Low-Bit Quantization Favors Undertrained LLMs

Xu Ouyang[1,2*] **Tao Ge**[2†] **Thomas Hartvigsen**[1] **Zhisong Zhang**[2] **Haitao Mi**[2] **Dong Yu**[2]
[1]University of Virginia [2]Tencent AI Lab Seattle
ftp8nr@virginia.edu sggetao@gmail.com

Abstract

Low-bit quantization improves machine learning model efficiency but surprisingly favors undertrained large language models (LLMs). Larger models or those trained on fewer tokens exhibit less quantization-induced degradation (QiD), while smaller, well-trained models face significant performance losses. To gain deeper insights into this trend, we study over 1500+ quantized LLM checkpoints of various sizes and at different training levels (undertrained or fully trained) in a controlled setting, deriving scaling laws for understanding the relationship between QiD and factors: the number of training tokens, model size and bit width.

With our derived scaling laws, we propose a novel perspective that we can use QiD to measure an LLM's training levels and determine the number of training tokens required for fully training LLMs of various sizes. Moreover, we use the scaling laws to predict the quantization performance of different-sized LLMs trained with **100 trillion** tokens. Our projection shows that the low-bit quantization performance of future models, which are expected to be trained with over 100 trillion tokens, may NOT be desirable. This poses a potential challenge for low-bit quantization in the future and highlights the need for awareness of a model's training level when evaluating low-bit quantization research. To facilitate future research on this problem, we release all the 1500+ quantized checkpoints used in this work at https://huggingface.co/Xu-Ouyang.

1 Introduction

Quantization (Jacob et al., 2018; Krishnamoorthi, 2018; Banner et al., 2019; Frantar et al., 2022; Shen et al., 2024; Lin et al., 2024; Zhang et al., 2024) is one of the most popular techniques for efficiently deploying large language models (LLMs) by reducing the model's disk size, memory footprint, and improving inference efficiency through lower precision weights and activations. As model sizes have continued to grow over the past years, researchers have moved beyond conventional 8-bit quantization (Zafrir et al., 2019; Dettmers et al., 2022; Zhong et al., 2024) and begun exploring even lower bit width (Bai et al., 2020; Zhang et al., 2020; Wang et al., 2023; Liu et al., 2023; Egiazarian et al., 2024; Liu et al., 2024; Huang et al., 2024), sparking a surge of research interest in low-bit quantization.

While low-bit quantization works well on some LLM checkpoints with very little quantization-induced degradation (QiD), we have observed that these checkpoints typically with either larger model sizes or fewer training tokens. In contrast, smaller models or those trained with substantially more tokens exhibit notable QiD when low-bit quantization is applied. As shown in Figure 1(right), 3-bit quantization results in negligible QiD for a 12 billion parameter LLM up to 10^{11} training tokens, but beyond this point, QiD begins to become pronounced; For smaller models (e.g., 160M and 1B parameters), QiD degradation occurs much earlier and is more severe. With even more extreme 2-bit quantization as shown in Figure 1(left), the trend is similar, but QiD worsens sooner and more significantly. This observation suggests that low-bit quantization tends to favor undertrained LLMs and is less compatible with fully trained LLMs.

To gain deeper insights into this trend, we study over 1500 quantized LLM checkpoints of various sizes (ranging from 160M to 12B) and at different training levels[1] (trained with from 1B to 206B training tokens), analyzing the impact of low-bit quantization on them in a controlled setting. We de-

[1]Training levels in this work refer to the extent to which an LLM has been trained (e.g., undertrained, fully trained, or overtrained), which are related to both the number of training tokens and the model size.

[*]Work done while interning at Tencent AI Lab Seattle.
[†]Corresponding author

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 32338–32348
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

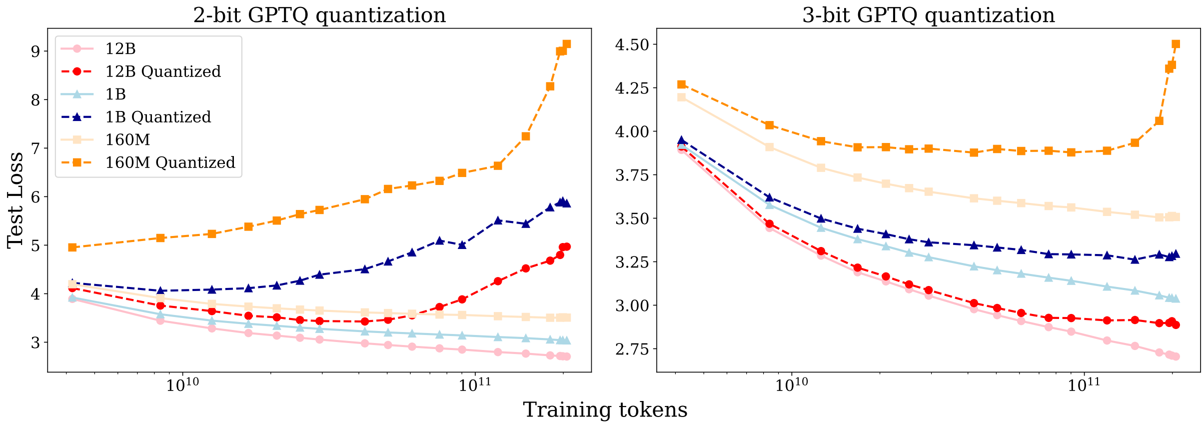

Figure 1: Performance of LLMs after low-bit quantization at different sizes and training levels. It is obvious that the models which are smaller or trained with more tokens suffer from greater quantization-induced degradation.

rive scaling laws to model QiD with respect to the number of training tokens, model size, bit width. According to the derived scaling laws, we propose a novel perspective that we can use QiD to measure an LLM's training levels and determine the number of training tokens required for fully training an LLM given its size. Moreover, we use the scaling laws to predict the performance of different-sized LLMs with 100 trillion training tokens when applying low-bit quantization. Our projection shows that low-bit quantization of future models, which are expected to be trained with over 100 trillion tokens, may not be desirable, which indicates a potential challenge for low-bit quantization in the future and suggests that a model's training level should be considered in the evaluation of future low-bit quantization research.

