Exploring the Trade-Offs: Quantization Methods, Task Difficulty, and Model Size in Large Language Models From Edge to Giant

Jemin Lee[1], **Sihyeong Park**[2], **Jinse Kwon**[1], **Jihun Oh**[3] and **Yongin Kwon**[1*]
[1]Electronics and Telecommunications Research Institute
[2]Korea Electronics Technology Institute
[3]Neubla
{leejaymin,kwonse,yongin.kwon}@etri.re.kr
sihyeong@keti.re.kr, oj9040@gmail.com

Abstract

Quantization has gained attention as a promising solution for the cost-effective deployment of large and small language models. However, most prior work has been limited to perplexity or basic knowledge tasks and lacks a comprehensive evaluation of recent models like Llama-3.3. In this paper, we conduct a comprehensive evaluation of instruction-tuned models spanning 1B to 405B parameters, applying four quantization methods across 13 datasets. Our findings reveal that (1) quantized models generally surpass smaller FP16 baselines, yet they often struggle with instruction-following and hallucination detection; (2) FP8 consistently emerges as the most robust option across tasks, and AWQ tends to outperform GPTQ in weight-only quantization; (3) smaller models can suffer severe accuracy drops at 4-bit quantization, while 70B-scale models maintain stable performance; (4) notably, *hard* tasks do not always experience the largest accuracy losses, indicating that quantization magnifies a model's inherent weaknesses rather than simply correlating with task difficulty; and (5) an LLM-based judge (MT-Bench) highlights significant performance declines in Coding and STEM tasks, though it occasionally reports improvements in reasoning.

1 Introduction

Despite the remarkable performance of recent large and small language models (LLMs and SLMs), deploying them in resource-constrained environments remains challenging. Even models like Llama-3.3-70B (released in December 2024) and Llama-3.2-1B (released in September 2024) still involve billions of parameters, making them costly to run in both server and mobile-edge scenarios. Low-bit quantization has emerged as a popular solution to reduce the memory and computational overhead of these models. In particular, Post-Training Quantization (PTQ) [Frantar *et al.*, 2022; Lin *et al.*, 2024; Xiao *et al.*, 2023; Micikevicius *et al.*, 2022] is widely adopted, as Quantization Aware Training (QAT) often requires extensive retraining [Zhu *et al.*, 2023; Wan *et al.*, 2023].

However, existing research on quantization has largely relied on perplexity-based metrics [Frantar *et al.*, 2022; Lin *et al.*, 2024; Xiao *et al.*, 2023] and older benchmarks (e.g., ARC, HellaSwag, Winogrande, MMLU) [Yao *et al.*, 2023; Dettmers and Zettlemoyer, 2023; Liu *et al.*, 2023; Jin *et al.*, 2024; Li *et al.*, 2024; Dutta *et al.*, 2024], which have become too easy for current models and risk data contamination in recently trained LLMs. Moreover, more recent architectures like Llama-3.3, Llama-3.2, and Llama-3.1 have not been thoroughly investigated. This gap includes extreme scales, from 1B to over 405B parameters, and omits detailed category-level analysis using LLM-as-judge evaluation methods. Additionally, there has been limited manual inspection to refine and confirm evaluation results [Li *et al.*, 2024; Dutta *et al.*, 2024].

In this paper, we present a comprehensive evaluation of how quantization affects instruction-tuned LLMs. In particular, we aim to address the following research questions (RQs): **(RQ1)** Do quantized LLMs outperform smaller original models in most benchmarks, and how do they perform across diverse architectures and small language models (SLMs)? **(RQ2)** How do different quantization methods influence performance across a broad range of tasks, and are there significant differences in how specific approaches (e.g., GPTQ, AWQ, SmoothQuant, FP8) affect task accuracy? **(RQ3)** In what ways do model size and architecture affect the accuracy of quantized models? **(RQ4)** Does higher task difficulty necessarily correlate with greater accuracy degradation under quantization? and **(RQ5)** How does quantization impact the free-form conversation quality of LLMs when evaluated using the MT-Bench framework, which relies on LLMs as judges?

We perform our evaluation in a multi-cluster GPU environment, as illustrated in Figure 1. This setup consists of four servers, each with a distinct GPU configuration, and ensures consistent measurement conditions across all experiments (details in Appendix D).

Our study applies four quantization methods—GPTQ [Frantar *et al.*, 2022], AWQ [Lin *et al.*, 2024], SmoothQuant [Xiao *et al.*, 2023], and FP8 [Micikevicius *et al.*, 2022]—to instruction-tuned models ranging from 1B to 405B param-

*Corresponding author
Appendix available at: https://arxiv.org/abs/2409.11055

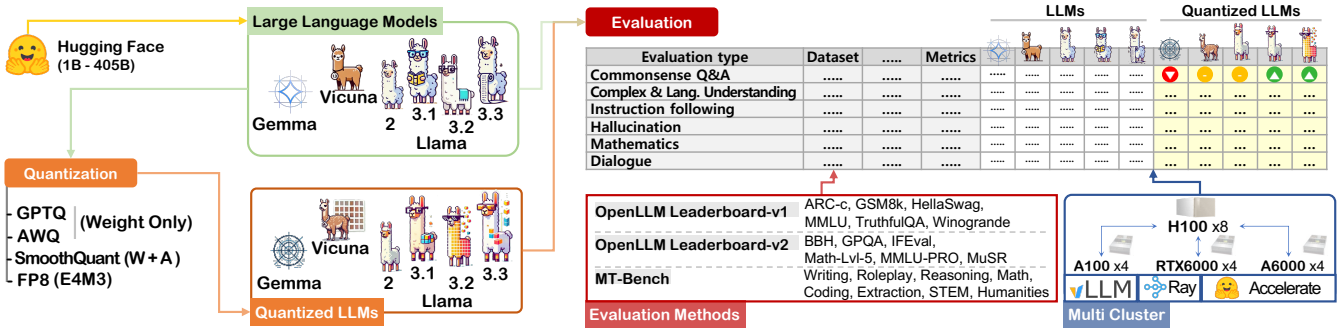

Figure 1: Overview of the evaluation pipeline for quantized LLMs, using a multi-node cluster setup to ensure fast and reliable assessments across multiple benchmarks.

eters, including Vicuna [Zheng *et al.*, 2023], Gemma [Team *et al.*, 2024], and the Llama family [Dubey *et al.*, 2024]. We evaluate these models on 13 datasets covering six task categories: commonsense Q&A, complex knowledge and language understanding, instruction following, hallucination detection, mathematics, and dialogue. Further dataset details are provided in Appendix A (Table 3).

