TurboBoA: Faster and Exact Attention-aware Quantization without Backpropagation

Junhan Kim, Yeo Jeong Park, Seungwoo Son,
Chungman Lee, Ho-young Kim, Joonyoung Kim, Yongkweon Jeon

Samsung Research, Korea

Background) GPTQ (ICLR'23)

- **TL;DR) Fast LLM PTQ algorithm without relying on backpropagation**
  - Procedure) iteratively conduct quantization and Hessian-guided error correction.
    - Whenever one weight is quantized, it updates the remaining full-precision weights based on the approximated Hessian $\mathbf{H}$ to compensate for the quantization error.

$$\boldsymbol{\delta}\mathbf{w} = \frac{\mathrm{Q}(w_i) - w_i}{[\mathbf{U}]_{i,i}}[\mathbf{U}]_{i,:} \quad \text{where } \mathbf{U} = \mathrm{Chol}(\mathbf{H}^{-1})^T$$

- **Main Limitation: Ignorance of Cross-layer Dependencies**
  - To maintain simplicity in the Hessian approximation, GPTQ assumes layer-wise independence and uses layer-wise reconstruction loss ($\|\Delta\mathbf{W}\mathbf{X}\|_F^2$) for the Hessian approximation (i.e., $\mathbf{H} \approx \frac{\partial\|\Delta\mathbf{W}\mathbf{X}\|_F^2}{\partial^2\Delta\mathbf{W}}$).

    → GPTQ cannot consider cross-layer dependencies, limiting its low-bit performance.

Background) BoA (ICML'25)

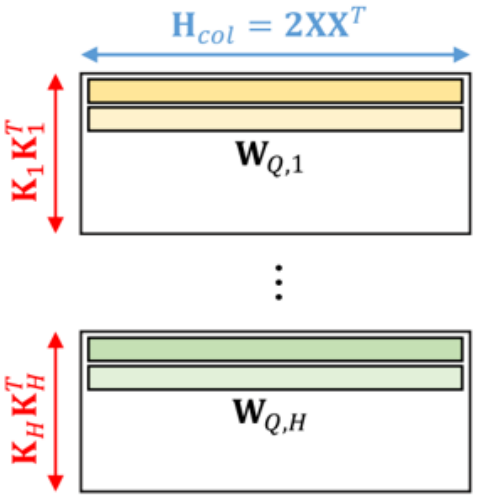
- **TL;DR) BoA improves GPTQ by considering dependencies within attention modules**
  - Key Contribution) Construction of attention-aware Hessians by exploiting attention-wise reconstruction loss

Table 1: Loss used to approximate Hessians and the corresponding Hessians in GPTQ and BoA.

Method	Layer	Loss ($\|\mathbf{G}\Delta\mathbf{W}\mathbf{X}\|_F^2$)	$\mathbf{H} = \mathbf{H}_{in} \otimes \mathbf{H}_{out}$
GPTQ	$\mathbf{W}_{\{Q,K,V\}}$	$\|\Delta\mathbf{W}\mathbf{X}\|_F^2$	$\mathbf{X}\mathbf{X}^T \otimes \mathbf{I}$
BoA	$\mathbf{W}_{Q,h}$	$\|\mathbf{K}_h\Delta\mathbf{W}_{Q,h}\mathbf{X}\|_F^2$	$\mathbf{X}\mathbf{X}^T \otimes \mathbf{K}_h^T\mathbf{K}_h$
	$\mathbf{W}_{K,h}$	$\|\mathbf{Q}_h\Delta\mathbf{W}_{K,h}\mathbf{X}\|_F^2$	$\mathbf{X}\mathbf{X}^T \otimes \mathbf{Q}_h^T\mathbf{Q}_h$
	$\mathbf{W}_{V,h}$	$\|\mathbf{W}_{out,h}\Delta\mathbf{W}_{V,h}\mathbf{X}\mathbf{A}_h^T\|_F^2$	$\mathbf{X}\mathbf{A}_h^T\mathbf{A}_h\mathbf{X}^T \otimes \mathbf{W}_{\text{out},h}^T\mathbf{W}_{\text{out},h}$

* h denotes the index of the attention head.

- **Main Limitation: Long Processing Time Incurred by Sequential Processing**
  - Since the attention-aware Hessians model dependencies between output channels, the quantization error of a certain output channel can be corrected using others.
  - To do so, each output channel must be quantized sequentially (NOT simultaneously).
    - e.g., the 2nd channel can be quantized after participating in the error correction for the 1st output channel.