The contributions of this work are threefold:

- We reveal that low-bit quantization favors undertrained LLMs but suffers from significant quantization-induced degradation (QiD) when applied to fully trained LLMs. This insight has been largely overlooked in previous low-bit quantization research: very few studies have considered the training level of a quantized LLM when evaluating their proposed low-bit quantization approaches.
- We derive scaling laws to model QiD with the number of training tokens, model size and bit width. Using these scaling laws, we propose to use QiD as a signal to measure whether an LLM is fully trained and estimate the number of training tokens required for LLMs of different sizes to reach a fully trained state. Moreover, we use the scaling law to predict the performance of low-bit quantization for different-sized LLMs trained with 100 trillion tokens. Our projection indicates potential challenges for the future application of low-bit quantization.
- We release all the 1500+ quantized checkpoints used in this work to facilitate future research.

2 Preliminary: Scaling Laws for Large Language Models

Scaling laws for large language models (Kaplan et al., 2020; Hoffmann et al., 2022) are crucial for understanding how these models' performance improves with increased scale, including the number of parameters and training tokens:

Number of Parameters LLMs' performance typically follows a power-law improvement as the number of parameters increases, allowing larger models to better fit on the same dataset:

$$L(N) = \frac{a}{N^\alpha} + \epsilon \quad (1)$$

where $L(N)$ is the loss function[2] dependent on N (the number of non-embedding parameters), a is a constant (i.e., coefficient), α is the scaling exponent, and ϵ represents the error term. This relationship indicates larger models are generally more capable of capturing the complexities of language, leading to better generalization and lower loss.

Training Tokens Increasing the number of training tokens enhances performance in a power-law manner, enabling better capture of languages.

$$L(D) = \frac{b}{D^\beta} + \epsilon \quad (2)$$

where D denotes the number of training tokens, b is a constant (i.e., coefficient) and β is the scaling

[2]We mainly discuss cross entropy loss for language modeling in this paper.

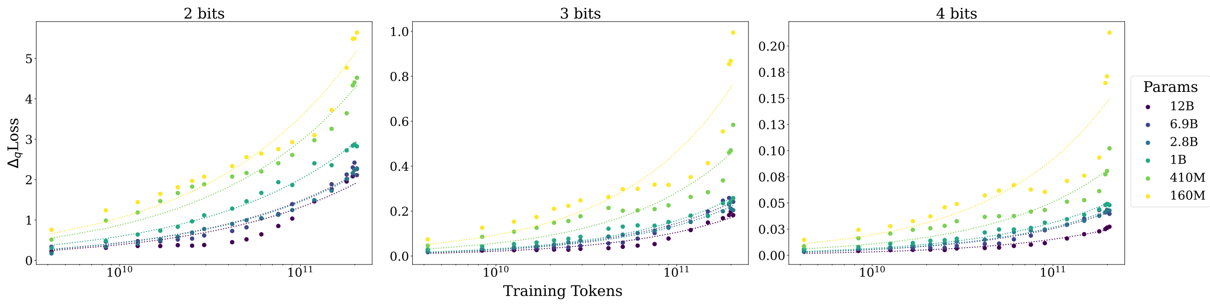

Figure 2: The fitted scaling law of QiD with respect to the number of training tokens in the form of (Equation 5), where β is fitted to be 0.5316.

exponent for training tokens. More training tokens enhance an LLM's ability to learn and generalize, allowing it to achieve better language modeling performance with lower loss.

When scaling both the number of parameters N and the amount of training data D simultaneously, the scaling law can be expressed as a function that accounts for the combined effects of both:

$$L(N, D) = [(\frac{N_c}{N})^{\frac{\alpha_N}{\alpha_D}} + \frac{D_c}{D}]^{\alpha_D} \tag{3}$$

This scaling law allows us to estimate the performance of language models at unprecedented scales of model size and training data effectively before conducting actual training runs.

3 Scaling Laws for Low-bit Quantization

In this section, we propose scaling laws for low-bit quantization. Unlike the scaling laws discussed in section 2, the focus here is to understand how QiD changes when low-bit quantization is applied to LLMs of varying training scales. Formally, QiD is defined as follows:

$$\Delta_q Loss = Loss_q - Loss_{\text{16-bit}} \tag{4}$$

where $Loss_q$ is the cross-entropy loss of a quantized LLM, and $Loss_{\text{16-bit}}$ is the cross-entropy loss of its pre-quantized, fp16 or bf16 counterparts. $\Delta_q Loss$, denoting QiD, represents the loss difference before and after applying low-bit quantization.

Inspired by conventional scaling laws for language modeling, we investigate the impact of model size and the number of training tokens on QiD. Additionally, we consider bit width (i.e., the precision of quantized weight values).

3.1 Experimental Setting

We utilize open-sourced LLMs from the Pythia suite (Biderman et al., 2023) in our experiments. The Pythia suite offers diverse model sizes and provides complete access to checkpoints across its training trajectory (spanning from initialization to 300 billion tokens). This comprehensive access enables controlled experimentation and facilitates the derivation of scaling laws for low-bit quantization.

We choose 6 different sizes of Pythia LLMs: 160M, 410M, 1B, 2.8B, 6.9B, and 12B. For each size, we sample 20 checkpoints (see Appendix A.1) up to 98k steps.[3]

For quantization, we employ one of the most popular LLM quantization techniques – GPTQ (Frantar et al., 2022) – to quantize the Pythia checkpoints to 2-bit, 3-bit and 4-bit levels.

We evaluate QiD on 1,000 randomly sampled texts from RefinedWeb (Penedo et al., 2023).

3.2 Training Tokens

In contrast to prior language modeling scaling laws where the number of training tokens D appears in the denominator, we propose the relationship between training tokens and QiD as follows:

$$\Delta_q Loss(D) \approx b \cdot D^{\beta} \tag{5}$$

As observed in Figure 1, QiD becomes increasingly pronounced with a greater number of training tokens, emphasizing its growing significance.