To compare our results with ongoing community efforts, we synchronize our benchmarks with Huggingface OpenLLM Leaderboard-v1 (covering April 2023 to June 2024) and OpenLLM Leaderboard-v2 (launched on June 26, 2024). However, both versions currently provide only limited data on quantized models, highlighting the need for our comprehensive evaluation. Our key findings are as follows:

- Quantized LLMs generally perform better than smaller models on most benchmarks and maintain their advantage across different architectures, showing significant improvements in both large and small language models. However, they still struggle with instruction-following (IFEval) and detecting hallucinations (TruthfulQA).
- FP8 is the most reliable method for all model sizes and tasks, especially for LLMs with 405B parameters, where SmoothQuant encounters problems. AWQ usually performs better than GPTQ in weight-only quantization, and hardware support makes FP8 even more advantageous.
- In smaller LLMs, using 4-bit quantization can lead to significant accuracy drops, especially with GPTQ. However, 70B models usually maintain good performance when quantized to 4 bits. While model size is the main factor affecting quantization difficulty, differences in LLM architecture within the same parameter size can also affect accuracy. Nevertheless, AWQ consistently outperforms GPTQ across different tasks and model types.
- Difficult tasks do not always have the biggest accuracy drops when quantized. The impact depends on the model design and the quantization method used, causing some *hard* tasks to remain stable while some *easy* tasks see bigger decreases. Overall, quantization highlights a model's existing weaknesses, especially in commonsense, logical, or mathematical reasoning.
- Quantization greatly reduces performance in Coding and STEM tasks, although it sometimes improves reasoning accuracy. Additionally, GPT4-based evaluators can sometimes incorrectly judge wrong answers as correct.

2 Related Work

Quantization for LLMs. There are two main types of quantization methods for LLMs: post-training quantization (PTQ) and quantization-aware training (QAT). Due to the size and training complexity of LLMs, QAT is challenging to apply, and as a result, only limited research has been conducted in this area. Consequently, the majority of quantization research for LLMs has focused on PTQ approaches [Zhu *et al.*, 2023; Wan *et al.*, 2023].

LLM.int8() [Dettmers *et al.*, 2022] is a post-training quantization method that uses 8-bit weights and activations to reduce the memory footprint of large models while maintaining performance. This dynamically adapts to ensure sensitive components of the computation retain higher precision when needed. GPTQ [Frantar *et al.*, 2022] is a layer-wise quantization that uses inverse Hessian information to reduce the number of bits per weight while maintaining low accuracy loss. AWQ [Lin *et al.*, 2024] proposed that preserving a small portion of important weights is a key part of reducing quantization errors. As part of an activation-aware strategy, AWQ focused on channels with larger activation magnitudes and used per-channel scaling. SmoothQuant [Xiao *et al.*, 2023] is a method that smooths activation outliers before quantization, improving robustness in large-scale models and enabling more effective 8-bit quantization. Outlier Suppression+ [Wei *et al.*, 2023] reduces the impact of extreme outliers in activations, allowing for more efficient quantization by normalizing problematic values without degrading model accuracy. QLoRA [Dettmers *et al.*, 2024] combines low-rank adaptation with quantization to achieve efficient fine-tuning of large models while minimizing computational costs and memory usage.

However, these quantization algorithm works have been evaluated only on basic datasets such as perplexity, ARC, and MMLU, which were released 2-3 years ago, and they do not sufficiently take into account the recent advancements in LLMs and SLMs. Therefore, for a safe application of quantization in LLM services, a more comprehensive performance analysis is necessary.

