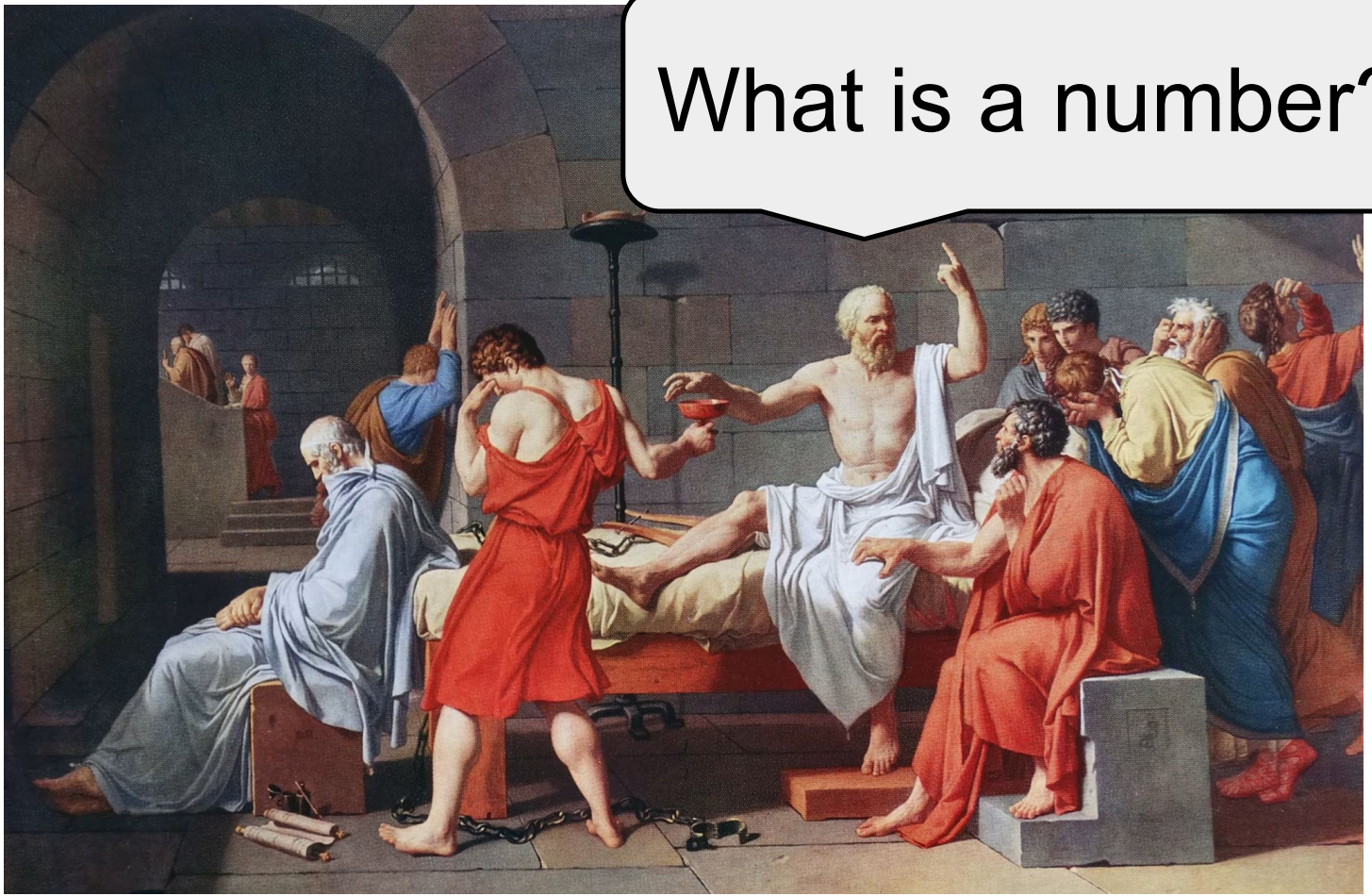
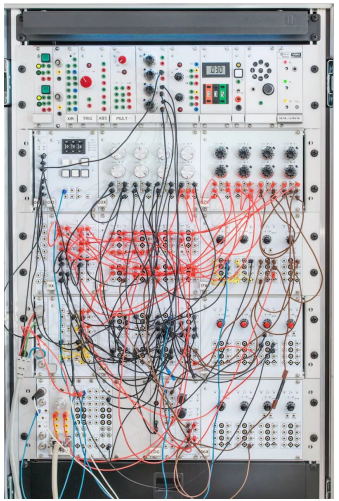
Training Dynamics Impact Post-Training Quantization Robustness

Albert Catalan-Tatjer, Niccolò Ajroldi, Jonas Geiping


What is a number?

Analog vs Digital

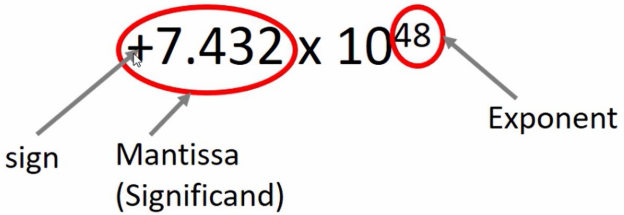
+ Infinite resolution
- Measurement error

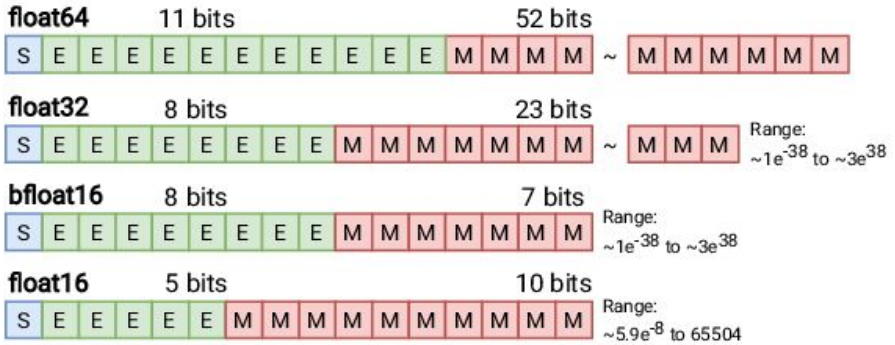


+ Precise (0 or 1)
- Limited resolution

Floating point formats

The IEEE-754 floating-point standard





How many bits do we need for “intelligence”

THE WALL STREET JOURNAL.

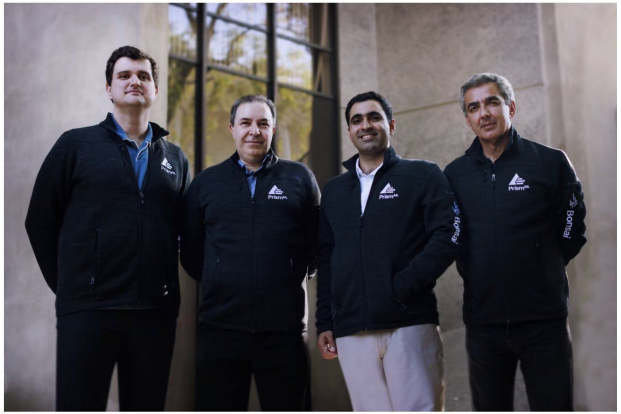
EXCLUSIVE STEVEN ROSENBUSH

Caltech Researchers Claim Radical Compression of High-Fidelity AI Models

PrismML says its 1-bit large language model achieves radical compression without sacrificing performance, lowering energy consumption

