What Makes Quantization for Large Language Models Hard?
An Empirical Study from the Lens of Perturbation

Zhuocheng Gong[1], Jiahao Liu[2], Jingang Wang[2], Xunliang Cai[2], Dongyan Zhao[1,4†], Rui Yan[3†]

[1] Wangxuan Institute of Computer Technology, Peking University
[2] Meituan
[3] Gaoling School of Artificial Intelligence, Renmin University of China
[4] State Key Laboratory of Media Convergence Production Technology and Systems
{gzhch,zhaody}@pku.edu.cn, ruiyan@ruc.edu.cn, {liujiahao12,wangjingang02,caixunliang}@meituan.com

Abstract

Quantization has emerged as a promising technique for improving the memory and computational efficiency of large language models (LLMs). Though the trade-off between performance and efficiency is well-known, there is still much to be learned about the relationship between quantization and LLM performance. To shed light on this relationship, we propose a new perspective on quantization, viewing it as perturbations added to the weights and activations of LLMs. We call this approach "the lens of perturbation". Using this lens, we conduct experiments with various artificial perturbations to explore their impact on LLM performance. Our findings reveal several connections between the properties of perturbations and LLM performance, providing insights into the failure cases of uniform quantization and suggesting potential solutions to improve the robustness of LLM quantization. To demonstrate the significance of our findings, we implement a simple non-uniform quantization approach based on our insights. Our experiments show that this approach achieves minimal performance degradation on both 4-bit weight quantization and 8-bit quantization for weights and activations. These results validate the correctness of our approach and highlight its potential to improve the efficiency of LLMs without sacrificing performance.

Introduction

Recently, large language models (LLMs) have gained a lot of attention in the deep learning community due to their remarkable performance (Touvron et al. 2023; OpenAI 2022). However, their size and computational demands pose challenges for deployment on resource-constrained devices or in real-time applications. To address this issue and improve the accessibility and efficiency of these models, researchers have explored quantization techniques that enable high-precision data to be represented in lower-precision formats. These techniques can significantly reduce the memory footprint and computational requirements of large models, making them more practical for deployment and use.

Quantizing large language models presents unique challenges that require careful consideration. First, LLMs can be incredibly large, with billions of parameters and multiple layers, resulting in significant quantization error accumulation. Second, due to the large model scale, training can be extremely costly, making post-training quantization (PTQ) the most appealing approach for LLMs, as it requires much less or even zero calibration data to perform (Bondarenko, Nagel, and Blankevoort 2021; Wu, Yao, and He 2023; Xiao et al. 2023). In summary, effectively quantifying LLMs requires a tailored approach that balances model complexity, computational resources, and performance. Despite recent studies investigating the quantization of LLMs, a comprehensive investigation into the performance of quantization has not been satisfactorily undertaken. Specifically, exploring the impact of different quantization strategies on different model families during the quantization process would be beneficial to the development of LLM quantization.

In this paper, our goal is to investigate the impact of quantization on LLMs in a systematic manner, with a focus on understanding the challenges of quantizing large language models. Specifically, we examine the effects of **zero-shot post-training uniform quantization** on LLMs, as it is a practical and representative quantization setting that requires no additional training or data. To begin, we apply standard uniform quantization to various LLMs and analyze the resulting performance degradation (as shown in Figure 1). Our findings indicate that the impact of quantization varies significantly across different model families and sizes. The BLOOM and LLAMA model families exhibit greater robustness to quantization, while OPT models are more sensitive to quantization. Furthermore, we observe that the scale of the model is a crucial factor that affects the quality of quantization. Finally, we find that activation quantization has a significant effect on performance.

To account for the various factors that impact quantization, it is important to establish an analytical framework. This is where **"the lens of perturbation"** comes to play. We view quantization as the addition of small perturbations to the weights and/or activations of the model. By perturbing these components and measuring the resulting performance degradation, we can analyze the impact of quantization on LLMs. This approach enables us to gain insights into how quantization influences the model's performance in different ways. To be more specific, we subject LLMs to different kinds of artificial perturbations and investigate which perturbation causes the most performance degradation. By
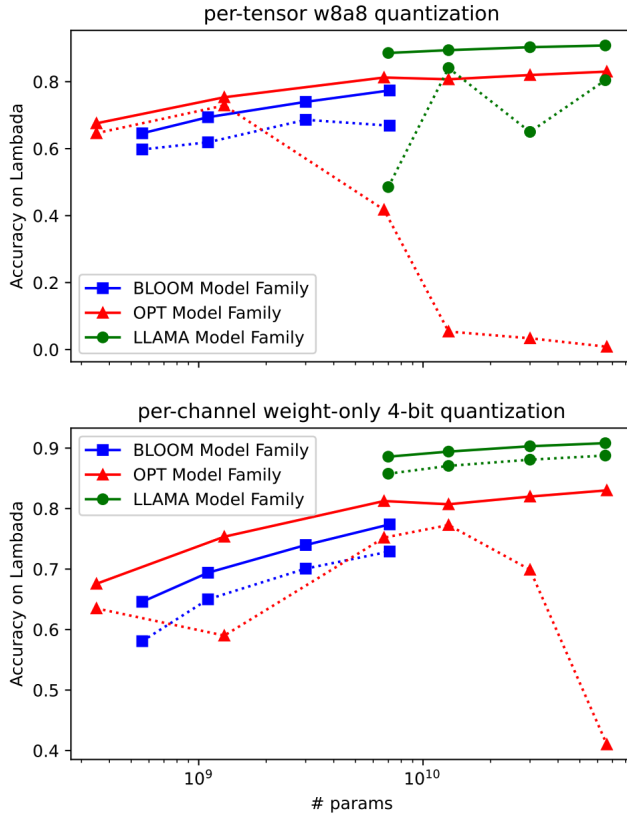
[†]Corresponding authors: Dongyan Zhao and Rui Yan.