Evaluating LLMs. Several studies have explored the effects

Model	Method	W/A	Storage (GB)	OpenLLM Leaderboard-v1 ↑ ARC-c (25-shot) acc_norm	GSM8k (5-shot) acc	HellaSwag (10-shot) acc_norm	MMLU (5-shot) acc	TruthfulQA (0-shot) mc2	Winogrande (5-shot) acc	Avg.
Llama-2-7B-Chat	FP16	16 / 16	14	53.16	21.91	78.92	47.24	45.32	72.13	53.11
	GPTQ*	4 / 16	3.5	51.28 (↓1.88)	13.87 (↓8.04)	72.17 (↓6.75)	43.10 (↓4.14)	44.12 (↓1.20)	71.27 (↓0.86)	49.30 (↓3.81)
	AWQ	4 / 16	3.5	52.47 (↓0.69)	19.63 (↓2.28)	78.13 (↓0.79)	45.34 (↓1.90)	44.28 (↓1.04)	71.19 (↓0.94)	51.84 (↓1.27)
Llama-2-13B-Chat	FP16	16 / 16	26	58.87	35.55	82.45	53.55	43.95	75.29	58.28
	GPTQ*	4 / 16	6.5	57.51 (↓1.36)	32.22 (↓3.33)	81.35 (↓1.10)	52.36 (↓1.19)	41.74 (↓2.21)	75.76 (↑0.47)	56.82 (↓1.46)
	AWQ	4 / 16	6.5	57.94 (↓0.93)	34.79 (↓0.76)	81.58 (↓0.87)	53.76 (↑0.21)	43.64 (↓0.31)	74.90 (↓0.39)	57.77 (↓0.51)
Llama-2-70B-Chat	FP16	16 / 16	140	66.30	50.64	85.61	63.18	52.76	80.50	66.50
	GPTQ*	4 / 16	35	62.88 (↓3.42)	50.27 (↓0.37)	84.98 (↓0.63)	61.57 (↓1.61)	51.13 (↓1.63)	79.32 (↓1.18)	65.03 (↓1.47)
	AWQ	4 / 16	35	65.27 (↓1.03)	48.14 (↓2.50)	85.29 (↓0.32)	62.65 (↓0.53)	52.75 (↓0.01)	79.87 (↓0.63)	65.66 (↓0.84)
Llama-3.1-8B-it	FP16	16 / 16	16	60.24	76.65	80.21	68.10	54.03	76.16	69.23
	FP8	8 / 8	8	61.52 (↑1.28)	74.75 (↓1.90)	80.12 (↓0.09)	68.52 (↑0.42)	53.81 (↓0.22)	77.43 (↑1.27)	69.36 (↑0.13)
	GPTQ*	4 / 16	4	61.43 (↑1.19)	72.33 (↓4.32)	78.36 (↓1.85)	66.85 (↓1.25)	53.60 (↓0.43)	75.22 (↓0.94)	67.97 (↓1.26)
	GPTQ**	4 / 16	4	59.81 (↓0.43)	69.98 (↓6.67)	78.53 (↓1.68)	66.07 (↓2.03)	50.45 (↓3.58)	76.64 (↑0.48)	66.91 (↓2.32)
	GPTQ**	8 / 16	8	61.01 (↑0.77)	75.81 (↓0.84)	80.27 (↓0.06)	68.21 (↑0.11)	54.03 (0.00)	77.19 (↑1.03)	69.42 (↑0.19)
	SmoothQuant	8 / 8	8	60.75 (↑0.51)	76.12 (↓0.53)	80.08 (↓0.13)	68.22 (↑0.12)	53.85 (↓0.18)	77.11 (↑0.95)	69.36 (↑0.13)
	AWQ	4 / 16	4	58.53 (↓1.71)	73.39 (↓3.26)	79.10 (↓1.11)	66.26 (↓1.84)	51.87 (↓2.16)	75.37 (↓0.79)	67.42 (↓1.81)
Llama-3.1-70B-it	FP16	16 / 16	140	69.54	88.70	86.74	82.30	59.85	85.40	78.76
	FP8	8 / 8	70	69.45 (↓0.09)	88.25 (↓0.45)	86.69 (↓0.05)	82.02 (↓0.28)	59.80 (↓0.05)	85.08 (↓0.32)	78.55 (↓0.21)
	GPTQ*	4 / 16	35	69.80 (↑0.26)	89.54 (↑0.84)	86.28 (↓0.46)	81.40 (↓0.90)	59.37 (↓0.48)	84.69 (↓0.71)	78.51 (↓0.25)
	GPTQ**	4 / 16	35	69.97 (↑0.43)	89.76 (↑1.06)	86.26 (↓0.48)	81.97 (↓0.33)	58.74 (↓1.11)	84.53 (↓0.87)	78.54 (↓0.22)
	GPTQ**	8 / 16	70	69.03 (↓0.51)	87.95 (↓0.75)	86.29 (↓0.45)	82.17 (↓0.13)	58.94 (↓0.91)	84.53 (↓0.87)	78.15 (↓0.61)
	SmoothQuant	8 / 8	70	70.05 (↑0.51)	88.55 (↓0.15)	86.56 (↓0.18)	82.10 (↓0.20)	60.39 (↑0.54)	85.24 (↓0.16)	78.82 (↑0.06)
	AWQ	4 / 16	35	69.80 (↑0.26)	90.83 (↑2.13)	86.18 (↓0.56)	81.33 (↓0.97)	59.68 (↓0.17)	84.37 (↓1.03)	78.70 (↓0.06)
Llama-3.1-405B-it	FP16	16 / 16	810	73.72	94.84	88.40	83.98	65.42	85.00	81.89
	FP8	8 / 8	405	73.12 (↓0.60)	95.38 (↑0.54)	88.32 (↓0.08)	85.91 (↑1.93)	64.79 (↓0.63)	85.63 (↑0.63)	82.19 (↑0.30)
	GPTQ**	4 / 16	202.5	72.10 (↓1.62)	94.24 (↓0.60)	88.17 (↓0.23)	85.79 (↑1.81)	64.80 (↓0.62)	85.48 (↑0.48)	81.76 (↓0.13)
	SmoothQuant	8 / 8	405	72.01 (↓1.71)	92.72 (↓2.12)	87.53 (↓0.87)	73.28 (↓10.70)	65.19 (↓0.23)	85.95 (↑0.95)	79.45 (↓2.44)
	AWQ	4 / 16	202.5	73.98 (↑0.26)	94.84 (0.00)	88.04 (↓0.36)	85.71 (↑1.73)	64.25 (↓1.17)	86.35 (↑1.35)	82.20 (↑0.31)
Llama-3.2-1B-it	FP16	16 / 16	2	42.06	33.36	59.62	45.44	43.82	62.59	47.81
	FP8	8 / 8	1	41.98 (↓0.08)	34.04 (↑0.68)	59.20 (↓0.42)	45.38 (↓0.06)	43.18 (↓0.64)	62.67 (↑0.08)	47.74 (↓0.07)
	GPTQ***	4 / 16	0.5	34.90 (↓7.16)	8.04 (↓25.32)	43.14 (↓16.48)	40.20 (↓5.24)	41.92 (↓1.90)	57.85 (↓4.74)	37.67 (↓10.14)
	SmoothQuant	8 / 8	0.5	42.06 (0.00)	33.13 (↓0.23)	59.51 (↓0.11)	45.19 (↓0.25)	43.40 (↓0.42)	61.48 (↓1.11)	47.46 (↓0.35)
	AWQ	4 / 16	0.5	38.48 (↓3.58)	17.97 (↓15.39)	53.82 (↓5.80)	40.51 (↓4.39)	42.61 (↓1.21)	59.12 (↓3.47)	42.09 (↓5.72)
Llama-3.2-3B-it	FP16	16 / 16	6	52.13	64.52	73.08	59.59	47.79	69.61	61.45
	FP8	8 / 8	3	51.37 (↓0.76)	63.31 (↓1.21)	73.04 (↓0.04)	59.74 (↑0.15)	49.98 (↑0.19)	69.14 (↓0.47)	61.10 (↓0.35)
	GPTQ***	4 / 16	1.5	50.26 (↓1.87)	60.20 (↓4.32)	71.19 (↓1.89)	57.90 (↓1.69)	49.52 (↓0.27)	68.75 (↓0.86)	59.64 (↓1.81)
	SmoothQuant	8 / 8	1.5	51.37 (↓0.76)	63.76 (↓0.76)	72.61 (↓0.47)	56.69 (↑0.10)	49.72 (↓0.07)	69.61 (0.00)	61.13 (↓0.32)
	AWQ	4 / 16	1.5	50.51 (↓1.62)	61.41 (↓3.11)	71.27 (↓1.81)	58.94 (↓0.65)	49.03 (↓0.76)	67.4 (↓2.21)	59.76 (↓1.69)
Llama-3.3-70B-it	FP16	16 / 16	140	71.67	90.83	86.39	82.20	60.90	83.98	79.33
	GPTQ***	4 / 16	35	69.71 (↓1.96)	89.39 (↓1.44)	85.58 (↓0.81)	81.63 (↓0.57)	61.25 (↑0.35)	84.21 (↑0.23)	78.63 (↓0.70)
	AWQ	4 / 16	35	70.82 (↓0.85)	88.17 (↓2.66)	85.73 (↓0.66)	81.45 (↓0.75)	60.82 (↓0.08)	83.98 (0.00)	78.50 (↓0.83)

Table 1: Evaluation of Llama families on OpenLLM Leaderboard-v1. *, **, and *** denote the use of AutoGPTQ, llmcompressor, and AutoRound for GPTQ quantization, respectively.

of model quantization on the performance of LLMs, focusing on various aspects. For instance, Yao et al. [Yao *et al.*, 2023] investigated the impact of quantization on both weights and activations in language modeling tasks. In contrast, Liu et al. [Liu *et al.*, 2023] concentrated solely on evaluating three emergent abilities of quantized LLMs, neglecting crucial tasks such as trustworthiness, dialogue, and long-context processing.

Hong et al. [Hong *et al.*, 2024] expanded the scope by examining trustworthiness dimensions in the assessment of LLM compression techniques. However, most studies have predominantly relied on accuracy as the primary evaluation metric, with limited attention paid to alternative metrics. For example, Zhang et al. [Zhang *et al.*, 2024] proposed additional evaluation metrics, including fluency, informativeness, coherence, and harmlessness, alongside accuracy.

Efforts to establish more comprehensive evaluation benchmarks have also been made. Jaiswal et al. [Jaiswal *et al.*, 2023] developed a benchmark from existing datasets to evaluate compressed models, while Li et al. [Li *et al.*, 2024] and Jin et al. [Jin *et al.*, 2024] assessed various quantization techniques across different tasks. Namburi et al. [Namburi *et al.*, 2023] explored how compression and pruning affect the parametric knowledge of LLMs.