We use the above functional form to fit the QiD observed in the quantized Pythia checkpoints in Figure 2, obtaining $\beta = 0.5316$, which fits the trend of QiD with respect to the change in training tokens quite well.

3.3 Model Size

As mentioned in Figure 1, the larger the size of the model, the smaller the QiD tends to be. Therefore, we propose the relationship between model size

[3]98k steps correspond to approximately 206 billion tokens, which is equivalent to one epoch of Pythia's training data. Although Pythia was trained for 143k steps, we skipped checkpoints beyond 98k steps to avoid the influence of duplicated data, as the data beyond 98k steps represents the second epoch.

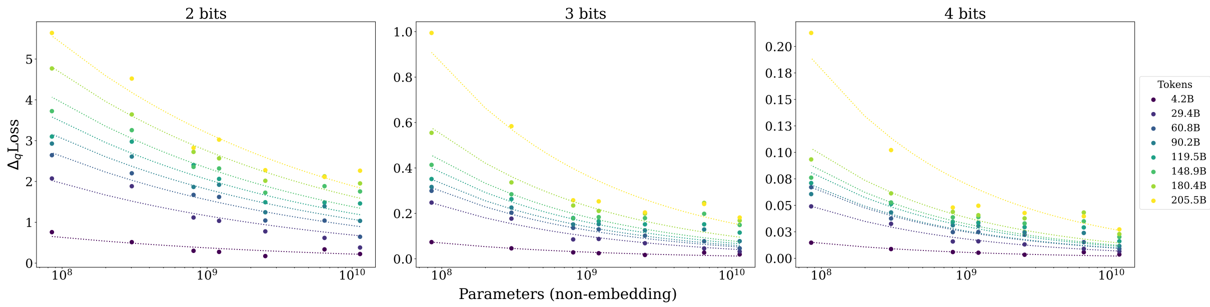

Figure 3: The fitted scaling law of QiD with respect to the model size (i.e., the number of non-embedding parameters) in the form of Equation 6, where α is fitted to be 0.2276.

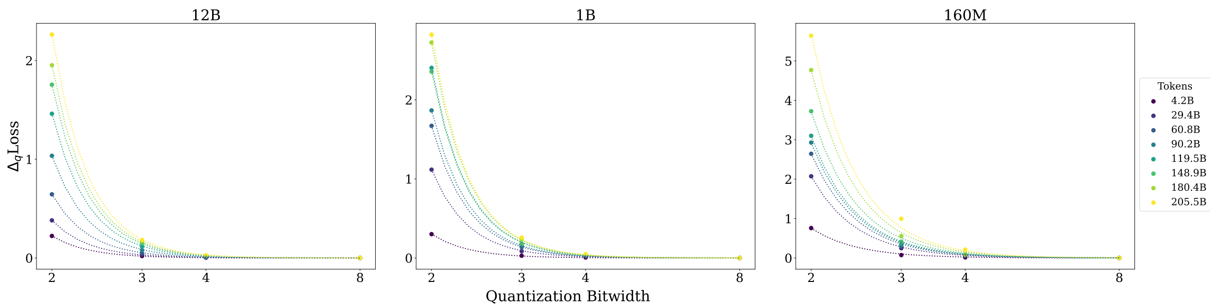

Figure 4: The fitted scaling law of QiD with respect to the bit width in the form of Equation 7, where γ is fitted to be 5.4812.

(i.e., the number of non-embedding parameters) and QiD as follows:

$$\Delta_q Loss(N) \approx \frac{a}{N^\alpha} \tag{6}$$

Using the described functional form, we fit the QiD of quantized Pythia checkpoints shown in Figure 3, yielding $\alpha = 0.2276$.

3.4 Bit Width

Bit width is a factor not present in conventional scaling laws. Considering that the role of bit width is similar to that of the number of parameters (both aim to increase the model's expressiveness), we propose a similar functional form as in subsection 3.3 to model bit width in Equation 7:

$$\Delta_q Loss(P) \approx \frac{c}{P^\gamma} \tag{7}$$

We fit the QiD of quantized Pythia checkpoints shown in Figure 4, yielding $\gamma = 5.4812$.

3.5 Unified Scaling Law

With the basic scaling laws derived in Sections 3.2 (the number of training tokens), 3.3 (model size), and 3.4 (bit width), we study how to model QiD with all three factors together. Inspired by Kaplan et al. (2020), we consider the following four principles for unifying the factors:

- Fixing D and P, sending $N \to \infty$, we expect $\Delta_q Loss \to 0$.
- Fixing N and P, sending $D \to 0$, we expect $\Delta_q Loss \to 0$.
- Fixing N and D, sending $P \geq 16$, we expect $\Delta_q Loss \to 0$.
- Fixing N and D, sending $P \to 0$, we expect $\Delta_q Loss \to \infty$.

We propose the unified scaling law for low-bit quantization as follows:

$$\Delta_q Loss(N, D, P) = k \cdot \frac{D^\beta}{N^\alpha P^\gamma} \tag{8}$$

where k is the joint coefficient, and both the coefficient and exponents (α, β, γ) are positive. Figure 5 displays the fitted curves using this functional form. The jointly fitted exponents α, β, and γ closely match those obtained by fitting these variables independently, further validating the effectiveness of the joint function form $\Delta_q Loss(N, D, P)$.

Given the unified scaling law for $\Delta_q Loss$ and the definition of $\Delta_q Loss$ in Equation 4, we can easily predict a quantized LLM's performance as $Loss_q = Loss_{\text{16-bit}} + \Delta_q Loss$, as illustrated in Figure 6, which fits well with the observations.

