RETHINKING THE QUANTIZATION ERROR LIFECYCLE IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-training quantization (PTQ) is essential for efficient deployment of large language models (LLMs), yet aggressive low-bit quantization can severely degrade model accuracy. Existing GPTQ-style compensation-based methods sequentially quantize weight columns, typically determining the quantization decision of the current column using round-to-nearest (RTN) and then updating the remaining weights to compensate the induced error. However, the RTN-based quantization decision is made independently of subsequent compensation, thereby treating error formation and error compensation as separate stages. In this work, we uncover the overlooked coupling between error formation and error compensation and formalize this coupling through the quantization error lifecycle, a systematic perspective on compensation-based LLM quantization. Building on this perspective, we derive a shared compensation representation that reveals complementary information for error formation and error compensation. From this shared representation, we develop CARE-Q, a unified compensation-based quantization framework comprising DiCR for compensation-aware rounding and ODPP for enhanced error compensation. Extensive experiments across diverse LLMs and quantization settings show that CARE-Q consistently outperforms leading compensation-based quantization methods and complements representative transformation-based approaches, including QuaRot and FlatQuant, while introducing no additional inference-time overhead. Our code is provided in the supplementary material and at `https://anonymous.4open.science/r/CARE-Q/`.

1 INTRODUCTION

Large language models (LLMs) have achieved remarkable success across a wide range of applications, but their growing scale imposes substantial memory and computational costs. Post-training quantization (PTQ) alleviates these costs by converting high-precision models into low-bit formats without additional training. However, aggressive quantization introduces substantial errors, making accuracy preservation increasingly difficult at low bit widths.

Recent LLM PTQ methods mainly follow two complementary directions: transformation-based and compensation-based quantization. Transformation-based methods, such as SmoothQuant (Xiao et al., 2023), QuaRot (Ashkboos et al., 2024), SpinQuant (Liu et al., 2025), and FlatQuant (Sun et al., 2025), reshape weight or activation distributions before quantization to improve quantization-friendliness. Compensation-based methods instead reduce reconstruction errors introduced during quantization. Representative methods such as GPTQ (Frantar et al., 2023), GPTAQ (Li et al., 2025), and ResComp (Li et al., 2026) sequentially quantize weight columns, typically using RTN for the current quantization decision and second-order updates to compensate the induced error in the remaining weights. GPTAQ further extends this framework with asymmetric calibration to account for errors accumulated along the quantized path.

Despite their effectiveness, existing compensation-based methods leave the interaction between error formation and error compensation largely unexplored. As illustrated in Figure 1(a), the current quantization decision is made independently of subsequent compensation, treating the two stages separately. However, the quantization decision determines the error that compensation must subsequently address, indicating that error formation and error compensation are inherently coupled. We capture this coupling through the *quantization error lifecycle* and derive a compensation represen-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

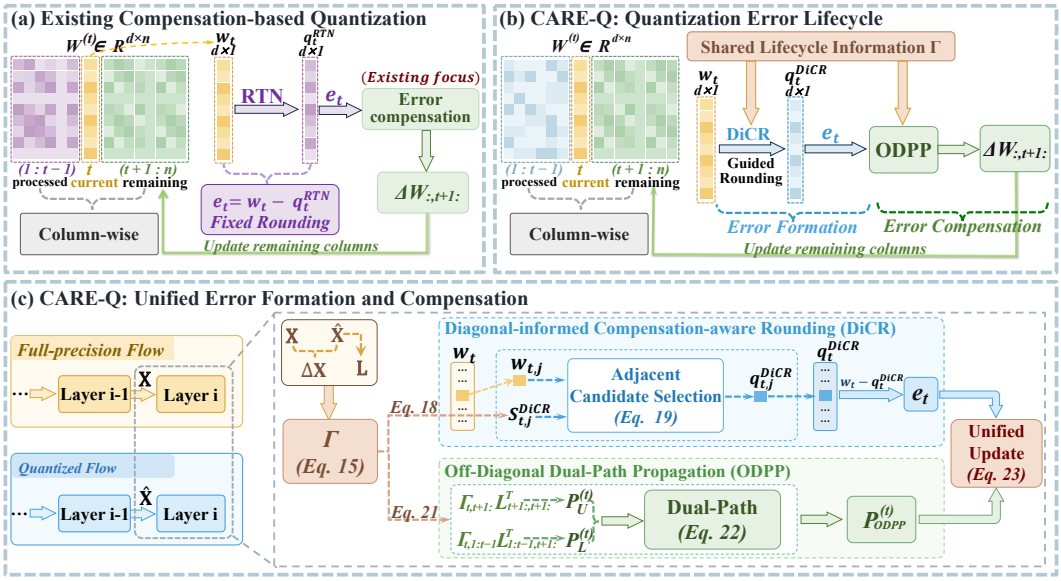

Figure 1: Overview of the quantization error lifecycle and CARE-Q. (a) Existing GPTQ-style methods use RTN for current-column quantization independently of subsequent compensation. (b) CARE-Q models the interaction between error formation and error compensation, where the compensation representation $\mathbf{\Gamma}$ provides shared lifecycle information across both stages. (c) CARE-Q derives DiCR and ODPP from $\mathbf{\Gamma}$ for error formation and error compensation.

