PB-LLM: PARTIALLY BINARIZED LARGE LANGUAGE MODELS

ICLR 2024

BILLM: PUSHING THE LIMIT OF POST-TRAINING QUANTIZATION FOR LLMS

Shared by: **Jiaqi Zhao**

2024.3.12

Introduction:

大模型庞大的参数量在实际应用场景下对硬件要求极高，如何在应用场景下降低模型大小、加快推理速度并保持足够的精度是重要的研究方向。

模型量化：

一种有效的模型压缩方法，通过将网络的权值（weights）、激活值（activations）等由浮点数（如 Float32）转换为低比特数据（如 Int8、Int4）进行计算实现网络瘦身和加速。

1bit量化方法：

主要是weight-only的，QAT如BitNet、OneBit。PTQ如PB-LLM、BiLLM。

Aiming at 1-bit weight quantization.

PTQ:

Under PTQ, combining the concepts from GPTQ, the paper partially reconstructs the binarized weight matrix guided by the Hessian matrix.

Salient metric: $w_i^2 / [\mathbf{H}^{-1}]_{ii}^2$

Using the metric to search for salient weights (element-wise) and keep them into higher bits (8bit).

For non-salient weights, PB-LLM binarizes them using the following formula with GPTQ.

$$\texttt{sign}(x) = \begin{cases} +1 & x \geq 0 \\ -1 & x < 0. \end{cases}$$

GPTQ：Quantize the weights column by column during which time update the remaining weights in the block to reduce quantization error.

$$\delta_{-q} = \frac{w_q - \hat{w}_q}{[\mathbf{H}^{-1}]_{qq}} \cdot (\mathbf{H}^{-1})_{:,q},$$

$$w_{-q} := w_{-q} + \delta_{-q},$$

QAT:

For 2% salient weights selected by magnitude, PB-LLM freezes them (required_grad = False) and keeps them into FP.

For non-salient weights. a column-wise scaling factor to binarized weights is applied to reduce the binarization error:

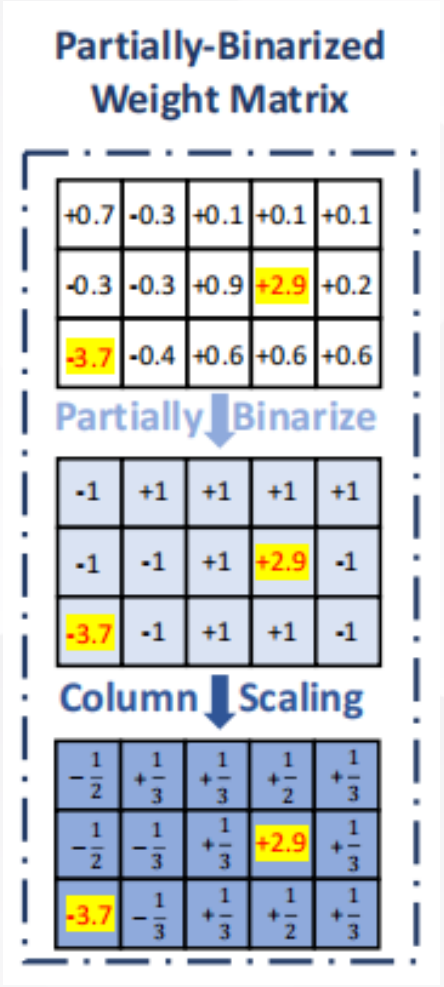
$$\mathbf{w}_F = \alpha \bar{\mathbf{w}}_B$$

The optimal values of scaling factor α can be calculated by minimizing the L2 error:

$$\alpha^\star = \arg\min_{\alpha \in \mathbb{R}_+} \mathcal{J}(\alpha), \text{ in which } \mathcal{J}(\alpha) = \|\mathbf{w}_F - \alpha \bar{\mathbf{w}}_B\|_2^2$$

$$\mathcal{J}(\alpha) = \alpha^2 \bar{\mathbf{w}}_B^T \bar{\mathbf{w}}_B - 2\alpha \mathbf{w}_F^T \bar{\mathbf{w}}_B + \mathbf{w}_F^T \mathbf{w}_F$$

$$\alpha^* = \frac{\mathbf{w}_F^T \bar{\mathbf{w}}_B}{n_{\mathbf{w}_F}} = \frac{\|\mathbf{w}_F\|_1}{n_{\mathbf{w}_F}}. \qquad \bar{\mathbf{w}}_B^T \bar{\mathbf{w}}_B = n_{\mathbf{w}_F}$$