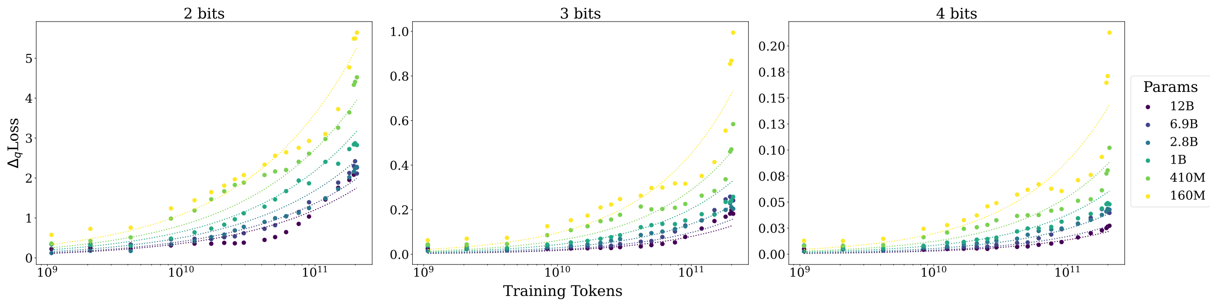

Figure 5: The unified scaling law we fit based on Equation 8 with the GPTQ-quantized LLMs from the Pythia suite: $\Delta_q Loss(N, D, P) = 0.017 D^{0.5251} / (N^{0.2261} \cdot P^{5.4967})$

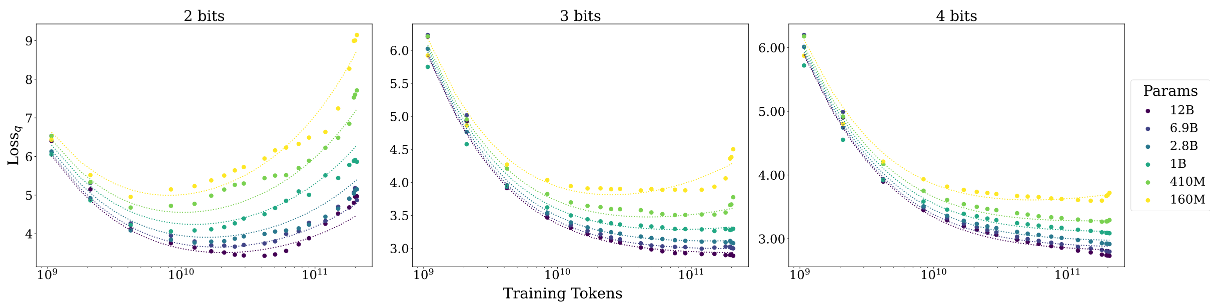

Figure 6: We can predict the performance of a quantized LLM as $Loss_q = Loss_{\text{16-bit}} + \Delta_q Loss$, where $Loss_{\text{16-bit}}$ can be predicted by the conventional LLM's scaling law which is fitted based on the function form of Equation 3 with the LLMs in the Pythia suite as $Loss_{\text{16-bit}} = [(4.74e^{19}/N)^{(0.045/0.399)} + 7.63e^{10}/D]^{0.399}$.

3.6 Validation with Ablation Studies

We validate the scaling law derived in subsection 3.5 with different test data, quantization methods and foundation models.

3.6.1 Test Data

We compare the results obtained using RefinedWeb and Wikitext-2 (Merity et al., 2016) as test data in Figure 7, demonstrating that the QiD results on these two test datasets are almost identical. This suggests that the trends of QiD are largely independent of the test data.

3.6.2 Quantization Methods

We quantize Pythia checkpoints using two other popular quantization methods – AWQ (Lin et al., 2024) and bitandbytes[4] in addition to GPTQ. We show the QiD results and fitted scaling laws in Figure 8, and we find that QiD trends remain nearly identical across different quantization methods, despite slight variations in the fitted scaling laws.

3.6.3 Foundation Models

Figure 9 shows the fitting results of our scaling laws function form, Equation 8, on the Spectra suite (Kaushal et al., 2024) as well as the popular open-sourced Llama (Touvron et al., 2023; Dubey et al., 2024) and Qwen (Yang et al., 2024) models, which verifies our laws are not only valid for Pythia but are broadly applicable.

4 Discussion: Low-bit Quantization Favors Undertrained LLMs

4.1 Intuition

Based on the scaling laws we derived in section 3, we confirm low-bit quantization tends to favor models with fewer training tokens or larger model sizes, which are essentially undertrained LLMs.

Figure 10 demonstrates the relationship between QiD, model size, training token number, and bit width. Points located in the upper-left region are more fully trained, resulting in a substantially higher QiD, while those in the bottom-right are undertrained, exhibiting a lower QiD.

To understand this observation intuitively, we illustrate changes in sampled model weights between adjacent checkpoints in Figure 11. It can be observed that the early checkpoints exhibit significant changes in weights. Due to the significant fluctuations in weights during training, the model becomes inherently robust to weight variations, meaning that even if low-bit quantization introduces some precision loss, the overall impact

[4] https://github.com/bitsandbytes-foundation/bitsandbytes

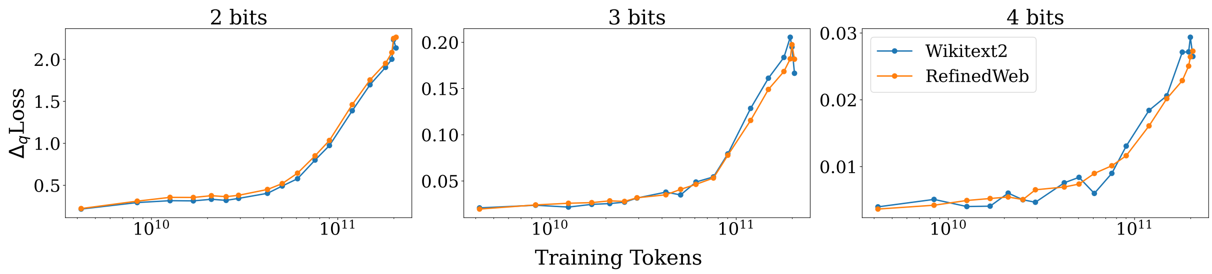

Figure 7: QiD results evaluated on RefinedWeb and Wikitext-2 with the 12B Pythia model.