tation that reveals complementary information for both stages. This derivation leads to CARE-Q, a unified compensation-based quantization framework comprising DiCR for compensation-aware rounding and ODPP for enhanced error compensation, as illustrated in Figure 1(b,c).

CARE-Q modifies the quantization procedure rather than reshaping weight or activation distributions, making it complementary to transformation-based approaches such as QuaRot and FlatQuant. All additional computation is confined to offline quantization, while the resulting model preserves the standard low-bit quantization format and introduces no additional inference-time overhead.

Our contributions are summarized as follows:

- We uncover the overlooked coupling between error formation and error compensation and establish the *quantization error lifecycle* as a systematic perspective for compensation-based LLM quantization.
- We derive a compensation representation that reveals complementary information for error formation and error compensation, from which we develop CARE-Q with DiCR for compensation-aware rounding and ODPP for enhanced error compensation.
- Extensive experiments across diverse LLMs and quantization settings show that CARE-Q consistently outperforms leading compensation-based quantization methods, remains complementary to transformation-based methods such as QuaRot and FlatQuant, and introduces no additional inference-time overhead.

2 Related Work

Recent PTQ methods for LLMs can be broadly categorized into two orthogonal directions: transformation-based quantization and compensation-based quantization.

Transformation-based methods improve low-bit quantization by reshaping weight and activation distributions before quantization. SmoothQuant (Xiao et al., 2023) reduces activation outliers through equivalent scaling, while OmniQuant (Shao et al., 2024) learns such equivalent transformations from calibration data. AWQ (Lin et al., 2024b) leverages activation-aware scaling to preserve

salient weights, while QuIP (Chee et al., 2023) and QuIP# (Tseng et al., 2024) apply incoherence-processing transformations for ultra-low-bit weight quantization, with QuIP# further introducing lattice-based quantization. QuaRot (Ashkboos et al., 2024) introduces Hadamard-based rotations for outlier-free low-bit inference, while DuQuant (Lin et al., 2024a) further reduces quantization difficulty through distribution transformation. SpinQuant (Liu et al., 2025) learns rotation transformations to obtain more quantization-friendly representations, and FlatQuant (Sun et al., 2025) further generalizes learnable transformations for low-bit quantization.

Complementary to transformation-based methods, compensation-based quantization formulates quantization as a reconstruction problem and reduces quantization errors through sequential compensation. GPTQ (Frantar et al., 2023) performs sequential second-order error compensation while retaining RTN for the current quantization decision. GPTAQ (Li et al., 2025) extends this framework by accounting for the mismatch between full-precision and quantized activation paths, while ResComp (Li et al., 2026) further improves reconstruction through refined residual modeling. Despite these advances, existing methods still determine the quantization decision independently of subsequent compensation, leaving the interaction between error formation and error compensation largely unexplored. CARE-Q addresses this limitation by introducing the quantization error lifecycle as a unified perspective and deriving a shared compensation representation that connects error formation with subsequent error compensation.

3 METHOD

3.1 REVISITING COMPENSATION-BASED QUANTIZATION

Consider a linear layer. Let $\mathbf{W}$ and $\hat{\mathbf{W}}$ denote the full-precision and quantized weights, and let $\mathbf{X}$ and $\hat{\mathbf{X}}$ denote calibration inputs from the full-precision and quantized paths. GPTQ (Frantar et al., 2023) minimizes the output reconstruction error under the quantized-path inputs:

$$\min_{\hat{\mathbf{W}}} \left\| \mathbf{W}\hat{\mathbf{X}} - \hat{\mathbf{W}}\hat{\mathbf{X}} \right\|_F^2. \tag{1}$$

Since jointly optimizing discrete weights is intractable, GPTQ adopts column-wise sequential quantization and compensates quantization error through second-order updates to remaining columns.

GPTAQ (Li et al., 2025) further considers the mismatch between the full-precision and quantized paths and uses the full-precision output as the reconstruction target:

$$\min_{\hat{\mathbf{W}}} \left\| \mathbf{W}\mathbf{X} - \hat{\mathbf{W}}\hat{\mathbf{X}} \right\|_F^2. \tag{2}$$

ResComp (Li et al., 2026) retains this reconstruction framework and further improves compensation through refined residual modeling.

