Joint MoE Scaling Laws: Mixture of Experts Can Be Memory Efficient

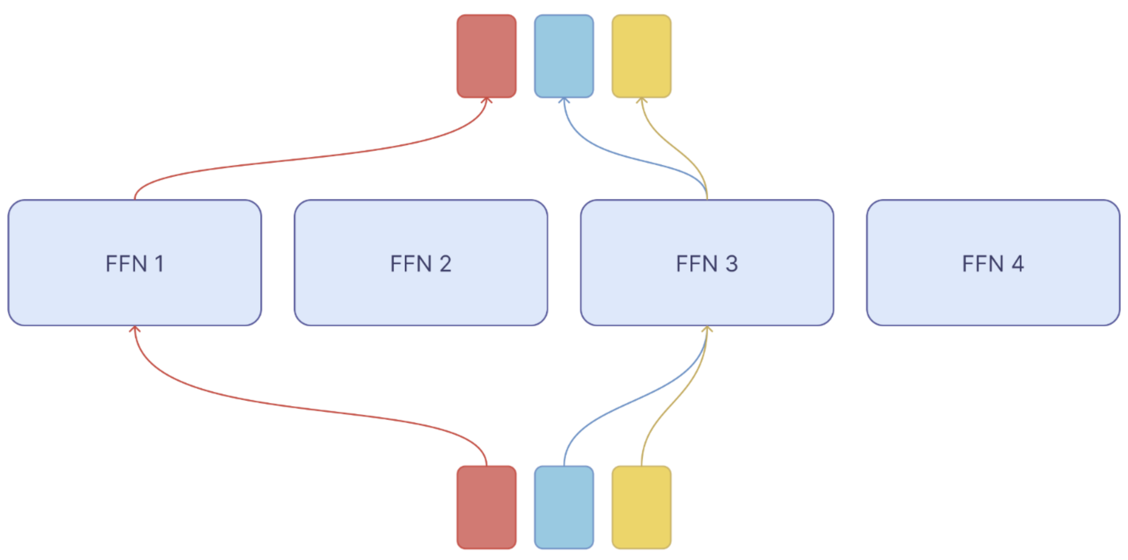
IDEAS NCBR

Jan Ludziejewski* Maciej Pióro* Jakub Krajewski* Maciej Stefaniak Michał Krutul
Jan Małaśnicki Marek Cygan Piotr Sankowski Kamil Adamczewski Piotr Miłoś Sebastian Jaszczur