Figure 1: Performance of quantized LLMs on Lambada (Paperno et al. 2016) on different model families (BLOOM (Scao et al. 2022), OPT (Zhang et al. 2022), and LLAMA (Touvron et al. 2023)), the parameter number scaling from 350M to 66B. We implement two uniform quantization settings: one is to quantize both weights and activations to 8 bits (W8A8), and the other is to perform 4-bit channel-wise for weights only (W4A16).

experimenting with artificial perturbations, we can gain insights into the robustness of the LLM quantization method and identify potential areas for improvement.

By using the lens of perturbation, we are able to delve deeper into the mechanism of quantization and identify several factors that impact the performance of LLM quantization. Our first finding is that the ability of data to withstand perturbations is closely tied to its magnitude. Specifically, larger values can endure more significant perturbations with minimal performance degradation, while smaller values can only tolerate minor perturbations. Our second observation pertains to the outlier phenomenon, which involves abnormally large values in weights and activations (Bondarenko, Nagel, and Blankevoort 2021). These insights obtained through the lens of perturbation explain why uniform quantization may not always be optimal. Uniform quantization treats all weights and activations equally, without considering their sensitivity to perturbations, and is obsessed with outlier values. Instead, we propose a potential improvement by taking into account the sensitivity to perturbations.

Through experiments, we demonstrate that this modification significantly reduces the quantization error, where uniform quantization falls short. These findings provide a more nuanced understanding of the impact of quantization on LLMs and offer a promising direction for improving the quantization process.

In summary, our contributions are as follows:

- We propose a new perspective of investigating uniform quantization for LLMs, namely the lens of perturbation. This lens provides new insights into the challenges of quantization for LLMs.

- We conduct a comprehensive comparison of the quantization performance of LLMs across various model scales and three model families (LLAMA, BLOOM, and OPT), using three quantization settings (W4A16, W8A8, and W4A8)

- Drawing on the insights through the lens of perturbation, we conduct preliminary experiments to leverage the properties of quantization-friendly perturbations and explore the potential of non-uniform quantization for LLMs, which significantly reduces quantization error. These experiments demonstrate that non-uniform quantization can significantly reduce quantization error.

Related Works
Large Language Model Quantization
Quantization, which represents the weights and activations of neural networks with low-bit precision, has been widely studied in computer vision and natural language processing (NLP) communities (Gholami et al. 2021). Recently, some researchers explore quantizing large language models to reduce the deployment cost (Shen et al. 2020; Wu et al. 2022; Kim et al. 2021; Bondarenko, Nagel, and Blankevoort 2021; Wu, Yao, and He 2023; Liu et al. 2023; Dettmers et al. 2023; Xiao et al. 2023). One major factor distinguishing LLM quantization is that LLMs with their enormous parameters are too expensive to train. Thus, most current approaches choose few-shot or zero-shot post-training quantization, which directly quantizes the already-trained model without much calibration data. These methods mostly build on uniform quantization, the most widespread, practical, and easy-to-use quantization strategy. Up to now, it has been validated that quantizing the weight of LLMs to 8 or even 4 bits is not too hard (Dettmers and Zettlemoyer 2023), however, activation quantization is still a major obstacle. Multiple studies have revealed the outlier phenomenon in weights and activations of transformer-based models, a tiny fixed set of embedding dimensions consistently exhibits large values across multiple layers (Bondarenko, Nagel, and Blankevoort 2021; Puccetti et al. 2022). Statistical analysis of LLMs shows that activations suffer from outliers several orders of magnitude larger than weights (Xiao et al. 2023). As a result, under the setting of uniform quantization, activations encounter much larger quantization loss. Some methods have been proposed to alleviate this problem, for example, by separating the quantization of outlier dimensions and regular dimensions (Gong et al. 2023), or by partially migrating outlier effect from activations to weights (Xiao et al.
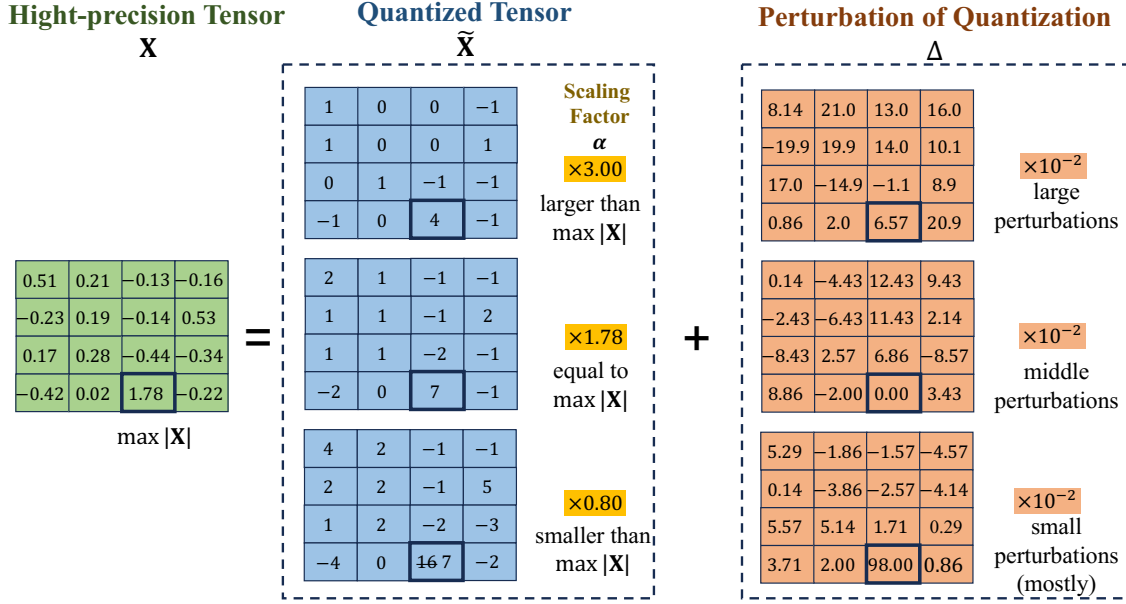
Hight-precision Tensor
X