Dutta et al. [Dutta *et al.*, 2024] proposed a new metric from the perspective of "Flip" errors, emphasizing the importance of evaluating model robustness by accounting for inconsistencies and reversals in predictions, thereby going beyond traditional accuracy-focused metrics. Kurtic et al. [Kurtic *et al.*, 2024] investigated the accuracy-performance trade-offs in quantizing LLMs, evaluating formats like FP8, INT8, and INT4 across various tasks and proposing practical guidelines for efficient LLM deployment at different model scales.

Xu et al. [Xu *et al.*, 2024] explored the challenges of evaluating multilingual LLMs across diverse languages and cultures,

Model	Method	W/A	Storage (GB)	OpenLLM Leaderboard-v2 ↑ BBH (3-shot)	GPQA (0-shot)	IFEval (0-shot)	Math-Lvl-5 (4-shot)	MMLU-PRO (5-shot)	MuSR (0-shot)	Avg.
Llama-2-7B-Chat	FP16	16 / 16	14	12.23	1.59	35.31	1.93	11.04	8.89	11.83
	GPTQ*	4 / 16	3.5	7.96 (↓4.27)	0.85 (↓0.74)	30.59 (↓4.72)	1.43 (↓0.50)	7.58 (↓3.46)	3.73 (↓5.16)	8.69 (↓3.14)
	AWQ	4 / 16	3.5	10.78 (↓1.45)	3.54 (↑1.95)	31.57 (↓3.74)	1.81 (↓0.12)	7.75 (↓3.29)	10.62 (↑1.73)	11.01 (↓0.82)
Llama-2-13B-Chat	FP16	16 / 16	26	16.87	4.03	37.45	1.56	15.36	10.00	14.21
	GPTQ*	4 / 16	6.5	16.92 (↑0.05)	4.27 (↑0.24)	33.40 (↓4.05)	2.21 (↑0.65)	15.44 (↑0.08)	8.30 (↓1.70)	13.42 (↓0.79)
	AWQ	4 / 16	6.5	16.54 (↓0.33)	6.96 (↑2.93)	33.28 (↓4.17)	1.80 (↑0.24)	15.57 (↑0.21)	10.39 (↑0.39)	14.09 (↓0.12)
Llama-2-70B-Chat	FP16	16 / 16	140	29.42	6.72	44.11	2.64	25.01	6.32	19.04
	GPTQ*	4 / 16	35	25.80 (↓3.62)	5.25 (↓1.47)	41.52 (↓2.59)	3.27 (↑0.63)	23.87 (↓1.14)	7.06 (↑0.74)	17.80 (↓1.24)
	AWQ	4 / 16	35	28.63 (↓0.79)	6.72 (0.00)	42.89 (↓1.22)	1.95 (↓0.69)	24.44 (↓0.57)	5.56 (↓0.76)	18.37 (↓0.67)
Llama-3.1-8B-it	FP16	16 / 16	16	30.11	6.23	50.09	11.69	30.90	8.88	22.98
	FP8	8 / 8	8	29.20 (↓0.91)	5.49 (↓0.74)	49.16 (↓0.93)	12.01 (↑0.32)	30.92 (↑0.02)	6.95 (↓1.93)	22.29 (↓0.69)
	GPTQ*	4 / 16	4	25.86 (↓4.25)	7.20 (↑0.97)	47.95 (↓2.14)	9.49 (↓2.20)	29.60 (↓1.30)	6.03 (↓2.85)	21.02 (↓1.96)
	GPTQ**	4 / 16	4	25.83 (↓4.28)	6.72 (↑0.49)	44.81 (↓5.28)	8.85 (↓2.84)	28.16 (↓2.74)	10.58 (↑1.70)	20.83 (↓2.15)
	GPTQ**	8 / 16	8	29.97 (↓0.14)	6.23 (0.00)	50.53 (↑0.44)	11.94 (↑0.25)	31.19 (↑0.29)	7.80 (↓1.08)	22.94 (↓0.04)
	SmoothQuant	8 / 8	8	30.19 (↑0.08)	2.56 (↓3.67)	50.25 (↑0.16)	12.77 (↑1.08)	30.75 (↓0.15)	8.12 (↓0.76)	22.44 (↓0.54)
	AWQ	4 / 16	4	25.73 (↓4.38)	5.98 (↓0.25)	47.97 (↓2.12)	10.02 (↓1.67)	29.08 (↓1.82)	6.74 (↓2.14)	20.92 (↓2.06)
Llama-3.1-70B-it	FP16	16 / 16	140	55.90	16.48	75.48	28.68	48.00	19.32	40.64
	FP8	8 / 8	70	55.54 (↓0.36)	16.24 (↓0.24)	75.75 (↑0.27)	28.92 (↑0.24)	47.84 (↓0.16)	19.35 (↑0.03)	40.61 (↓0.03)
	GPTQ*	4 / 16	35	53.65 (↓2.25)	17.70 (↑1.22)	73.26 (↓2.22)	27.26 (↓1.42)	47.49 (↓0.51)	20.33 (↑1.01)	39.95 (↓0.69)
	GPTQ**	4 / 16	35	55.79 (↓0.11)	14.04 (↓2.44)	72.71 (↓2.77)	26.16 (↓2.52)	46.97 (↓1.03)	16.93 (↓2.39)	38.77 (↓1.87)
	GPTQ**	8 / 16	70	54.79 (↓1.11)	2.81 (↓13.67)	66.66 (↓8.82)	29.06 (↑0.38)	47.56 (↓0.44)	20.42 (↑1.10)	36.88 (↓3.76)
	SmoothQuant	8 / 8	70	55.06 (↓0.84)	16.24 (↓0.24)	74.78 (↓0.70)	27.90 (↓0.78)	47.20 (↓0.80)	20.32 (↑1.00)	40.25 (↓0.39)
	AWQ	4 / 16	35	54.08 (↓1.82)	16.48 (0.00)	75.15 (↓0.33)	27.85 (↓0.83)	47.07 (↓0.93)	21.69 (↑2.37)	40.39 (↓0.25)
Llama-3.1-405B-it	FP16	16 / 16	810	66.81	26.25	76.18	37.06	60.01	19.86	47.70
	FP8	8 / 8	405	65.22 (↓1.59)	27.37 (↑1.22)	72.44 (↓3.74)	35.86 (↓1.20)	59.67 (↓0.34)	17.83 (↓2.03)	46.42 (↓1.28)
	GPTQ**	4 / 16	202.5	66.21 (↓0.60)	23.57 (↓2.68)	72.90 (↓3.28)	34.74 (↓2.32)	59.11 (↓0.90)	18.92 (↓0.94)	45.91 (↓1.79)
	SmoothQuant	8 / 8	405	54.46 (↓12.35)	16.73 (↓9.52)	70.34 (↓5.84)	35.01 (↓2.05)	18.24 (↓41.77)	18.60 (↓1.26)	35.56 (↓12.14)
	AWQ	4 / 16	202.5	65.50 (↓1.31)	26.50 (↑0.25)	47.52 (↓28.66)	38.13 (↑1.07)	58.63 (↓1.38)	19.69 (↓0.17)	42.66 (↓5.04)
Llama-3.2-1B-it	FP16	16 / 16	2	8.32	1.34	41.61	4.20	10.60	3.70	11.63
	FP8	8 / 8	1	8.83 (↑0.51)	1.59 (↑0.25)	43.18 (↑1.57)	3.51 (↓0.69)	10.28 (↓0.32)	3.49 (↓0.21)	11.81 (↑0.18)
	GPTQ***	4 / 16	0.5	3.98 (↓4.34)	0.85 (↓0.49)	25.60 (↓16.01)	0.30 (↓3.90)	6.72 (↓3.88)	1.86 (↓1.84)	6.55 (↓5.08)
	SmoothQuant	8 / 8	0.5	8.22 (↓0.10)	4.03 (↑2.69)	41.68 (↑0.07)	3.25 (↓0.95)	10.17 (↓0.43)	3.27 (↓0.43)	11.77 (↑0.14)
	AWQ	4 / 16	0.5	5.21 (↓3.11)	4.27 (↑2.93)	31.21 (↓10.40)	1.56 (↓2.64)	7.61 (↓2.99)	2.84 (↓0.86)	8.78 (↓2.85)
Llama-3.2-3B-it	FP16	16 / 16	6	20.77	7.69	53.56	11.20	22.24	6.92	20.40
	FP8	8 / 8	3	21.00 (↑0.23)	7.69 (0.00)	53.32 (↓0.24)	9.78 (↓1.42)	21.99 (↓0.25)	7.30 (↑0.38)	20.18 (↓0.22)
	GPTQ***	4 / 16	1.5	18.82 (↓1.95)	5.98 (↓1.71)	52.81 (↓0.75)	8.05 (↓3.14)	19.31 (↓2.93)	4.19 (↓2.73)	18.19 (↓2.21)
	SmoothQuant	8 / 8	1.5	21.01 (↑0.24)	9.65 (↓1.96)	53.96 (↑0.40)	9.58 (↓1.62)	21.83 (↓0.41)	6.07 (↓0.85)	20.35 (↓0.05)
	AWQ	4 / 16	1.5	19.45 (↓1.32)	8.18 (↑0.49)	51.5 (↓2.06)	8.3 (↓2.9)	20.79 (↓1.45)	7.19 (↓0.27)	19.23 (↓1.17)
Llama-3.3-70B-it	FP16	16 / 16	140	56.78	30.40	69.03	30.95	49.78	22.28	43.20
	GPTQ***	4 / 16	35	52.48 (↓4.31)	28.94 (↓1.46)	65.71 (↓3.32)	28.55 (↓2.40)	48.24 (↓1.54)	21.38 (↓0.90)	40.88 (↓2.32)
	AWQ	4 / 16	35	55.89 (↓0.89)	28.69 (↓1.71)	69.61 (↑0.58)	29.42 (↓1.53)	48.57 (↓1.21)	20.67 (↓1.61)	42.14 (↓1.06)

