Huang et al. *Visual Intelligence* (2024) 2:36
https://doi.org/10.1007/s44267-024-00070-x

Visual Intelligence

RESEARCH Open Access

An empirical study of LLaMA3 quantization: from LLMs to MLLMs

Wei Huang[1†], Xingyu Zheng[2†], Xudong Ma[2†], Haotong Qin[3*], Chengtao Lv[2], Hong Chen[2], Jie Luo[2], Xiaojuan Qi[1], Xianglong Liu[2] and Michele Magno[3]

Abstract

The LLaMA family, a collection of foundation language models ranging from 7B to 65B parameters, has become one of the most powerful open-source large language models (LLMs) and the popular LLM backbone of multi-modal large language models (MLLMs), widely used in computer vision and natural language understanding tasks. In particular, LLaMA3 models have recently been released and have achieved impressive performance in various domains with super-large scale pre-training on over 15T tokens of data. Given the wide application of low-bit quantization for LLMs in resource-constrained scenarios, we explore LLaMA3's capabilities when quantized to low bit-width. This exploration can potentially provide new insights and challenges for the low-bit quantization of LLaMA3 and other future LLMs, especially in addressing performance degradation issues that suffer in LLM compression. Specifically, we comprehensively evaluate the 10 existing post-training quantization and LoRA fine-tuning (LoRA-FT) methods of LLaMA3 on 1-8 bits and various datasets to reveal the low-bit quantization performance of LLaMA3. To uncover the capabilities of low-bit quantized MLLM, we assessed the performance of the LLaMA3-based LLaVA-Next-8B model under 2-4 ultra-low bits with post-training quantization methods. Our experimental results indicate that LLaMA3 still suffers from non-negligible degradation in linguistic and visual contexts, particularly under ultra-low bit widths. This highlights the significant performance gap at low bit-width that needs to be addressed in future developments. We expect that this empirical study will prove valuable in advancing future models, driving LLMs and MLLMs to achieve higher accuracy at lower bit to enhance practicality.

Keywords: Model quantization, Large language model, Multi-modal, Deep learning

1 Introduction

Launched by Meta in February 2023, the LLaMA [1] series,[1] a collection of foundation language models ranging from 7B to 65B parameters, represents a breakthrough in autoregressive large language models (LLMs) using the Transformer [2] architecture. From its first release, with 13 billion parameters, it outperformed the much larger, closed-source GPT-3 model with 175 billion parameters. On April 18, 2024, Meta introduced the LLaMA3 model, offering 8 billion and 70 billion parameter configurations. Thanks to extensive pre-training on more than 15 trillion data tokens, the LLaMA3 models [3] have achieved state-of-the-art performance across a wide range of tasks, establishing the LLaMA family as one of the best open-source LLMs available for a wide variety of applications and deployment scenarios. Recently, the LLaVA team [4] has launched the new LLaVA-Next-8B[2] model based on LLaMA3, giving the stronger general multi-modal capabilities of multi-modal large language models (MLLMs).

---

*Correspondence: haotong.qin@pbl.ee.ethz.ch
[3]Department of Information Technology and Electrical Engineering, ETH Zurich, Sternwartstrasse 7, Zürich, Switzerland
Full list of author information is available at the end of the article †Equal contributors

[1] https://llama.meta.com.

[2] https://llava-vl.github.io/blog/2024-05-10-llava-next-stronger-llms.

Springer

© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Despite their impressive performance, deploying LLaMA3 models still poses significant challenges due to resource limitations in many scenarios. Fortunately, low-bit quantization [5–8] has emerged as one of the most popular techniques for compressing LLMs. This technique reduces the memory and computational requirements of LLMs during inference, enabling them to run on resource-limited devices. Addressing the performance drop after compression is a major concern for current LLM quantization approaches. While numerous low-bit quantization methods have been proposed, their evaluations have primarily focused on the earlier and less capable LLaMA models (LLaMA and LLaMA2). Thus, LLaMA3 presents a new opportunity for the LLM community to assess the performance of quantization on cutting-edge LLMs and MLLMs and understand existing methods' strengths and limitations. In this empirical study, we aim to analyze the capability of LLaMA3 to handle the challenges associated with degradation due to quantization.

Our study delineates the outcomes of two principal techniques for quantizing LLaMA3 across three evaluation tracks: post-training quantization (PTQ) of LLMs, quantization of LLMs via LoRA-FineTuning (LoRA-FT), and PTQ of LLaMA3-based MLLM, aiming to conduct a comprehensive assessment of the LLaMA3 model's capabilities in language and visual-language tasks. We explore a range of cutting-edge quantization methods across technical tracks (RTN [9], GPTQ [10], AWQ [11], SmoothQuant [5], PB-LLM [12], QuIP [13], DB-LLM [14], BiLLM [15], and SliM-LLM [8] for PTQ; QLoRA [16] and IR-QLoRA [17] for LoRA-FT), covering a wide spectrum from 1 to 8 bits and utilizing a diverse array of evaluation datasets, including WikiText2 [18], C4 [19], PTB [20], CommonSenseQA datasets (PIQA [21], ARC-e [22], ARC-c [22], HellaSwag [23], Winogrande [24]), and MMLU [25] benchmark. For multi-modal tasks, we follow a common practice [11], performing low-bit post-training quantization on the LLM component of LLaVA-Next-8B using GPTQ and AWQ. We then validate the quantized MLLM inference capabilities on 6 visual language benchmarks, including AI2D [26], ChartQA [27], DocVQA [28], MME [29], and MMBench(English) [30]. These evaluations assess the capabilities and limitations of the LLaMA3 model under current LLM quantization techniques and serve as a source of inspiration for designing future large language and large visual-language model quantization methods. The decision to focus specifically on the LLaMA3 model is motivated by its superior performance among all current open-source instruction-tuned LLMs on a variety of datasets, including 5-shot MMLU, 0-shot GPQA, 0-shot HumanEval, 8-shot CoT GSM-8K, and 4-shot CoT MATH. The overview of our study is presented as Fig. 1.

This not only helps advance the research within the LLM and MLLM quantization community, but also facilitates a broader understanding and application of effective quantization.

We evaluate the low-bit quantization of LLaMA3-8B, -70B, and LLaVA-Next-8B, where the pre-trained models were obtained from their official repositories[2].

Quantization methods To evaluate the performance of low-bit quantized LLaMA3, we select representative LLM quantization methods with extensive influence and functionality, including 9 PTQ methods and 2 LoRA-FT methods. The implementations of our evaluated quantization

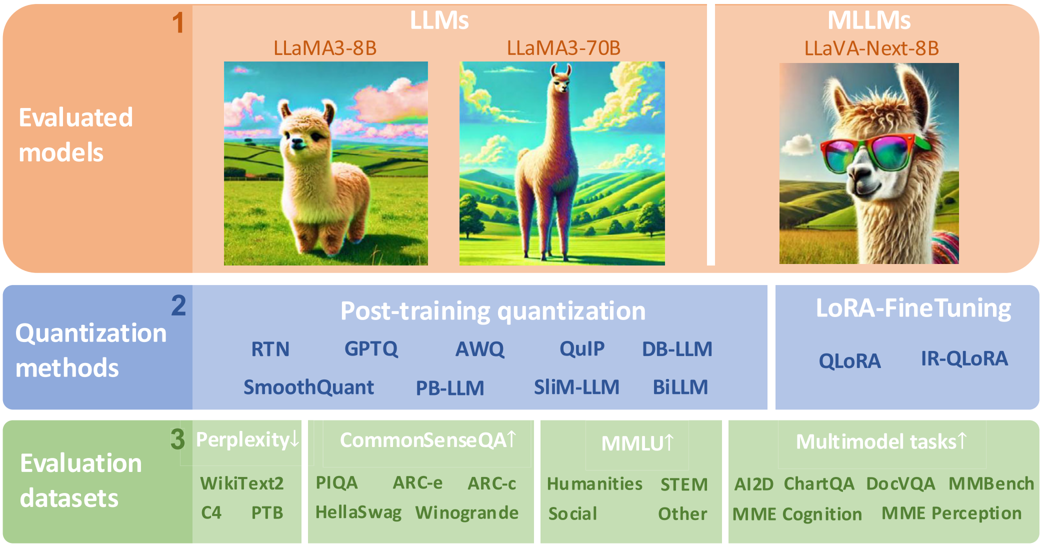


Figure 1 The overview of our empirical study

methods follow their open-source repositories.[3] We also used 8 NVIDIA A800 with 80 GB GPU memory for quantitative evaluation.