Quantized Tensor
$\widetilde{\mathbf{X}}$

Perturbation of Quantization
Δ

Figure 2: A toy example of 4-bit uniform quantization with different scale factors α. The top line shows a choice of α that is too large, leading to too much information lost in the quantization process. The middle represents the most common choice, where α is set to the maximum value of the tensor. The bottom line uses a smaller α, which results in less perturbation at the cost of clipping out-of-range values.

2023) . Recently, some researchers have analyzed why outliers appear and proposed a method to remove outliers and produce quantization-friendly models (Bondarenko, Nagel, and Blankevoort 2023). While most existing works target proposing new quantization methods for LLMs, some focus on empirical analysis of LLM quantization, among which Yao et al. (2023) thoroughly compares the effect of different quantization schemes, model families, and quantization coverage, e.g.

Preliminaries

Uniform Quantization
Quantization is a technique that involves mapping high-precision values into low-precision discrete values. This process reduces the memory and computational requirements of the model, making it more efficient to deploy on devices with limited resources. Given a float-point tensor $\mathbf{X}$ (either a weight tensor or activation tensor), the quantization and de-quantization process can be viewed as:

$$\mathbf{X}^{\mathbb{Z}} = Q(\mathbf{X}), \quad \widetilde{\mathbf{X}} = Q^{-1}(\mathbf{X}^{\mathbb{Z}}) \quad (1)$$

where $\mathbf{X}^{\mathbb{Z}}$ is the quantized tensor, and $\widetilde{\mathbf{X}}$ is the recovered tensor. In this paper, we target the setting of uniform quantization (Hubara et al. 2017), which is the most common quantization because of its efficient implementation and satisfactory performance. The quantization function $Q(\cdot)$ of uniform quantization is defined as a rounding-to-nearest operation over the scaled input:

$$Q_{\text{uni}}(\mathbf{X}) = \text{clip}\left(\left\lfloor \frac{\mathbf{X}}{\alpha} \cdot 2^b \right\rceil + z; 0, 2^b - 1\right) \quad (2)$$

where $b \in \mathbb{N}$ is the bit-width, $\alpha \in \mathbb{R}$ is the scale factor, and $z \in \mathbb{N}$ is zero-point. $\lfloor \cdot \rceil$ denotes the round-to-nearest-integer operator. The corresponding de-quantization function is as follows:

$$Q_{\text{uni}}^{-1}(\mathbf{X}) = (\mathbf{X}^{\mathbb{Z}} - z) \cdot \alpha / 2^b \quad (3)$$

In addition to the quantization strategy, the quality of quantization is also influenced by the bit-width and quantization granularity. The former is straightforward to comprehend: the number of quantization bins increases exponentially with the bit-width. For example, 8-bit quantization has 256 unique bins for each $\mathbf{X}$. The quantization granularity determines the size of $\mathbf{X}$. Tensor-wise quantization involves quantizing the entire tensor. Channel-wise or group-wise quantization, on the other hand, divides the tensor along a specific dimension and then quantizes each channel or group independently. Fine-grained granularity typically leads to better performance.