Proposed TurboBoA

TurboBoA: Faster and Exact Attention-aware Quantization without Backpropagation

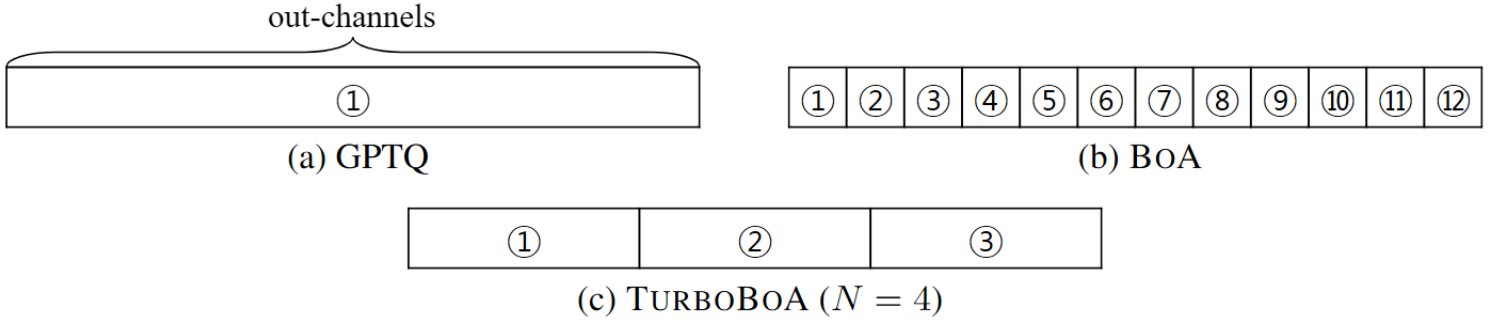
Junhan Kim, Yeo Jeong Park, Seungwoo Son, Chungman Lee,
Ho-young Kim, Joonyoung Kim, Yongkweon Jeon

- **TL;DR**
  - Improved BoA: Faster and More Accurate BoA
  - Acceleration via Joint Quantization of Multiple Output Channels
  - Performance Enhancement via Error Correction of Previously Quantized Layers and Adaptive Grid Selection

Contribution 1 – Joint Quantization of Multiple Output Channels

Key Idea

- quantize **multiple output channels jointly**, thereby reducing the number of sequential operations



- While multiple output channels are quantized together as if they were independent, their dependencies are still considered via error correction.

$$\min_{\Delta \mathbf{W}} \|\mathbf{G}\Delta\mathbf{W}\mathbf{X}\|_F^2 \; s.t. \; \mathbf{e}_1^T \Delta\mathbf{W} = \mathbf{Q}_{1,:} - \mathbf{W}_{1,:}$$

BoA

$$\min_{\Delta \mathbf{W}} \|\mathbf{G}\Delta\mathbf{W}\mathbf{X}\|_F^2 \; s.t. \; \mathbf{e}_i^T \Delta\mathbf{W} = \mathbf{Q}_{i,:} - \mathbf{W}_{i,:} \; (1 \le i \le N)$$

TurboBoA

  - New error correction rule

$$\Delta\mathbf{W} = -[\mathbf{U}_{out}^T]_{:,:N}[\mathbf{U}_{out}^T]_{:N,:N}^{-1}(\mathbf{W}_{:N,:} - \mathbf{Q}_{:N,:})$$

Contribution 1 – Joint Quantization of Multiple Rows

Effectiveness of Joint Quantization

- Through joint quantization, TurboBoA achieves more than a three-fold speedup over BoA when $N = 16$.
  - e.g., for 70B, 9~12 hours can be saved.
- Although the number of output channels participating in the error correction decreases, the observed degradation is negligible.
  - We hypothesize that this is because a large number of weights still remain, providing sufficient flexibility for error correction.

Table 2: Ablation of multiple-row processing (INT2 quantization)

Method	N	Llama3.2-1B		Llama3.2-3B		Llama3-8B		Llama3.1-70B	
		Time (min)	Wiki2 (↓)	Time (min)	Wiki2 (↓)	Time (min)	Wiki2 (↓)	Time (hr)	Wiki2 (↓)
BoA	1	13.32	40.40	59.94	32.26	94.75	15.20	16.99	7.726
BoA + **F1**	4	**6.255**	41.09	**22.68**	32.21	**39.46**	15.27	**7.683**	7.721
	8	**5.002**	41.53	**16.01**	31.66	**30.55**	15.30	**6.274**	7.714
	16	**4.363**	41.85	**12.70**	31.99	**25.30**	15.41	**5.636**	7.758
	32	**3.985**	41.75	**11.01**	32.15	**22.95**	15.22	**5.060**	7.746
	64	-	-	**10.29**	32.31	**21.56**	15.44	**4.885**	7.774

* Following BoA (Kim et al., 2025), QuaRot has been applied before quantizing weights. We note that TurboBoA reduces to GPTQ under $N = 64$ for Llama3.2-1B and $N = 128$ for other models.

