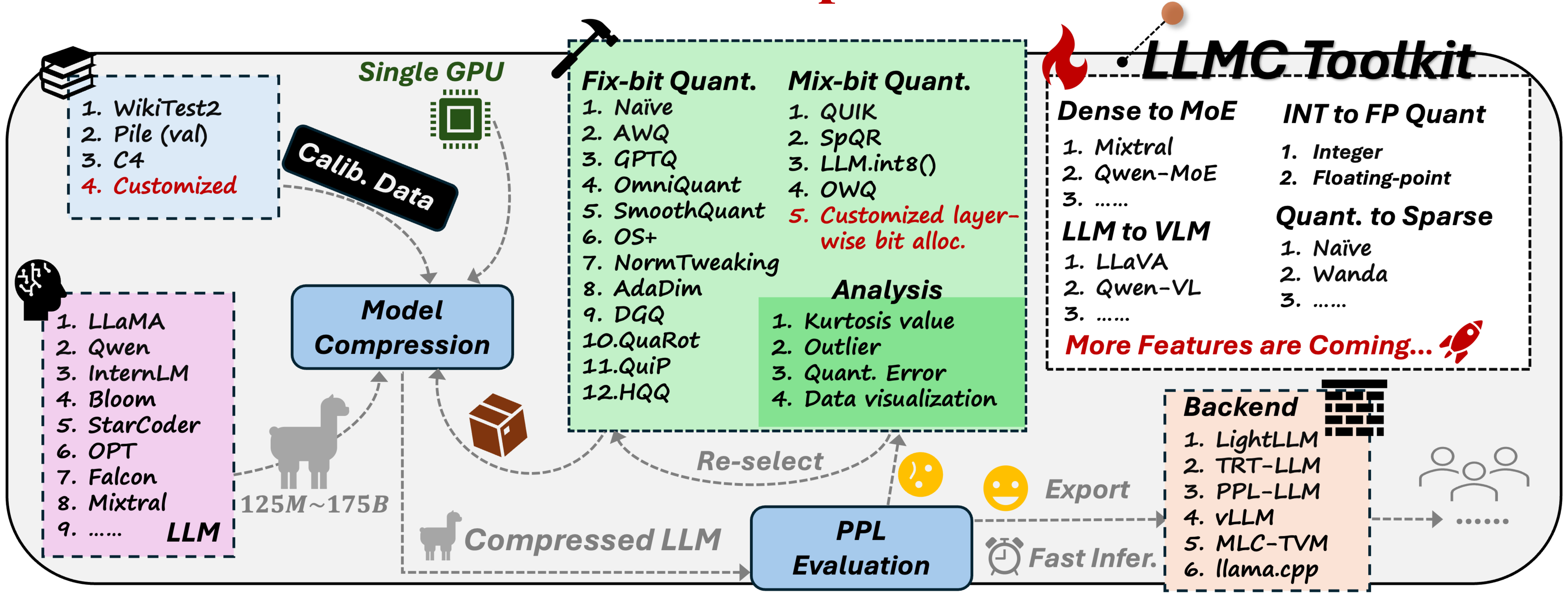
LLMC: Benchmarking Large Language Model Quantization with a Versatile Compression Toolkit

Ruihao Gong[1,2*], Yang Yong[2*], Shiqiao Gu[2*], Yushi Huang[1,2*], Chengtao Lv[1,2], Yunchen Zhang[2], Dacheng Tao[3], Xianglong Liu[1†]

[1]Beihang University & [2]SenseTime Research & [3]Nanyang Technological University

* indicates equal contribution,
† indicates corresponding author

LLMC: A Versatile LLM Compression Toolkit



Diverse algorithms support. **16** different methods covering weight-only, weight activation, and mixed-precision quantization.

Quantization with an ultra-low cost. Only one **40GB** A100 NVIDIA GPU is required to calibrate and evaluate **100B+** LLMs.