The Lens of Perturbation
While the quantized low-precision values are expected to be close to their high-precision counterparts, they usually may not be exactly equal due to the loss of precision. The extent of deviation from the high-precision values is influenced by the bit-width and the quantization scheme used. One way to view quantization is as perturbations added to the original values, which can be expressed as:

$$\mathbf{X} = \widetilde{\mathbf{X}} + \Delta, \quad (4)$$

where Δ is the difference matrix between the quantized and original tensors, representing the noise introduced by the
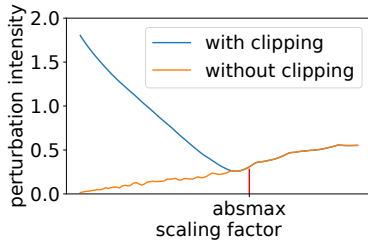
Figure 3: An illustration of the relationship between the scale factor and the intensity of perturbation. To calculate the perturbation intensity, we use the L2 norm of Δ, which represents the distance between the high-precision tensor and the quantized one.

quantization process. We call this perspective as **the lens of perturbation**.

The Choice of the Scale Factor

In uniform quantization, the scale factor α plays a critical role as it determines the scaling of original values. Usually, the scale factor is computed by optimizing the performance of a few calibration samples. However, in the zero-shot setting, it is often set to the absolute maximum value, represented $\alpha_{\max} = \max(|\mathbf{X}|)$.

Figure 2 illustrates how the choice of scale factor impacts uniform quantization. If the scale factor is greater than $\max(|\mathbf{X}|)$, values are overscaled, resulting in an insufficient utilization of the range of integers. Conversely, if the scale factor is smaller than $\max(|\mathbf{X}|)$, quantized values exceed the range so out-of-range values would be clipped. In the example depicted in Figure 2, the maximum value of the input tensor Smaller scale factors result in smaller quantization intervals and thus slighter perturbations for in-range values. Figure 3 provides a rough relationship between the intensity of perturbation and the scale factor. Throughout this paper, we use $\alpha_{\max}$ as the scale factor for uniform quantization, unless otherwise specified.

Uniform Quantization under the Lens of Perturbation
Before conducting experiments, we first examine uniform quantization through the lens of perturbation. For simplicity, we initially disregard the clipping operation for out-of-range values and later discuss how it affects quantization. In uniform quantization, high-precision values are mapped to a series of equal interval bins. Through the lens of perturbation, we make the following observations: (1) the intensity of perturbation is positively correlated with the scale factor (as depicted in Figure 3, the larger the scale factor, the greater the intensity of perturbation); (2) due to the uniform intervals, the magnitude of perturbation is not dependent on the magnitude of the values being quantized.

Looking into LLM Quantization through the Lens of Perturbation

In this section, we investigate the relationships between various types of perturbations and their impact on performance

degradation. Analyzing the statistical properties of perturbation can provide a more profound comprehension of the effects of quantization. Our goal is to determine which type of perturbation is more quantization-friendly (leading to less performance degradation).

What Kind of Perturbation Is More Quantization-Friendly?

To explore the correlation between perturbations and their impact on performance, we propose to use manually constructed perturbations instead of relying on the natural perturbations caused by quantization, which can be intricate and inconvenient to manipulate. Our approach involves controlling the properties of artificial perturbations and examining how they influence the model's behavior.

Our approach constructs perturbations from two perspectives: the distribution of the perturbation (i.e., the type of distribution from which the perturbation is sampled) and the magnitude of the perturbation (i.e., whether the perturbation's size is proportional to the input's size). By manipulating these properties, we can enhance our comprehension of how perturbations impact the model's performance. We consider three types of distributions for our perturbations:

- **Gaussian Perturbation** $\Delta_{\mathrm{G}} \sim \mathcal{N}(\mu, \sigma^2)$: where the perturbations are sampled from a Gaussian distribution.
- **Uniform Perturbation** $\Delta_{\mathrm{U}} \sim \mathcal{U}(-c, c)$: where the perturbations follow the uniform distribution between $-c$ and c.
- **Rademacher Perturbation** $\Delta_{\mathrm{R}} \sim \mathcal{R}$: where the perturbations have constant sizes but random signs.

Apart from the distribution, we also take into account whether the magnitude of perturbations should be linked to the magnitude of the input (i.e., whether larger values should have greater (or lesser) perturbations). In this sense, the three methods mentioned above are classified as "magnitude-independent perturbation," signifying that the perturbation's magnitude is not contingent on the input's magnitude To investigate the impact of perturbation magnitude, we introduce two "magnitude-aware perturbation" settings, in which the perturbation's size is dynamically determined by the value's size.

- **Positively-correlated Perturbation** $\Delta_{\mathrm{M+}}$: the positive correlation setting scales the perturbation magnitude positively with the input, meaning that higher weights have larger perturbations, formally, $\delta \propto |\mathbf{x}|$.
- **Negatively-correlated Perturbation** $\Delta_{\mathrm{M-}}$: conversely, the negative correlation setting scales the perturbation magnitude inversely with the input, meaning that higher weights lead to smaller perturbations, formally, we choose $\delta \propto (|\mathbf{x}| + \epsilon)^{-1}$.