We next analyze the common sequential compensation framework underlying these methods. The working weight matrix before step t is denoted as $\mathbf{W}^{(t)}$, with current column $\mathbf{w}_t = \mathbf{W}^{(t)}_{:,t}$, and $\Delta\mathbf{X} = \mathbf{X} - \hat{\mathbf{X}}$. The current reconstruction residual is

$$\mathbf{R}^{(t)} = \mathbf{W}\mathbf{X} - \mathbf{W}^{(t)}\hat{\mathbf{X}}. \tag{3}$$

This residual captures the reconstruction mismatch at quantization step t. After quantizing the current column as $\mathbf{q}_t$, the current quantization error is defined as $\mathbf{e}_t = \mathbf{w}_t - \mathbf{q}_t$. The compensation for subsequent weight columns is denoted as $\Delta\mathbf{W}_{:,t+1:}$. The residual evolves as

$$\mathbf{R}^{(t+1)} = \mathbf{R}^{(t)} + \mathbf{e}_t\hat{\mathbf{X}}_{t,:} - \Delta\mathbf{W}_{:,t+1:}\hat{\mathbf{X}}_{t+1:,:}. \tag{4}$$

Let $\mathbf{H} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ denote the Hessian matrix of the reconstruction objective up to a constant factor, and let $\widetilde{\mathbf{H}}_t^{-1}$ denote the effective inverse Hessian after eliminating previously quantized columns. The inverse-Hessian update in Eq. (5) determines how the current quantization error is compensated through second-order updates:

$$\mathbf{W}^{(t+1)}_{:,t+1:} = \mathbf{W}^{(t)}_{:,t+1:} - \frac{\mathbf{e}_t}{[\widetilde{\mathbf{H}}_t^{-1}]_{tt}}[\widetilde{\mathbf{H}}_t^{-1}]_{t,t+1:}. \tag{5}$$

Although GPTAQ and ResComp further improve error compensation, error formation remains independent of subsequent compensation, leaving their interaction unexplored.

Sequential quantization can be viewed as an error lifecycle: quantizing $\mathbf{w}_t$ produces error $\mathbf{e}_t = \mathbf{w}_t - \mathbf{q}_t$, which is compensated through second-order updates on remaining columns $\Delta\mathbf{W}_{:,t+1:}$.

$$\underbrace{\mathbf{w}_t \to \mathbf{q}_t \to \mathbf{e}_t}_{\text{error formation}} \to \underbrace{\Delta\mathbf{W}_{:,t+1:}}_{\text{error compensation}} . \tag{6}$$

This observation raises a fundamental question: if the current quantization error will be compensated through subsequent updates, should its formation still be independent of subsequent compensation? CARE-Q reveals a previously overlooked coupling between error formation and error compensation and formulates them within a unified optimization framework.

3.2 Quantization Error Lifecycle: An Ideal Joint Objective

Building on the residual evolution in Eq. (4), we formulate an ideal objective that jointly considers the current error formation and future error compensation:

$$\min_{\mathbf{q}_t \in \mathcal{Q}_t} \min_{\Delta\mathbf{W}_{:,t+1:}} \left\| \mathbf{R}^{(t)} + \mathbf{e}_t \hat{\mathbf{X}}_{t,:} - \Delta\mathbf{W}_{:,t+1:} \hat{\mathbf{X}}_{t+1:,:} \right\|_F^2 . \tag{7}$$

Eq. (7) evaluates each quantization decision $\mathbf{q}_t$ under optimal compensation, thereby coupling error formation and error compensation.

Proposition 3.1 (Compensation-Conditioned Quantization Decision)**.** *Eliminating the optimal compensation in Eq. (7) gives the objective induced by different quantization decisions:*

$$J_t(\mathbf{q}_t) = c_t + \omega_t \left\| \mathbf{e}_t - \mathbf{s}_t^\star \right\|_2^2 , \qquad \omega_t > 0. \tag{8}$$

where c_t is independent of the quantization decision, and $\mathbf{s}_t^\star$ denotes the compensation-induced shift that reflects how future compensation affects the current quantization decision. The optimal quantization decision therefore satisfies

$$\mathbf{q}_t^\star = \arg\min_{\mathbf{q} \in \mathcal{Q}_t} \left\| \mathbf{w}_t - \mathbf{q} - \mathbf{s}_t^\star \right\|_2^2 . \tag{9}$$

The proof and the closed-form expressions of ω_t and $\mathbf{s}_t^\star$ are provided in the appendix.