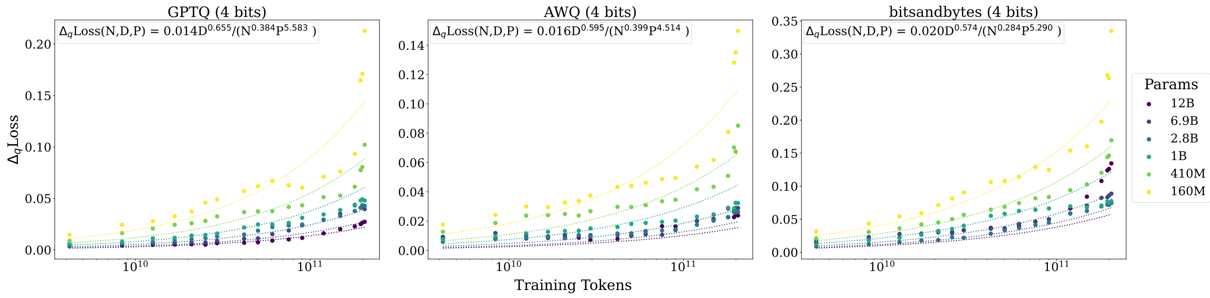

Figure 8: QiD results and fitted scaling laws for different quantization methods. Note that the GPTQ function here differs slightly from that in Figure 5, as it is fitted exclusively with 4-bit quantized Pythia checkpoints, whereas the function in Figure 5 is fitted using all quantized Pythia checkpoints.

on the model remains limited. On the other hand, checkpoints from the later stages of training, which are more fully trained, show very small changes in weights (often at a very small scale, even beyond the 3rd-4th decimal place). In such cases, low-bit quantization is very likely to shift weights outside the small range of recent variations, potentially causing the model to degrade or even collapse.

From another perspective, during the under-trained stage, the model's weights undergo significant changes and have not yet fully exploited the precision dimension. In the later, more fully trained stage, as weight adjustments stabilize, the model increasingly relies on precision to continue optimizing the training objective and improving language modeling performance. This aligns with the two phrases of representation learning in the information bottleneck theory (Shwartz-Ziv and Tishby, 2017): during the early training phase, gradients have a large mean and small variance, making high precision unnecessary. However, in the later training phase, gradients have a small mean and large variance, requiring higher precision for the model to converge effectively.

4.2 QiD: A Signal that Measures an LLM's Training Level

Unlike previous work that often uses the inability of the loss to decrease further as a signal to determine whether an LLM is fully trained (i.e., saturated), we introduce a novel perspective that we can use QiD to determine whether an LLM is fully trained. If an LLM exhibits QiD ≈ 0 after low-bit quantization, it suggests that the LLM is likely undertrained, as it has not yet exploited higher precision, as discussed in subsection 4.1.

With the scaling law in Equation 8 derived in subsection 3.5, we can estimate how many training tokens are needed for a given LLM size to be considered fully trained based on QiD predictions. Table 1 shows the number of training tokens required for different model sizes to achieve $\Delta_q Loss$ = $\{0.2, 0.3, 0.4, 0.5\}$ when applying low-bit quantization. For a 70B scale model, achieving a QiD greater than 0.2 (corresponding to likelihood decrease by 20%) under 4-bit quantization requires over 17 trillion training tokens. In contrast, for a 405B scale LLM, achieving a QiD above 0.2 under 4-bit quantization requires nearly 50 trillion training tokens – a scale far beyond what has been achieved by now, indicating that current training efforts for extremely large LLMs may be still far from sufficient.

4.3 QiD Prediction When Scaling to 100 Trillion Training Tokens

Figure 13 shows the trend in the number of training tokens for state-of-the-art 7B-scale LLMs from

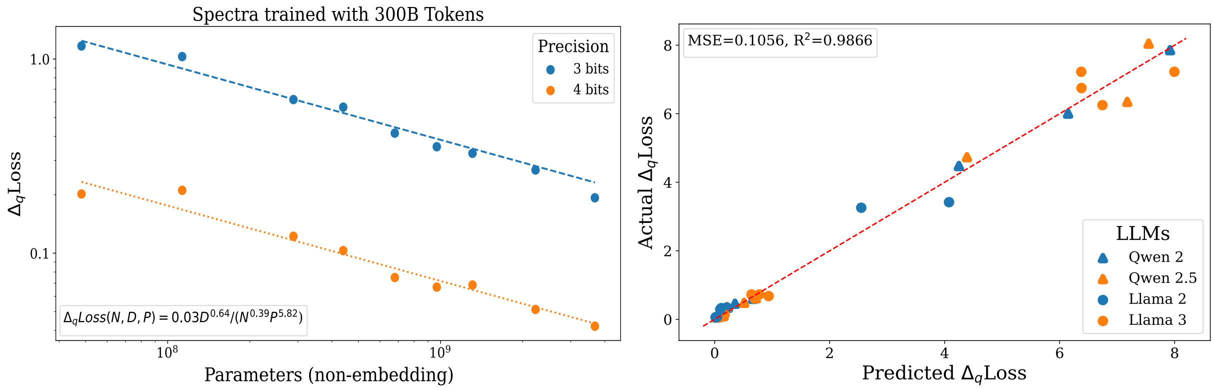

Figure 9: **Left:** Scaling laws for low-bit quantization, fitted on the LLM checkpoints of the Spectra suite, which are all trained with 300B tokens; **Right:** Actual $\Delta_q Loss$ **VS** Predicted $\Delta_q Loss$ that is computed based on the scaling laws fitted on Llama and Qwen.