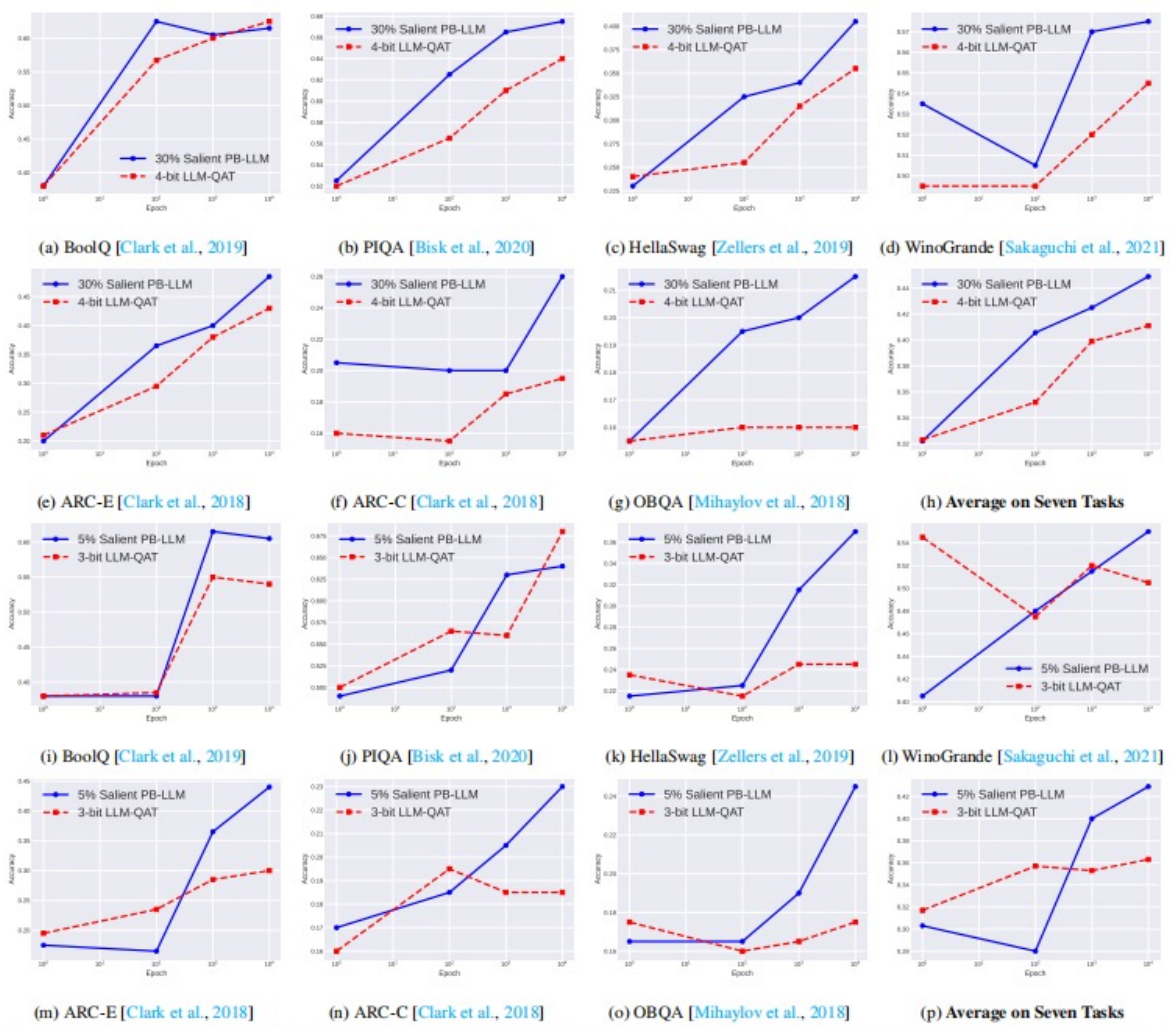


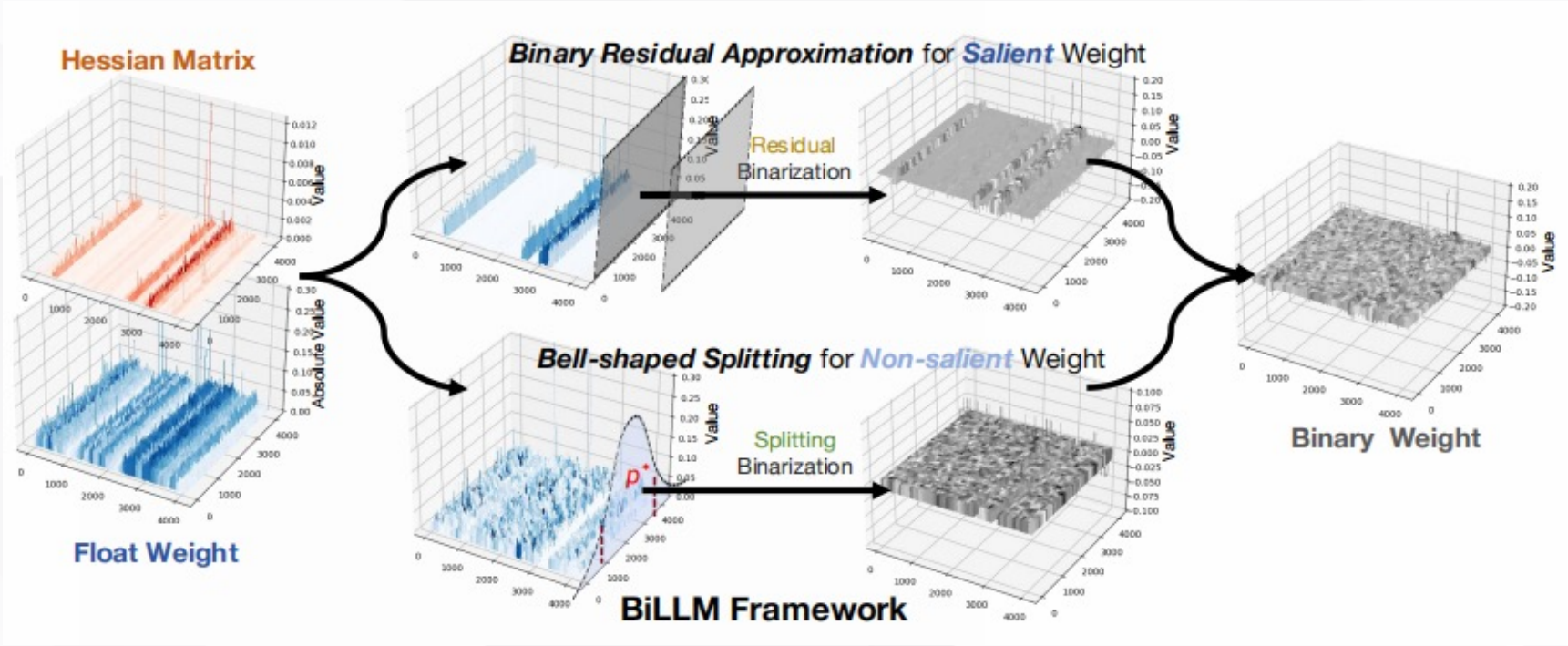
Figure 7: **QAT training results with 30% salient weights PB-LLM (upper two lines):** As fine-tuning epochs increase, quantized models swiftly regain their reasoning capacities, demonstrating the resilience and adaptability of PB-LLMin sustaining cognitive functionalities within models, despite substantial quantization; **QAT training results with 5% salient weights PB-LLM (bottom two lines)**: Existing LLM QAT methods exhibit an absolute failure when subjected to extremely-low bit conditions. In contrast, PB-LLMtriumphs in restoring the reasoning capacities of low-bit quantized LLMs. This underlines the efficacy of PB-LLM in balancing quantization and

Table 2: Zero-shot performance on Common Sense Reasoning tasks within a 4-bit setting. Reported results of previous works are documented in their papers. PB-LLM 30% denotes the preservation of 30% salient weights, and PB-LLM 10% implies the preservation of 10% salient weights.