We conducted experiments using different types of perturbations on the BLOOM-560M, BLOOM-3B, LLAMA-7B, and LLAMA-13B models. Due to the high computational cost, we did not experiment with more LLMs. Nevertheless, we are confident that our current experiments provide sufficient observations to draw reliable conclusions. To simulate the W8A8 quantization setting, we add perturbations to
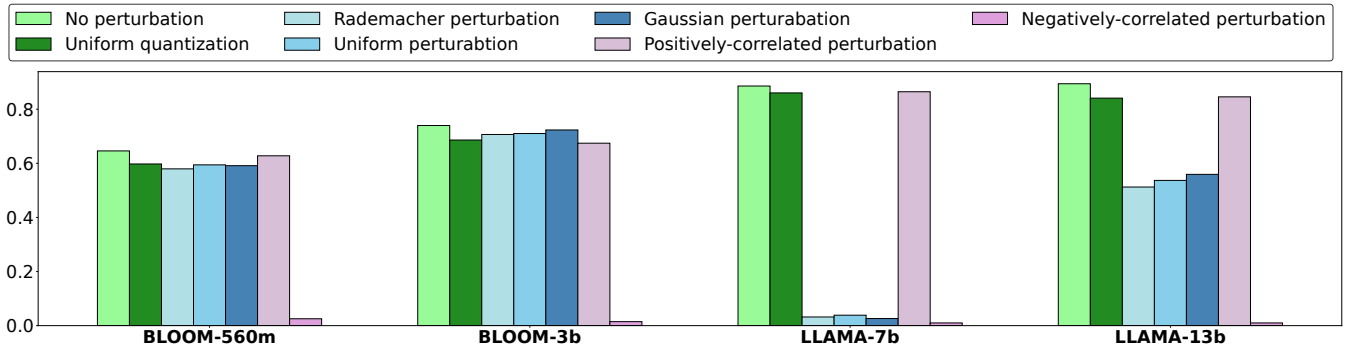
Figure 4: Lambada (Paperno et al. 2016) performance comparison on different types of perturbations. Here *No perturbation* represents the vanilla LLM (without quantization and without perturbation). *Uniform quantization* represents for W8A8 uniform quantization. For a fair comparison, all artificial perturbations are set to have the same variance with the native perturbation caused by *Uniform quantization*. The perturbation methods are run four times with different random seeds.

both weights and activations in this experiment. To ensure a fair comparison between artificial perturbations and quantization perturbations, we control the perturbation intensity (measured by the L2 norm of Δ) to be the same for each matrix being quantized.

Observations   The results of perturbations are shown in Figure 4. As expected, perturbations lead to non-negligible decreases in performance, regardless of the properties of the perturbations or the backbone models. The extent of performance degradation varies across different model families. In the case of the BLOOM model family, artificial perturbations exhibit similar performance to standard uniform quantization, whereas LLAMA models experience a significant loss in performance. This contrast highlights the distinctive characteristics of different LLM model families, which warrants further investigation. Regarding our primary objective of establishing a link between the properties of perturbations and performance degradation, we have made an interesting discovery. We have found that the sampling distribution has little impact on performance, as all three distributions we tested exhibit comparable performance. On the other hand, the magnitude of perturbations plays a crucial role. Specifically, the Positively-correlated perturbation Δ_{M+} (where larger values have larger perturbations) exhibits a dominant advantage over all perturbations on all models, while the negatively-correlated perturbation Δ_{M+} (where smaller values have larger perturbations) almost completely destroys performance. This phenomenon has led us to a significant finding: **larger values in LLM activations and weights are more robust to perturbations, whereas smaller values are more sensitive to perturbations.**

Clipping as a Special Perturbation
The experiment conducted on artificial perturbations mentioned above has not considered the effect of the clipping operation, which can have a considerable impact on realistic quantization scenarios, however. According to the formula for uniform quantization (Eq. 2), if α is smaller than $\max(|\mathbf{X}|)$, out-of-range values will be clipped, and the deviation caused by the clipping operation is much greater
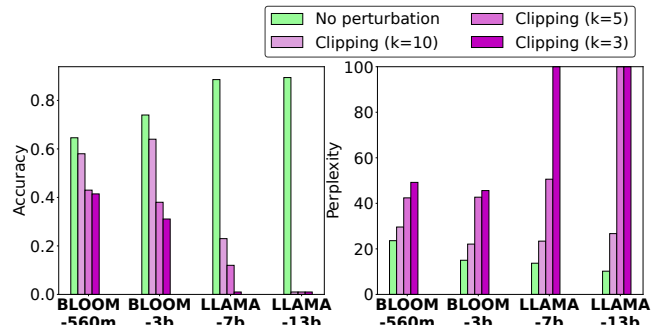

Figure 5: Effects of clipping on Lambada dataset (left) and Wikitext-2 (Merity et al. 2016) and C4 (Raffel et al. 2019) (averaged ppl.) (right). We compare three options of k, which determines the clipping threshold: $k \in \{3, 5, 10\}$. It is worth noting that even with the most stringent clipping setting, where $k = 3$, on average, only less than 0.1% of values are affected by clipping.

than that caused by regular perturbations. Through the lens of perturbation, clipping can be viewed as an exceedingly large perturbation applied to out-of-range values. Our current finding shows that larger values can tolerate much greater perturbations. Therefore, the question arises as to whether a perturbation as large as clipping is acceptable in terms of model performance.