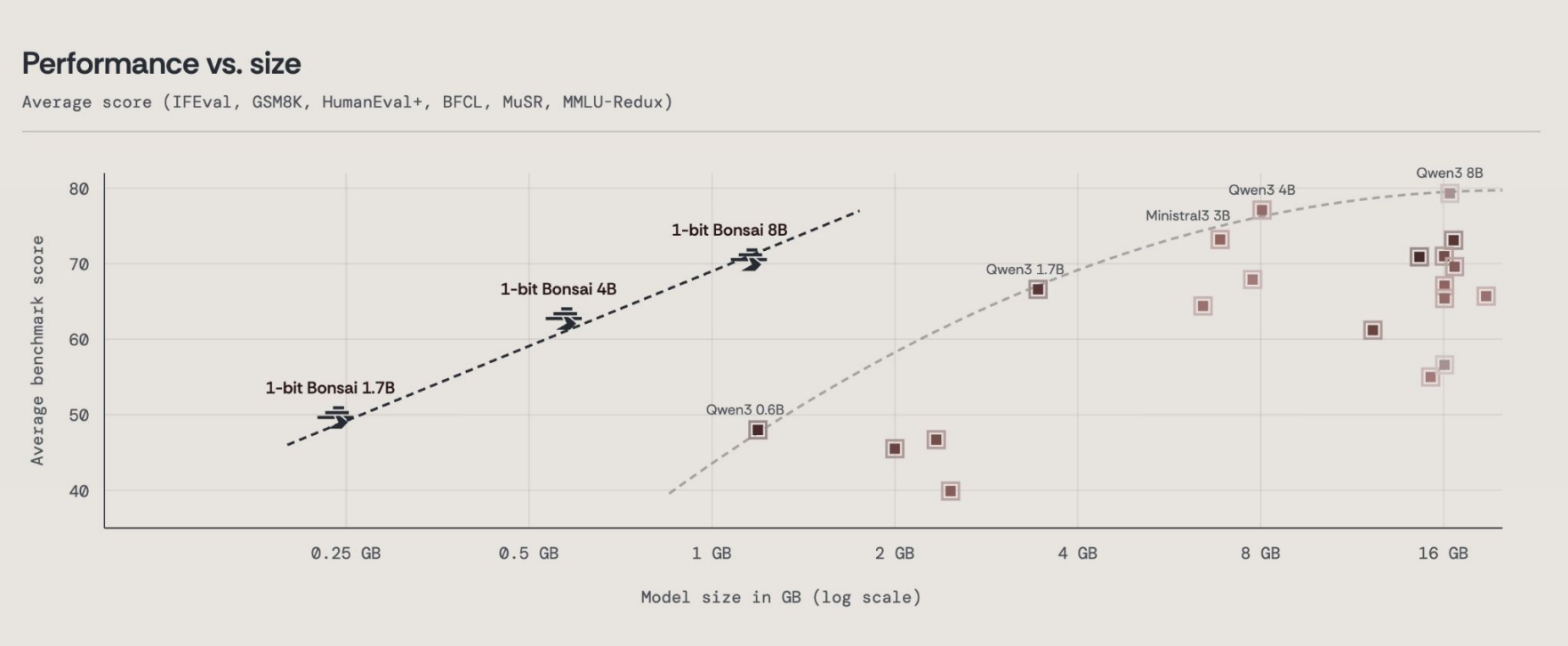
By Steven Rosenbush Follow

March 31, 2026 2:00 pm ET



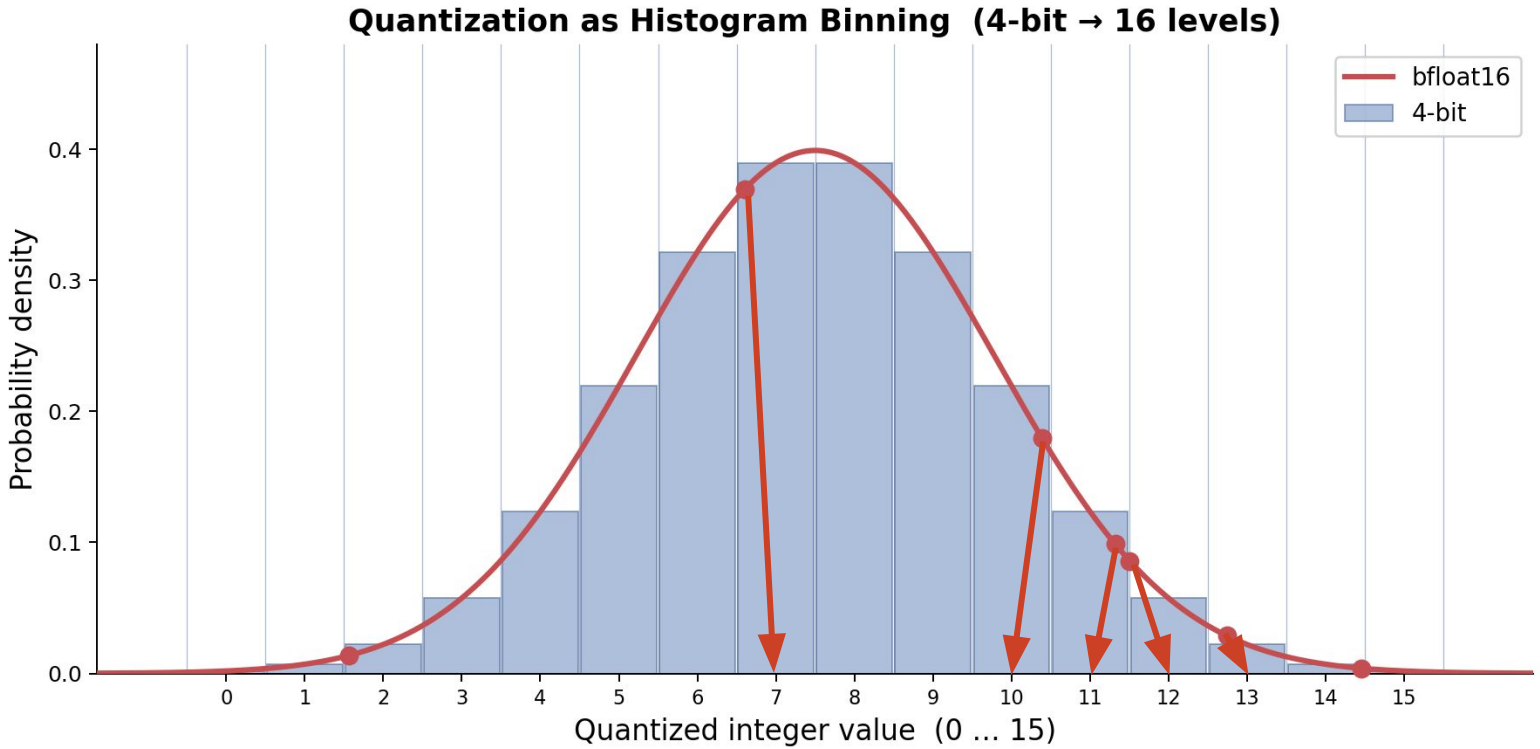
The PrismML team, from left: Sahin Lale, co-founder, Babak Hassibi, co-founder and CEO, Omead Pooladzandi, co-founder, and Reza Sadri, co-founder and vice president, strategy. ENOCH KIM

How many bits do we need for “intelligence”



What is Quantization?

Scaling and shifting required to match the original and quantized ranges.



Frontier of Quantization

Modify multiplication operands to ease quantization:

$$\mathbf{W}\mathbf{X}^T \longrightarrow \mathbf{Q}(\mathbf{W}\mathbf{R})\mathbf{Q}(\mathbf{R}^{-1}\mathbf{X}^\mathbf{T})$$

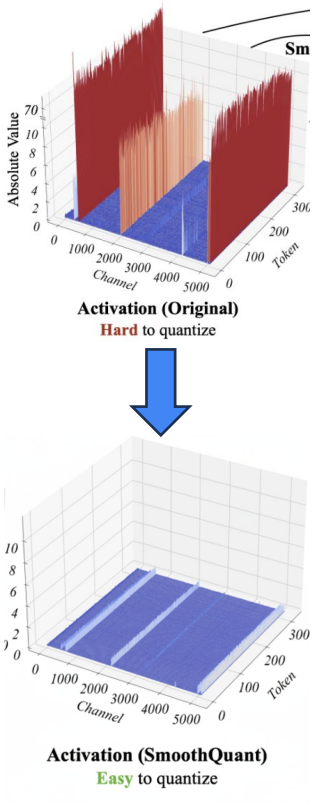
Where R is a rescaling, permutation, rotation matrix.