University of Warsaw

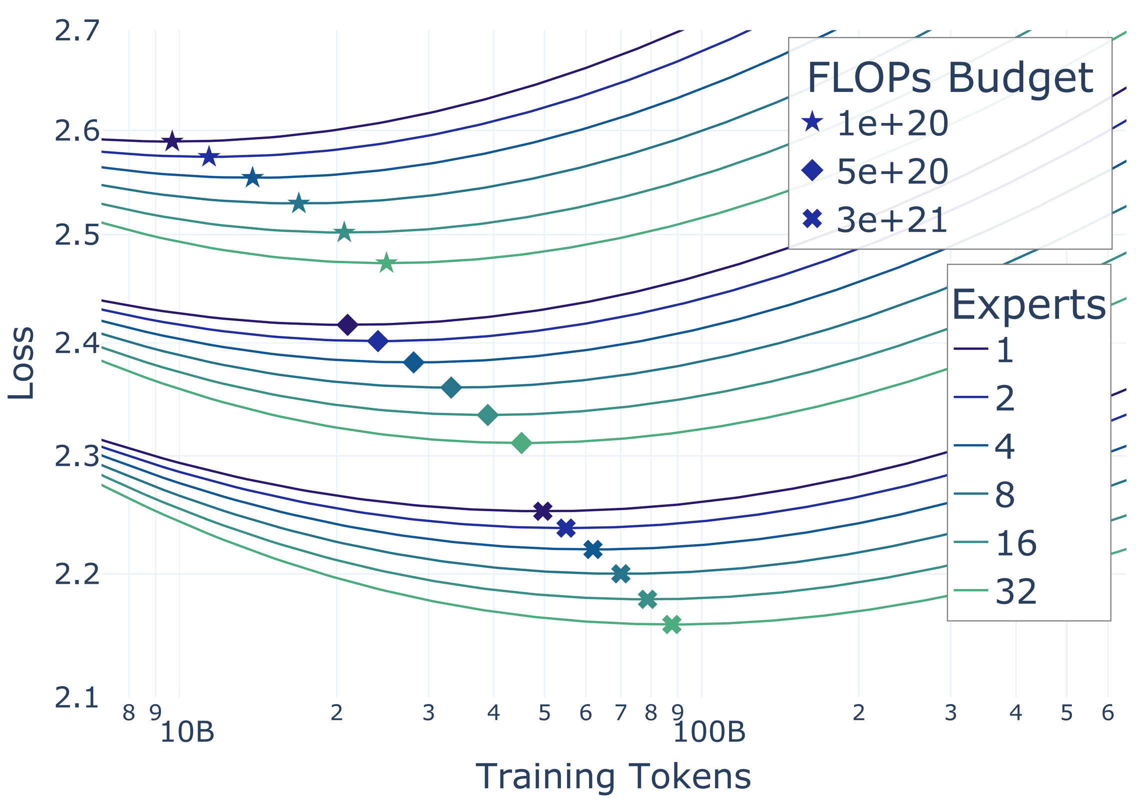
Motivation

- Key question: Can a Mixture of Experts (MoE) model be the optimal choice if the inference memory is limited (e.g. a 16 GB GPU)?
- MoE layers route each token through only a fraction of the model's parameters, lowering both inference and training FLOPs.
- With the same device memory, MoE has fewer active parameters than a dense model, but those parameters can be trained for longer (more tokens).
- Is that enough to offset the capacity gap? Our scaling law analysis says yes.
- Joint scaling with variable model size N, training tokens D, and number of experts E reveals optimal token-to-param ratios and performance under memory/compute constraints.



1. More experts ⇒ higher tokens-to-param ratio

Assuming a fixed compute budget, increasing experts means you should decrease N_{act} and increase D. See Table 1.