To discuss the impact of extreme perturbations caused by clipping, we design and experiment with this special perturbation:

• **Clipping Perturbation** Δ_C: where large values exceeding the pre-set threshold are clipped to the threshold. We set the clipping threshold as $\mu_{\mathbf{X}} + k \cdot \sigma_{\mathbf{X}}$.

Figure 3 illustrates the effect of clipping perturbations on performance. Overall, clipping has a considerably more adverse impact than regular perturbations. This finding aligns with the outlier phenomenon, which posits that extremely large values are crucial for LLM performance. Clipping these values results in significant information loss, thus lead-

Quant. Methods	Bits (W/A)	Bloom				OPT						Llama			
		560M	1.1B	3B	7.1B	350M	1.3B	6.7B	13B	30B	66B	7B	13B	30B	65B
Wiketext-2 (ppl↓)															
full	16/16	19.36	24.33	11.83	10.32	15.89	11.35	8.84	8.32	7.93	7.14	19.7	13.3	10.3	10.18
Uniform	4/16	24.19	29.97	13.41	11.45	15.95	11.34	8.86	8.30	7.93	8.65	25.05	14.98	11.78	11.42
Non uniform	4/16	22.02	27.67	12.87	10.98	15.89	11.35	8.84	8.30	7.93	7.14	20.92	15.55	11.03	11.29
Uniform	8/8	26.14	36.88	14.98	15.64	18.28	12.2	18.83	3300	1328	2254	100.81	52.41	57.91	26.88
Non uniform	8/8	19.78	24.42	11.90	10.36	15.95	11.51	8.86	8.34	7.93	7.38	22.28	16.14	12.62	15.13
Uniform	4/8	33.31	44.56	17.28	17.52	18.33	12.16	18.58	3352	1262	2379	453.05	75.81	711	1095
Non uniform	4/8	22.67	27.72	12.91	11.03	15.92	11.49	8.86	8.32	7.93	7.38	21.02	15.58	11.07	12.01
C4 (ppl↓)															
full	16/16	27.83	22.98	18.08	15.83	22.94	16.20	12.91	12.38	11.78	9.57	7.67	7.14	6.50	6.12
Uniform	4/16	34.22	27.41	20.56	17.52	22.98	16.20	12.95	12.38	11.78	10.25	9.12	7.94	7.09	6.83
Non uniform	4/16	31.84	25.39	19.55	16.75	22.98	16.20	12.91	12.38	11.78	10.21	8.23	7.56	6.79	6.34
Uniform	8/8	34.84	35.81	21.59	20.64	26.88	17.94	98.88	6116	2790	4931	20.05	18.00	19.36	18.22
Non uniform	8/8	28.33	23.08	18.14	15.89	22.98	16.42	12.95	12.40	11.78	10.21	7.68	7.16	6.52	6.13
Uniform	4/8	43.44	45.19	25.09	22.8	26.92	17.86	100.81	6116	2812	5191	39.09	28.94	30.39	28.05
Non uniform	4/8	32.53	25.48	19.62	16.81	22.98	16.42	12.95	12.38	11.78	10.34	8.26	7.58	6.81	6.38
Lambada (accuracy↑)															
full	16/16	64.59	69.40.	73.98	77.39	67.57	75.35	81.25	80.71	81.99	83.01	88.58	89.42	90.31	90.82
Uniform	4/16	58.08	65.00	72.85	76.69	63.50	59.01	75.19	77.28	69.89	41.01	85.77	87.06	88.09	88.77
Non uniform	4/16	61.88	68.27	71.68	76.96	64.37	70.79	79.22	79.44	80.55	81.63	87.08	88.64	89.98	90.29
Uniform	8/8	59.77	61.90	68.62	66.87	64.61	72.91	41.75	5.38	3.37	0.84	86.03	84.10	65.00	80.45
Non uniform	8/8	64.70	68.99	73.77	77.39	67.65	74.76	80.94	80.47	81.84	82.49	88.52	89.46	90.37	90.32

Table 1: Comparison of uniform and non-uniform method on three datasets. In addition to evaluating W4A16 and W8A8 quantization, we also evaluate the much more challenging w4a8 quantization setting where weights are quantized to 4 bits and activations to 8 bits. For this experiment, we have not processed the entire test set for Wikitext-2 and C4. Instead, we use the first 2000 samples to calculate perplexity.
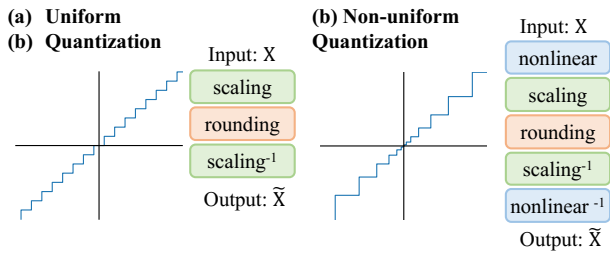

Figure 6: Illustration of uniform quantization and its non-uniform adaptation. The key difference is that non-uniform quantization utilizes two mutually inverse transformations to shape the input in a more quantization-friendly manner, one before the scaling operation and the other after the scaling-back operation.