Rescaling: SmoothQuant, BASE-Q.

Rotations/Permutations: QuaRot, OSTAQuant, DuQuant.

Learned Rotations: SpinQuant, FlatQuant.

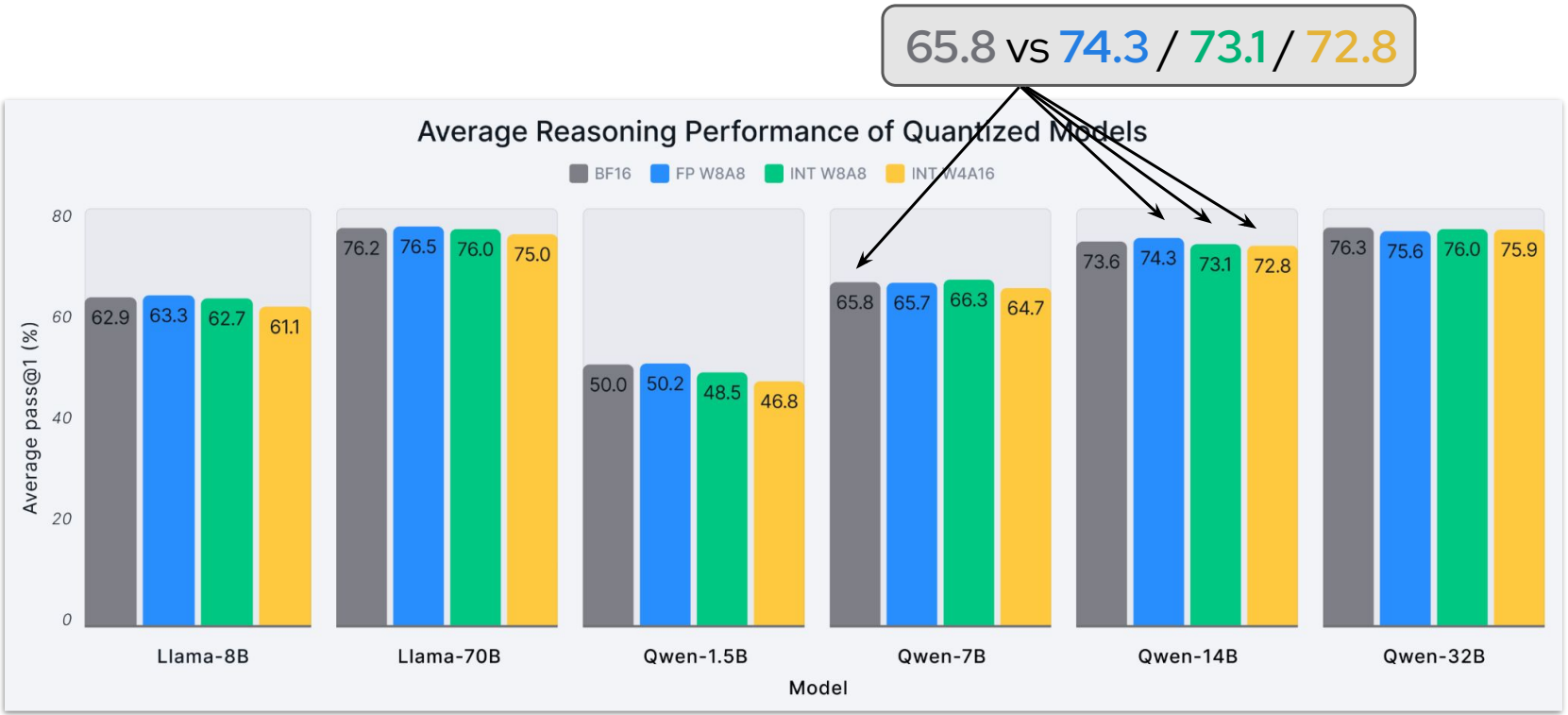
Outlier Preservation: Atom, QUIK.



Adapted from vLLM Europe Meetup 2025

Larger quantized is better than smaller unquantized.

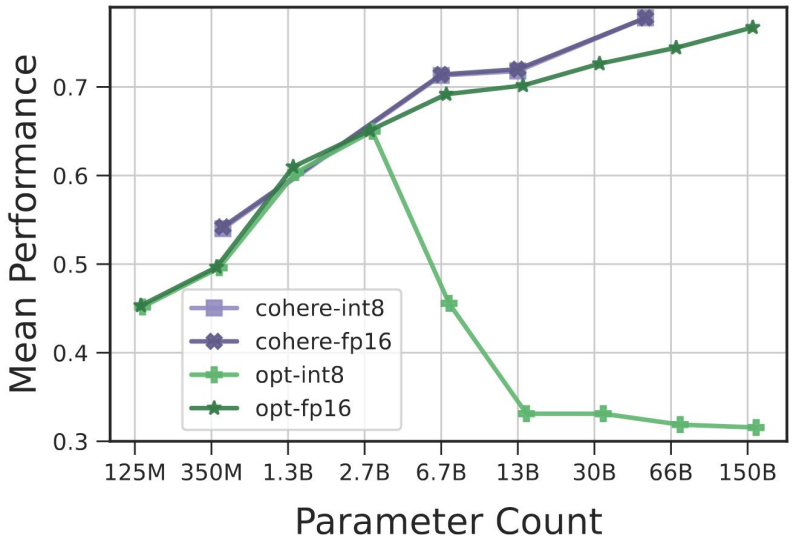
Reasoning performance = average pass@1 score on AIME24, MATH-500, GPQA-Diamond.



Adapted from vLLM Europe Meetup 2025

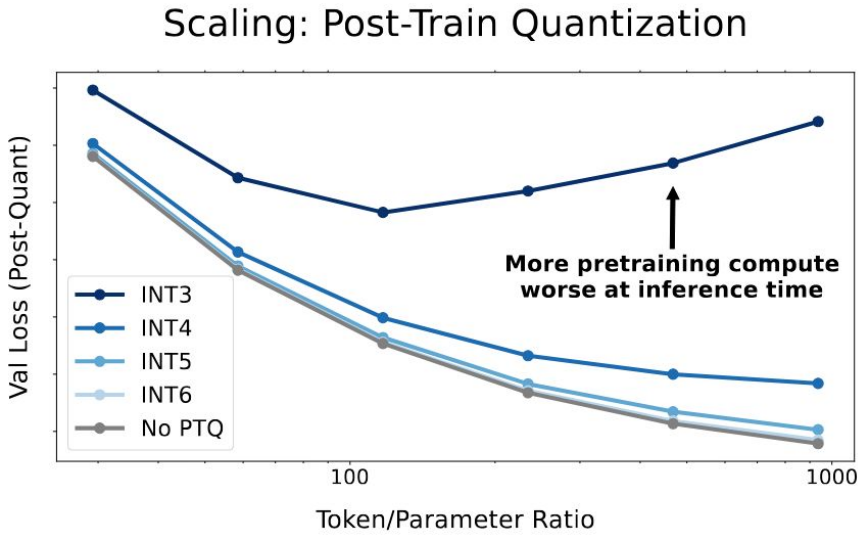
RQ: What makes a model robust to quantization?

Is it model size?



Intriguing Properties of Quantization at Scale, Ahmadian et al., 2023

Is it the token budget?