Table 2: Evaluation of Llama families on OpenLLM Leaderboard-v2. *, **, and *** denote the use of AutoGPTQ, llmcompressor, and AutoRound for GPTQ quantization, respectively.

emphasizing the development of culturally and linguistically inclusive benchmarks for fair evaluation. Liu et al. [Liu *et al.*, 2024] provided a comprehensive examination of the generalization ability, focusing on their performance across various tasks and datasets. This study highlights the importance of designing benchmarks and metrics that accurately reflect real-world applications while identifying the limitations of current evaluation strategies.

To our knowledge, no prior study has comprehensively examined the effects of quantization across a wide range of model sizes—from 1B to 405B parameters—encompassing both SLMs and LLMs, including the latest architectures such as Llama-3.1, Llama-3.2, and Llama-3.3. Furthermore, existing research has not conducted detailed category-level analyses through cross-architecture comparisons or employed manual inspection using LLM-as-judge qualitative methods. Additionally, no work has compared the trends observed in LLM-as-judge (MT-Bench) evaluations with leaderboard results to identify new trends in quantization impacts.

3 Evaluation Procedure

To handle LLMs, which cannot be processed on a single server, and to ensure fast and reliable evaluations, we developed a structured evaluation pipeline based on a multi-node cluster setup. Figure 1 presents an overview of the implemented pipeline for evaluating quantized LLMs. The evaluated LLMs include the Vicuna, Gemma, and Llama families, ranging in size from 1B to 405B. Each model is quantized using GPTQ, AWQ, SmoothQuant, and FP8 methods. The evaluation is conducted using lm-eval and MT-Bench as benchmarking tools. The multi-node cluster used for evaluation is implemented with vLLM and consists of four servers: H100-80Gx8, A100-80Gx4, RTX 6000-48Gx4, and A6000-48Gx4. Additionally, the Huggingface library is integrated into the pipeline to support model hosting and benchmarking. The evaluation is distributed across a multi-cluster environment to ensure a

thorough performance assessment. If vLLM cannot be used for processing, we used the Huggingface Accelerate library instead, which is slower but shows better comparability. The versions of all tools used are provided in Appendix D.1.

4 Experimental Setup

4.1 Datasets

We conducted a comprehensive evaluation of the quantized LLMs on widely adopted benchmarks, grouped into six main categories: CommonSenseQA (ARC, HellaSwag, Winogrande), Knowledge and Language Understanding (MMLU, GPQA, MMLU-PRO, BBH, MuSR), Instruction Following (IFEval), Hallucination (TruthfulQA), Mathematics (GSM8K, MATH-Lvl-5), and Dialogue (MT-Bench). Additional information about these datasets can be found in Appendix A, and an overview of all benchmarks is provided in Table 3 (Appendix).

4.2 Reproducibility Details

Both OpenLLM Leaderboard V1 and V2 follow the same methodology outlined on the HuggingFace Leaderboard's page, including identical normalization procedures. Additional information on leaderboard calculations is provided in Appendix D.2.

We used the *greedy decoding* strategy to maintain deterministic output tokens across runs. The detailed configuration for the *greedy decoding* strategy is presented in Appendix D.1, which also lists the specific versions of each package used. Furthermore, we record all random seeds for Python, NumPy, Torch, and few-shot setups in Appendix D.1, ensuring the reproducibility of our experimental results.

4.3 Quantization Methods and Calibration Data

We evaluated multiple PTQ methods, including GPTQ, AWQ, SmoothQuant, and FP8. GPTQ and AWQ focus on weight-only quantization, while SmoothQuant and FP8 apply to both weights and activations. For a detailed overview of each quantization method, refer to Appendix D.3, with configuration details and group sizes described in Appendix D.4.

The selection and configuration of calibration datasets are crucial for maintaining consistent performance across models. We used default settings for the number of samples and sequence lengths, as detailed in Appendix D.5 and summarized in Table 9. These settings may vary depending on the specific algorithms and tools but generally ensure stable results for our experiments.

4.4 Models

We applied quantization techniques to 12 instruction-tuned open LLMs, including the Vicuna [Zheng *et al.*, 2023], Gemma [Team *et al.*, 2024], and Llama [Dubey *et al.*, 2024] families, with model sizes ranging from 1B to 405B. These models were released between June 2023 and December 2024 and were downloaded from HuggingFace's model sources.