In contrast, existing compensation-based methods determine the current quantization decision using RTN without incorporating subsequent compensation:

$$\mathbf{q}_t^{\mathrm{RTN}} = \arg\min_{\mathbf{q} \in \mathcal{Q}_t} \left\| \mathbf{w}_t - \mathbf{q} \right\|_2^2 . \tag{10}$$

When $\mathbf{s}_t^\star = \mathbf{0}$, Eq. (9) reduces to Eq. (10), showing that RTN is the special case where the compensation-induced shift vanishes and the decision relies only on local quantization distance.

More generally, a nonzero $\mathbf{s}_t^\star$ indicates that the optimal quantization decision should account for the effect of future compensation, leading to compensation-conditioned error formation. Proposition 3.1 therefore establishes the theoretical basis for coupling error formation and error compensation.

3.3 From Ideal Objective to Efficient Realization

Proposition 3.1 shows that the ideal quantization decision depends on $\mathbf{s}_t^\star$. Directly evaluating this residual-conditioned shift at every sequential step is computationally prohibitive. CARE-Q instead constructs a reusable compensation representation that can be shared across quantization steps.

With $\Delta\mathbf{X} = \mathbf{X} - \hat{\mathbf{X}}$, the current reconstruction residual admits the exact decomposition

$$\mathbf{R}^{(t)} = (\mathbf{W} - \mathbf{W}^{(t)})\mathbf{X} + \mathbf{W}^{(t)}\Delta\mathbf{X}. \tag{11}$$

This decomposition separates the weight-state mismatch from the input-path discrepancy contribution. As shown by the closed-form ideal shift in Eq. (34), $\mathbf{s}_t^\star$ is linear in $\mathbf{R}^{(t)}$ at a fixed sequential step, so the two residual components contribute additively to the ideal shift. While both depend

on the evolving quantization state, the input-path component admits a factorization that separates $\mathbf{W}^{(t)}$ from an operator determined solely by fixed calibration statistics. We therefore formulate the following input-path compensation regression:

$$\Delta\mathbf{W}^{\star} = \arg\min_{\Delta\mathbf{W}} \left\| \mathbf{W}^{(t)}\Delta\mathbf{X} - \Delta\mathbf{W}\hat{\mathbf{X}} \right\|_F^2 . \tag{12}$$

Its closed-form solution factorizes the input-path compensation into the current quantization state $\mathbf{W}^{(t)}$ and a reusable operator determined by calibration statistics and inverse-Hessian geometry:

$$\Delta\mathbf{W}^{\star} = \mathbf{W}^{(t)}\Delta\mathbf{X}\hat{\mathbf{X}}^{\top}\mathbf{H}^{-1}, \qquad \mathbf{H} = \hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}. \tag{13}$$

Following the inverse-Hessian formulation, we write

$$\mathbf{H}^{-1} = \mathbf{L}\mathbf{L}^{\top}. \tag{14}$$

The Cholesky factor exposes the triangular structure required by sequential propagation and can be precomputed before quantizing the current layer. Substituting this decomposition gives

$$\Delta\mathbf{W}^{\star}_{:,t+1:} = \mathbf{W}^{(t)}\mathbf{\Gamma}(\mathbf{L}^{\top})_{:,t+1:}, \tag{15}$$

where $\mathbf{\Gamma} = \Delta\mathbf{X}\hat{\mathbf{X}}^{\top}\mathbf{L}$ is determined by the fixed calibration statistics and inverse-Hessian geometry of the current layer, and can therefore be precomputed and shared across sequential quantization steps. This representation forms the common basis for the error-formation and error-compensation mechanisms introduced next.

3.4 CARE-Q: Unified Error Formation and Compensation

Building upon the compensation representation in Eq. (15), CARE-Q reveals that $\mathbf{\Gamma}$ provides shared lifecycle information connecting error formation and error compensation. For a sequential quantization step t, we define the propagated compensation information as

$$\mathcal{F}_t = \mathbf{\Gamma}_{t,:}(\mathbf{L}^{\top})_{:,t+1:}. \tag{16}$$

Proposition 3.2 (Quantization Error Lifecycle Representation via Cholesky Decomposition)**.** *For the compensation representation defined in Eq. (15), the diagonal structure associated with the current quantization step yields the signal $g_t = [\mathrm{diag}(\mathbf{\Gamma}) \odot \mathrm{diag}(\mathbf{L})]_t = \Gamma_{tt}L_{tt}$. Meanwhile, the propagated compensation information $\mathcal{F}_t$ admits the following lifecycle representation:*

$$\mathcal{F}_t = \underbrace{\Gamma_{tt}(\mathbf{L}^{\top})_{t,t+1:}}_{\textit{diagonal propagation}} + \underbrace{\left(\Gamma_{t,t+1:}(\mathbf{L}^{\top})_{t+1:,t+1:} + \Gamma_{t,1:t-1}(\mathbf{L}^{\top})_{1:t-1,t+1:}\right)}_{\textit{off-diagonal propagation}}. \tag{17}$$

The diagonal structure provides the signal g_t for the current quantization decision, while the off-diagonal structure captures complementary cross-column interactions relevant to subsequent compensation. Therefore, $\mathbf{\Gamma}$ serves as a unified representation connecting error formation and error compensation throughout the quantization lifecycle.