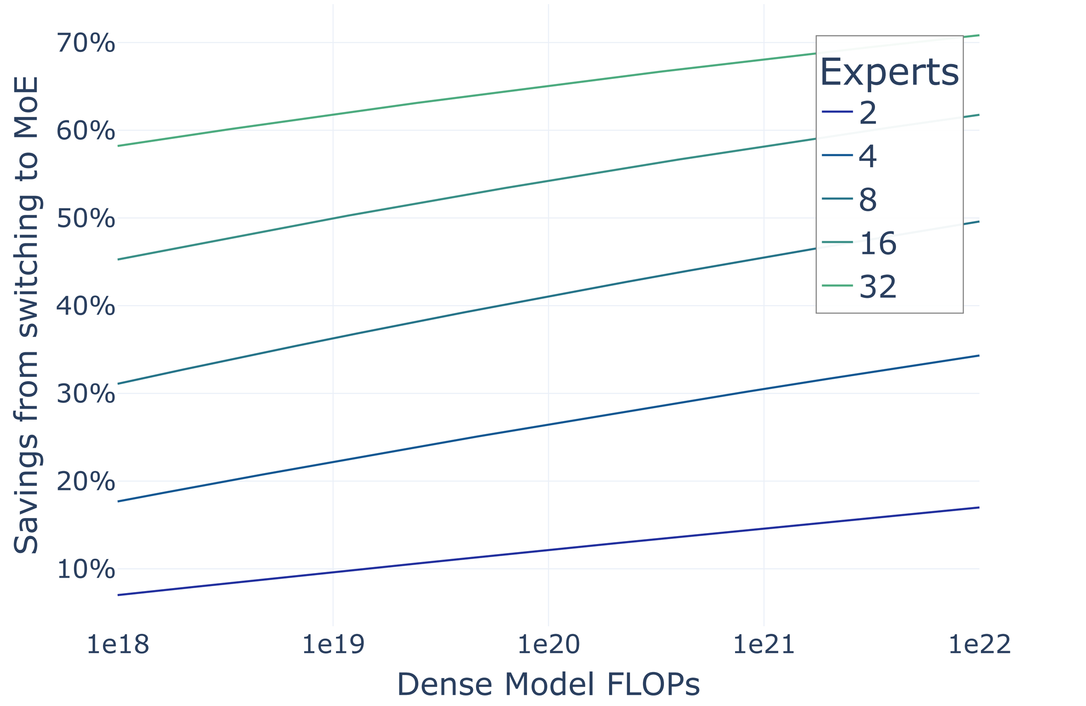


IsoFLOP curves from the joint fit

Training Budget (FLOPs)	Experts	$N_{\text{act}}^{\text{opt}}$	D^{opt}
10^{20}	1	1.7B	9.7B
	2	1.5B	11.4B
	4	1.2B	13.9B
	8	990M	17B
	16	810M	20.7B
	32	669M	24.9B
10^{22}	1	18.8B	88.6B
	2	17.4B	96B
	4	15.8B	105.4B
	8	14.4B	115.8B
	16	13.2B	126.5B
	32	12.2B	136.9B

Table 1. Compute-optimal settings.

Dense vs MoE (Compute Savings)



Joint Scaling Law

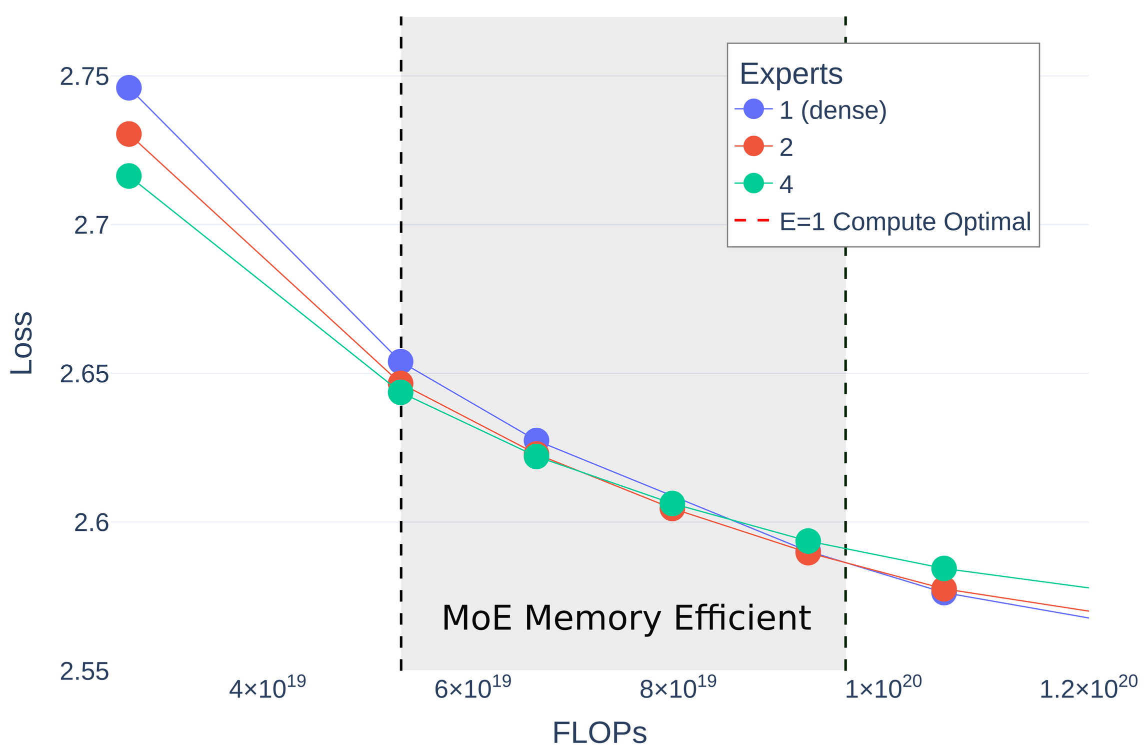
$$
\mathcal{L}(N_{\text{act}}, D, \hat{E}) = a\hat{E}^{\delta}\, N_{\text{act}}^{\alpha+\gamma \ln \hat{E}} + b\hat{E}^{\omega}\, D^{\beta+\zeta \ln \hat{E}} + c
$$