Quant. Methods	Bits	Wikitext-2	C4
Full Precision	16/16	19.70	7.67
Non-uniform (Ours)	4/16	20.92	8.23
RTN (alias of Uniform Quant.)	4/16	25.05	9.12
GPTQ (Frantar et al. 2022)	4/16	21.09	8.48
AWQ (Lin et al. 2023)	4/16	20.77	8.03

Table 2: Comparison of different methods on LLaMA-7b.

ing to severe performance degradation. Another interesting observation is the comparison between LLAMA and BLOOM, where models from the LLAMA family seem to be more vulnerable to clipping.

Improving LLM Quantization

Having identified the preferred types of perturbations, we can now explore how these properties can be leveraged to improve existing quantization methods. Specifically, we have discovered that smaller values in a tensor should be subject to smaller perturbations, while larger values can handle greater perturbations, and that the magnitude of the perturbations should be constrained and clipping should be avoided. These properties can be useful in designing a quantization method that takes them into account.

To this end, we have implemented a quantization method to verify the effectiveness of the properties we discovered.

Method	Humanities	STEM	Social Sciences	Other	Avg.
Full Prec.	34.80	35.12	30.29	38.07	37.95
Uniform	31.63	34.13	27.55	33.38	33.92
Non-uniform	34.43	33.29	31.48	38.40	35.83

Table 3: LLaMA-7B 4-shot ICL results on MMLU.

Our goal is to ensure that smaller values in a tensor are subject to smaller perturbations, while larger values bear more perturbations. We achieve this by applying a nonlinear transformation before quantization, which amplifies smaller values around zero and shrinks larger values. The approach is straightforward: perform a nonlinear transformation before quantization, and inverse transform it back after quantization, as illustrated in Figure 6. For simplicity, we select $f(x) = x^{1/3}$ as the nonlinear transformation as it effectively amplifies values near zero and shrinks larger values.

We conducted experiments on three model families: BLOOM, OPT, and LLAMA. Table 1 summarizes results on Lambada, Wikitext-2, and C4. In the W4A16 setting, where activations are not quantized, uniform quantization performs well on most of the models and tasks, suggesting that weight quantization is not overly challenging. However, when it comes to quantizing activations, uniform quantization exhibits weaknesses, while the non-uniform method maintains near-lossless performance. We observe that for OPT models, as the model scales up, standard min-max uniform quantization struggles to perform accurately with explosive perplexity scores, whereas the non-uniform method maintains the model's performance close to that of the unquantized model. Another interesting finding is that for non-uniform quantization, as the model size scales up, the performance gap between before and after quantization decreases, indicating that larger models are easier to quantize. This phenomenon contrasts with observations made in uniform quantization, where model performance deteriorates as the model size grows.

We conducted more experiments on LLaMA-7b. Table 2 shows the comparison of non-uniform quantization and some recent post-training quantization methods and Table 3 is the performance on MMLU (Hendrycks et al. 2021), an advanced benchmark for evaluating the abilities of LLMs.

Limitations and Practical Value Quantization offers two sources of efficiency improvements: reduced memory costs through the use of low-precision formats, which accelerates data transfer in Memory/GPUs (memory efficiency), and faster matrix multiplication of low-precision formats (computation efficiency). However, the latter advantage is only realized when both the weight and activation are quantized to low precision. In the commonly used W4A16 setting, matrix calculation is still performed with high precision. Despite non-uniform quantization being a dominant approach in signal processing, it has not received sufficient attention in the realm of neural network compression. One major obstacle to its application in model compression is efficiency concerns. Although our non-uniform quantization method demonstrates reliable performance, it is important to note that the introduction of non-linear transformation entails additional computations. Consequently, non-uniform quantization only offers memory efficiency without computation efficiency. However, as current LLMs are not computation-constrained but rather memory-constrained, the model wastes a significant amount of time waiting for variables to be loaded on specific hardware rather than performing calculations. This imbalance of memory capacity and computation is particularly outstanding when generating long text, as the loading/offloading of KV-Cache for each generated token significantly slows down inference. This scenario is where most of the current quantization approaches glow.

Further Insights Through our experimentation with the non-uniform quantization method, which amplifies smaller values and compresses larger ones, we have confirmed a general principle regarding model quantization. Specifically, larger values can tolerate more error, while smaller values are more sensitive to larger errors. Based on this observation, we have identified the limitations of uniform quantization, which employs uniformly distributed quantization intervals to scale all values of varying sizes. We propose that is preferable to establish dense bins for small values and sparser bins for large values. We believe that this finding is highly thought-provoking and warrants further exploration in future research on model quantization. There are two potential avenues to pursue: one is to implement hardware-friendly non-uniform quantization for efficient deployment, while the other entails shaping the weight and activation distributions through training to make them easier to quantize. Some prior research has already been conducted in this direction (Bondarenko, Nagel, and Blankevoort 2023).