Evaluation datasets For the PTQ methods, we evaluate quantized LLaMA3 on the WikiText2 [18], PTB [20], and a portion of the C4 dataset [19], using perplexity (PPL) as the evaluation metric. Subsequently, we further conduct experiments on five zero-shot evaluation tasks (PIQA [21], Winogrande [24], ARC-e [22], ARC-c [22], and Hellaswag [23]) to fully validate the quantized performance of LLaMA3. We further conduct the evaluation on 5 visual language benchmarks (AI2D, ChartQA, DocVQA, MME, and MMBench(English)) for quantized LLaVA-Next-8B. To ensure fairness in evaluation of different PTQ methods, we set WikiText2 as the calibration dataset for all quantization methods, with a sample size of 128 and a sequence length of 2048. Additionally, for methods requiring grouped quantization, we standardize the block size at 128 to balance performance and inference efficiency, a common practice in existing studies. For the LoRA-FT methods, we conduct the evaluation on the 5-shot MMLU benchmark [31] while also validating the aforementioned 5 zero-shot datasets for the LoRA-FT methods. To ensure fairness in the evaluation of different LoRA-FT methods, we fine-tune all models using the same training data and consistent hyperparameters, including learning rate, batch size, number of training epochs, and LoRA configurations such as rank and scaling factors.

2 Track1: post-training quantization

Quantization framework We begin by outlining the general uniform quantization process for LLMs, following standard practices as described in Refs. [5, 10, 32]. This process involves mapping floating-point weights, distributed within the range $[w_{\min}, w_{\max}]$, to an integer range of 2^N, where N denotes the target bit-width. The quantization function for a weight matrix $\boldsymbol{w}_f \in \mathbb{R}^{n\times m}$ is defined as follows:

$$\hat{\boldsymbol{w}}_q = \text{clamp}(\lfloor \frac{\boldsymbol{w}_f}{\Delta} \rceil + z, 0, 2^N - 1) \tag{1a}$$

$$\Delta = \frac{w_{\max} - w_{\min}}{2^N - 1} \tag{1b}$$

$$z = -\lfloor \frac{w_{\min}}{\Delta} \rceil \tag{1c}$$

where $\hat{\boldsymbol{w}}_q$ indicates quantized weight, which is integer, N denotes the target bit-width, $\lfloor\cdot\rceil$ is round operation and clamp$(\cdot)$ constrains the value within integer range (e.g. $[0, 1, 2, 3]$, $N = 2$). Δ is scale factor and z is quantization zero point, respectively. As shown in Table 1 to Table 4, we provide the performance of the low-bit LLaMA3-8B and LLaMA3-70B with 8 different PTQ methods, respectively, covering a wide bit-width spectrum from 1 to 8 bits. In addition, the performance of LLaMA1 and LLaMA2 under the same setting are summarized in Table 5.

PTQ methods Among them, round-to-nearest (RTN) is a vanilla rounding quantization method that directly applies the statistical approach from Eq. (1a)–(1c) to obtain quantization parameters for immediate quantization. GPTQ [10] is one of the most effective weight-only quantization methods, utilizing an error compensation strategy based on second-order loss. By using the inverse of the Hessian matrix, it reduces compression errors during quantization. AWQ [11] employs an activation-aware outlier suppression approach, introducing a scaling factor s to smooth the weight distribution of LLMs, thereby easing the quantization difficulty. QuIP [13] ensures consistency between weights and the Hessian by optimizing matrix computations and adopts codebook encoding to quantize weight parameters, further enhancing the mapping accuracy between continuous and discrete parameter spaces. Recently, Huang et al. [8] proposed a grouped mixed-precision quantization method that leverages the clustering characteristics of significant weights. This method uses mixed precision group quantization to achieve high-precision low-bit quantization in a hardware-friendly manner. Both approaches preserve LLaMA3's 3-bit quantization capability, with the potential to bring 2-bit quantization to higher performance levels.

The recent emergence of binarized LLM quantization methods has realized ultra-low bit-width LLM weight compression. PB-LLM [12] employs a mixed-precision quantization strategy, retaining a small portion of significant weight full-precision while quantizing most of the weights to 1 bit. DB-LLM [14] achieves efficient LLM compression through double binarization weight splitting and proposes a deviation-aware distillation strategy to further improve 2-bit LLM performance. BiLLM [15] pushes the LLM quantization limit further down to 1.1 bit by residual approximation of salient weights and grouped quantization of non-salient weights. These LLM quantization methods, which are specially designed for ultra-low bit-width, can achieve higher accuracy of quantized LLaMA3-8B at ≤ 2 bits, far outperforming methods such as GPTQ, AWQ, and QuIP below 2 bits (even 3 bits in some cases). We also perform the evaluation on quantized activations using SmoothQuant [5], which shifts the quantization difficulty offline from activations to weights to smooth out activation outliers. Our evaluation shows that SmoothQuant can maintain the accuracy of LLaMA3

---

[3] https://github.com/IST-DASLab/gptq,https://github.com/mit-han-lab/llm-awq,https://github.com/mit-han-lab/smoothquant, https://github.com/Cornell-RelaxML/QuIP, https://github.com/Aaronhuang-778/SliM-LLM, https://github.com/hahnyuan/PB-LLM, https://github.com/Aaronhuang-778/BiLLM, https://github.com/artidoro/qlora, https://github.com/htqin/IR-QLoRA.

Table 1 Evaluation results of post-training quantization on the LLaMA3-8B model (1/2). #W, #A, and #G represent the bit-width for weight, activation, and group size, respectively, '–' indicates no grouping required, and ↓ denotes that the lower is better

Method	#W	#A	#G	PPL↓		
				WikiText2	C4	PTB
LLaMA3	16	16	–	6.1	9.2	10.6
RTN	4	16	128	8.5	13.4	14.5
	3	16	128	27.9	1.1e2	95.6
	2	16	128	1.9×10^3	2.5×10^4	1.8×10^4
	8	16	–	6.2	9.5	11.2
	4	16	–	8.7	14.0	14.9
	3	16	–	2.2×10^3	5.6×10^2	2.0×10^3
	2	16	–	2.7×10^6	7.4×10^6	3.1×10^6
GPTQ [10]	4	16	128	6.5	10.4	11.0
	3	16	128	8.2	13.7	15.2
	2	16	128	2.1×10^2	4.1×10^4	9.1×10^2
	8	16	–	6.1	9.4	10.6
	4	16	–	7.0	11.8	14.4
	3	16	–	13.0	45.9	37.0
	2	16	–	5.7×10^4	1.0×10^5	2.7×10^5
AWQ [11]	4	16	128	6.6	9.4	11.1
	3	16	128	8.2	11.6	13.2
	2	16	128	1.7×10^6	2.1×10^6	1.8×10^6
	8	16	–	6.1	8.9	10.6
	4	16	–	7.1	10.1	11.8
	3	16	–	12.8	16.8	24.0
	2	16	–	8.2×10^5	8.1×10^5	9.0×10^5
SliM-LLM [8]	4	16	128	6.4	9.5	10.9
	3	16	128	7.7	13.1	14.7
	2	16	128	39.7	1.1×10^2	1.6×10^2
QuIP [13]	4	16	–	6.5	11.1	9.5
	3	16	–	7.5	11.3	12.6
	2	16	–	85.1	1.3×10^2	1.8×10^2
DB-LLM [14]	2	16	128	13.6	19.2	23.8
PB-LLM [12]	2	16	128	24.7	79.2	65.6
	1.7	16	128	41.8	2.6×10^2	1.2×10^2
BiLLM [15]	1.1	16	128	28.3	2.9×10^2	94.7
SmoothQuant [5]	8	8	–	6.3	9.2	10.8
	6	6	–	7.7	11.8	12.5
	4	4	–	4.3×10^3	4.0×10^3	3.6×10^3
OmniQuant [33]	6	6	–	7.0	10.1	–
	4	4	–	4.4×10^2	3.2×10^2	–
I-LLM [34]	6	6	–	6.6	9.8	–
	4	4	–	21.2	30.9	–
SpinQuant [35]	4	8	–	6.5	–	–
	4	4	–	7.1	–	–