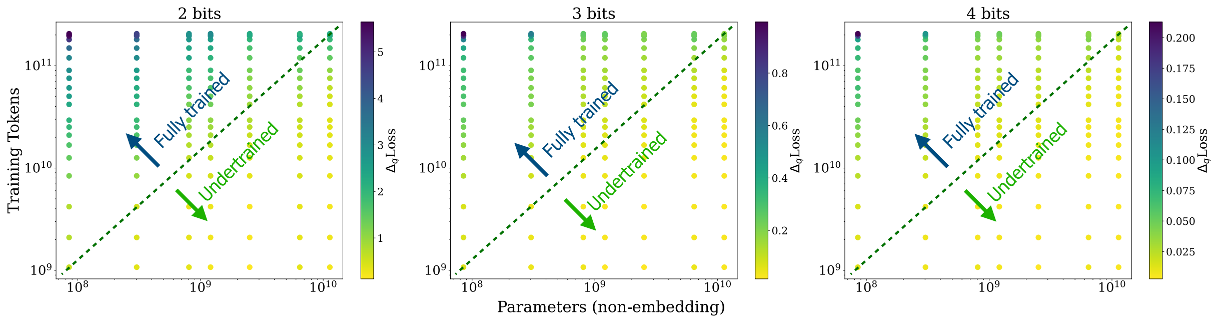

Figure 10: Fully trained LLMs suffer from much greater QiD (i.e., $\Delta_q Loss$) than undertrained LLMs.

2020 to the present, showing that the number of training tokens has increased nearly 50× over the past 4 years. Based on this trend, it is very likely that LLMs in 2025-2026 will be trained with up to 100 trillion (10^{14}) tokens[5].

Using the scaling laws derived, we predict the performance of quantized LLMs trained on 100 trillion tokens, as illustrated in Figure 12. In particular, performance degradation with 2-bit and 3-bit quantization at the unprecedented training scale of 100 trillion tokens is predicted to be severe, which is in stark contrast to the acceptable performance at the current training scale of 10^{13} tokens. This indicates a challenge for the practical application of low-bit quantization to future LLMs.

4.4 From Low-bit Quantization to Low-bit LLMs

Although this work mainly focuses on low-bit (post-)quantization, we suspect that native low-bit LLMs are also likely to favor undertrained LLMs. We replicated the popular 1-bit LLM – *BitNet b1.58* (Ma et al., 2024) – to compare it with its bf16 counterpart throughout training. Specifically, we trained 120M and 1.2B decoder-only models with both bf16 and BitNet. Figure 14 shows the comparison of training losses between BitNet and its 16-bit counterparts in the early- and mid-training steps. It can be observed that, in the early stages of training, the training loss curves of BitNet closely match (and even outperform) those of bf16, as BitNet tends to use a higher learning rate than bf16 training according to its training recipe. As training continues, the 120M BitNet gradually begins to lag behind its bf16 counterpart, and after further training steps, a noticeable gap starts to appear in the 1.2B models, which is consistent with our observations for low-bit quantization. This indicates that native low-bit LLMs such as BitNet[6] may also favor undertrained LLMs, though the gap manifests later compared to post-quantization, as native low-

[5]Although there have been claims that internet data is nearing exhaustion, recent continuous innovations in synthetic data creation (Ge et al., 2024) lead us to believe that the milestone of 100 trillion training tokens is achievable in the next few years.

[6]We reviewed the original BitNet paper and some open-sourced reimplementations, and found that their numbers of training tokens were up to 100 billion. Considering their model sizes and the fact that the performance gap of native low-bit LLMs tends to emerge later compared to post-quantization, we express concerns about their performance at larger training scales (i.e., with more training tokens). We call for results of native low-bit LLMs at larger training scales to better justify their practical value.

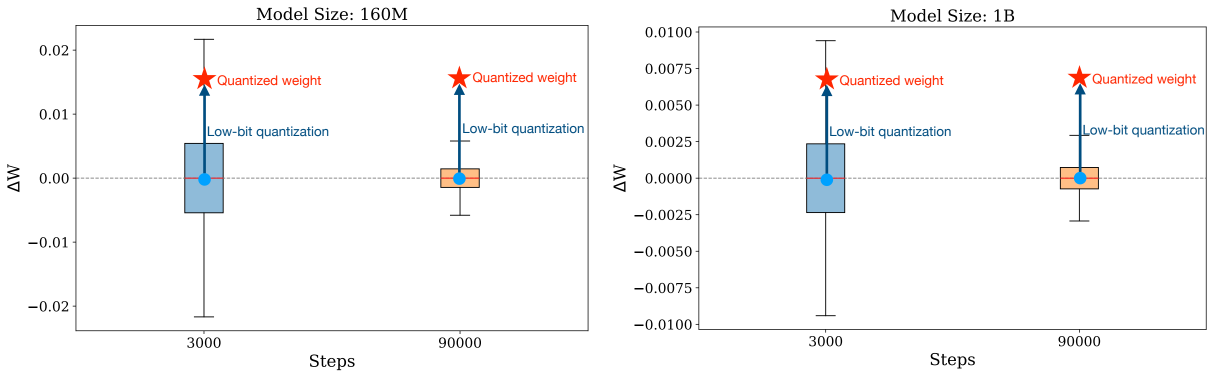

Figure 11: Changes in model weights between adjacent checkpoints. Early (undertrained) checkpoints exhibit significant weight fluctuations during training, making the model relatively robust to weight variations. Therefore, small changes introduced by quantization have a limited impact on the model's performance. In contrast, fully trained checkpoints demonstrate very little weight fluctuations during training. As a result, low-bit quantization is likely to push weights **beyond the narrow range of recent variations**, leading to performance degradation or even model collapse.

Model Size	Δ_qLoss = 0.2			Δ_qLoss = 0.3			Δ_qLoss = 0.4			Δ_qLoss = 0.5		
	2 bits	3 bits	4 bits	2 bits	3 bits	4 bits	2 bits	3 bits	4 bits	2 bits	3 bits	4 bits
1B	0.001	0.109	1.442	0.003	0.199	2.679	0.004	0.305	4.156	0.007	0.425	5.842
7B	0.003	0.304	4.507	0.006	0.555	8.369	0.010	0.851	12.984	0.015	1.186	18.253
70B	0.007	1.023	**17.350**	0.015	1.869	32.219	0.027	2.866	49.985	0.041	3.993	70.272
405B	0.015	2.581	**48.486**	0.033	4.715	90.040	0.057	7.231	139.689	0.087	10.075	196.383

Table 1: Prediction of the number of training tokens (in trillion) needed to achieve a given training level measured by $\Delta_q Loss$ for different model sizes and bit widths. Note that $\Delta_q Loss = 0.2$ means the likelihood is reduced to 80% of its original value ($e^{-0.2} \approx 0.8$), while $\Delta_q Loss = 0.5$ means the likelihood is reduced to 60% ($e^{-0.5} \approx 0.6$).