- Unifies dense (E=1) and MoE models; reduces to Chinchilla-like scaling law for each E.
- Captures interaction among active params N_{act}, training tokens D, and experts E.

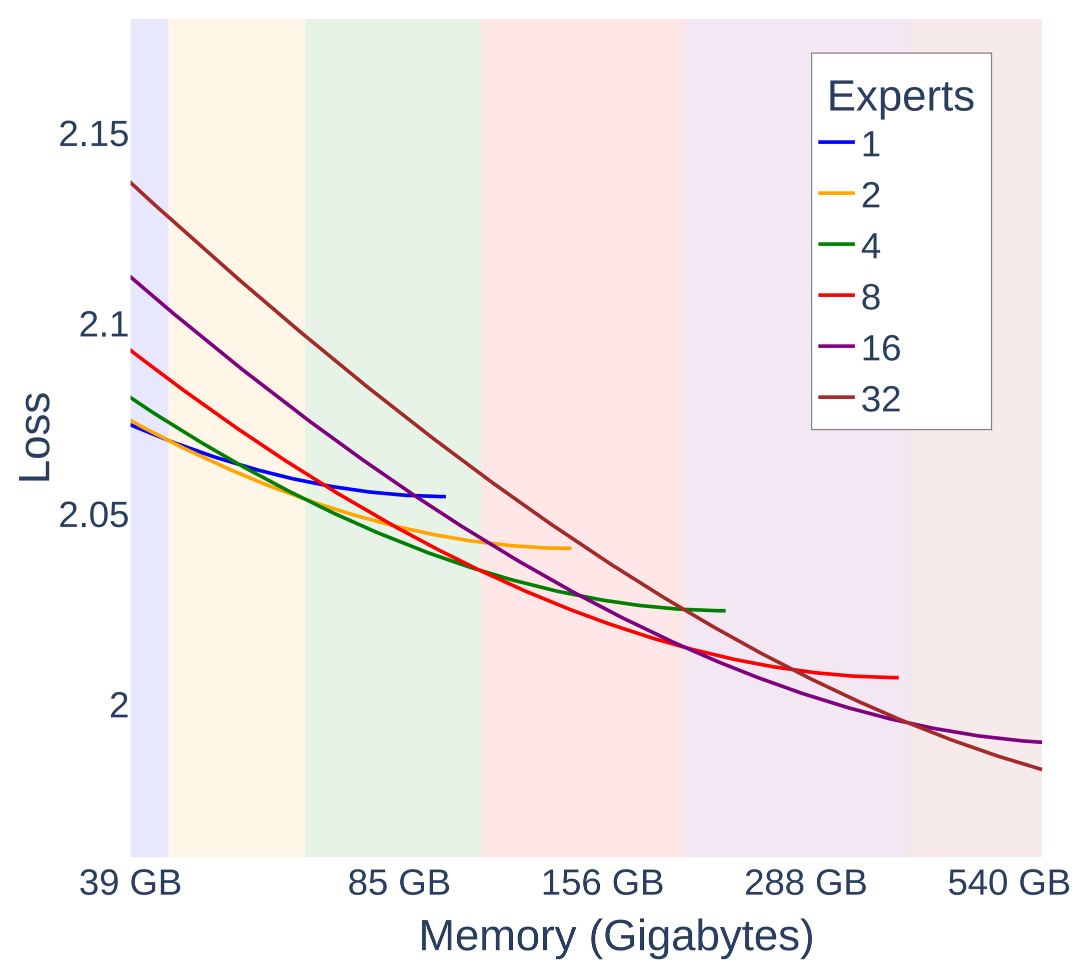