Conclusion

Our work introduces a new perspective on quantization, which we refer to as "the lens of perturbation". Using this approach, we conduct a comprehensive investigation of uniform quantization on LLMs, evaluating the performance of various models under different quantization settings. We also probe the models with manually constructed perturbations, which provide valuable insights into how quantization impacts the model performance. These insights help us understand the difficulties associated with quantization for LLMs. Based on our findings, we propose a non-uniform quantization approach that significantly reduces the performance degradation caused by quantization. We hope that our study, conducted through the lens of perturbation, will contribute to a better understanding of the challenges associated with quantization for LLMs and inspire the development of more efficient and effective quantization methods.

Acknowledgments

References

Bondarenko, Y.; Nagel, M.; and Blankevoort, T. 2021. Understanding and Overcoming the Challenges of Efficient Transformer Quantization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 7947–7969. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Bondarenko, Y.; Nagel, M.; and Blankevoort, T. 2023. Quantizable Transformers: Removing Outliers by Helping Attention Heads Do Nothing. *arXiv preprint arXiv:2306.12929*.

Dettmers, T.; Pagnoni, A.; Holtzman, A.; and Zettlemoyer, L. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Dettmers, T.; and Zettlemoyer, L. 2023. The case for 4-bit precision: k-bit inference scaling laws. In *International Conference on Machine Learning*, 7750–7774. PMLR.

Frantar, E.; Ashkboos, S.; Hoefler, T.; and Alistarh, D. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Gholami, A.; Kim, S.; Dong, Z.; Yao, Z.; Mahoney, M. W.; and Keutzer, K. 2021. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*.

Gong, Z.; Liu, J.; Wang, Q.; Yang, Y.; Wang, J.; Wu, W.; Xian, Y.; Zhao, D.; and Yan, R. 2023. PreQuant: A Task-agnostic Quantization Approach for Pre-trained Language Models. In *Findings of the Association for Computational Linguistics: ACL 2023*, 8065–8079. Toronto, Canada: Association for Computational Linguistics.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Hubara, I.; Courbariaux, M.; Soudry, D.; El-Yaniv, R.; and Bengio, Y. 2017. Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research*, 18(1): 6869–6898.

Kim, S.; Gholami, A.; Yao, Z.; Mahoney, M. W.; and Keutzer, K. 2021. I-BERT: Integer-only BERT Quantization. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 5506–5518. PMLR.

Lin, J.; Tang, J.; Tang, H.; Yang, S.; Dang, X.; and Han, S. 2023. AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration. *arXiv preprint arXiv:2306.00978*.

Liu, Z.; Oguz, B.; Zhao, C.; Chang, E.; Stock, P.; Mehdad, Y.; Shi, Y.; Krishnamoorthi, R.; and Chandra, V. 2023. LLM-QAT: Data-Free Quantization Aware Training for Large Language Models. *arXiv preprint arXiv:2305.17888*.

Merity, S.; Xiong, C.; Bradbury, J.; and Socher, R. 2016. Pointer Sentinel Mixture Models. arXiv:1609.07843.

OpenAI. 2022. OpenAI chatgpt. https://openai.com/blog/chatgpt. Accessed: 2024-02-28.

Paperno, D.; Kruszewski, G.; Lazaridou, A.; Pham, N. Q.; Bernardi, R.; Pezzelle, S.; Baroni, M.; Boleda, G.; and Fernandez, R. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1525–1534. Berlin, Germany: Association for Computational Linguistics.

Puccetti, G.; Rogers, A.; Drozd, A.; and Dell'Orletta, F. 2022. Outlier Dimensions that Disrupt Transformers are Driven by Frequency. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 1286–1304.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv e-prints*.

Scao, T. L.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A. S.; Yvon, F.; Gallé, M.; et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Shen, S.; Dong, Z.; Ye, J.; Ma, L.; Yao, Z.; Gholami, A.; Mahoney, M. W.; and Keutzer, K. 2020. Q-bert: Hessian based ultra low precision quantization of bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8815–8821.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Wu, X.; Yao, Z.; and He, Y. 2023. ZeroQuant-FP: A Leap Forward in LLMs Post-Training W4A8 Quantization Using Floating-Point Formats. *arXiv preprint arXiv:2307.09782*.

Wu, X.; Yao, Z.; Zhang, M.; Li, C.; and He, Y. 2022. Extreme Compression for Pre-trained Transformers Made Simple and Efficient. *arXiv preprint arXiv:2206.01859*.

Xiao, G.; Lin, J.; Seznec, M.; Wu, H.; Demouth, J.; and Han, S. 2023. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, 38087–38099. PMLR.

Yao, Z.; Wu, X.; Li, C.; Youn, S.; and He, Y. 2023. ZeroQuant-V2: Exploring Post-training Quantization in LLMs from Comprehensive Study to Low Rank Compensation. arXiv:2303.08302.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.