Diagonal-informed Compensation-aware Rounding (DiCR). Proposition 3.1 shows that the ideal quantization decision depends on the compensation-induced shift $\mathbf{s}_t^{\star}$, whose direct evaluation throughout sequential quantization is computationally prohibitive. Proposition 3.2 identifies the diagonal signal g_t from the shared lifecycle representation $\mathbf{\Gamma}$. Appendix A.3 further shows that the same scalar g_t appears exactly in the current-column self term of the input-path contribution to $\mathbf{s}_t^{\star}$. Thus, rather than reconstructing the complete ideal shift, DiCR uses this structurally grounded signal to form an efficient compensation-aware surrogate:

$$\mathbf{s}_t^{\mathrm{DiCR}} = \beta g_t \mathbf{w}_t^{\mathrm{ref}}. \tag{18}$$

Here, β is a signed calibration coefficient and $\mathbf{w}_t^{\mathrm{ref}}$ denotes the fixed full-precision reference column before sequential quantization. Appendix C.11 compares this fixed reference with the evolving working column $\mathbf{w}_t$.

With g_t precomputed, DiCR only requires column-wise candidate evaluation with $O(mn)$ complexity. Let $\mathcal{C}_t$ denote the element-wise adjacent legal candidates. The quantization decision is

$$\mathbf{q}_t^{\mathrm{DiCR}} = \arg\min_{\mathbf{q}\in\mathcal{C}_t} \left\| \mathbf{w}_t - \mathbf{q} - \mathbf{s}_t^{\mathrm{DiCR}} \right\|_2^2 . \tag{19}$$

When $\mathbf{s}_t^{\mathrm{DiCR}} = \mathbf{0}$, DiCR reduces to conventional RTN. Appendix B gives a sufficient margin condition for preserving the restricted ideal candidate decision and, without requiring this condition, bounds the excess objective induced by the surrogate–ideal shift discrepancy.

Algorithm 1 CARE-Q Quantization for One Layer

Input: Full-precision weights $\mathbf{W}$, calibration inputs $\mathbf{X}, \hat{\mathbf{X}}$, block size B
Output: Quantized weights $\hat{\mathbf{W}}$

$\mathbf{W}^{(0)} \leftarrow \mathbf{W}, \quad \Delta\mathbf{X} \leftarrow \mathbf{X} - \hat{\mathbf{X}}, \quad \hat{\mathbf{W}} \leftarrow \mathbf{0}$
$\mathbf{H} \leftarrow \hat{\mathbf{X}}\hat{\mathbf{X}}^\top, \quad \mathbf{L} \leftarrow \mathrm{chol}\left((\mathbf{H} + \mathrm{damping} \cdot \mathbf{I})^{-1}\right)$
$\mathbf{\Gamma} \leftarrow \Delta\mathbf{X}\hat{\mathbf{X}}^\top\mathbf{L}, \quad \mathbf{g} \leftarrow \mathrm{diag}(\mathbf{\Gamma}) \odot \mathrm{diag}(\mathbf{L}), \quad \mathbf{P}_{\mathrm{ODPP}} \leftarrow \mathrm{BuildODPP}(\mathbf{\Gamma}, \mathbf{L})$
for $i_1 = 0, B, 2B, \ldots$ **do**
  $i_2 \leftarrow \min(i_1 + B, n), \quad \mathcal{B} \leftarrow [i_1, i_2), \quad \mathbf{E}^{\mathrm{DiCR}} \leftarrow \mathbf{0}^{m \times |\mathcal{B}|}$
  for $t = i_1, \ldots, i_2 - 1$ **do**
    $\mathbf{w}_t \leftarrow \mathbf{W}^{(t)}_{:,t}, \quad \mathcal{C}_t \leftarrow \mathrm{AdjacentCandidates}(\mathbf{w}_t; \mathcal{Q}_t)$
    $\mathbf{s}_t^{\mathrm{DiCR}} \leftarrow \beta g_t \mathbf{W}_{:,t}, \quad \mathbf{q}_t^{\mathrm{DiCR}} \leftarrow \arg\min_{\mathbf{q} \in \mathcal{C}_t} \left\|\mathbf{w}_t - \mathbf{q} - \mathbf{s}_t^{\mathrm{DiCR}}\right\|_2^2$
    $\hat{\mathbf{W}}_{:,t} \leftarrow \mathbf{q}_t^{\mathrm{DiCR}}, \quad \mathbf{E}^{\mathrm{DiCR}}_{:,t-i_1} \leftarrow (\mathbf{w}_t - \mathbf{q}_t^{\mathrm{DiCR}})/L_{tt}$
    $\mathbf{W}^{(t+1)}_{:,t:i_2} \leftarrow \mathbf{W}^{(t)}_{:,t:i_2} - \mathbf{E}^{\mathrm{DiCR}}_{:,t-i_1}(\mathbf{L}^\top)_{t,t:i_2} + \mathbf{w}_t[\mathbf{P}_{\mathrm{ODPP}}]_{t,t:i_2}$
  end for
  $\mathbf{W}^{(i_2)}_{:,i_2:} \leftarrow \mathbf{W}^{(i_2)}_{:,i_2:} - \mathbf{E}^{\mathrm{DiCR}}(\mathbf{L}^\top)_{\mathcal{B},i_2:} + \mathbf{W}^{(i_2)}_{:,\mathcal{B}}[\mathbf{P}_{\mathrm{ODPP}}]_{\mathcal{B},i_2:}$
end for
return $\hat{\mathbf{W}}$