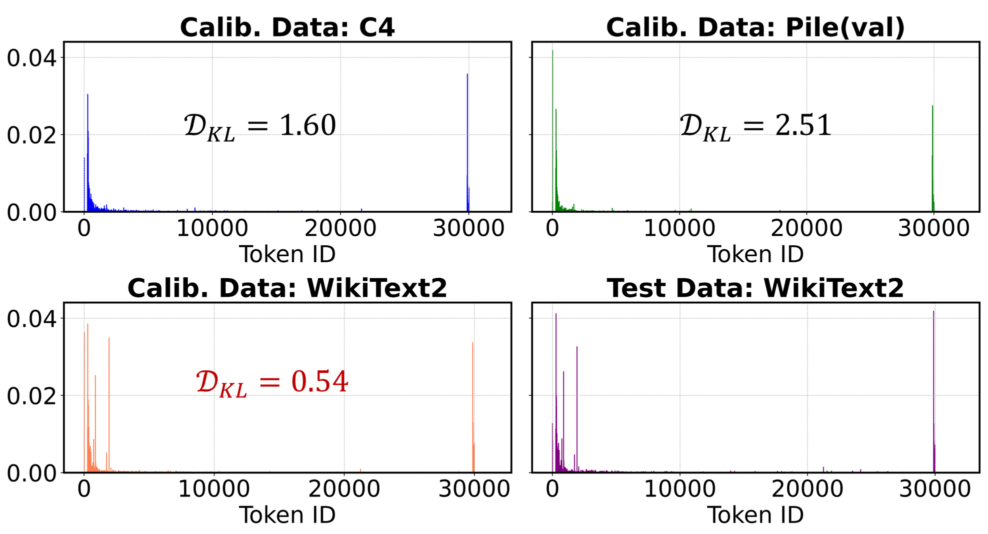
Multi-backend compatibility. **6** backends, *i.e.*, LightLLM, TensorRT-LLM, PPL-LLM, vLLM, MLC-TVM and `llama.cpp`.

High extensibility. Easy adaptation from integer quantization to **floating-point** quantization, from LLMs to **VLMs**, from quantization to **sparsification**, and from dense models to Mixture-of-Expert (**MoE**) models.

Comprehensive Evaluation. **PPL** and data **visualization** analysis, *e.g.*, Kurtosis value, quantization error, and outlier distribution.

Benchmarking LLM Quantization

Impact of Calibration Data



Token distribution consistency. It's important to select calibration data with an aligned distribution for the data in practice.

Calib. Data	GPTQ	AWQ	OmniQuant
C4	6.323	6.173	5.717
Pile (val)	6.330	6.195	5.753
WikiText2	**6.133** +0.568	**6.144** +0.156	**5.697** +0.516

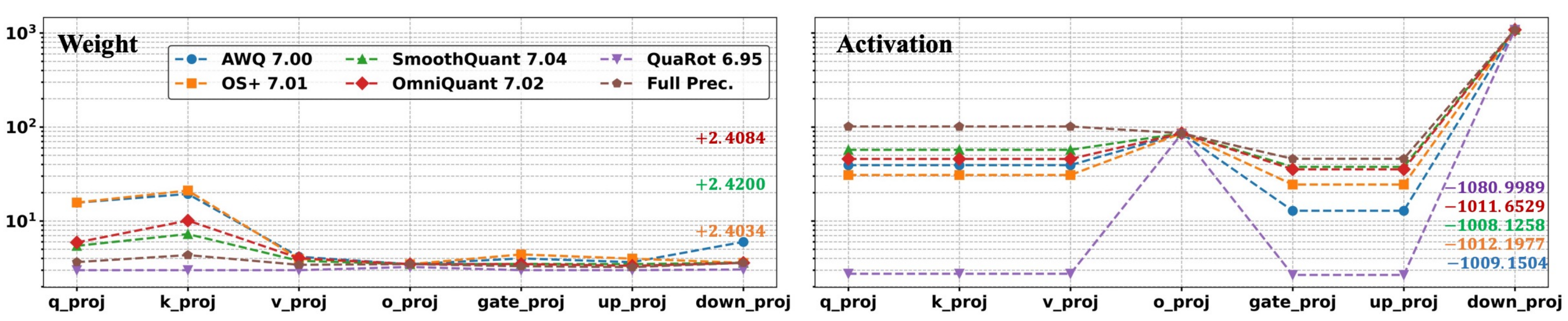
Intra-sentence logic. Break the logic within the calibration sentences can cause a non-negligible accuracy drop (data indices show differences in results from randomly shuffling token order within each data entry).