Table 2 Evaluation results of post-training quantization on the LLaMA3-70B model (1/2)

Method	#W	#A	#G	PPL↓		
				WikiText2	C4	PTB
LLaMA3	16	16	–	2.9	6.9	8.2
RTN	4	16	128	3.6	8.9	9.1
	3	16	128	11.8	22.0	26.3
	2	16	128	4.6×10^5	4.7×10^5	3.8×10^5
GPTQ [10]	4	16	128	3.3	6.9	8.3
	3	16	128	5.2	10.5	9.7
	2	16	128	11.9	22.8	31.6
AWQ [11]	4	16	128	3.3	7.0	8.3
	3	16	128	4.8	8.0	9.0
	2	16	128	1.7×10^6	1.4×10^6	1.5×10^6
SliM-LLM [8]	4	16	128	3.3	7.0	8.3
	3	16	128	4.1	7.9	9.0
	2	16	128	9.5	16.2	18.7
QuIP [13]	4	16	–	3.4	7.1	8.4
	3	16	–	4.7	8.0	8.9
	2	16	–	13.0	22.2	24.9
PB-LLM [12]	2	16	128	11.6	34.5	27.2
	1.7	16	128	18.6	65.2	55.9
BiLLM [15]	1.1	16	128	17.1	77.7	54.2
SmoothQuant [5]	8	8	–	2.9	6.9	8.2
	6	6	–	2.9	6.9	8.2
	4	4	–	9.6	16.9	17.7

with 6/8-bit weights and activations, but collapses at 4 bits. Moreover, we find that the LLaMA3-70B model shows significant robustness to different quantization methods, even for ultra-low bit-width quantization.

In the evaluation metrics of PPL (Table 1 and Table 2) and CommonSenseQA (Table 3 and Table 4), we found that, overall, the 4-bit methods had a slight performance decrease (approximately 2%) compared to the original 16-bit LLM, with no significant differences between the different methods. In the context of 3-bit scenarios, traditional RTN quantization methods faced substantial performance losses (over 10% lower than 4 bits), while methods such as GPTQ, AWQ, SliM-LLM, and QuIP were able to maintain performance close to that of 4 bits (with less than 5% performance degradation). Interestingly, both DB-LLM and BiLLM were able to achieve reasonable results at ultra-low bit-width settings of 2 bits and even 1.1 bits, possibly due to the large-batch fine-tuning strategy and BiLLM's fine-grained salience partitioning. When quantifying both weight and activation simultaneously, both the 8B and 70B models demonstrated near lossless performance at 8 bits. As the bit-width was further reduced, the performance loss decreased significantly for the 8B model, while it decreased slowly for 70B models, indicating the presence of information redundancy within 70B models.

For practical deployment, we recorded the GPU memory usage and training time consumption for some PTQ methods on different sizes of the LLaMA model, as shown in Table 6. It demonstrates that methods such as SmoothQuant and AWQ are highly efficient in terms of memory usage and training time, with SmoothQuant requiring only 13.5 GB of GPU memory and 7 min for LLaMA2-7B, making it an ideal choice for memory-constrained environments. In contrast, OmniQuant, while effective for model compression, shows significantly higher quantization time consumption. Meanwhile, we tested the inference latency of the quantized 4-bit models resulting from the above methods in real-world deployment, as shown in Table 6. In fact, GPTQ, AWQ, and Omniquant all use block-wise quantization techniques, and theoretically, the upper bound of real inference speed optimization for these three methods is the same. To ensure a fair comparison of latency, we conducted tests using the deployment methods provided in the original methodology. In the case of LLaMA2-7B, GPTQ, AWQ, and Omniquant all exhibited speeds exceeding 100 tokens per second. However, in the case of LLaMA3-8B, the overall speed ranged between 50 to 80 tokens per second, with AWQ's quantization kernel achieving an inference speed of 89.8 tokens per second, surpassing that of other methods.

Table 3 Evaluation results of post-training quantization on LLaMA3-8B model (2/2). ↑ indicates that the higher value is better

Method	#W	#A	#G	CommonSenseQA↑					
				PIQA	ARC-e	ARC-c	HellaSwag	Wino	Avg.
LLaMA3	16	16	–	79.9	80.1	50.4	60.2	72.8	68.6
RTN	4	16	128	76.6	70.1	45.0	56.8	71.0	63.9
	3	16	128	62.3	32.1	22.5	29.1	54.7	40.2
	2	16	128	53.1	24.8	22.1	26.9	53.1	36.0
	8	16	–	79.7	80.8	50.4	60.1	73.4	68.9
	4	16	–	75.0	68.2	39.4	56.0	69.0	61.5
	3	16	–	56.2	31.1	20.0	27.5	53.1	35.6
	2	16	–	53.1	24.7	21.9	25.6	51.1	35.3
GPTQ [10]	4	16	128	78.4	78.8	47.7	59.0	72.6	67.3
	3	16	128	74.9	70.5	37.7	54.3	71.1	61.7
	2	16	128	53.9	28.8	19.9	27.7	50.5	36.2
	8	16	–	79.8	80.1	50.2	60.2	72.8	68.6
	4	16	–	76.8	74.3	42.4	57.4	72.8	64.8
	3	16	–	60.8	38.8	22.3	41.8	60.9	44.9
	2	16	–	52.8	25.0	20.5	26.6	49.6	34.9
AWQ [11]	4	16	128	79.1	79.7	49.3	59.1	74.0	68.2
	3	16	128	77.7	74.0	43.2	55.1	72.1	64.4
	2	16	128	52.4	24.2	21.5	25.6	50.7	34.9
	8	16	–	79.6	80.3	50.5	60.2	72.8	68.7
	4	16	–	78.3	77.6	48.3	58.6	72.5	67.0
	3	16	–	71.9	66.7	35.1	50.7	64.7	57.8
	2	16	–	55.2	25.2	21.3	25.4	50.4	35.5
SliM-LLM [8]	4	16	128	78.9	79.9	49.4	58.7	72.6	67.9
	3	16	128	77.8	73.7	42.9	55.5	72.8	64.5
	2	16	128	57.1	35.4	26.1	28.9	56.6	40.8
QuIP [13]	4	16	–	78.2	78.2	47.4	58.6	73.2	67.1
	3	16	–	76.8	72.9	41.0	55.4	72.5	63.7
	2	16	–	52.9	29.0	21.3	29.2	51.7	36.8
DB-LLM	2	16	128	68.9	59.1	28.2	42.1	60.4	51.8
PB-LLM [12]	2	16	128	57.0	37.8	17.2	29.8	52.5	38.8
	1.7	16	128	52.5	31.7	17.5	27.7	50.4	36.0
BiLLM [15]	1.1	16	128	56.1	36.0	17.7	28.9	51.0	37.9
SmoothQuant [5]	8	8	–	79.5	79.7	49.0	60.0	73.2	68.3
	6	6	–	76.8	75.5	45.0	56.9	69.0	64.6
	4	4	–	54.6	26.3	20.0	26.4	50.3	35.5
SpinQuant [35]	4	8	–	79.6	76.5	54.0	78.1	72.4	72.1
	4	4	–	77.5	75.0	50.9	75.9	68.5	69.6