Off-Diagonal Dual-Path Propagation (ODPP). Complementing DiCR on error formation, ODPP exploits propagation information encoded in the same compensation representation to improve error compensation. The off-diagonal components of $\mathbf{\Gamma}$ encode cross-column compensation information under the Cholesky-induced propagation structure. For the current quantization step t, the corresponding row of $\mathbf{\Gamma}$ can be partitioned as

$$\mathbf{\Gamma}_{t,:} = [\mathbf{\Gamma}_{t,1:t-1}, \mathbf{\Gamma}_{tt}, \mathbf{\Gamma}_{t,t+1:}]. \tag{20}$$

Under the causal propagation structure induced by the Cholesky factor, this propagation information consists of two complementary propagation paths:

$$\mathbf{P}_U^{(t)} = \mathbf{\Gamma}_{t,t+1:}(\mathbf{L}^\top)_{t+1:,t+1:}, \qquad \mathbf{P}_L^{(t)} = \mathbf{\Gamma}_{t,1:t-1}(\mathbf{L}^\top)_{1:t-1,t+1:}. \tag{21}$$

The first path captures propagation through the future-side off-diagonal structure, while the second path captures the causal contribution accumulated from previously processed columns. These two paths provide complementary propagation information for constructing the compensation update. ODPP combines both paths to construct the compensation operator:

$$\mathbf{P}_{\mathrm{ODPP}}^{(t)} = \mathbf{P}_U^{(t)} + \rho\mathbf{P}_L^{(t)}. \tag{22}$$

The coefficient ρ controls the contribution of the accumulated propagation path $\mathbf{P}_L^{(t)}$.

Unified CARE-Q Update. At each quantization step, DiCR incorporates compensation information into error formation through compensation-aware rounding, while ODPP exploits propagation information to perform error compensation. Sharing the same compensation representation, CARE-Q performs the unified update:

$$\mathbf{W}^{(t+1)}_{:,t+1:} = \mathbf{W}^{(t)}_{:,t+1:} - \frac{\mathbf{w}_t - \mathbf{q}_t^{\mathrm{DiCR}}}{L_{tt}}(\mathbf{L}^\top)_{t,t+1:} + \mathbf{w}_t\mathbf{P}^{(t)}_{\mathrm{ODPP}}. \tag{23}$$

This update realizes the quantization error lifecycle by coupling the DiCR decision and ODPP compensation through the shared representation $\mathbf{\Gamma}$. The layerwise quantities required by DiCR and ODPP are precomputed once and reused throughout sequential quantization. Following the GPTQ-style lazy-batch schedule, CARE-Q preserves the column-wise sequential dependency within each block, while cross-block propagation is deferred and uses the post-update block working state, as detailed in Algorithm 1. All reported experiments use this blockwise implementation.

CARE-Q is confined to offline weight reconstruction. Its compensation structures are used only to guide the reconstruction process and are not retained in the final quantized model, which preserves the standard low-bit quantization format. Consequently, CARE-Q requires no additional inference-time operators or auxiliary states, incurring **zero additional inference overhead**.

Table 1: W2A16KV16 weight-only quantization results. Each entry reports WikiText PPL / C4 PPL / 8-task zero-shot average accuracy. FP denotes the full-precision. Best results are in bold.