All models were evaluated using 13 benchmark datasets, applying GPTQ, AWQ, SmoothQuant, and FP8 quantization. However, due to runtime limitations (over 30 days), we did not measure the original model accuracy or test all GPTQ configurations for Llama-3.1-405B. Also, due to space limitations, the experimental results for the Vicuna and Gemma models are provided in the Appendix B

5 Experimental Results

This section presents experimental results addressing five research questions, detailing the performance impact of quantization across 13 datasets for three model families of varying sizes. Table 1 and Table 2 summarize the experimental results from OpenLLM Leaderboard-v1 and v2, respectively.

5.1 RQ1: Do quantized LLMs outperform smaller original models on most benchmarks, and how do they fare across different architectures and SLMs?

We observe that quantized LLMs generally outperform smaller, uncompressed models across a wide range of benchmarks. For instance, a 4-bit Llama-2-13B (6.5 GB) outperforms an FP16 Llama-2-7B (14 GB) on most tasks, despite its reduced size. However, in TruthfulQA (hallucination testing) and IFEval (instruction-following), the FP16 Llama-2-7B still performs better, indicating that quantization can compromise alignment and adherence to instructions.

Similarly, quantizing Llama-3.1-405B to 4 bits (202.5 GB) yields higher accuracy than the FP16 Llama-3.1-70B (140 GB) across various tasks; yet, the instruction-following IFEval benchmark again highlights a shortfall in the quantized model. This performance gap holds across different model architectures: although Llama-3.3-70B demonstrates improvements over Llama-3.1-70B, a 4-bit Llama-3.1-405B can still outperform the uncompressed Llama-3.3-70B.

In edge-focused SLMs, quantization produces significantly larger improvements. For example, quantizing Llama-3.2-3B (SmoothQuant) improves accuracy by 13.32% on OpenLLM Leaderboard-v1 and 8.72% on OpenLLM Leaderboard-v2 compared to the FP16-Llama-3.2-1B. Such improvements exceed the margin typically observed in larger models (7B to 405B).

RQ1 Findings: Quantized LLMs consistently outperform smaller models across most benchmarks and maintain this advantage across different architectures, with significant gains observed in both large models and edge-focused SLMs. However, tasks like instruction-following (IFEval) and hallucination detection (TruthfulQA) remain challenging for quantized models.

5.2 RQ2: How do different quantization methods affect the performance of models across diverse tasks? Are there noticeable differences in how specific methods (e.g., GPTQ, AWQ, SmoothQuant, FP8) impact task accuracy?

In most cases, weight-only quantization methods (GPTQ and AWQ) and activation quantization methods (SmoothQuant and FP8) exhibit similar performance. However, for Llama-3.1-405B, SmoothQuant's activation quantization resulted in

a significant accuracy drop compared to other methods, with an average decrease of up to 10.86% compared to FP8 on the OpenLLM Leaderboard-v2 datasets. This decline occurs because SmoothQuant was originally designed to handle the high activation ranges observed in models up to the size of OPT-175B. Consequently, at the 405B scale of Llama-3.1, the algorithm likely did not account for certain factors that are critical at this larger scale. In contrast, for smaller models, such as SLMs with 1B and 3B parameters, the average accuracy drop remains below 1%.

When comparing weight-only quantization methods, AWQ consistently outperforms GPTQ across various LLMs on overall benchmark scores. Additionally, different implementations of GPTQ, such as AutoGPTQ and llmcompressor, demonstrate notable performance differences, with the oldest GPTQ implementation library, AutoGPTQ, still maintaining stable and consistent performance.

When both weight and activation quantization are required at 8 bits, FP8 proves to be highly effective, even on challenging tasks from the OpenLLM Leaderboard-v2. This effectiveness spans models of all sizes, from the largest, such as Llama-3.1-405B, to the smallest, like Llama-3.2-1B. Therefore, FP8 offers greater stability compared to SmoothQuant and is advantageous to use when supported by hardware such as the NVIDIA H100 GPU and RTX 6000 Ada, as it provides benefits in both latency and throughput. Additionally, applying FP8 to both weights and activations allows for a reduction of the KV cache size by half, which is highly beneficial during LLM decoding phases where I/O bottlenecks are a concern. This FP8 KV cache feature is supported by vLLM.

RQ2 Findings: FP8 is the most stable option across all model sizes and tasks, particularly in large LLMs where SmoothQuant performs poorly, whereas AWQ regularly outperforms GPTQ in weight-only quantization, and specialized hardware enhances FP8's advantages.

5.3 RQ3: How do model size and architecture influence the accuracy of quantized models?

We evaluated GPTQ across 13 datasets and observed that its accuracy can degrade significantly under 4-bit quantization—a behavior not reflected in perplexity-based evaluations alone. For smaller models, such as Llama-3.2-1B, 4-bit quantization causes particularly severe accuracy drops (e.g., -25.32% on GSM8k and -16.01% on IFEval). These declines tend to be more pronounced in GPTQ than in AWQ, suggesting that SmoothQuant or FP8 at 8-bit may be necessary to maintain accuracy for 1B-scale models.

With mid-sized models like Llama-3.1-8B (GPTQ-8bit), we noted average accuracy improvements of +2.51% and +2.11% over 4-bit on OpenLLM Leaderboard-v1 and OpenLLM Leaderboard-v2, respectively. However, at the 70B scale, 4-bit outperformed 8-bit by +1.89%, indicating a reverse trend for larger models. To examine architectural differences at a fixed scale, we compared Llama-2, Llama-3.1, and Llama-3.3 at the 70B size and found that AWQ consistently surpassed GPTQ, delivering stable accuracy even at 4-bit quantization.

RQ3 Findings: In smaller LLMs, 4-bit quantization often leads to significant accuracy loss (especially with GPTQ), whereas 70B-scale models can maintain stable performance with 4-bit. Although size primarily drives quantization difficulty, architectural variations within the same scale can also influence accuracy. Nonetheless, AWQ consistently outperforms GPTQ across diverse tasks and model families.

5.4 RQ4: Does higher task difficulty always correlate with greater accuracy degradation under quantization?

Contrary to common assumptions, tasks widely considered challenging (e.g., GSM8K, MMLU, Math-Lvl-5) did not consistently show the greatest performance drops when quantized. For instance, among 12 tested models quantized with AWQ, MMLU accuracy remained largely unchanged, with some models even exhibiting slight improvements. In contrast, tasks generally regarded as less knowledge-intensive (ARC-challenge, TruthfulQA, Winogrande) occasionally experienced declines exceeding -3%.

Rather than strictly depending on task difficulty, quantization appears to magnify existing weaknesses in a model's ability to handle specific forms of reasoning, such as commonsense or mathematical inference. As highlighted in RQ3, smaller models (2B–7B) are especially vulnerable to computational reasoning tasks like GSM8K, exhibiting sharper performance drops. Moreover, as noted in RQ2, different quantization methods can produce varying degrees of accuracy loss, making it difficult to draw definitive conclusions from a single task perspective.