3 Track2: LoRA-FineTuning quantization

Quantization framework The LoRA-FT quantization process involves applying low-bit quantization to the original model weights, adding low-rank matrices to the pretrained model weights, and fine-tuning the low-rank matrices with the training data, allowing model updates without modifying the core parameters. In addition to using the integer quantization commonly applied in PTQ, LoRA-FT can also use NormalFloat quantization. The NormalFloat quantization function for a weight matrix $\boldsymbol{w}_q \in \mathbb{R}^{n\times m}$ is defined as follows:

$$\hat{\boldsymbol{w}}_q = \mathrm{NF}_k(\frac{\boldsymbol{w}}{s}) \tag{2}$$

where $\hat{\boldsymbol{w}}_q$ indicates quantized weight, s is the scale factor, typically set to the maximum value of $\boldsymbol{w}$ and NF_k denotes the NormalFloat quantization operator at k bit-width, mapping each value in $\boldsymbol{w}_{\mathrm{norm}}$ to the nearest quantile in the normal distribution for a bit-width k.

Table 4 Evaluation results of post-training quantization on the LLaMA3-70B model (2/2)

Method	#W	#A	#G	CommonSenseQA↑					
				PIQA	ARC-e	ARC-c	HellaSwag	Wino	Avg.
LLaMA3	16	16	–	82.4	86.9	60.3	66.4	80.6	75.3
RTN	4	16	128	82.3	85.2	58.4	65.6	79.8	74.3
	3	16	128	64.2	48.9	25.1	41.1	60.5	48.0
	2	16	128	53.2	23.9	22.1	25.8	53.0	35.6
GPTQ [10]	4	16	128	82.9	86.3	58.4	66.1	80.7	74.9
	3	16	128	80.6	79.6	52.1	63.5	77.1	70.6
	2	16	128	62.7	38.9	24.6	41.0	59.9	45.4
AWQ [11]	4	16	128	82.7	86.3	59.0	65.7	80.9	74.9
	3	16	128	81.4	84.7	58.0	63.5	78.6	73.2
	2	16	128	52.2	25.5	23.1	25.6	52.3	35.7
SliM-LLM [8]	4	16	128	82.9	86.5	59.0	66.2	80.7	75.1
	3	16	128	81.6	83.1	58.5	64.7	78.4	73.3
	2	16	128	76.2	66.3	45.7	55.4	63.7	61.5
QuIP [13]	4	16	–	82.5	86.0	58.7	65.7	79.7	74.5
	3	16	–	82.3	83.3	54.9	63.9	78.4	72.5
	2	16	–	65.3	48.9	26.5	40.9	61.7	48.7
PB-LLM [12]	2	16	128	65.2	40.6	25.1	42.7	56.4	46.0
	1.7	16	128	56.5	49.9	25.8	34.9	53.1	44.1
BiLLM [15]	1.1	16	128	58.2	46.4	25.1	37.5	53.6	44.2
SmoothQuant [5]	8	8	–	82.2	86.9	60.2	66.3	80.7	75.3
	6	6	–	82.4	87.0	59.9	66.1	80.6	75.2
	4	4	–	76.9	75.8	43.5	52.9	58.9	61.6

LoRA-FT methods Except for the PTQ methods, we also provide the performance of 4-bit LLaMA3-8B with 2 different LoRA-FT quantization methods as shown in Table 7 and Table 8, including QLoRA [16] and IR-QLoRA [17]. In addition, the performance of LLaMA-7B under the same setting is summarized in Table 9. QLoRA [16] is the first LoRA-FT method that uses 4-bit NormalFloat quantization for base model weights, achieving significant memory reduction with minimal impact on model performance. Building on QLoRA, IR-QLoRA [17] introduces information calibration quantization and information elastic connection from the information inspection, resulting in high-performance adaptation with low-bit precision.

On the MMLU dataset, the most notable observation with LLaMA3-8B under LoRA-FT quantization is that low-rank fine-tuning on the Alpaca [36] dataset not only fails to compensate for the errors introduced by quantization, but actually exacerbates the degradation. Specifically, various LoRA-FT quantization methods yield worse performance for quantized LLaMA3 below 4 bits compared with their 4-bit counterparts without LoRA-FT. This is in stark contrast to similar phenomena on LLaMA and LLaMA2, where the 4-bit low-rank fine-tuned quantized versions for the front panel could even easily outperform the original FP16 counterpart on MMLU. According to our intuitive analysis, the main reason for this phenomenon is LLaMA3's strong performance due to its massive pre-scale training. This means that the performance loss due to the quantization of the original model cannot be compensated by fine-tuning on a tiny data set with low-rank parameters (which can be seen as a subset of the original model [16, 37]). Despite the significant quantization loss that cannot be compensated by fine-tuning, the 4-bit LoRA-FT quantized LLaMA3-8B significantly outperforms LLaMA-7B and LLaMA2-7B using different quantization methods. For instance, with the QLoRA method, the 4-bit LLaMA3-8B has an average accuracy of 57.0 (FP16: 64.8), exceeding the 4-bit LLaMA-7B's 38.4 (FP16: 34.6) by 18.6, and surpassing the 4-bit LLaMA2-7B's 43.9 (FP16: 45.5) by 13.1 [17, 38]. This implies that a new LoRA-FT quantization paradigm is needed in the era of LLaMA3.

A similar phenomenon occurs with the CommonSenseQA benchmark. Compared to the 4-bit counterparts without LoRA-FT, the performance of the models fine-tuned using QLoRA and IR-QLoRA also declined (e.g. QLoRA 2.8% vs. IR-QLoRA 2.4% on average). This further demonstrates the strength of using high-quality datasets in LLaMA3, as the general dataset, Alpaca, does not contribute to the model's performance in other tasks. Moreover, IR-QLoRA consistently outperforms QLoRA, due

Table 5 PPL results of post-training quantization on the LLaMA1/2-7B model

Method	#W	#A	#G	LLaMA-7B↓		LLaMA2-7B↓	
				WikiText2	C4	WikiText2	C4
FP	16	16	–	5.7	7.1	5.5	7.0
RTN	4	16	128	6.0	7.4	5.7	7.2
	3	16	128	7.0	8.6	6.7	8.4
	2	16	128	1.9×10^3	1.0×10^3	4.2×10^3	4.9×10^3
GPTQ [10]	4	16	128	6.2	–	5.7	–
	3	16	128	6.6	7.9	6.3	7.9
	2	16	128	1.5×10^2	34.6	60.5	33.7
AWQ [11]	4	16	128	5.8	–	5.6	–
	3	16	128	6.5	7.9	6.2	7.8
	2	16	128	2.6×10^5	1.9×10^5	2.2×10^5	1.75
SliM-LLM [8]	3	16	128	6.4	6.1	6.2	7.7
	2	16	128	14.6	32.9	16.0	16.0
QuIP [13]	2	16	–	29.7	33.7	39.7	31.9
DB-LLM [14]	2	16	128	7.6	9.7	7.2	–
PB-LLM [12]	2	16	128	24.6	49.7	25.4	29.8
	1.7	16	128	1.0×10^2	1.0×10^2	69.2	80.2
BiLLM [15]	1.1	16	128	35.0	39.6	32.5	40.5
SmoothQuant [5]	6	6	–	6.0	7.5	6.2	7.8
	4	4	–	22.3	32.3	83.1	77.3
OmniQuant [33]	6	6	–	6.0	7.4	5.9	7.5
	4	4	–	11.3	14.5	14.3	18.0
I-LLM [34]	6	6	–	5.8	7.3	5.7	7.3
	4	4	–	9.1	12.3	10.4	12.9
SpinQuant [35]	4	8	–	–	–	5.7	–
	4	4	–	–	–	5.9	–