Config	Method	Llama3-8B Wiki ↓	Llama3-8B C4 ↓	Llama3-8B Avg ↑	Llama3.1-8B Wiki ↓	Llama3.1-8B C4 ↓	Llama3.1-8B Avg ↑	Llama3.2-3B Wiki ↓	Llama3.2-3B C4 ↓	Llama3.2-3B Avg ↑	Qwen3-4B Wiki ↓	Qwen3-4B C4 ↓	Qwen3-4B Avg ↑	Qwen3-8B Wiki ↓	Qwen3-8B C4 ↓	Qwen3-8B Avg ↑	Qwen3-14B Wiki ↓	Qwen3-14B C4 ↓	Qwen3-14B Avg ↑
–	FP	6.1	9.4	67.1	7.2	11.4	68.2	11.0	16.5	60.9	13.7	19.9	64.6	9.7	15.4	67.2	8.6	13.8	70.3
QuaRot (PC)	GPTQ	40.8	52.0	43.7	38.3	52.3	46.3	93.0	111.6	39.9	61.1	73.1	41.0	22.0	29.3	**53.6**	15.0	21.3	60.5
	GPTAQ	23.1	27.5	49.1	22.2	29.2	50.0	42.7	46.0	42.6	42.6	49.9	42.6	21.9	27.9	51.9	15.5	21.1	59.4
	ResComp	23.3	27.7	48.8	23.8	30.9	49.9	44.5	46.8	43.1	42.1	50.3	42.1	21.3	27.3	52.5	16.2	21.3	58.8
	CARE-Q	**21.3**	**25.2**	**49.7**	**18.5**	**24.7**	**51.8**	**39.0**	**42.1**	**44.5**	**40.3**	**47.8**	**42.7**	**19.9**	**26.6**	52.8	**14.2**	**20.3**	**60.8**
QuaRot (G128)	GPTQ	35.4	47.3	45.2	29.6	41.9	47.9	75.0	84.3	39.2	39.1	52.0	41.6	18.2	25.7	54.0	13.1	19.6	62.0
	GPTAQ	19.5	25.0	50.6	19.9	27.2	52.2	33.7	38.6	44.1	32.5	40.8	43.0	18.3	25.2	53.0	13.7	19.6	60.5
	ResComp	21.1	26.4	49.4	19.3	25.7	52.4	37.3	40.9	43.7	32.0	39.1	43.3	17.8	24.6	52.3	13.7	19.3	61.0
	CARE-Q	**17.0**	**21.5**	**51.9**	**17.3**	**23.8**	**52.6**	**33.0**	**36.6**	**44.9**	**30.5**	**39.0**	**43.8**	**17.4**	**24.1**	**54.9**	**13.1**	**19.0**	**62.3**
FlatQuant (PC)	GPTQ	21.5	26.9	50.6	24.8	33.0	53.5	48.5	53.5	46.1	28.4	34.6	52.0	19.3	26.3	56.0	15.1	19.9	62.7
	GPTAQ	18.3	22.8	51.4	19.8	25.7	53.2	33.5	36.5	47.3	31.2	34.5	50.6	19.6	24.9	55.5	15.7	19.9	62.2
	ResComp	18.7	22.9	51.4	20.7	26.3	51.8	35.0	37.9	46.8	30.9	33.8	50.4	20.3	25.0	55.0	15.9	20.0	61.8
	CARE-Q	**14.7**	**19.9**	**52.8**	**16.4**	**23.3**	**54.8**	**26.6**	**32.2**	**47.6**	**26.6**	**31.4**	**52.2**	**16.7**	**23.4**	**57.2**	**14.1**	**19.2**	**62.7**
FlatQuant (G128)	GPTQ	18.7	24.5	51.1	21.1	29.7	53.1	46.5	43.7	46.9	25.7	31.9	51.8	16.6	23.3	57.1	13.0	18.9	63.5
	GPTAQ	16.1	21.1	53.0	17.8	23.7	54.3	36.1	33.8	47.4	25.5	30.9	51.2	17.6	23.8	55.6	14.0	18.9	62.8
	ResComp	16.8	21.2	51.8	17.8	23.7	54.0	31.6	34.5	46.6	24.8	30.8	50.6	18.3	23.8	55.9	13.8	19.0	62.4
	CARE-Q	**13.8**	**19.5**	**54.0**	**15.4**	**22.3**	**54.9**	**25.8**	**30.0**	**48.3**	**23.9**	**30.0**	**52.5**	**15.3**	**22.7**	**57.5**	**12.7**	**18.4**	**63.5**