Method	BoolQ	PIQA	HellaSwag	WinoGrande	ARC-E	ARC-C	OBQA	Avg
FP LLaMA-7B	76.8	79.3	76.1	70.0	73.0	48.0	57.6	68.7
RTN	71.2	77.3	72.7	66.9	68.8	46.4	52.8	65.2
SmoothQuant	67.7	76.0	69.4	66.7	66.9	43.0	50.6	63.0
LLM-QAT	75.5	78.3	74.0	69.0	70.0	45.0	55.4	66.6
PB-GPTQ 10%	62.3	55.9	27.7	49.3	29.3	20.1	10.6	36.5
PB-GPTQ 30%	73.5	74.9	47.5	64.9	61.3	32.4	25.2	54.2
PB-LLM 10%	68.9	67.8	68.1	67.4	58.7	42.9	50.6	60.6
PB-LLM 30%	75.7	78.0	74.3	69.7	69.0	45.6	55.8	66.9

Table 3: Perplexity of C4, wikitext2 and PTB on LLaMA-7b quantized with PTQ methods.

	C4	WIKI	PTB
FP	7.3435	5.6770	41.1509
GPTQ 4b	8.6977	8.1368	57.9951
SparseGPT 50%	15.5949	12.829483	505.1396
PB-GPTQ 50%	8.1466	6.3089	54.8674
PB-GPTQ 20%	20.6057	17.1929	280.4353
PB-GPTQ 10%	72.1115	85.7838	708.4120
PB-GPTQ 5%	401.6475	619.1054	1687.1815



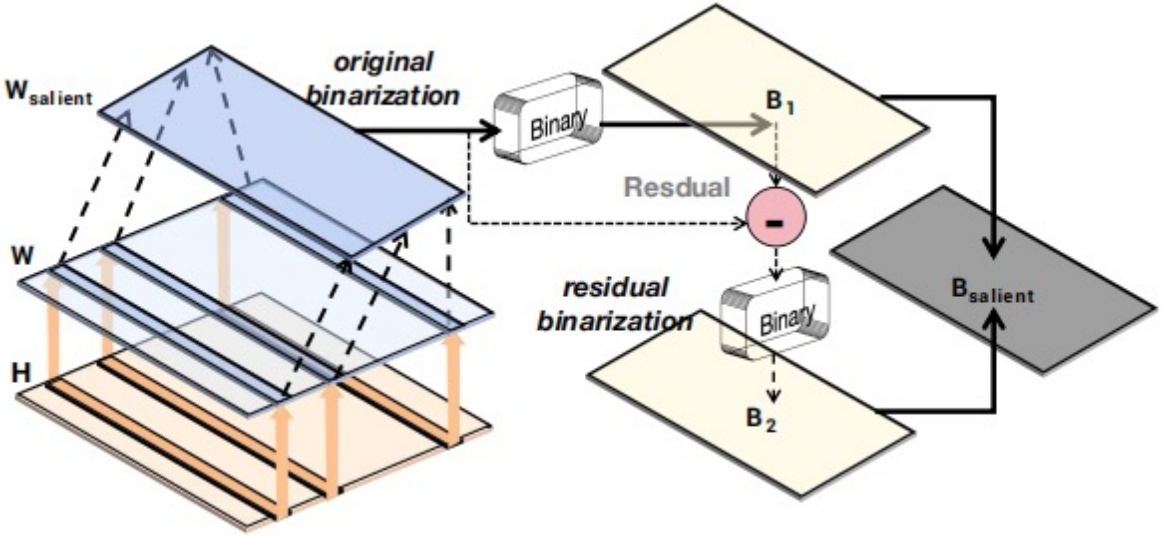
Binarize both salient weights and non-salient weights.