Table 6 GPU memory usage, quantization time, and inference latency for PTQ methods on LLaMA2-7B and LLaMA3-8B. Latency is determined under a group size of 128. '–' denotes that the current method did not provide the real quantization kernel for the latency test

Method	#W	LLaMA2-7B			LLaMA3-8B		
		Memory (GB)	Time (min)	Speed (token/s)	Memory (GB)	Time (min)	Speed (token/s)
GPTQ [10]	4	26.4	17	159.4	40.3	19	61.2
SmoothQuant [5]	4	13.5	7	–	16.0	15	–
AWQ [11]	4	11.7	12	112.9	20.1	10	89.8
OmniQuant [33]	4	29.45	325	147.2	30.61	307	54.9

Table 7 LoRA-FT on LLaMA3-8B with Alpaca dataset (1/2)

Method	#W	MMLU↑				
		Hums.	STEM	Social	Other	Avg.
LLaMA3	16	59.0	55.3	76.0	71.5	64.8
NormalFloat	4	56.8	52.9	73.6	69.4	62.5
QLoRA [16]	4	50.3	49.3	65.8	64.2	56.7
IR-QLoRA [17]	4	52.2	49.0	66.5	63.1	57.2

Table 8 LoRA-FT on LLaMA3-8B with Alpaca dataset (2/2)

Method	#W	CommonSenseQA↑					
		PIQA	ARC-e	ARC-c	HellaSwag	Wino	Avg.
LLaMA3	16	79.9	80.1	50.4	60.2	72.8	68.6
NormalFloat	4	78.6	78.5	46.2	58.8	74.3	67.3
QLoRA [16]	4	76.6	74.8	45.0	59.4	67.0	64.5
IR-QLoRA [17]	4	76.3	74.3	45.3	59.1	69.5	64.9

Table 9 LoRA-FT on LLaMA-7B with Alpaca dataset

Method	#W	MMLU↑				
		Hums.	STEM	Social	Other	Avg.
LLaMA	16	33.3	29.8	37.8	38.0	34.6
NormalFloat	4	33.1	30.6	38.8	38.8	35.1
QLoRA [16]	4	36.1	31.9	42.0	44.5	38.4
IR-QLoRA [17]	4	38.6	34.6	45.2	45.5	40.8

Table 10 GPU memory usage, training time, and inference latency for LoRA-FT Methods on LLaMA models

Method	#W	LLaMA2-7B			LLaMA3-8B		
		Memory (GB)	Time (hour)	Speed (token/s)	Memory (GB)	Time (hour)	Speed (token/s)
LLaMA	16	–	–	95.6	–	–	79.7
QLoRA [16]	4	7.2	15.3	88.6	13.2	16.1	72.8
IR-QLoRA [17]	4	7.4	15.4	83.1	14.2	16.3	69.2

to its incorporation of information calibration quantization and information elastic connection through information inspection. These mechanisms allow IR-QLoRA to achieve high-performance adaptation even at low-bit accuracy.

For practical deployment, we recorded the GPU memory usage and training time consumption for different sizes of the LLaMA model, as shown in Table 10. It demonstrates that both QLoRA and IR-QLoRA achieve significant memory efficiency, dramatically reducing the required memory footprint compared to the original LLaMA model. Nevertheless, both QLoRA and IR-QLoRA introduce inference bottlenecks primarily due to the dequantization process, which results in an increase in inference latency. The trade-off between the reduced memory footprint and the slight increase in latency is often acceptable for deployment in resource-constrained environments where memory is the limiting factor. Further optimizations, such as hardware-specific tuning and algorithmic improvements, could mitigate this bottleneck and improve overall inference speed.

4 Track3: multi-modal large language model quantization

For the MLLM model, we follow a common practice by conducting post-training quantization on the LLaMA3 part [11, 39]. As shown in Table 11 and Table 12, we compare the ultra-low bit-width performance of LLaVA-Next-8B under GPTQ and AWQ in six visual-language benchmarks.

We initially evaluate the pure language capabilities of LLaVA-Next-8B, as illustrated in Table 11. The fp16 precision PPL metrics of the LLaMA3 model, after being fine-tuned for visual tasks, worsened across three datasets compared to its performance on language tasks. This also suggests that when fine-tuned for visual-language tasks, the introduction of image tokens leads to a partial loss and forgetting of LLaMA3's inherent language abilities. The language capabilities of multi-modal LLMs (MLLMs) show a loss trend consistent with pure LLMs under low-bit quantization. Subsequently, we tested the quantized LLaMA3 within the MLLM model on visual QA tasks. As shown in Table 12, under several advanced PTQ methods, the 4-bit MLLM exhibits a loss of less than 2% on multi-modal benchmarks, efficiently performing visual-language tasks with reduced model size.

At 3 bits, the performance loss ranges from 5% to 20%, with the highest loss, 20.75%, occurring on the MME cognition task. Notably, regardless of GPTQ or AWQ, we observe that the 2-bit LLaVA-Next-8B completely collapses in the six multi-modal QA tasks, with scores dropping to zero. Although SliM-LLM mitigates the performance collapse of LLaVA-Next-8B at 2 bits, it still shows a large performance degradation.

Table 11 Evaluation results of post-training quantization on LLaVA-Next-8B (1/2)

Method	#W	#G	PPL↓		
			WikiText2	C4	PTB
LLaVA-Next (LLaMA3-8B)	16	–	9.5	14.8	16.3
RTN	4	128	10.2	15.6	17.1
	3	128	23.2	26.5	36.1
	2	128	1.5×10^5	5.7×10^5	8.6×10^5
GPTQ [10]	4	128	9.5	14.8	17.1
	3	128	13.0	19.5	28.4
	2	128	83.7	3.1×10^3	2.0×10^2
AWQ [11]	4	128	9.9	15.3	16.9
	3	128	11.7	17.9	20.2
	2	128	1.6×10^6	2.0×10^6	2.2×10^6
SliM-LLM [8]	4	128	9.5	15.0	16.5
	3	128	11.1	16.8	18.5
	2	128	46.3	2.0×10^2	1.8×10^2

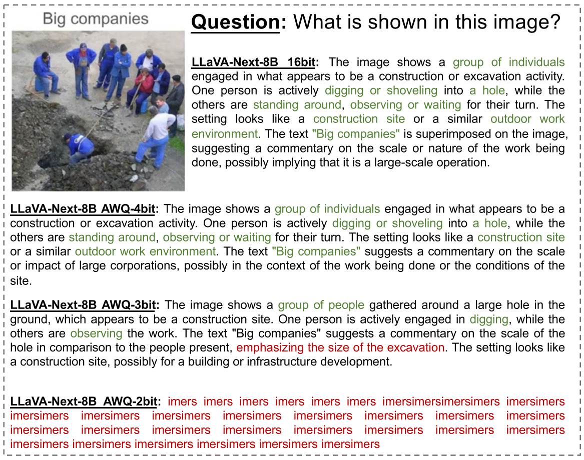
Table 12 Evaluation results of post-training quantization on LLaVA-Next-8B (2/2). N denots that the answer score is 0, or the outputs are unexpected characters

Method	#W	#G	Multimodel Tasks↑					
			AI2D	Chart QA	Doc VQA	MM Bench	MME Cognition	MME Perception
LLaVA-Next (LLaMA3-8B)	16	–	71.7	69.2	78.2	72.2	376.8	1588.3
RTN	4	128	70.4	65.7	77.0	70.1	304.4	1550.9
	3	128	58.7	63.2	69.7	64.3	247.2	1526.2
	2	128	N	N	N	N	N	N
GPTQ [10]	4	128	70.7	67.4	77.4	71.0	331.6	1563.4
	3	128	66.2	65.1	75.6	67.4	290.1	1541.7
	2	128	N	N	N	N	N	N
AWQ [11]	4	128	70.6	68.0	77.2	71.1	325.7	1562.7
	3	128	67.7	65.4	74.4	68.0	298.6	1541.7
	2	128	N	N	N	N	N	N
SliM-LLM [8]	4	128	70.6	68.0	77.2	71.1	342.5	1563.9
	3	128	68.2	67.5	74.8	68.9	321.0	1554.3
	2	128	57.2	49.3	60.6	60.9	282.1	1515.8