Contribution 2 – Error Correction of Preceding Quantized Layers

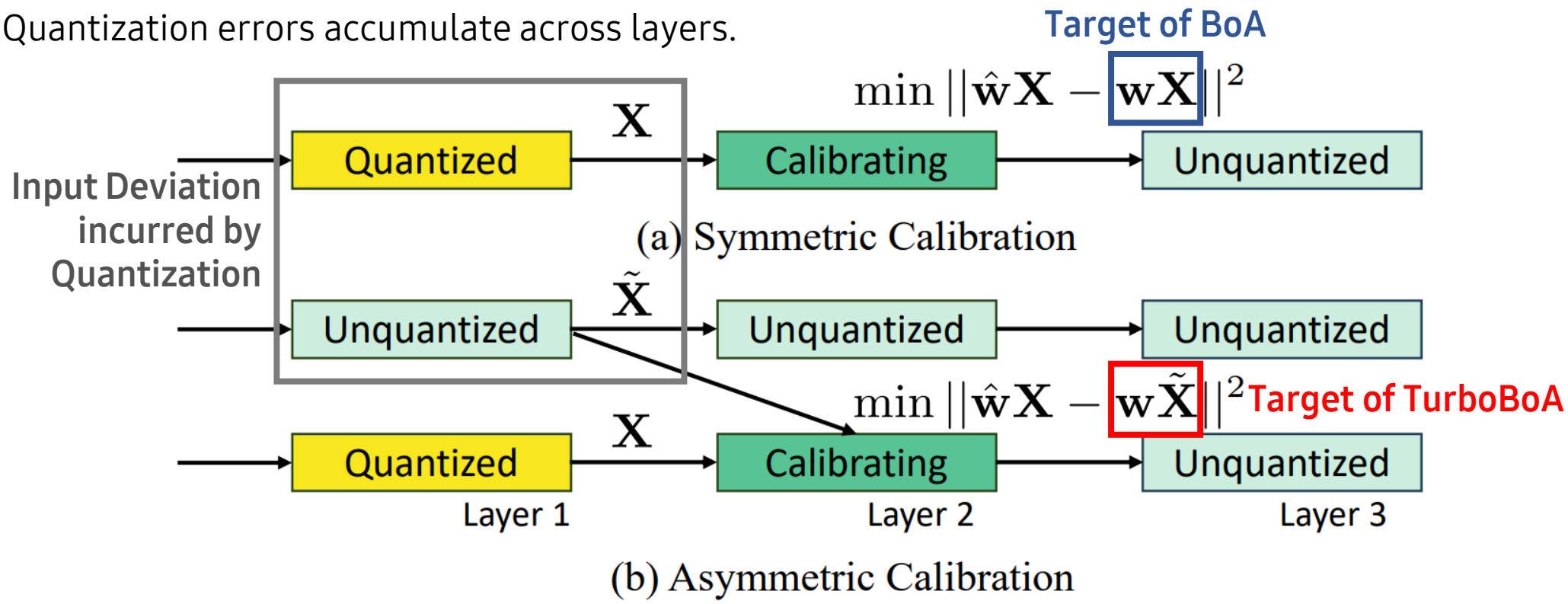
Observation

- An error produced in one Transformer block alters the inputs of subsequent blocks.

  → For blocks beyond the first, **the input $\mathbf{X}$ differs from the full precision input $\tilde{\mathbf{X}}$**.

- BoA aims to preserve the output corresponding to the quantized input $\mathbf{X}$.

  → Quantization errors accumulate across layers.



Contribution 2 – Error Correction of Preceding Quantized Layers

Correction of Preceding Quantized Layers

- The input deviation $\Delta\mathbf{X} = \mathbf{X} - \widetilde{\mathbf{X}}$ introduces additional distortion in the final output:

$$\mathbf{GQX} - \mathbf{GW}\widetilde{\mathbf{X}} = \mathbf{G}(\mathbf{Q} - \mathbf{W})\mathbf{X} + \mathbf{GW}\left(\mathbf{X} - \widetilde{\mathbf{X}}\right) = \mathbf{G}\Delta\mathbf{WX} + \mathbf{GW}\Delta\mathbf{X}$$

- When correcting the quantization errors of $\mathbf{W}_{:N,:}$, we compensate for
  - Error incurred by the weight perturbation: $\mathbf{G}_{:,:N}\Delta\mathbf{W}_{:N,:}\mathbf{X}$
  - Error incurred by the input deviation: $\mathbf{G}_{:,:N}\mathbf{W}_{:N,:}\Delta\mathbf{X}$