Scaling Laws for Precision, Kumar et al., 2024

Quantization Error in the Wild - Settings

1. We define quantization error as the loss degradation induced by quantization.

$$\Delta_Q \mathcal{L} = \frac{\mathcal{L}(\hat{\theta})}{\mathcal{L}(\theta)} - 1$$

2. Use GPTQ to quantize checkpoints to 3- and 4- bits along the training trajectories of open-source models {SmolLM3, OLMo2, Pythia, Apertus…}

3. What happens?

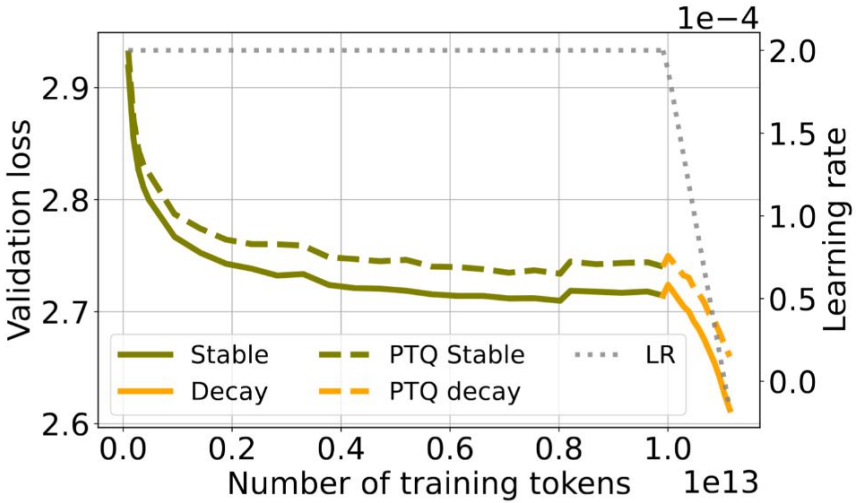
GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers, Frantar et al., 2023

Quantization Error in the Wild - SmolLM3

We define quantization error as the performance loss induced by quantization.

$$\Delta_Q \mathcal{L} = \frac{\mathcal{L}(\hat{\theta})}{\mathcal{L}(\theta)} - 1$$

(a) 4-bit quantization error vs training tokens.



(b) Validation loss vs training tokens.

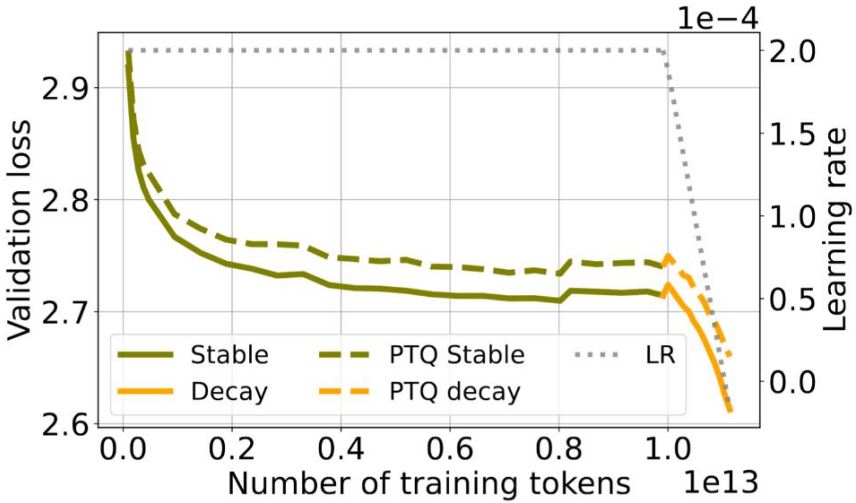
SmolLM3: smol, multilingual, long-context reasoner. Bakouch et al., 2025

Quantization Error in the Wild - SmolLM3

As learning rates decay, validation loss and quantization error diverge.

How much does pretraining influence quantization robustness?

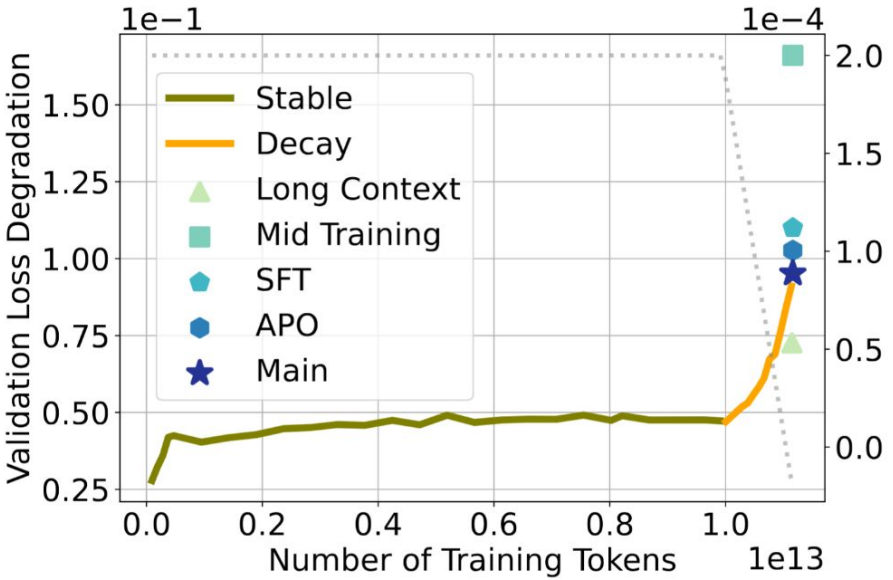
(a) 4-bit quantization error vs training tokens.



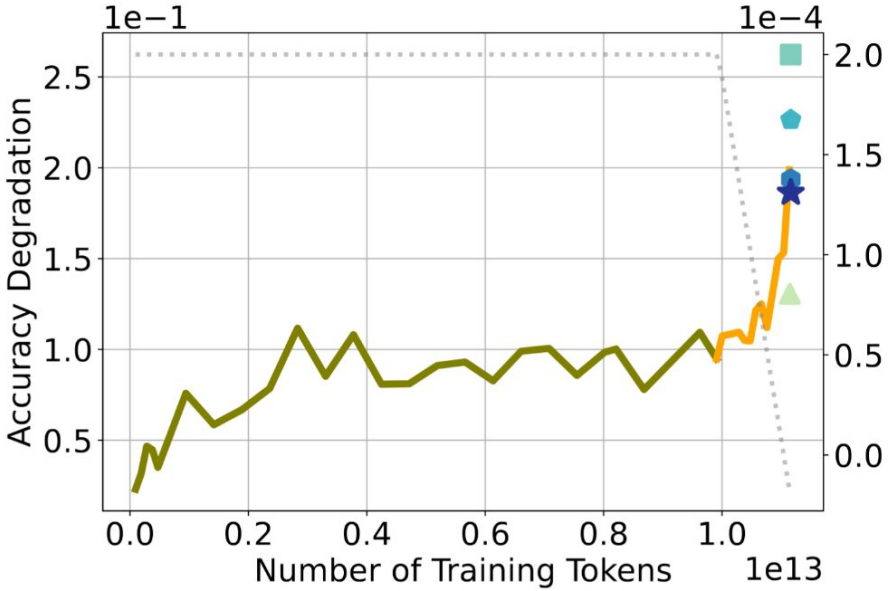
(b) Validation loss vs training tokens.

SmolLM3: smol, multilingual, long-context reasoner. Bakouch et al., 2025

Quantization Error in the Wild - Loss vs Task accuracy



(a) 3-bit validation loss degradation.



(b) 3-bit accuracy degradation.