RQ4 Findings: Difficult tasks do not always suffer the largest accuracy loss under quantization. The impact varies by model architecture and the chosen quantization method, leading some *hard* tasks to remain stable while *easier* tasks occasionally show bigger drops. In essence, quantization amplifies a model's existing weaknesses, particularly in commonsense, logical, or mathematical reasoning.

5.5 RQ5: How does quantization impact the free-form conversation quality of LLMs when evaluated using the MT-Bench framework, which employs LLMs as judges?

Category-Level Analysis. Figure 2 presents a detailed breakdown of three models (Llama-3.1, Llama-3.2, and Llama-3.3) across MT-Bench categories. Quantized LLMs suffer the largest score degradation in *Coding* and *STEM*. For *Coding*, manual inspections reveal that GPT4 often assigns lower scores when the generated code contains fewer examples or insufficient comments. In *STEM* tasks, correctness matters when a definite solution exists, and concise logical explanations are important when an exact answer does not exist. Quantized models frequently either provided overly verbose justifications or produced incorrect statements, leading to lower scores. In

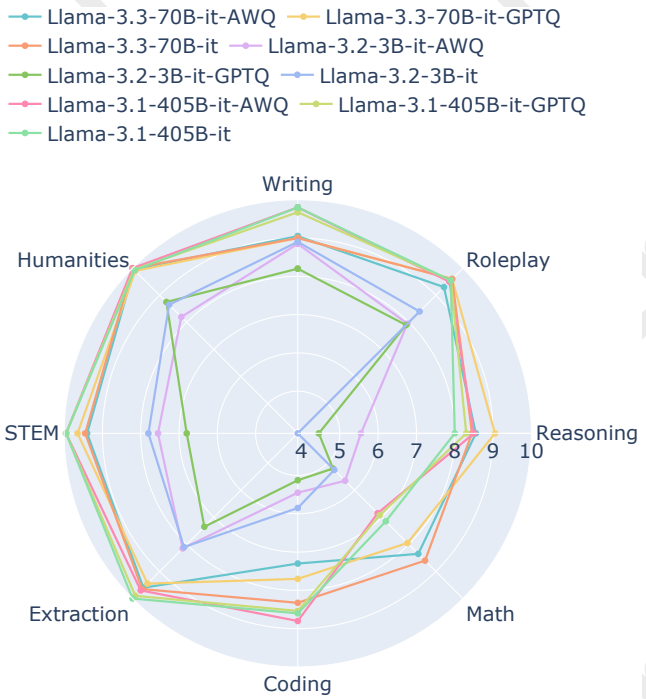

Figure 2: Category-wise MT-Bench scores of three quantized LLMs (Llama-3.3, Llama-3.2, and Llama-3.1) evaluated using AWQ and GPTQ methods. It highlights the performance differences across categories, including Writing, Roleplay, Reasoning, Math, Coding, Extraction, STEM, and Humanities, demonstrating the impact of quantization on diverse tasks.

contrast, the *Reasoning* category showed an increase of about 1 point with quantization. Manual checks reveal that concise answers tend to receive higher GPT4 scores, and quantized models often responded more concisely than their original counterparts. The detailed results of this manual inspection, along with the full text responses, can be found in Appendix C.4. Also, Table 6 in the Appendix C.1 lists category-wise MT-Bench scores for all models.

Limitations of GPT4-Based Evaluation. Consistent with the findings in MT-Bench, GPT4 sometimes misjudges incorrect responses as correct, particularly for math and reasoning tasks. Although reference-guided judging and chain-of-thought prompting can mitigate such errors [Zheng *et al.*, 2023], they do not eliminate them entirely. In *Reasoning* task, GPT4 erroneously considered a wrong answer correct, boosting quantized models' scores. Conversely, there were *STEM* questions where both original and quantized models provided accurate answers, yet GPT4 mistakenly marked them as incorrect. **These misjudged cases are described in the Appendix C.5.**

For models like Llama-3.3-70B, categories such as *Humanities* consistently scored near or at the maximum (10 points). In these cases, quantized versions also achieved scores in the high 9-point range, making it difficult to discern meaningful quality differences through manual inspection. This is because it is difficult to clearly understand the reasoning behind the results by only looking at GPT4's judgment statements, especially when the score differences are marginal, such as 1-2 points. Hence, for the latest large-scale models, more sensitive metrics or superior judging models may be required to evaluate subtle quality gaps.

Multi-Turn Analysis. Table 7 in the Appendix C.2 presents the average scores of multi-turn conversations across 12 models. Among smaller models (e.g., 2B, 7B, 8B), some exhibit slight score improvements; however, this trend is not consistent across all small-scale models. In contrast, larger models (e.g., 13B, 70B) generally experience score declines, although there are exceptions where AWQ enhances performance. Additionally, accuracy losses become more noticeable in the second turn of multi-turn interactions.

These observations indicate that establishing a clear trend based solely on model size or type is challenging, as the impact of quantization depends on a combination of factors, including model architecture, quantization method, and task complexity. Furthermore, when comparing quantization methods, AWQ typically outperforms GPTQ, which is consistent with the results observed on OpenLLM Leaderboard-v1 and v2.

Comparison with Leaderboard Results. The trends observed in MT-Bench do not always align with leaderboard outcomes. For instance, the Quantized Llama-3.1-405B model outperformed a newer Llama-3.3-70B-FP16 model on certain leaderboards, yet scored similarly or slightly lower in MT-Bench. Unlike the leaderboard tasks, which may not be particularly sensitive to *Coding* and *Math* challenges, the free-form and demanding nature of MT-Bench conversations highlights performance drops in more complex categories such as *Coding* and *STEM*. Thus, although computational metrics suggest that quantization does not uniformly degrade accuracy (RQ4), MT-Bench's qualitative LLM-based evaluation reveals significant performance reductions in tasks known to be difficult.

RQ5 Findings: We observe that quantization considerably reduces performance in Coding and STEM tasks, while occasionally improving reasoning. The impact of quantization on multi-turn conversation quality does not consistently correlate with model size or type Additionally, GPT4-based assessments sometimes misjudge incorrect answers as correct.

6 Conclusion

We evaluated instruction-tuned quantized LLMs across 13 datasets and 6 task types, using models ranging from 1B to 405B and 4 quantization methods, including GPTQ, AWQ, SmoothQuant, and FP8. We found that quantized LLMs generally outperformed smaller models in most tasks, except for hallucination detection and instruction-following. Performance varied by quantization method and precision, with weight-only quantization performing better in the 405B model. Task difficulty had little impact on accuracy loss. Our MT-Bench evaluation revealed that quantization significantly reduces performance in Coding and STEM tasks while occasionally enhancing reasoning. Additionally, GPT4-based assessments can misjudge incorrect answers as correct.

Acknowledgments

We thank the former Neubla ML team members — Minwook Ahn, Minho Park, Jinsol Kim, Raegeun Park, and Byonghwa Oh — for their valuable discussions and feedback. This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2023-00277060, Development of open edge AI SoC hardware and software platform, No.RS-2024-00459797, Development of ML compiler framework for on-device AI, RS-2025-02214497, Development of low-level optimization program API technology for AI semiconductors)

Ethical Statement

There are no ethical issues.