Dive into the Quantization Algorithms

Technique	Approach	Strategy	Eq. Trans.	Algorithm
TRANSFORMATION	*Rule-based*	$\boldsymbol{s} = \max(\lvert\boldsymbol{X}\rvert^{\gamma})/\max(\lvert\boldsymbol{W}\rvert^{1-\gamma}), \gamma = 0.5, 0.75, ...$	✓	SmoothQuant(Xiao et al., 2023)
		$\boldsymbol{Q}$, where $\boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{I}$ and $\lvert\boldsymbol{Q}\rvert = 1$	✓	QuaRot (Ashkboos et al., 2024)
	Search-based	$\boldsymbol{s} = \max(\lvert\boldsymbol{X}\rvert^{\gamma})/\max(\lvert\boldsymbol{W}\rvert^{1-\gamma})$, grid search for $\gamma \in [0,1]$	✓	AWQ(Lin et al., 2023)
		$\boldsymbol{s} = \max(1.0, \max(\boldsymbol{X})/t)$, grid search for t	✓	OS+(Wei et al., 2023b)
	Learnining-based	$\boldsymbol{s} = \arg\min_{\boldsymbol{s}} \mathcal{L}, \boldsymbol{s} \leftarrow \boldsymbol{s} - \eta\frac{\partial\mathcal{L}(\boldsymbol{s})}{\partial\boldsymbol{s}}$	✓	OmniQuant(Shao et al., 2023)
CLIPPING	*Rule-based*	$\alpha = 1, \beta = 1$	✓	SmoothQuant(Xiao et al., 2023), OS+(Wei et al., 2023b), GPTQ(Frantar et al., 2022), QuaRot (Ashkboos et al., 2024)
	Search-based	grid search for $\alpha = \beta \in [0,1]$	✗	AWQ(Lin et al., 2023)
	Learning-based	$\alpha, \beta = \arg\min_{\alpha,\beta}\mathcal{L}, \alpha \leftarrow \alpha - \eta\frac{\partial\mathcal{L}(\alpha)}{\partial\alpha}, \beta \leftarrow \beta - \eta\frac{\partial\mathcal{L}(\beta)}{\partial\beta}$	✗	OmniQuant(Shao et al., 2023)
RECONSTRUCTION	*Hessian-based*	$\boldsymbol{W} \leftarrow \boldsymbol{W} - \boldsymbol{E}\boldsymbol{H}^{-1}, \boldsymbol{H}^{-1} = \left(2\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}\right)^{-1}$	✗	GPTQ(Frantar et al., 2022)

How Does Transformation Influence Activation and Weight Outlier?

Definition. *Kurtosis value is defined as* $K = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{X_i-\mu}{\sigma}\right)^4$ *, where μ and σ represent mean and variance of a tensor $\boldsymbol{X}$, to reflect outlier conditions.*



1. Scaling-based transformation methods achieve lower K for activations at the cost of higher K for weights compared with full precision, which would induce a non-negligible performance degradation for lower-bit weight quantization (w6a6 > w4a8).
2. K for some specific positions like `down_proj` layers is significantly higher than others. These positions have a pronounced impact on accuracy. For example, with `down_proj` transformed (evident lower K), salient improvements are gained as exhibited.
3. Although the rotation-based transformation reduces outliers by directly optimizing the tensor's outliers, it may not realize obvious accuracy improvement in some cases. It is evident that the quantization error of output tensors is not minimized, as optimization did not focus on reducing output error, leading to a higher PPL.