Task accuracy is the mean of ARC-C, ARC-E, OBQA, PIQA, HSwag, WinoG, MathQA, PubMedQA, SciQ, SIQA, CSQA, MMLU

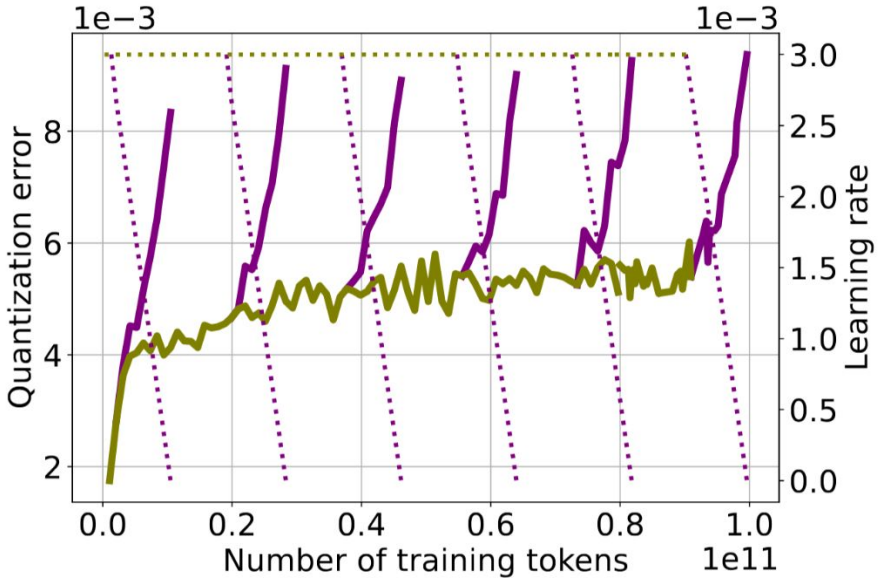
Controlled experiments

Train 160M parameter transformer on up to 100B tokens. Isolating,

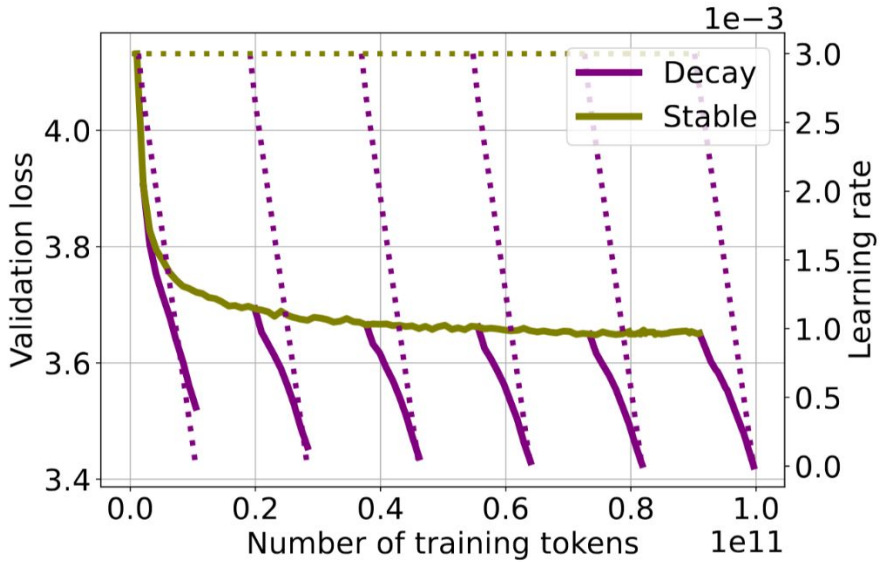
- **Token budget**
- **Learning rate magnitude**
- Learning rates shape
- Weight decay
- **Weight averaging**

Token budget

LR: 3e-3 | Schedule: WSD | Token budget: {16, 33, 49, 65, 82, 100} e9



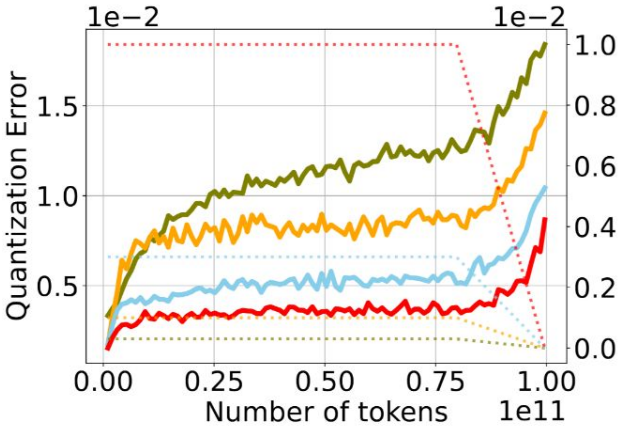
(a) 4-bit quantization error vs training tokens.



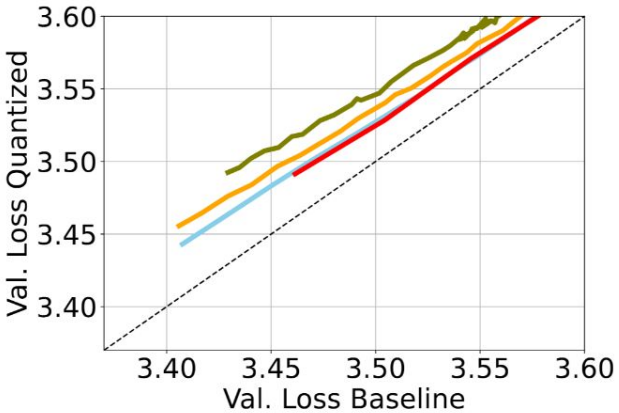
(b) Validation loss vs training tokens.

Learning rate magnitude

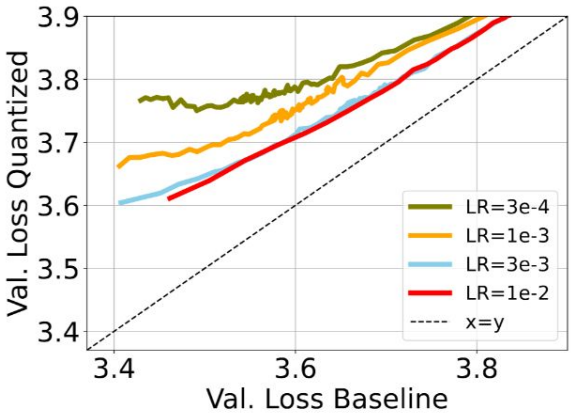
LR: {3e-4, 1e-3, 3e-3, 1e-2} | Schedule: WSD | Token budget: 100B



(a) 4-bit quantization error.



(b) FP to 4-bit validation loss.

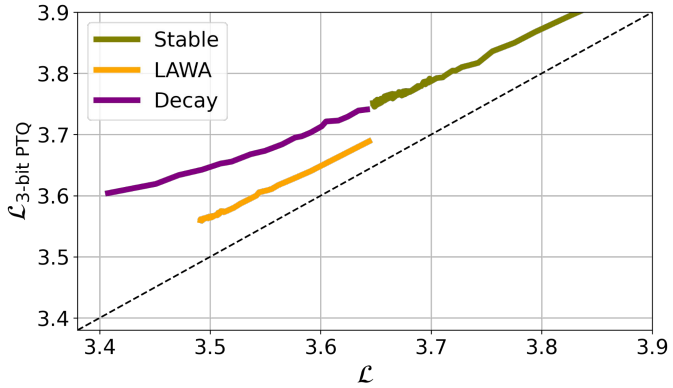


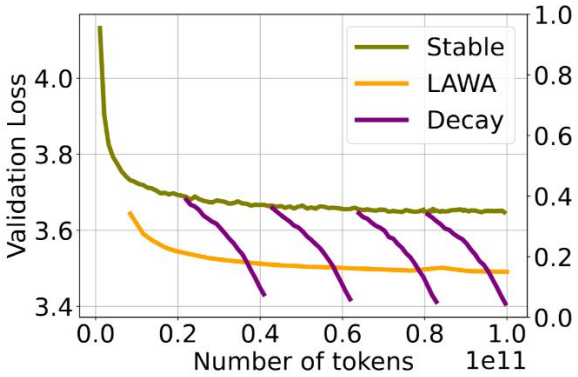
(c) FP to 3-bit validation loss.