2. MoE can also be *memory* optimal

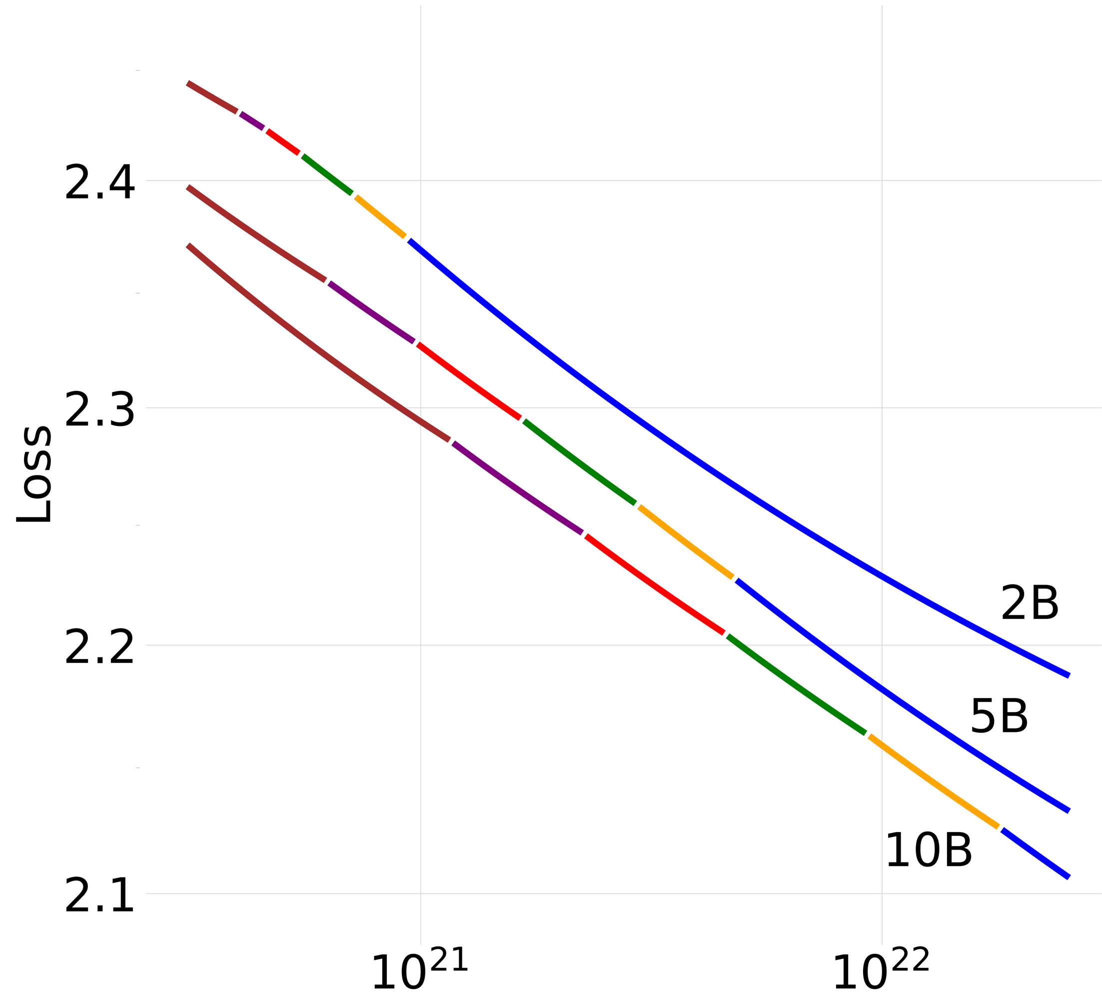
A total-parameter-matched MoE can outperform a dense model trained with the same compute. Such an MoE is more compute and memory efficient at inference.