Method	q_proj	k_proj	v_proj	o_proj	gate_proj	up_proj	down_proj	PPL↓
Full Prec.	3.6505	4.3354	3.4174	3.4720	3.2991	3.2300	3.5845	6.14
AWQ	4.9219	6.1633	3.4602	3.4720	3.3190	3.2438	4.3083	8.57
	0.9960	0.9960	0.9784	0.9387	0.9882	0.9628	0.9479	
QuaRot	2.9051	2.9050	2.9069	2.9075	2.9074	2.9073	2.9075	40.81
	0.9962	0.9967	0.9797	0.8286	0.9764	0.9579	0.9230	

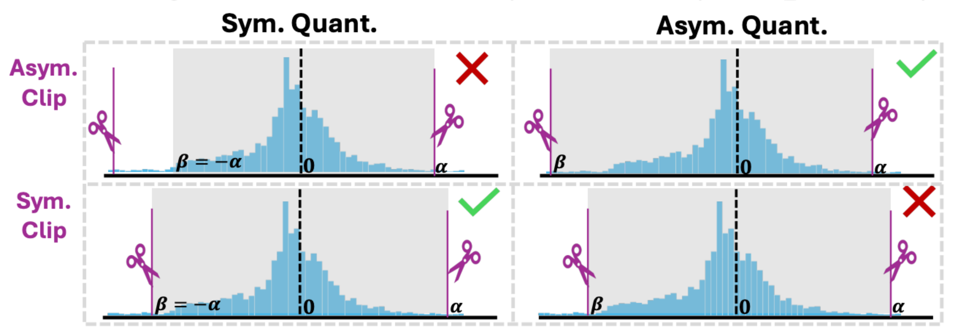
AWQ		SmoothQuant		OS+		OmniQuant		QuaRot	
w4a8	w6a6	w4a8	w6a6	w4a8	w6a6	w4a8	w6a6	w4a8	w6a6
8.60	7.00	8.85	7.04	8.55	7.01	8.83	7.02	9.77	6.95
7.77	6.79	7.92	6.85	7.76	6.81	7.92	6.83	9.43	6.74

(Upper) Due to the neglect of optimizing output quantization error (cosine similarity in the gray cells), QuaRot results in higher PPL even with fewer outlier issues. (Down) The gray raw indicates the results are obtained with `down_proj` layers transformed.

Code **Document** **Paper**

When Should We Utilize the Weight Clipping?

Symmetric or asymmetric. Symmetric clipping with symmetric quantization maintains more information (*i.e.*, solid gray box) than with asymmetric quantization, and for asymmetric clipping vice versa. This finding can help improve current methods with significant accuracy recovery, especially for extremely lower bit-width.



#Bits	Method	LLaMA-2-7B Avg. PPL↓	LLaMA-2-7B Avg. Acc.↑	LLaMA-2-70B Avg. PPL↓	LLaMA-2-70B Avg. Acc.↑
w3a16g128	AWQ	7.25	61.18	4.90	80.95
	AWQ *w/* asym. clip	7.21	61.59	4.89	81.07
w2a16g64	AWQ	1.8e5	37.69	6.8e4	32.84
	AWQ *w/* asym. clip	13.26	48.77	6.49	75.31

Bit-width.

1. For higher bit (4-bit) weight-only quantization, clipping has a side-effect, unlike improvement for lower-bit (3-bit).
2. For weight activation quantization, suitable clipping shows positive effects whatever bit-width.

Model	w3a16g128		w4a16g128		w6a6		w8a8	
	w/ clip	*w/o* clip	*w/* clip	*w/o* clip	*w/* clip	*w/o* clip	*w/* clip	*w/o* clip
LLaMA-3-8B	11.74	11.23	11.99	17.42	10.35	9.46	10.73	10.35
	30.60	24.80	40.60	42.20	40.60	39.40	43.80	43.80
LLaMA-3-70B	8.08	7.57	9.09	11.62	26.38	25.75	16.79	16.66
	54.00	54.20	59.20	60.00	58.20	58.20	60.20	57.60

Accuracy on GPQA is highlighted in gray rows, and the rest for MBPP.