In Figs. 2-6, we show some real visual-language results of LLaVA-Next-8B under different bit widths quantized with AWQ. The 4-bit quantized model can still generate precise descriptions in images, while the 3-bit model excels in overall multi-modal understanding but suffers from a loss of detail. For example, in Fig. 2, the descriptions of people and actions in images by the 4-bit and 3-bit models are largely consistent with those of the 16-bit model. Additionally, the 4-bit model aligns with the 16-bit model in abstract semantic understanding of "big companies"; however, the 3-bit model misinterprets "big companies" as a descriptor of hole size. Further, under 2-bit quantization, the model struggles to produce reasonable answers, resulting in repetitive character responses. This contrasts with the performance of 2-bit models in pure language tasks, where previous studies [8, 11, 15] have shown that 2-bit quantized models can still generate logically coherent sentences. However, in MLLM tasks, the 2-bit model fails to produce results close to expectations. This further indicates that the advanced PTQ method in the current LLM does not effectively perform equally well in the ultra-low bit MLLM models, which also inspires future work to propose better quantization solutions for this huge challenge in MLLM.

5 Conclusion

The recently released LLaMA3 family has quickly become the most powerful LLM backbones, attracting significant interest from LLM and MLLM researchers. Building on this momentum, our study aims to thoroughly



Question: What is shown in this image?

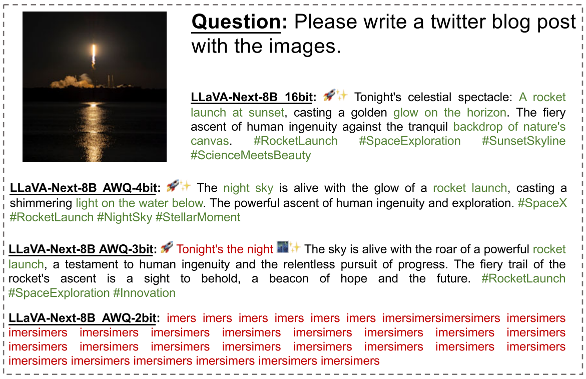
LLaVA-Next-8B 16bit: The image shows a group of individuals engaged in what appears to be a construction or excavation activity. One person is actively digging or shoveling into a hole, while the others are standing around, observing or waiting for their turn. The setting looks like a construction site or a similar outdoor work environment. The text "Big companies" is superimposed on the image, suggesting a commentary on the scale or nature of the work being done, possibly implying that it is a large-scale operation.

LLaVA-Next-8B AWQ-4bit: The image shows a group of individuals engaged in what appears to be a construction or excavation activity. One person is actively digging or shoveling into a hole, while the others are standing around, observing or waiting for their turn. The setting looks like a construction site or a similar outdoor work environment. The text "Big companies" suggests a commentary on the scale or impact of large corporations, possibly in the context of the work being done or the conditions of the site.

LLaVA-Next-8B AWQ-3bit: The image shows a group of people gathered around a large hole in the ground, which appears to be a construction site. One person is actively engaged in digging, while the others are observing the work. The text "Big companies" suggests a commentary on the scale of the hole in comparison to the people present, emphasizing the size of the excavation. The setting looks like a construction site, possibly for a building or infrastructure development.

LLaVA-Next-8B AWQ-2bit: imers imers imers imers imers imers imersimersimersimers imersimers

Figure 2 The VQA results of LLaVA-Next-8B for different quantization bit widths (1/5)



Question: Please write a twitter blog post with the images.

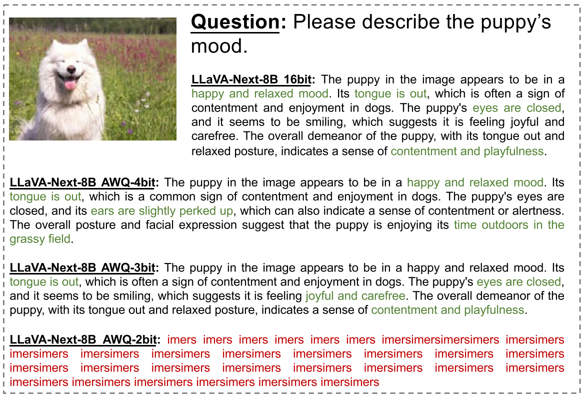
LLaVA-Next-8B 16bit: 🚀✨ Tonight's celestial spectacle: A rocket launch at sunset, casting a golden glow on the horizon. The fiery ascent of human ingenuity against the tranquil backdrop of nature's canvas. #RocketLaunch #SpaceExploration #SunsetSkyline #ScienceMeetsBeauty

LLaVA-Next-8B AWQ-4bit: 🚀✨ The night sky is alive with the glow of a rocket launch, casting a shimmering light on the water below. The powerful ascent of human ingenuity and exploration. #SpaceX #RocketLaunch #NightSky #StellarMoment

LLaVA-Next-8B AWQ-3bit: 🚀 Tonight's the night 🌃✨ The sky is alive with the roar of a powerful rocket launch, a testament to human ingenuity and the relentless pursuit of progress. The fiery trail of the rocket's ascent is a sight to behold, a beacon of hope and the future. #RocketLaunch #SpaceExploration #Innovation

LLaVA-Next-8B AWQ-2bit: imers imers imers imers imers imers imersimersimersimers imersimers

Figure 3 The VQA results of LLaVA-Next-8B for different quantization bit widths (2/5)



Question: Please describe the puppy's mood.

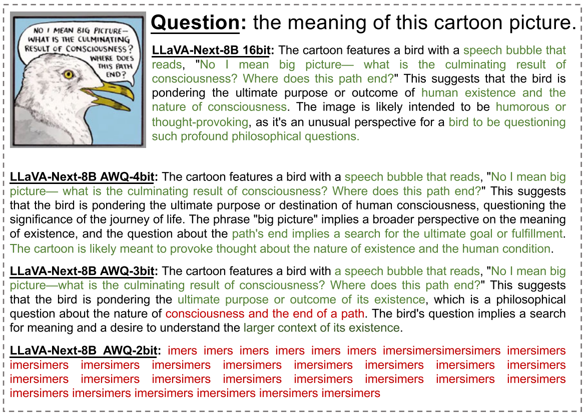
LLaVA-Next-8B 16bit: The puppy in the image appears to be in a happy and relaxed mood. Its tongue is out, which is often a sign of contentment and enjoyment in dogs. The puppy's eyes are closed, and it seems to be smiling, which suggests it is feeling joyful and carefree. The overall demeanor of the puppy, with its tongue out and relaxed posture, indicates a sense of contentment and playfulness.

LLaVA-Next-8B AWQ-4bit: The puppy in the image appears to be in a happy and relaxed mood. Its tongue is out, which is a common sign of contentment and enjoyment in dogs. The puppy's eyes are closed, and its ears are slightly perked up, which can also indicate a sense of contentment or alertness. The overall posture and facial expression suggest that the puppy is enjoying its time outdoors in the grassy field.

LLaVA-Next-8B AWQ-3bit: The puppy in the image appears to be in a happy and relaxed mood. Its tongue is out, which is often a sign of contentment and enjoyment in dogs. The puppy's eyes are closed, and it seems to be smiling, which suggests it is feeling joyful and carefree. The overall demeanor of the puppy, with its tongue out and relaxed posture, indicates a sense of contentment and playfulness.