Following PB-LLM, BiLLM selects salient weights based on hessian metric.

For salient weights, BiLLM develops a residual approximation approach to further minimize binarization error.

$$\begin{cases} \alpha_o^*, \mathbf{B}_o^* = \arg\min_{\alpha_o, \mathbf{B}_o} ||\mathbf{W} - \alpha_o \mathbf{B}_o||^2), \\ \alpha_r^*, \mathbf{B}_r^* = \arg\min_{\alpha_r, \mathbf{B}_r} ||(\mathbf{W} - \alpha_o^* \mathbf{B}_o^*) - \alpha_r \mathbf{B}_r||^2), \end{cases}$$

$$\mathbf{W} \approx \alpha_o^* \mathbf{B}_o^* + \alpha_r^* \mathbf{B}_r^*.$$



For non-salient weights, BiLLM observes that the remaining non-salient weights maintains a bell-shaped distribution which is closer to symmetric after the removal of salient weights.

The segmentation process identifies a break-point that categorizes non-salient weights into two groups: Ac[−p, p] for concentrated weights and As[−m, −p] ∪ [p, m] for sparse weights.

$$\theta_{q,p}^2 = ||\mathbf{W}_s - \alpha_s \mathbf{B}_s||^2 + ||\mathbf{W}_c - \alpha_c \mathbf{B}_c||^2,$$

$$\alpha_s = \frac{1}{n_s}||\mathbf{W}_s||_{\ell 1}, \alpha_c = \frac{1}{n_c}||\mathbf{W}_c||_{\ell 1},$$