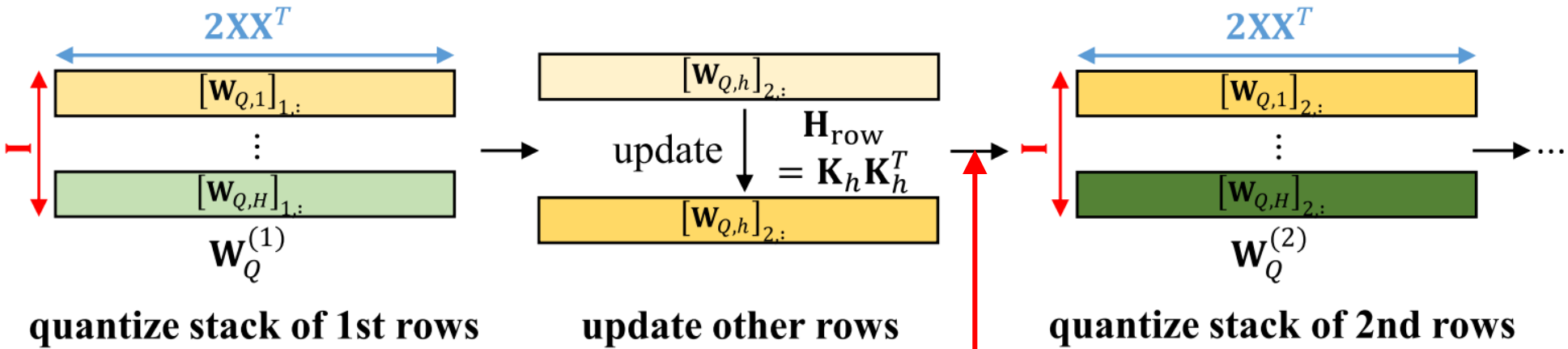
$$\min_{\Delta\mathbf{W}} \left\| \mathbf{G}\Delta\mathbf{WX} + \mathbf{G}_{:,:N}\mathbf{W}_{:N,:}\Delta\mathbf{X} \right\|_F^2 \quad s.t. \quad \mathbf{e}_i^T \Delta\mathbf{W} = \mathbf{Q}_{i,:} - \mathbf{W}_{i,:} \; (1 \le i \le N)$$

- New error correction rule

$$[\Delta\mathbf{W}]_{N:,:} = -[\mathbf{U}_{out}^T]_{N:,:N}[\mathbf{U}_{out}^T]_{:N,:N}^{-1}((\mathbf{W}_{:N,:} - \mathbf{Q}_{:N,:}) - \mathbf{W}_{:N,:}\Delta\mathbf{X}\mathbf{X}^T\mathbf{H}_{in}^{-1})$$

Contribution 3 – Adaptive Grid Selection

- **Key Idea**
  - determine quantization grids **right before** the quantization to align them with the updated weight distribution
    - In BoA, grids remain fixed throughout the iterative process despite successive updates



BoA: determine grids for all output channels

TurboBoA: determine grids for the 1st output channel

TurboBoA: determine grids for the 2nd output channel

Performance of TurboBoA

Performance Enhancement over BoA

- Error compensation of preceding Transformer blocks (F2) improves PPL greatly.
  - This feature incurs non-negligible overhead because computation of the full precision input $\widetilde{\mathbf{X}}$ is additionally needed to obtain the input deviation $\Delta\mathbf{X}$.
- Adaptive grid selection (F3) enhances PPL with a marginal cost.
- Combining the two features yields the best performance.
  - Thanks to the reduced number of sequential operations, TurboBoA is still faster than BoA (2.5~3.5x).

Table 3: Ablation of features targeting performance enhancement (INT2 quantization)

Method	F2	F3	Llama3.2-1B			Llama3.2-3B			Llama3-8B		
			Wiki2 (↓)	C4 (↓)	Time	Wiki2 (↓)	C4 (↓)	Time	Wiki2 (↓)	C4 (↓)	Time
BoA			40.40	104.9	13.32	32.26	79.17	59.94	15.20	36.95	94.75
TurboBoA ($N = 16$)			41.85	108.1	4.363	31.99	80.09	12.70	15.41	38.96	25.30
	✓		37.15	92.58	6.253	25.92	63.48	17.51	14.21	34.67	40.16
		✓	39.45	107.3	4.426	31.12	73.57	12.84	15.01	36.40	25.39
	✓	✓	33.33	85.55	6.263	24.10	54.20	17.71	13.54	32.99	40.20

* Time in minutes. QuaRot has been applied before quantizing weights.