LLaVA-Next-8B AWQ-2bit: imers imers imers imers imers imers imersimersimersimers imersimers

Figure 4 The VQA results of LLaVA-Next-8B for different quantization bit widths (3/5)


NO I MEAN BIG PICTURE— WHAT IS THE CULMINATING RESULT OF CONSCIOUSNESS? WHERE DOES THIS PATH END?

Question: the meaning of this cartoon picture.

LLaVA-Next-8B 16bit: The cartoon features a bird with a speech bubble that reads, "No I mean big picture— what is the culminating result of consciousness? Where does this path end?" This suggests that the bird is pondering the ultimate purpose or outcome of human existence and the nature of consciousness. The image is likely intended to be humorous or thought-provoking, as it's an unusual perspective for a bird to be questioning such profound philosophical questions.

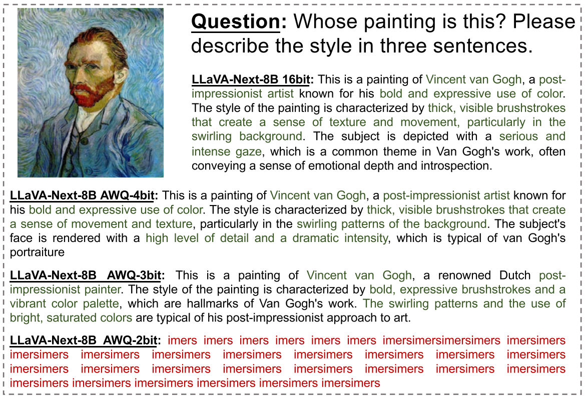
LLaVA-Next-8B AWQ-4bit: The cartoon features a bird with a speech bubble that reads, "No I mean big picture— what is the culminating result of consciousness? Where does this path end?" This suggests that the bird is pondering the ultimate purpose or destination of human consciousness, questioning the significance of the journey of life. The phrase "big picture" implies a broader perspective on the meaning of existence, and the question about the path's end implies a search for the ultimate goal or fulfillment. The cartoon is likely meant to provoke thought about the nature of existence and the human condition.

LLaVA-Next-8B AWQ-3bit: The cartoon features a bird with a speech bubble that reads, "No I mean big picture—what is the culminating result of consciousness? Where does this path end?" This suggests that the bird is pondering the ultimate purpose or outcome of its existence, which is a philosophical question about the nature of consciousness and the end of a path. The bird's question implies a search for meaning and a desire to understand the larger context of its existence.

LLaVA-Next-8B AWQ-2bit: imers imers imers imers imers imers imersimersimersimers imersimers

Figure 5 The VQA results of LLaVA-Next-8B for different quantization bit widths (4/5)

evaluate the performance of LLaMA3 for various low-bit quantization techniques, including post-training quantization and LoRA fine-tuning quantization for LLMs and MLLMs. Our goal is to assess the limits of its capabilities in resource-constrained scenarios using existing LLM and MLLM quantization techniques. We found that while LLaMA3 still demonstrates superior performance after quantization, the performance degradation associated with quantization is significant and can lead to larger declines. This decrease is mainly due to the fact that the powerful pre-training process allows LLaMA3 to learn more information to a similar extent as the previous LLaMA and LLaMA2, and its more sophisticated internal knowledge structure makes the effect of the quantization perturbation less obvious. The collapse of the ultra-low 2 bits also confirms that the quantized LLaMA3 backbone in MLLM exacerbates the performance loss caused by quantization when processing complex visual information. This discovery highlights the potential challenges of deploying LLaMA3 in resource-constrained environments and underscores the ample room for growth and improvement in low-bit quantization. The empirical results of our research are expected to be valuable in the development of future LLM quantization techniques, especially in narrowing the performance gap with the original models. By addressing the performance degradation caused by low-bit quantization, we anticipate that subsequent quantization paradigms will allow LLMs to achieve stronger capabilities at a lower computational cost, ultimately driving the progress of generative artificial intelligence, as represented by LLMs and MLLMs, to new heights.



Question: Whose painting is this? Please describe the style in three sentences.

LLaVA-Next-8B 16bit: This is a painting of Vincent van Gogh, a post-impressionist artist known for his bold and expressive use of color. The style of the painting is characterized by thick, visible brushstrokes that create a sense of texture and movement, particularly in the swirling background. The subject is depicted with a serious and intense gaze, which is a common theme in Van Gogh's work, often conveying a sense of emotional depth and introspection.

LLaVA-Next-8B AWQ-4bit: This is a painting of Vincent van Gogh, a post-impressionist artist known for his bold and expressive use of color. The style is characterized by thick, visible brushstrokes that create a sense of movement and texture, particularly in the swirling patterns of the background. The subject's face is rendered with a high level of detail and a dramatic intensity, which is typical of van Gogh's portraiture

LLaVA-Next-8B AWQ-3bit: This is a painting of Vincent van Gogh, a renowned Dutch post-impressionist painter. The style of the painting is characterized by bold, expressive brushstrokes and a vibrant color palette, which are hallmarks of Van Gogh's work. The swirling patterns and the use of bright, saturated colors are typical of his post-impressionist approach to art.

LLaVA-Next-8B AWQ-2bit: imers imers imers imers imers imers imersimersimersimers imersimers

Figure 6 The VQA results of LLaVA-Next-8B for different quantization bit widths (5/5)

Abbreviations

CV, computer vision; LLMs, large language models; LoRA-FT, LoRA-FineTuning; MLLMs, multi-modal large language model; NLU, natural language understanding; PTQ, post-training quantization.

Author contributions

All authors contributed to the study's conception and design. WH, XZ, XM, HQ, CL, and HC performed data collection and analysis. HQ wrote the first draft of the manuscript, and all authors commented on previous versions. All authors read and approved the final manuscript. We propose the original idea together.

Funding

This work was supported by the National Science and Technology Major Project (2021ZD0110503), the Swiss National Science Foundation (SNSF) project 200021E_219943 Neuromorphic Attention Models for Event Data (NAMED), the Baidu Scholarship, and the National Natural Science Foundation of China (Nos. 62306025 and 92367204).

Data availability

Availability of data and material: The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request. Our project is released on GitHub and quantized LLaMA3 models are released in HuggingFace.

Declarations

Competing interests

All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Author details

[1]Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, 999077, China. [2]School of Computer Science and Engineering, Beihang University, Xueyuan Road, Beijing, 100191, China. [3]Department of Information Technology and Electrical Engineering, ETH Zurich, Sternwartstrasse 7, Zürich, Switzerland.

Received: 27 August 2024 Revised: 16 December 2024
Accepted: 17 December 2024 Published online: 30 December 2024