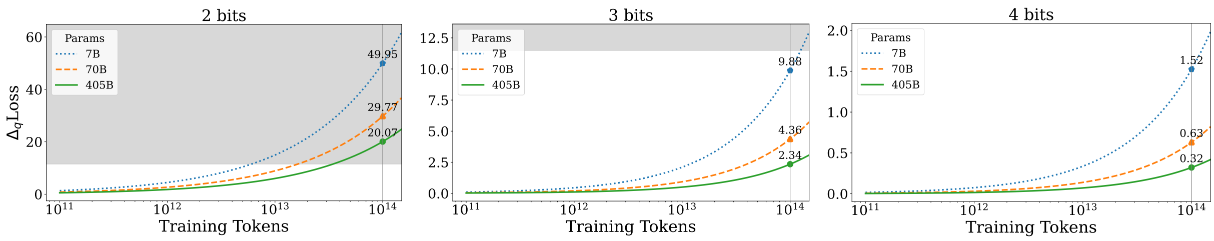

Figure 12: Scaling laws for predicting **Q**uantization-**i**nduced **D**egradation (QiD, denoted as $\Delta_q Loss$) in 7B, 70B, and 405B models trained on up to 100 trillion (10^{14}) tokens. While low-bit quantization yields acceptable QiD for undertrained LLMs (trained with $\leq 10^{12}$ tokens), it is predicted to become undesirable when applied to fully trained LLMs (e.g., trained with 100 trillion tokens, a milestone expected to be reached in the next few years), particularly for smaller models. Note that the gray areas in this figure indicate levels of QiD that degrade the model's predictions to a level worse than random guessing.

bit training keeps the model capable of operating under low bit throughout the training.

5 Conclusion

We derive scaling laws for low-bit quantization from over 1500 quantized LLM checkpoints and reveal that low-bit quantization favors undertrained LLMs. We provide an interpretation for this phenomenon and introduce a novel perspective of using QiD as a signal to determine a model's training level. Moreover, we use the derived scaling laws to predict the effect of low-bit quantization on LLMs trained with 100 trillion tokens. This, on one hand, challenges the future practical value of low-bit quantization, and on the other hand, suggests that future research on low-bit quantization should consider the model's training level during evaluation. Alongside concurrent research (Kumar et al., 2024; Feng et al., 2024) that takes a serious look at the limits of low-bit LLMs, we hope this

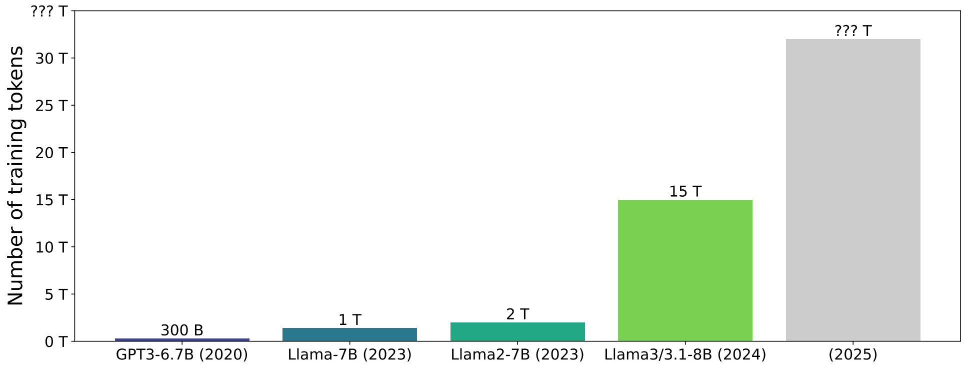

Figure 13: The number of training tokens for the state-of-the-art 7B-scale LLMs increase by nearly $50\times$ over the past 4 years. According to this trend, it is expected that the future models will have much more training tokens.

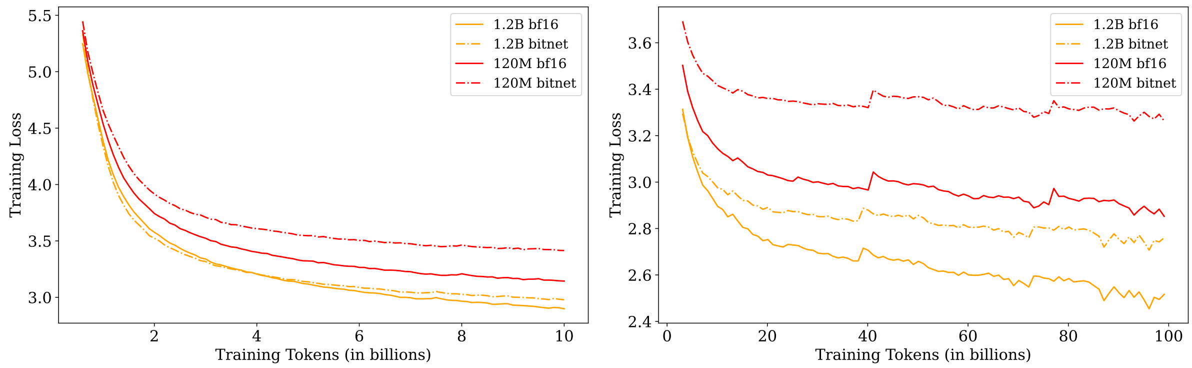

Figure 14: Training losses of BitNet and its 16-bit counterparts show a trend similar to that of low-bit quantization – they tend to perform well when undertrained but struggle to match the performance of fully trained LLMs.

work can help the community cool down from the surrounding hype, and foster deeper reflection and critical examination in this field.

6 Limitations

This work includes the following limitations:

- Although we have done our best to conduct extensive experiments and derive the scaling laws from over 1500 quantized checkpoints, it is still not extensive enough. For example, the training tokens used in our experiments with Pythia only amount to 300 billion. We expect more observations from a greater number of quantized checkpoints in the future to refine the scaling laws we have derived.