References

[Clark *et al.*, 2018] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[Cobbe *et al.*, 2021] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[contributors, 2024a] AutoAWQ contributors. Autoawq. https://github.com/casper-hansen/AutoAWQ, 2024.

[contributors, 2024b] AutoGPTQ contributors. Autogptq. https://github.com/AutoGPTQ/AutoGPTQ, 2024.

[Dettmers and Zettlemoyer, 2023] Tim Dettmers and Luke Zettlemoyer. The case for 4-bit precision: k-bit inference scaling laws. In *International Conference on Machine Learning*, pages 7750–7774. PMLR, 2023.

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

[Dettmers *et al.*, 2024] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

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

[Dutta *et al.*, 2024] Abhinav Dutta, Sanjeev Krishnan, Nipun Kwatra, and Ramachandran Ramjee. Accuracy is not all you need. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[Frantar *et al.*, 2022] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Optq: Accurate quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*, 2022.

[Gong *et al.*, 2024] Ruihao Gong, Yang Yong, Shiqiao Gu, Yushi Huang, Chentao Lv, Yunchen Zhang, Xianglong Liu, and Dacheng Tao. Llmc: Benchmarking large language model quantization with a versatile compression toolkit, 2024.

[Hendrycks *et al.*, 2021a] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[Hendrycks *et al.*, 2021b] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

[Hong *et al.*, 2024] Junyuan Hong, Jinhao Duan, Chenhui Zhang, Zhangheng Li, Chulin Xie, Kelsey Lieberman, James Diffenderfer, Brian Bartoldson, Ajay Jaiswal, Kaidi Xu, et al. Decoding compressed trust: Scrutinizing the trustworthiness of efficient llms under compression. *arXiv preprint arXiv:2403.15447*, 2024.

[Jaiswal *et al.*, 2023] Ajay Jaiswal, Zhe Gan, Xianzhi Du, Bowen Zhang, Zhangyang Wang, and Yinfei Yang. Compressing llms: The truth is rarely pure and never simple. *arXiv preprint arXiv:2310.01382*, 2023.

[Jin *et al.*, 2024] Renren Jin, Jiangcun Du, Wuwei Huang, Wei Liu, Jian Luan, Bin Wang, and Deyi Xiong. A comprehensive evaluation of quantization strategies for large language models. *arXiv preprint arXiv:2402.16775*, 2024.

[Kurtic *et al.*, 2024] Eldar Kurtic, Alexandre Marques, Shubhra Pandit, Mark Kurtz, and Dan Alistarh. "give me bf16 or give me death"? accuracy-performance trade-offs in llm quantization. *arXiv preprint arXiv:2411.02355*, 2024.

[Lewkowycz *et al.*, 2022] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022.

[Li *et al.*, 2024] Shiyao Li, Xuefei Ning, Luning Wang, Tengxuan Liu, Xiangsheng Shi, Shengen Yan, Guohao Dai, Huazhong Yang, and Yu Wang. Evaluating quantized large language models. *arXiv preprint arXiv:2402.18158*, 2024.

[Lin *et al.*, 2021] Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*, 2021.

[Lin *et al.*, 2024] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq:

Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of Machine Learning and Systems*, 6:87–100, 2024.

[Liu *et al.*, 2023] Peiyu Liu, Zikang Liu, Ze-Feng Gao, Dawei Gao, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. Do emergent abilities exist in quantized large language models: An empirical study. *arXiv preprint arXiv:2307.08072*, 2023.

[Liu *et al.*, 2024] Yijun Liu, Yuan Meng, Fang Wu, Shenhao Peng, Hang Yao, Chaoyu Guan, Chen Tang, Xinzhu Ma, Zhi Wang, and Wenwu Zhu. Evaluating the generalization ability of quantized llms: Benchmark, analysis, and toolbox. *arXiv preprint arXiv:2406.12928*, 2024.

[llmcompressor contributors, 2024] llmcompressor contributors. Llm compressor. https://github.com/vllm-project/llm-compressor, 2024.

[Micikevicius *et al.*, 2022] Paulius Micikevicius, Dusan Stosic, Neil Burgess, Marius Cornea, Pradeep Dubey, Richard Grisenthwaite, Sangwon Ha, Alexander Heinecke, Patrick Judd, John Kamalu, et al. Fp8 formats for deep learning. *arXiv preprint arXiv:2209.05433*, 2022.

[Namburi *et al.*, 2023] Satya Sai Srinath Namburi, Makesh Sreedhar, Srinath Srinivasan, and Frederic Sala. The cost of compression: Investigating the impact of compression on parametric knowledge in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5255–5273, 2023.

[Rein *et al.*, 2023] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

[Sakaguchi *et al.*, 2021] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

[Sprague *et al.*, 2023] Zayne Sprague, Xi Ye, Kaj Bostrom, Swarat Chaudhuri, and Greg Durrett. Musr: Testing the limits of chain-of-thought with multistep soft reasoning. *arXiv preprint arXiv:2310.16049*, 2023.

[Suzgun *et al.*, 2022] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

[Team *et al.*, 2024] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[Wan *et al.*, 2023] Zhongwei Wan, Xin Wang, Che Liu, Samiul Alam, Yu Zheng, Zhongnan Qu, Shen Yan, Yi Zhu, Quanlu Zhang, Mosharaf Chowdhury, et al. Efficient large language models: A survey. *arXiv preprint arXiv:2312.03863*, 1, 2023.

[Wang *et al.*, 2024] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*, 2024.

[Wei *et al.*, 2023] Xiuying Wei, Yunchen Zhang, Yuhang Li, Xiangguo Zhang, Ruihao Gong, Jinyang Guo, and Xianglong Liu. Outlier suppression+: Accurate quantization of large language models by equivalent and optimal shifting and scaling, 2023.

[Xiao *et al.*, 2023] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099. PMLR, 2023.

[Xu *et al.*, 2024] Zhichao Xu, Ashim Gupta, Tao Li, Oliver Bentham, and Vivek Srikumar. Beyond perplexity: Multi-dimensional safety evaluation of llm compression. *arXiv preprint arXiv:2407.04965*, 2024.

[Yao *et al.*, 2023] Zhewei Yao, Xiaoxia Wu, Cheng Li, Stephen Youn, and Yuxiong He. Zeroquant-v2: Exploring post-training quantization in llms from comprehensive study to low rank compensation. *arXiv preprint arXiv:2303.08302*, 2023.

[Zellers *et al.*, 2019] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

[Zhang *et al.*, 2024] Yue Zhang, Ming Zhang, Haipeng Yuan, Shichun Liu, Yongyao Shi, Tao Gui, Qi Zhang, and Xuanjing Huang. Llmeval: A preliminary study on how to evaluate large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19615–19622, 2024.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

[Zhou *et al.*, 2023] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023.

[Zhu *et al.*, 2023] Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. A survey on model compression for large language models. *arXiv preprint arXiv:2308.07633*, 2023.