Weight Averaging

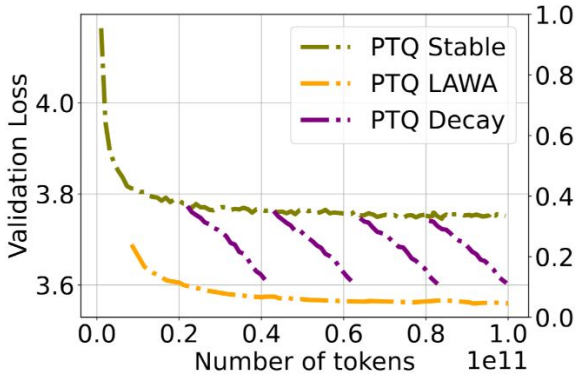
Aggregates checkpoints along a single trajectory.

LR: 3e-3 | Schedule: WS | Token budget: 100B

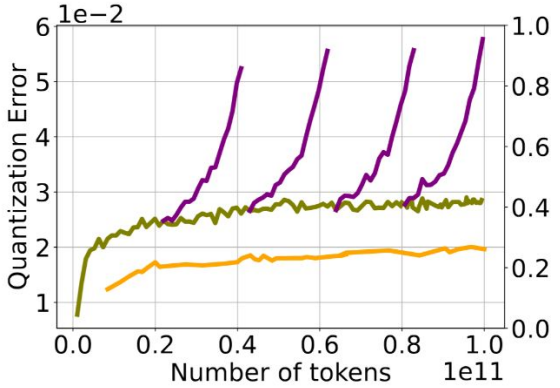




(a) FP validation loss.



(b) 3-bit validation loss.

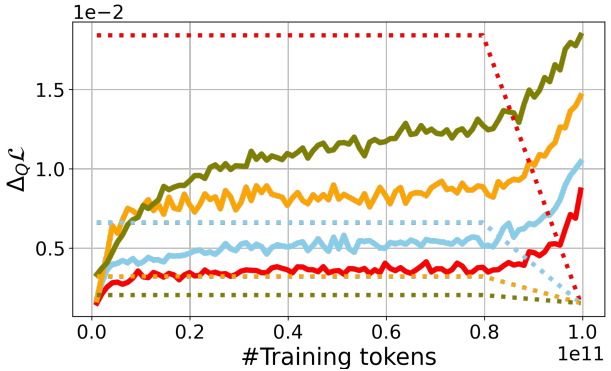


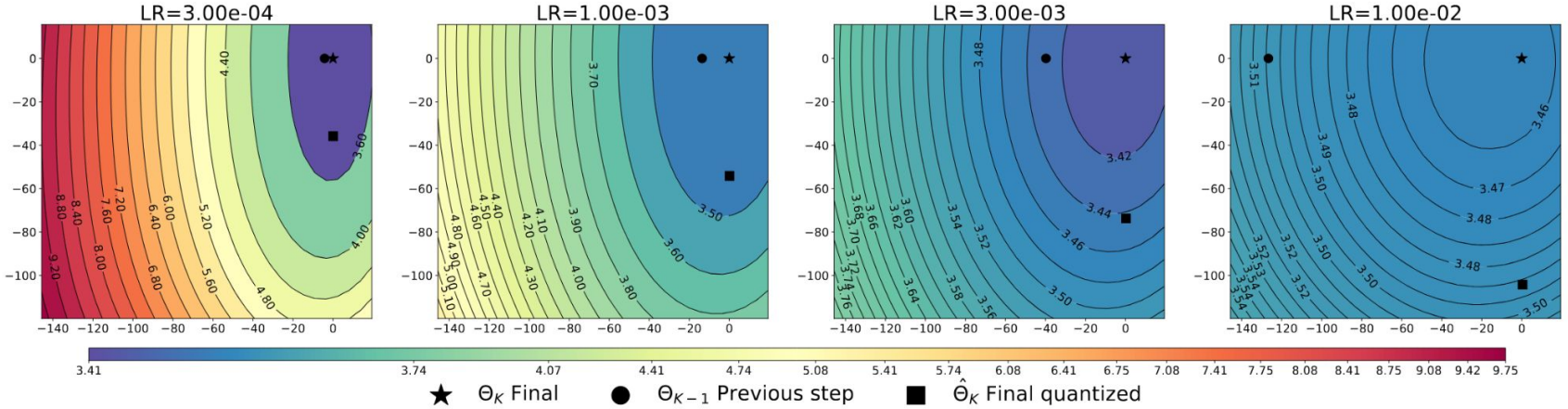
(c) 3-bit quantization error.

Averaging weights leads to wider optima and better generalization, Izmailov et al., 2018

Landscape of the loss

For the 4 learning rate experiments on 100B tokens.

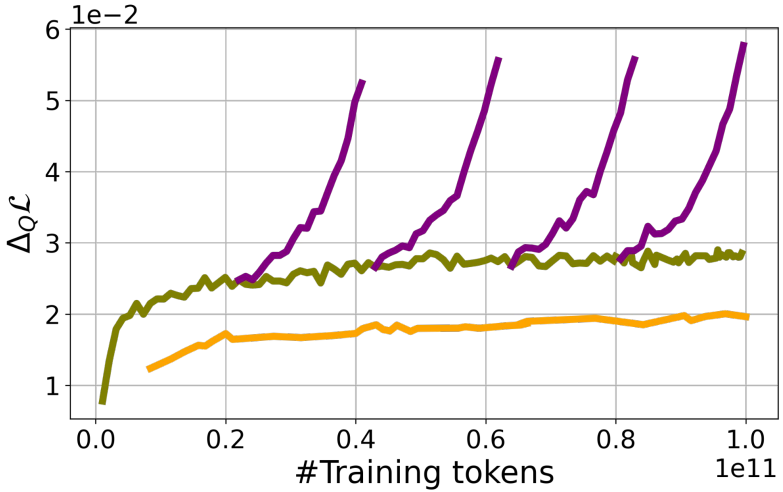


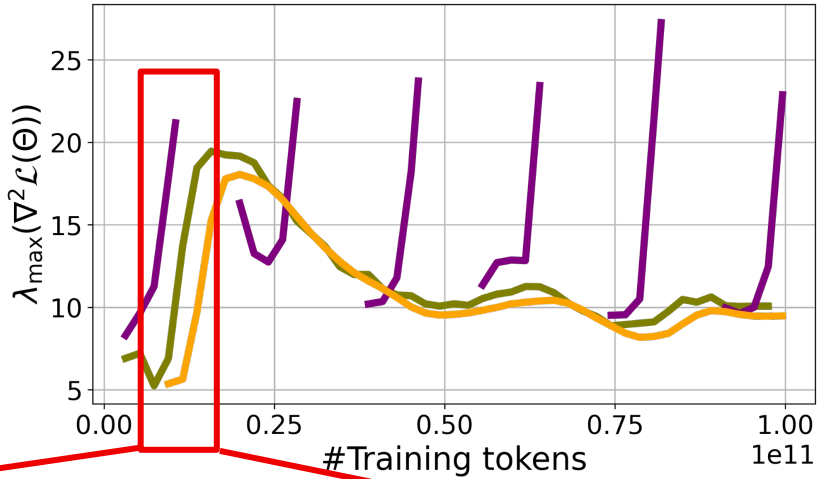


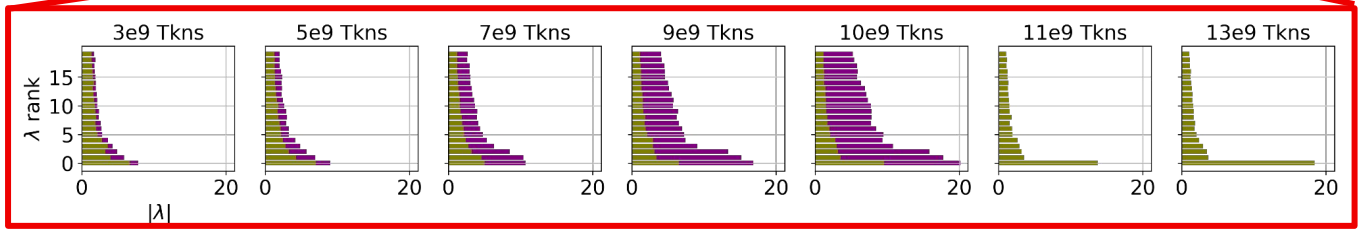
Geometry of the Loss perspective

Quantization error vs Top eigenvalue of the Hessian.