- The scaling laws derived in this work are primarily focused on single-stage pre-trained language models. However, advanced LLMs today often employ multi-stage training strategies including supervised fine-tuning and preference optimization, and even within pre-training, multiple stages are often involved (e.g., Llama-3.1 focuses more on high-quality text, math, reasoning, and code data during the final pre-training stages). Such multi-stage training strategies may cause the behavior of the model after quantization to be significantly different, which we plan to explore in future work.

- Our study focuses on dense models. However, mixture-of-experts (MoE) models are increasingly prevalent. Extending our laws to MoE architectures is an important direction for future work, as they may exhibit different pretraining dynamics.

- Alternative functional forms are also possible. Our scaling law, $\Delta_q Loss(N, D, P)$, is constructed based on power-law formulations, which demonstrate strong empirical performance in modeling QiD, both independently and collectively. Nonetheless, we experimented with other functional forms, including linear, exponential, and logarithmic laws. These forms may also provide reasonable empirical fits. We aim to conduct more comparisons based on experiment results in future work.

References

Haoli Bai, Wei Zhang, Lu Hou, Lifeng Shang, Jing Jin, Xin Jiang, Qun Liu, Michael Lyu, and Irwin King. 2020. Binarybert: Pushing the limit of bert quantization. *arXiv preprint arXiv:2012.15701*.

Ron Banner, Yury Nahshan, and Daniel Soudry. 2019. Post training 4-bit quantization of convolutional networks for rapid-deployment. *Advances in Neural Information Processing Systems*, 32.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in Neural Information Processing Systems*, 35:30318–30332.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Vage Egiazarian, Andrei Panferov, Denis Kuznedelev, Elias Frantar, Artem Babenko, and Dan Alistarh. 2024. Extreme compression of large language models via additive quantization. *arXiv preprint arXiv:2401.06118*.

Guhao Feng, Kai Yang, Yuntian Gu, Xinyue Ai, Shengjie Luo, Jiacheng Sun, Di He, Zhenguo Li, and Liwei Wang. 2024. How numerical precision affects mathematical reasoning capabilities of llms. *arXiv preprint arXiv:2410.13857*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Wei Huang, Yangdong Liu, Haotong Qin, Ying Li, Shiming Zhang, Xianglong Liu, Michele Magno, and Xiaojuan Qi. 2024. Billm: Pushing the limit of post-training quantization for llms. *arXiv preprint arXiv:2402.04291*.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. 2018. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Ayush Kaushal, Tejas Vaidhya, Arnab Kumar Mondal, Tejas Pandey, Aaryan Bhagat, and Irina Rish. 2024. Spectra: Surprising effectiveness of pretraining ternary language models at scale. *arXiv preprint arXiv:2407.12327*.

Raghuraman Krishnamoorthi. 2018. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*.

Tanishq Kumar, Zachary Ankner, Benjamin F Spector, Blake Bordelon, Niklas Muennighoff, Mansheej Paul, Cengiz Pehlevan, Christopher Ré, and Aditi Raghunathan. 2024. Scaling laws for precision. *arXiv preprint arXiv:2411.04330*.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of Machine Learning and Systems*, 6:87–100.

Jing Liu, Ruihao Gong, Xiuying Wei, Zhiwei Dong, Jianfei Cai, and Bohan Zhuang. 2023. Qllm: Accurate and efficient low-bitwidth quantization for large language models. *arXiv preprint arXiv:2310.08041*.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. 2024. Kivi: A tuning-free asymmetric 2bit quantization for kv cache. *arXiv preprint arXiv:2402.02750*.

Shuming Ma, Hongyu Wang, Lingxiao Ma, Lei Wang, Wenhui Wang, Shaohan Huang, Li Dong, Ruiping Wang, Jilong Xue, and Furu Wei. 2024. The era of 1-bit llms: All large language models are in 1.58 bits. *arXiv preprint arXiv:2402.17764*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Xuan Shen, Peiyan Dong, Lei Lu, Zhenglun Kong, Zhengang Li, Ming Lin, Chao Wu, and Yanzhi Wang. 2024. Agile-quant: Activation-guided quantization for faster inference of llms on the edge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18944–18951.

Ravid Shwartz-Ziv and Naftali Tishby. 2017. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Huaijie Wang, Lingxiao Ma, Fan Yang, Ruiping Wang, Yi Wu, and Furu Wei. 2023. Bitnet: Scaling 1-bit transformers for large language models. *arXiv preprint arXiv:2310.11453*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. 2019. Q8bert: Quantized 8bit bert. In *2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition (EMC2-NIPS)*, pages 36–39. IEEE.

Cheng Zhang, Jianyi Cheng, George A Constantinides, and Yiren Zhao. 2024. Lqer: Low-rank quantization error reconstruction for llms. *arXiv preprint arXiv:2402.02446*.

Wei Zhang, Lu Hou, Yichun Yin, Lifeng Shang, Xiao Chen, Xin Jiang, and Qun Liu. 2020. Ternarybert: Distillation-aware ultra-low bit bert. *arXiv preprint arXiv:2009.12812*.

Yunshan Zhong, Jiawei Hu, You Huang, Yuxin Zhang, and Rongrong Ji. 2024. Erq: Error reduction for post-training quantization of vision transformers. *arXiv preprint arXiv:2407.06794*.

A Appendix

A.1 Implementation Details

Checkpoints of the Pythia We choose the following 20 checkpoints of the Pythia models at the following steps for fitting the scaling laws: {512, 1k, 2k, 4k, 6k, 8k, 10k, 12k, 14k, 20k, 24k, 29k, 36k, 43k, 57k, 71k, 86k, 93k, 95k, 98k}.

Tokenization consistency To ensure consistency in token counts for computing cross entropy loss, which can vary with different tokenizers, we use the token counts generated by the Llama-3 8B (Dubey et al., 2024) tokenizer for all QiD calculations in this work.