$$p^* = \arg\min_p(\theta_{q,p}^2).$$

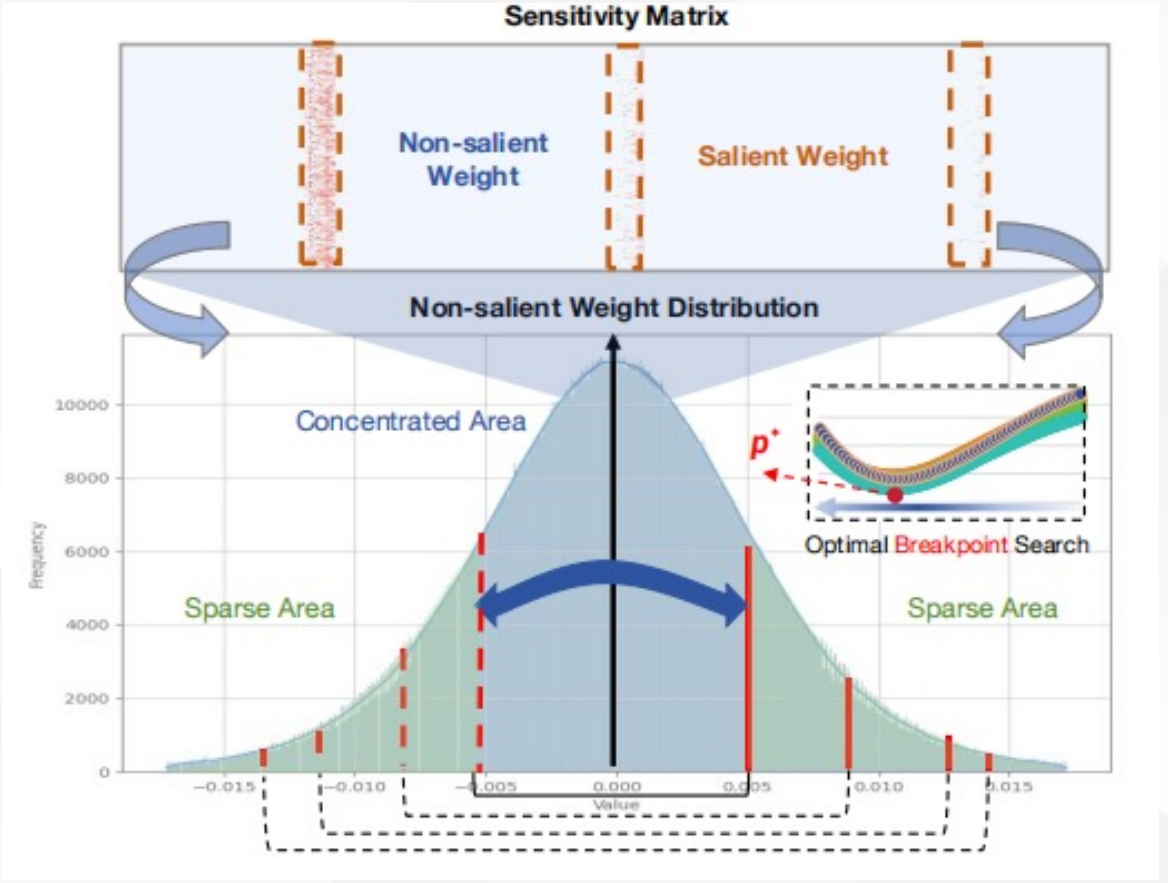


Table 2. Perplexity of RTN, GPTQ, PB-LLM, and BiLLM on OPT Family. The columns represent the perplexity results on Wikitext2 datasets with different model sizes.

Method	Block Size	Weight Bits	1.3B	2.7B	6.7B	13B	30B	66B
Full Precision	-	16.00	14.62	12.47	10.86	10.13	9.56	9.34
RTN	-	3.00	13337.38	15594.72	5797.32	3357.01	1566.00	6126.09
GPTQ	128	3.00	20.97	16.88	14.86	11.61	10.27	10.51
RTN	-	2.00	11272.65	9505.76	28363.14	194086.78	169616.47	1165864.25
GPTQ	128	2.00	115.17	61.59	50.19	21.36	15.71	82.10
RTN	-	1.00	17165.72	36516.69	11550.91	6986.35	6485.99	184796.30
GPTQ	128	1.00	14884.73	14144.58	10622.81	15196.96	12478.37	13106.45
PB-LLM †	128	1.70	265.52	124.35	105.16	81.92	25.14	29.09
BiLLM ‡	128	**1.11**	**69.97**	**49.55**	**35.36**	**18.82**	**12.71**	**12.06**

-: Vanilla RTN conducts layer-wise quantization. †: PB-LLM selects 10% elements in the original tensor as salient weights based on Hessian. ‡: BiLLM uses structural searching for salient weights. The table gives the average bit-width of the OPT family.

Table 5. Accuracy on 7 data sets, from binarization LLaMA, LLaMA2, and OPT, and we also compare the results among GPTQ, PB-LLM, and BiLLM to validate the quantization effect.

Model	Method	Weight Bits	Block Size	PIQA ↑	BoolQ ↑	OBQA ↑	Winogrande ↑	ARC-e ↑	ARC-c ↑	Hellaswag ↑
LLaMA-7B	GPTQ	2.00	128	52.8	50.0	28.2	49.3	26.6	29.5	26.3
	PB-LLM	1.70	128	54.6	59.7	30.4	50.6	28.2	24.6	28.7
	BiLLM	**1.09**	128	**61.2**	**62.7**	**31.8**	**51.1**	**36.0**	**25.7**	**36.8**
LLaMA2-7B	GPTQ	2.00	128	51.1	43.9	29.0	50.8	26.6	28.5	26.3
	PB-LLM	1.70	128	53.8	62.3	30.2	49.3	28.0	25.0	27.7
	BiLLM	**1.08**	128	**60.6**	**61.8**	**33.2**	**52.4**	**36.2**	**24.4**	**34.8**
OPT-6.7B	GPTQ	2.00	128	56.6	51.1	25.6	51.2	31.3	22.9	30.4
	PB-LLM	1.70	128	57.6	55.5	24.2	47.7	33.2	21.0	31.0
	BiLLM	**1.11**	128	**58.6**	**62.2**	**29.0**	**51.5**	**34.1**	**23.9**	**31.9**