Geometry of the Loss perspective







Real world case: OLMo2-7B

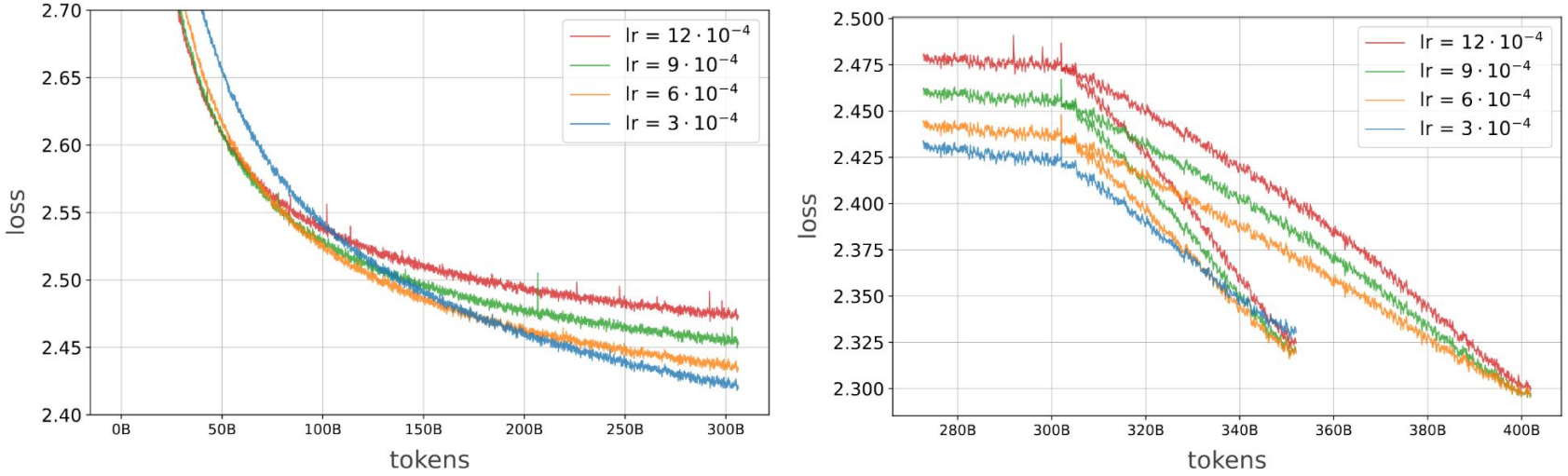


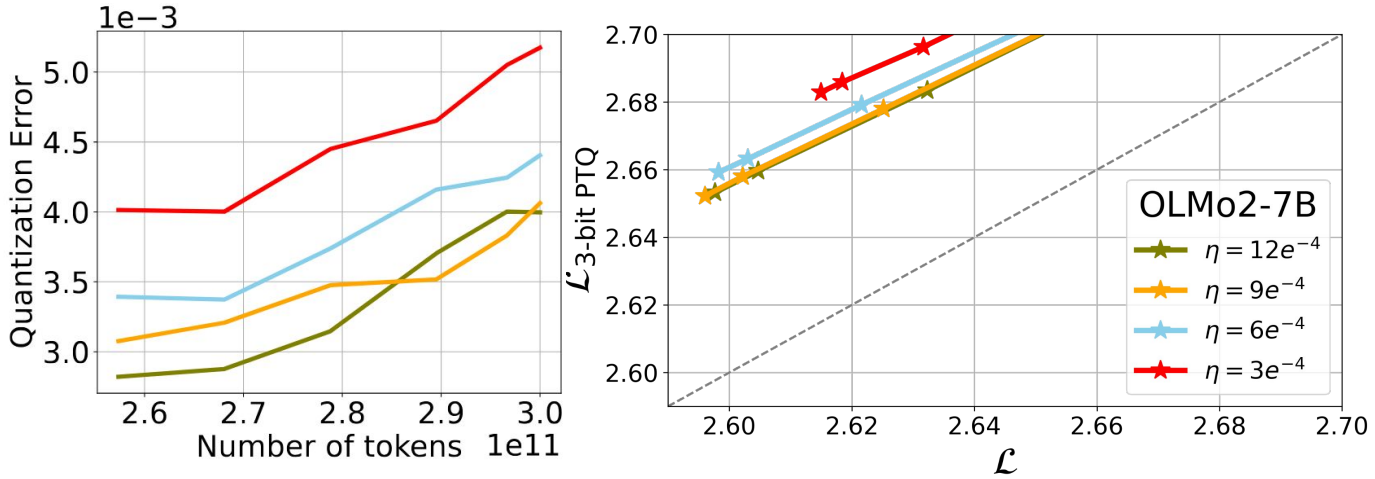
Figure 11 Higher learning rates perform better at first but are eventually overtaken by lower rates. However, linearly decaying the learning rate to zero over 50B or 100B tokens results in equivalent training loss.

2 OLMo 2 Furious, Team OLMo et al., 2025

Real world case: OLMo2-7B

300B token LR ablation of OLMo2-7B.

Same phenomenon happens!



Summary

Quantization is a **weight perturbation that benefits from general model robustness**:

1. Large learning rates.
2. Weight averaging (LAWA & Soups).
3. Weight decay.

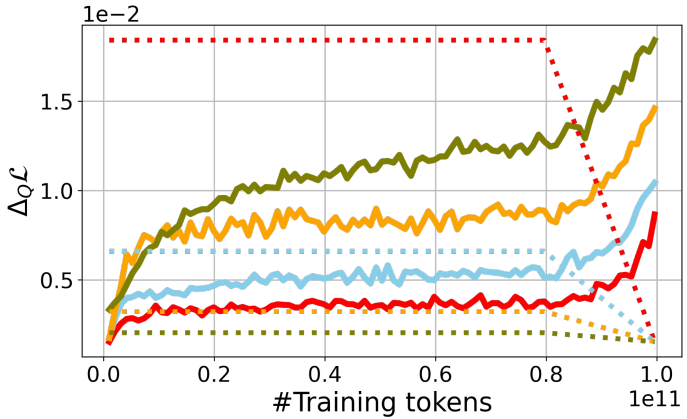
Quantization performance can be optimized for in routine hyperparameter exploration.

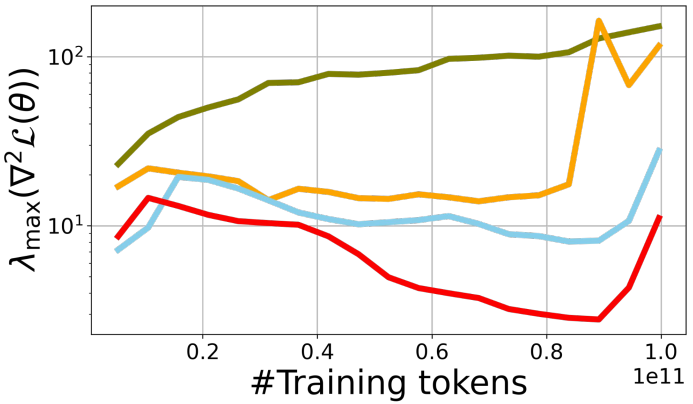
Any questions?