Should We Combine Transformation and Reconstruction?

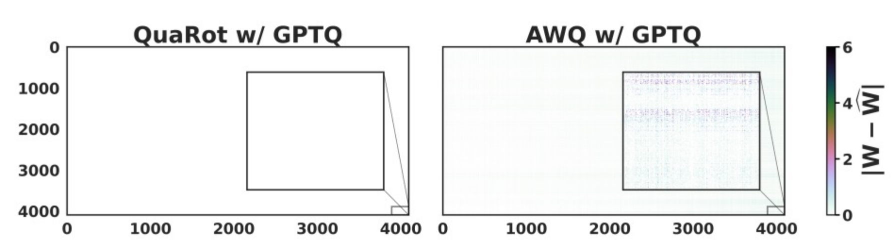
Observations.

1. The scaling-based transformation like AWQ *w/* GPTQ shows moderate improvement for LLaMA-3-8B.
2. However, The rotation-based method QuaRot *w/* GPTQ far surpasses QuaRot alone.

Metric	GPTQ	AWQ	AWQ *w/* GPTQ	QuaRot	QuaRot *w/* GPTQ
Avg. PPL↓	10.67	10.98	10.55	50.00	10.35
Avg. Acc.↑	71.96	70.72	72.72	45.90	74.84

Reasons.

1. Scaling-based transformation methods may amplify weight outliers . This gives rise to a larger challenge for iterative compensation during the reconstruction, especially weights in rear columns which GPTQ can not properly deal with. However, QuaRot, which effectively eliminates weight outliers, pairs well with GPTQ.
2. Rotation-based transformation only aims to decrease tensor outliers without considering output errors, so the kurtosis value is significantly reduced. GPTQ exactly considers the output error through approximated Hessian matrix, and thus can always complement rotation-based transformation.



Method	q_proj	k_proj	v_proj	o_proj	gate_proj	up_proj	down_proj
QuaRot	0.9962	0.9967	0.9797	0.8286	0.9764	0.9579	0.9230
QuaRot *w/* GPTQ	0.9971	0.9975	0.9847	0.9476	0.9895	0.9791	0.9529

Output cosine similarity between the original layer and the quantized layer.

Integer or Floating-point Quantization?

Full Prec.	w3a16g128		w4a16g128		w4a16		w4a4		w6a6	
	Naive	AWQ	Naive	AWQ	Naive	AWQ	Naive	SmoothQuant	Naive	SmoothQuant
5.47	6.66	6.19	5.78	5.59	6.11	5.81	NaN	NaN	6.86	6.77
	6.89	6.38	5.70	5.63	5.89	5.75	90.85	16.35	5.56	5.56

INT (gray rows)/FP (white rows) quantization. Naive means simple round-to-nearest quantization.

Observations.

1. For the weight-activation quantization, FP quantization consistently surpasses INT quantization by a large margin as it can better overcome the outlier issue.

2. Conversely, when applying weight-only quantization, the FP quantization achieves worse performance under ultra-low-bit ($\leq$ 3-bit) or small group size.

Insights.

1. The positive zero and negative zero in FP format constrain the representation capability of this quantization type, particularly under low-bit.
2. The range of small group size is more uniform, which is unsuitable for FP quantization.
3. The symmetric FP quantization struggles to deal with the asymmetry in LLMs.

Reproducibility of LLMC

w4a16g128	MMLU	BoolQ	ARC-e	PIQA
AWQ	46.36	71.25	54.14	77.04
AWQ-LLMC	46.47	71.62	53.96	77.26
GPTQ	43.36	72.81	51.50	77.86
GPTQ-LLMC	43.40	72.91	51.50	77.75

w8a8	MMLU	BoolQ	ARC-e	PIQA
SmoothQuant	46.17	69.76	49.03	77.26
SmoothQuant-LLMC	46.28	69.08	50.97	77.26
QuaRot *w/* GPTQ	46.38	71.50	52.73	77.75
QuaRot-LLMC + *w/* GPTQ-LLMC.	46.42	70.61	53.26	77.97