References

1. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., et al. (2023). LLaMA: open and efficient foundation language models. arXiv preprint. arXiv:2302.13971.
2. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, et al. (Eds.), *Proceedings of the 31st international conference on neural information processing systems* (pp. 5998–6008). Red Hook: Curran Associates.
3. Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al. (2024). The llama 3 herd of models. arXiv preprint. arXiv:2407.21783.
4. Liu, H., Li, C., Wu, Q., & Lee, Y.J. (2023). Visual instruction tuning. In A. Oh, T. Neumann, A. Globerson, et al. (Eds.), *Proceedings of the 37th international conference on neural information processing systems* (pp. 1–25). Red Hook: Curran Associates.
5. Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., & Han, S. (2023). SmoothQuant: accurate and efficient post-training quantization for large language models. In *Proceedings of the international conference on machine learning* (pp. 38087–38099). Retrieved November 10, 2024, from https://proceedings.mlr.press/v202/xiao23c.html.
6. Qin, H., Zhang, Y., Ding, Y., Liu, X., Danelljan, M., Yu, F., et al. (2023). QuantSR: accurate low-bit quantization for efficient image super-resolution. In A. Oh, T. Neumann, A. Globerson, et al. (Eds.), *Proceedings of the 37th international conference on neural information processing systems* (pp. 1–11). Red Hook: Curran Associates.
7. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., et al. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2704–2713). Piscataway: IEEE.
8. Huang, W., Qin, H., Liu, Y., Li, Y., Liu, X., Benini, L., et al. (2024). SliM-LLM: Salience-driven mixed-precision quantization for large language models. arXiv preprint. arXiv:2405.14917.
9. Nagel, M., Amjad, R.A., Van Baalen, M., Louizos, C., & Blankevoort, T. (2020). Up or down? Adaptive rounding for post-training quantization. In *International conference on machine learning* (pp. 7197–7206). PMLR.
10. Frantar, E., Ashkboos, S., Hoefler, T., & Alistarh, D. (2022). GPTQ: Accurate post-training quantization for generative pre-trained transformers. arXiv preprint. arXiv:2210.17323.
11. Lin, J., Tang, J., Tang, H., Yang, S., Chen, W.-M., Xiao, G., et al. (2024). AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. In P. B. Gibbons, G. Pekhimenko, & C. de Sa (Eds.), *Proceedings of machine learning and systems* (pp. 87–100). Retrieved November 10, 2024, from https://proceedings.mlsys.org/paper_files/paper/2024/hash/42a452cbafa9dd64e9ba4aa95cc1ef21-Abstract-Conference.html.
12. Shang, Y., Yuan, Z., Wu, Q., & Dong, Z. (2024). PB-LLM: partially binarized large language models. In *Proceedings of the 12th international conference on learning representations* (pp. 1–14). Retrieved November 10, 2024, from https://openreview.net/forum?id=BifeBRhikU.
13. Chee, J., Cai, Y., Kuleshov, V., & De Sa, C. (2024). QuIP: 2-bit quantization of large language models with guarantees. In A. Oh, T. Neumann, A. Globerson, et al. (Eds.), *Proceedings of the 37th international conference on neural information processing systems* (pp. 1–34). Red Hook: Curran Associates.
14. Chen, H., Lv, C., Ding, L., Qin, H., Zhou, X., Ding, Y., et al. (2024). DB-LLM: accurate dual-binarization for efficient LLMs. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *Findings of the association for computational linguistics* (pp. 8719–8730). Stroudsburg: ACL.
15. Huang, W., Liu, Y., Qin, H., Li, Y., Zhang, S., Liu, X., et al. (2024). BiLLM: pushing the limit of post-training quantization for LLMs. In *Proceedings of the 41st international conference on machine learning* (pp. 1–20). Retrieved November 10, 2024, from https://openreview.net/forum?id=qOI2WWOqFg.
16. Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2024). QLoRA: efficient finetuning of quantized LLMs. In A. Oh, T. Neumann, A. Globerson, et al. (Eds.), *Proceedings of the 37th international conference on neural information processing systems* (pp. 1–28). Red Hook: Curran Associates.
17. Qin, H., Ma, X., Zheng, X., Li, X., Zhang, Y., Liu, S., et al. (2024). Accurate lora-finetuning quantization of LLMs via information retention. In *Proceedings of the 41st international conference on machine learning* (pp. 1–19). Retrieved November 10, 2024, from https://openreview.net/forum?id=jQ92egz5Ym.
18. Merity, S., Xiong, C., Bradbury, J., & Socher, R. (2016). Pointer sentinel mixture models. In *Proceedings of the 5th international conference on learning representations* (pp. 1–15). Retrieved November 10, 2024, from https://openreview.net/forum?id=Byj72udxe.

19. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., et al. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, *21*(1), 5485–5551.
20. Marcus, M., Grace Kim, P., Marcinkiewicz, M. A., MacIntyre, R., Bies, A., Ferguson, M., et al. (1994). The Penn Treebank: annotating predicate argument structure. In *Proceedings of human language technology workshop* (pp. 114–119). San Francisco: Morgan Kaufmann.
21. Bisk, Y., Zellers, R., Le Bras, R., Gao, J., & Choi, Y. (2020). PIQA: reasoning about physical commonsense in natural language. In *Proceedings of the 34th AAAI conference on artificial intelligence* (pp. 7432–7439). Palo Alto: AAAI Press.
22. Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., et al. (2018). Think you have solved question answering? Try ARC-DA, the AI2 reasoning challenge. arXiv preprint. arXiv:1803.05457.
23. Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., & Choi, Y. (2019). HellaSwag: can a machine really finish your sentence? In A. Korhonen, D. R. Traum, & L. M'arquez (Eds.), *Proceedings of the 57th conference of the association for computational linguistics* (pp. 4791–4800). Stroudsburg: ACL.
24. Sakaguchi, K., Le Bras, R., Bhagavatula, C., & Choi, Y. (2021). Winogrande: an adversarial winograd schema challenge at scale. *Communications of the ACM*, *64*(9), 99–106.
25. Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., & Steinhardt, J. (2021). Measuring massive multitask language understanding. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
26. Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., & Farhadi, A. (2016). A diagram is worth a dozen images.
27. Masry, A., Long, D.X., Tan, J.Q., Joty, S., & Hoque, E. (2022). A benchmark for question answering about charts with visual and logical reasoning. arXiv preprint. arXiv:2203.10244.
28. Mathew, M., Karatzas, D., & Jawahar, C. V. (2021). Docvqa: a dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 2200–2209).
29. Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., & Ji, R. (2024). Mme: a comprehensive evaluation benchmark for multimodal large language models.
30. Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., et al. (2025). Mmbench: is your multi-modal model an all-around player? In *European conference on computer vision* (pp. 216–233). Berlin: Springer.
31. Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., et al. (2021). Measuring massive multitask language understanding. In *Proceedings of the 9th international conference on learning representations* (pp. 1–27). Retrieved November 10, 2024, from https://openreview.net/forum?id=d7KBjmI3GmQ.
32. Liu, Z., Oguz, B., Zhao, C., Chang, E., Stock, P., Mehdad, Y., et al. (2024). LLM-QAT: data-free quantization aware training for large language models. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *Findings of the association for computational linguistics* (pp. 467–484). Stroudsburg: ACL.
33. Shao, W., Chen, M., Zhang, Z., Xu, P., Zhao, L., Li, Z., Zhang, K., Gao, P., Qiao, Y., & Luo, P. (2023). Omniquant: Omnidirectionally calibrated quantization for large language models. arXiv preprint. arXiv:2308.13137.
34. Hu, X., Cheng, Y., Yang, D., Yuan, Z., Yu, J., Xu, C., & Zhou, S. (2024). I-llm: Efficient integer-only inference for fully-quantized low-bit large language models. arXiv preprint. arXiv:2405.17849.
35. Liu, Z., Zhao, C., Fedorov, I., Soran, B., Choudhary, D., Krishnamoorthi, R., Chandra, V., Tian, Y., & Blankevoort, T. (2024). Spinquant–llm quantization with learned rotations. arXiv preprint. arXiv:2405.16406.
36. Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., et al. (2023). Stanford alpaca: an instruction-following llama model. Retrieved November 10, 2024, from https://github.com/tatsu-lab/stanford_alpaca.
37. Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. (2022). LoRA: low-rank adaptation of large language models. In *Proceedings of the 10th international conference on learning representations* (pp. 1–13). Retrieved November 10, 2024, from https://openreview.net/forum?id=nZeVKeeFYf9.
38. Xu, Y., Xie, L., Gu, X., Chen, X., Chang, H., Zhang, H., et al. (2024). QA-LoRA: quantization-aware low-rank adaptation of large language models. In *Proceedings of the 12th international conference on learning representations* (pp. 1–18). Retrieved November 10, 2024, from https://openreview.net/forum?id=WvFoJccpo8.
39. Lin, J., Yin, H., Ping, W., Molchanov, P., Shoeybi, M., & Song, H. (2024). VILA: on pre-training for visual language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 26689–26699). Piscataway: IEEE.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com