More results and insights at https://arxiv.org/pdf/2510.06213

Sharpness of the loss

Top eigenvalue of the Hessian of the loss via power iterations.





Other models - Unpredictable behaviour

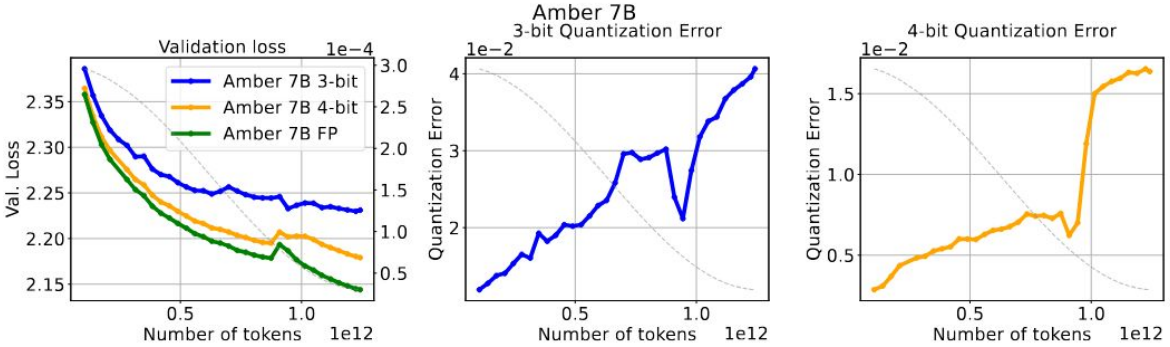


Figure 12: Quantization degradation for Amber-7B. 3 and 4-bit quantization with GPTQ.

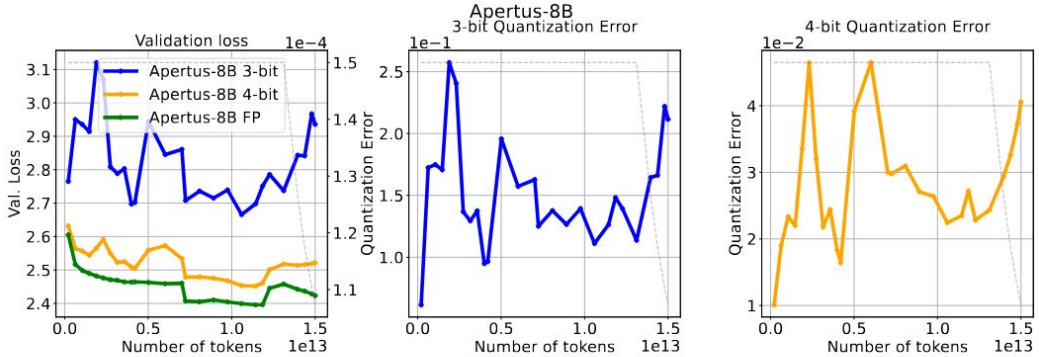


Figure 13: Quantization degradation for Apertus-8B. 3 and 4-bit quantization with GPTQ.

OLMo2 family results

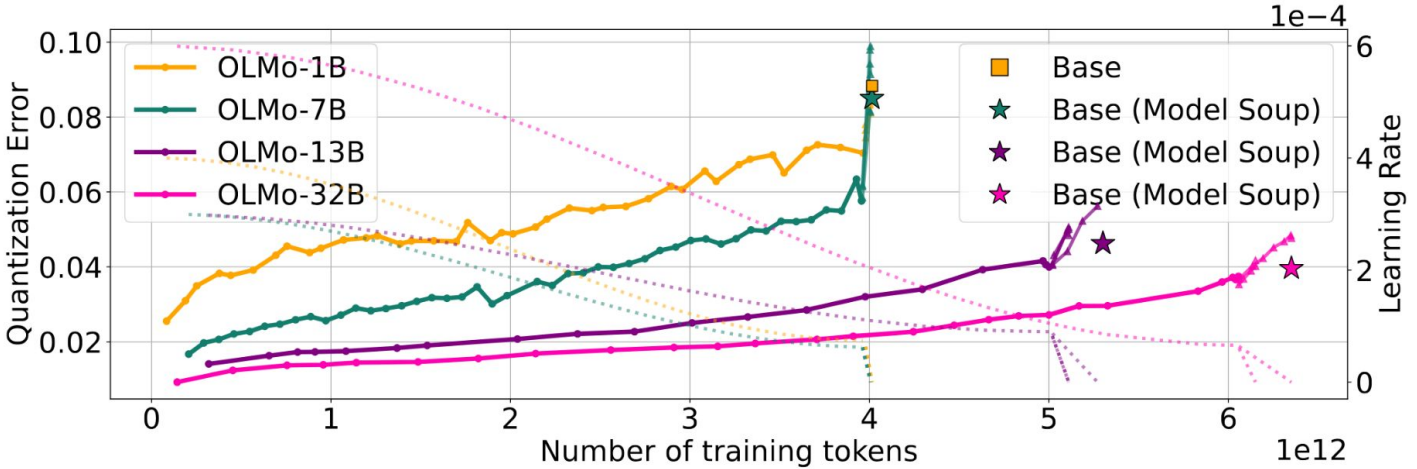
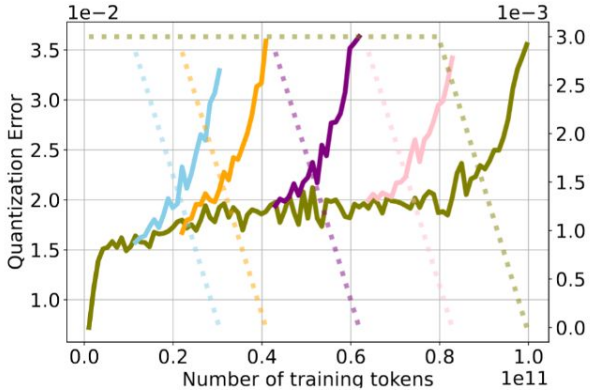
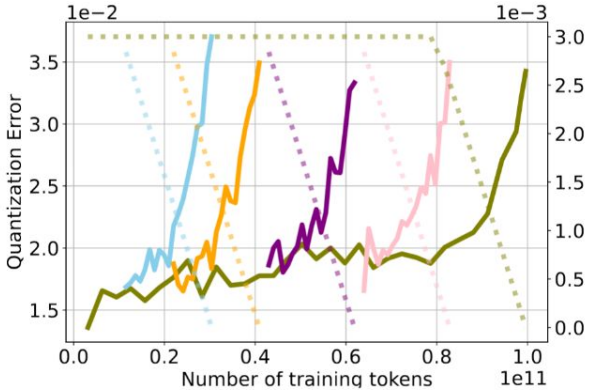


Figure 2: 3-bit quantization error along the training trajectories of OLMo2 models. Error grows gradually during cosine decay but spikes under the steep linear decay phase. Model souping ($\star$) reduces degradation, achieving lower PTQ error than any individual run.

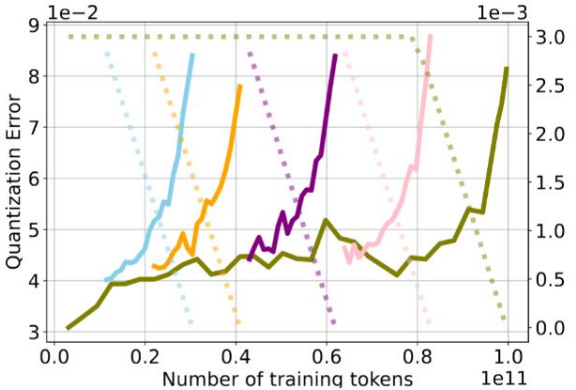
Similar results for other quantization methods



(a) GPTQ



(b) AWQ



(c) LLM.int8