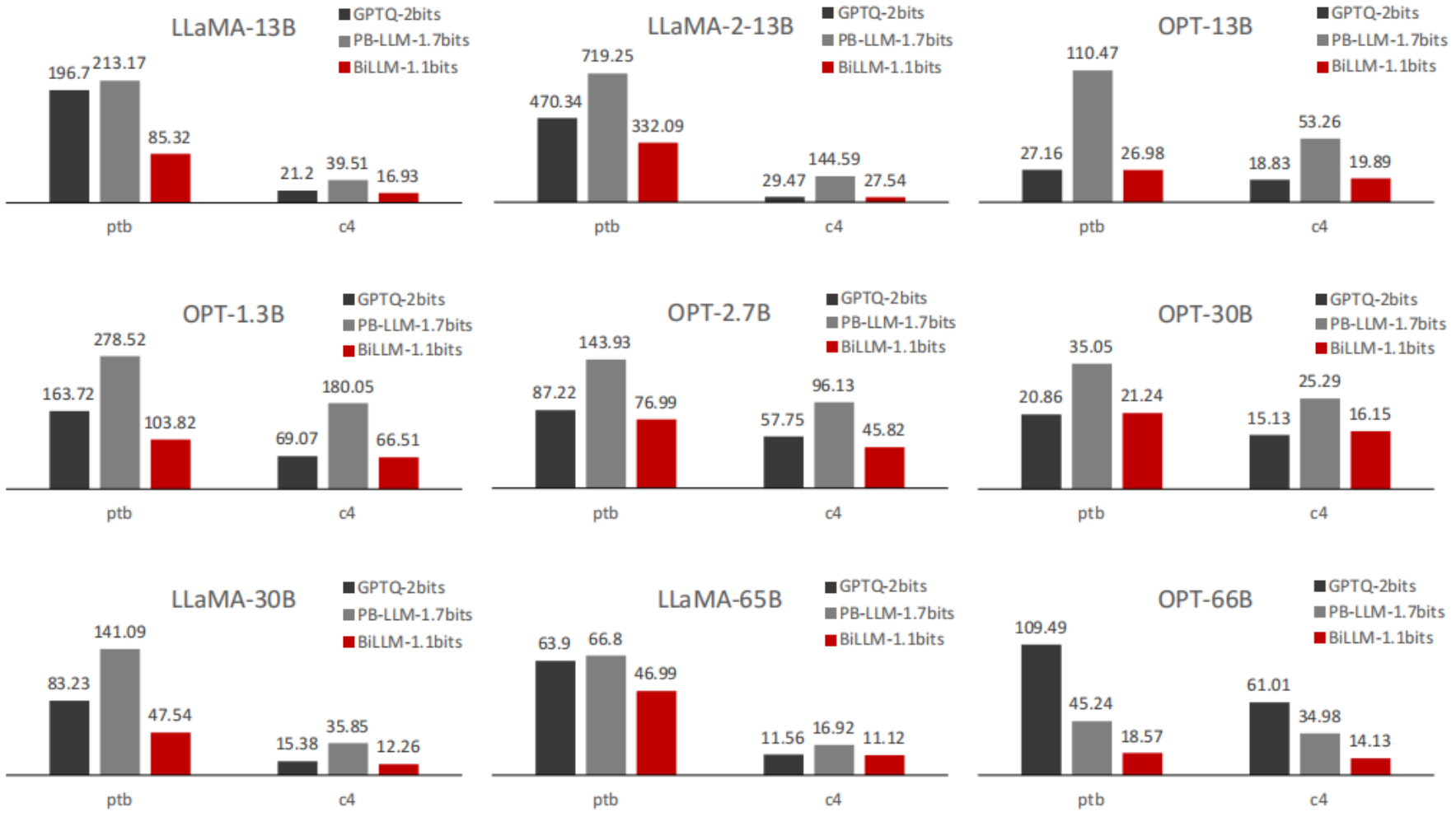


Figure 11. GPTQ, PB-LLM, BiLLM performed on the PTB and C4 datasets, mainly on LLaMA-13B, LLaMA2-13B, OPT-13B, and so on. The results showed that BiLLM performed relatively well.

THANKS FOR LISTENING