MoE Outperforming a Total-Parameter-Matched Dense Model



Varying Memory and Compute Constraints





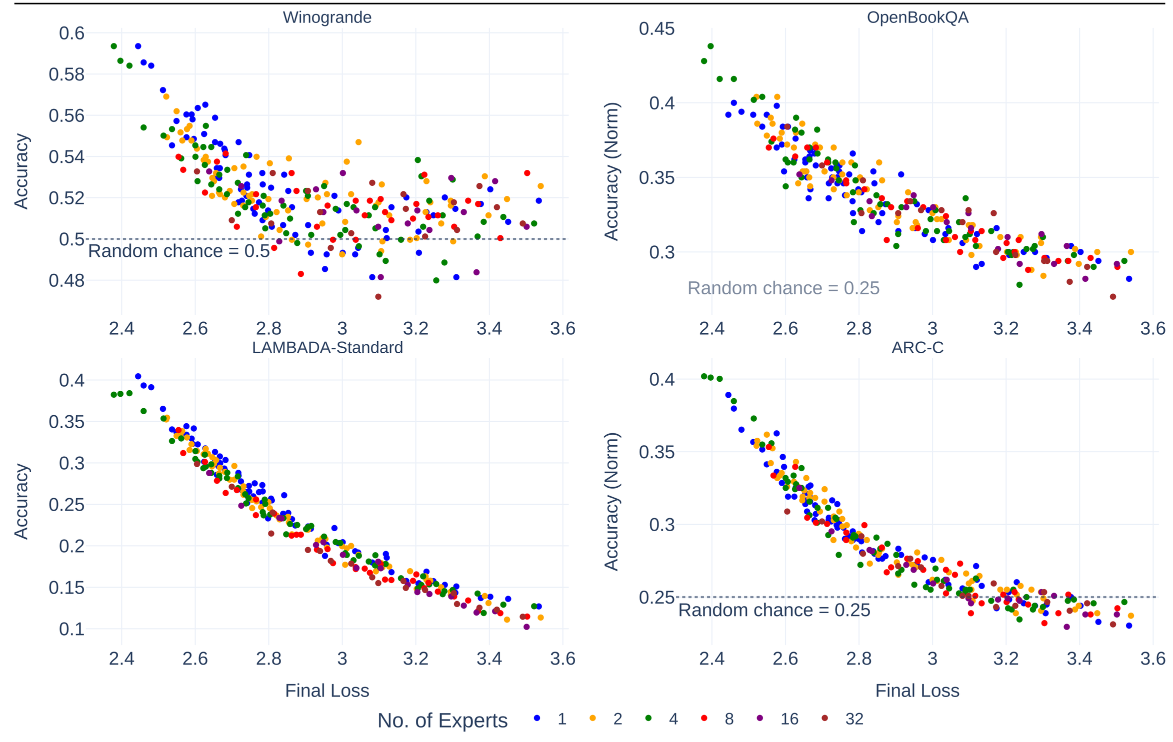
3. More experts → better performance.

For a given compute budget, increasing the number of experts always improves performance, provided the size of the model and the number of training tokens are adjusted.

Rule of Thumb

For a fixed total parameter count, an MoE with $E \leq 8$ beats a compute-optimal dense model if trained on $\times E$ more tokens while keeping the same memory footprint.

Downstream Performance



Depending on the task, we can see positive, negative, or no impact of MoE given the same pretraining perplexity. Generally, task performance correlates strongly with pretraining loss. Low expansion rates behave more like dense models.

Experiments & Fit Quality

- Runs: 270 unique training instances
- Parameter range: 80M–5B total
- Batch-size ramp-up + WSD LR schedule
- LR scaled optimally for each configuration

Fit diagnostics

4. More experts ⇒ lower learning rate

Increase E ⇒ decrease LR:

$$\ln LR_{\text{peak}} = 8.39 - 0.81 \ln N_{\text{act}} - 0.25 \ln E$$

Read the Paper & Access the Checkpoints

arXiv

Hugging Face