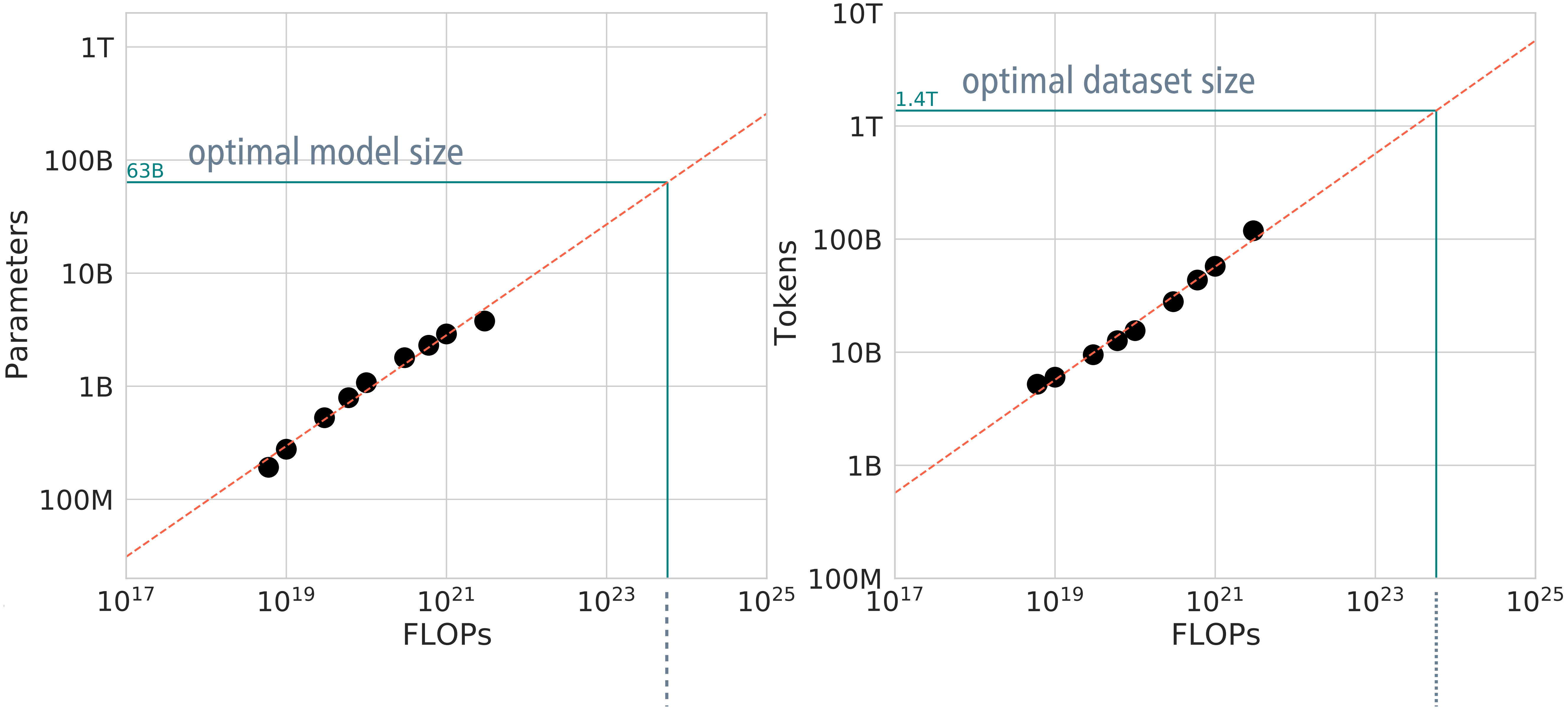
Compute-optimal models



isoflops

ca. 200M parameters

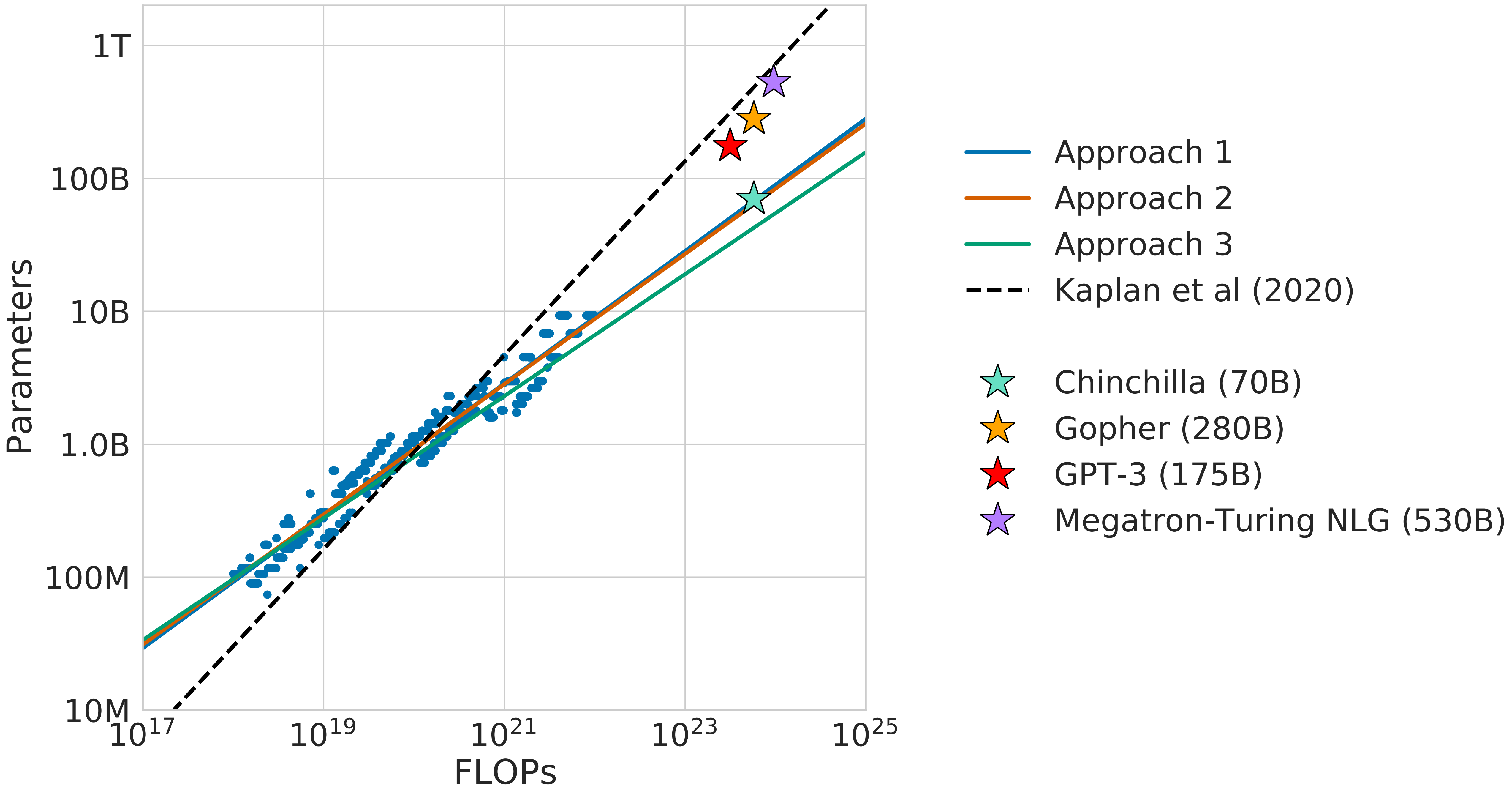
Compute-optimal models



576 billion TFLOPs

576 billion TFLOPs

Large language models can be too large


1T
100B
10B
1.0B
100M
10M
Parameters
10^{17}
10^{19}
10^{21}
10^{23}
10^{25}
FLOPs
Approach 1
Approach 2
Approach 3
Kaplan et al (2020)
Chinchilla (70B)
Gopher (280B)
GPT-3 (175B)
Megatron-Turing NLG (530B)