4 EXPERIMENTS

Models and Evaluation Protocol. To enable direct and fair comparison with ResComp (Li et al., 2026), we follow its benchmark protocol wherever applicable. We evaluate CARE-Q on eight representative LLMs spanning model scales from 3B to 70B, including Llama-3.2-3B-Instruct, Llama-3-8B, Llama-3.1-8B-Instruct, and Llama-3-70B (Grattafiori et al., 2024), together with Qwen3-4B, Qwen3-8B, Qwen3-14B, and Qwen3-32B (Yang et al., 2025). We compare against representative compensation-based PTQ methods, including GPTQ (Frantar et al., 2023), GPTAQ (Li et al., 2025), and ResComp (Li et al., 2026). Following the same protocol, we use 128 calibration samples with a sequence length of 2048, using C4 (Raffel et al., 2020) for weight-only quantization and WikiText (Merity et al., 2017) for full quantization. We evaluate perplexity on WikiText and C4 and report average zero-shot accuracy over eight downstream tasks: ARC-Easy, ARC-Challenge, PIQA, SIQA, HellaSwag, OBQA, WinoGrande, and BoolQ (Clark et al., 2018; Bisk et al., 2020; Sap et al., 2019; Zellers et al., 2019; Mihaylov et al., 2018; Sakaguchi et al., 2020; Clark et al., 2019).

Quantization Settings. We consider both weight-only and full quantization. For weight-only quantization, we evaluate direct quantization and two strong representative transformation-based methods: QuaRot (Ashkboos et al., 2024), a training-free rotation method, and FlatQuant (Sun et al., 2025), a learnable transformation method. For full quantization, we evaluate the challenging W2A4KV4 setting with FlatQuant preprocessing, where weights, activations, and KV cache are jointly quantized. Across all settings, weights are symmetrically quantized under both per-channel (PC) and group-wise (G128) configurations. Activations use per-token symmetric quantization, while the KV cache uses group-wise asymmetric quantization with group size 128.

4.1 WEIGHT-ONLY QUANTIZATION

We first evaluate CARE-Q under W2A16KV16 weight-only quantization. As shown in Table 1, CARE-Q consistently outperforms GPTQ, GPTAQ, and ResComp across the evaluated models, transformations, and quantization granularities. It achieves the best or tied-best results on 71 out of 72 reported metrics, covering WikiText and C4 perplexities as well as downstream average accuracy. For example, under FlatQuant-PC on Llama3-8B, the strongest baseline achieves WikiText/C4 perplexities of 18.3/22.8 and a downstream average of 51.4, whereas CARE-Q reaches 14.7/19.9 and 52.8, corresponding to 19.7%/12.7% lower perplexity and a 1.4-point accuracy gain.

Table 2: W2A4KV4 quantization results. Each entry reports WikiText PPL / C4 PPL / 8-task zero-shot average accuracy. FP denotes the full-precision model. Best quantized results are in bold.

Config	Method	Llama3-8B Wiki ↓	Llama3-8B C4 ↓	Llama3-8B Avg ↑	Llama3.1-8B Wiki ↓	Llama3.1-8B C4 ↓	Llama3.1-8B Avg ↑	Llama3.2-3B Wiki ↓	Llama3.2-3B C4 ↓	Llama3.2-3B Avg ↑	Qwen3-4B Wiki ↓	Qwen3-4B C4 ↓	Qwen3-4B Avg ↑	Qwen3-8B Wiki ↓	Qwen3-8B C4 ↓	Qwen3-8B Avg ↑	Qwen3-14B Wiki ↓	Qwen3-14B C4 ↓	Qwen3-14B Avg ↑
–	FP	6.1	9.4	67.1	7.2	11.4	68.2	11.0	16.5	60.9	13.7	19.9	64.6	9.7	15.4	67.2	8.6	13.8	70.3
PC	GPTQ	16.5	43.6	48.2	18.8	48.1	48.9	36.6	77.9	44.2	22.1	43.8	48.7	15.6	31.0	53.3	13.4	25.2	59.0
	GPTAQ	13.9	38.2	47.7	15.5	37.9	49.0	27.2	60.0	43.8	24.6	48.0	47.6	16.3	32.6	51.9	13.5	27.1	58.6
	ResComp	14.1	38.9	48.0	15.4	37.9	49.7	27.0	63.8	43.4	24.7	47.8	45.7	17.1	34.4	51.5	13.4	26.3	59.0
	CARE-Q	**12.5**	**29.3**	**49.4**	**14.1**	**31.9**	**51.1**	**22.8**	**43.6**	**44.8**	**21.8**	**41.2**	**48.8**	**14.7**	**27.3**	**54.9**	**12.3**	**23.1**	**59.8**
G128	GPTQ	15.3	37.2	48.1	17.7	43.9	49.9	33.7	73.3	44.5	20.8	38.9	47.5	14.4	28.2	53.8	12.4	22.6	59.7
	GPTAQ	13.0	32.2	49.2	14.1	33.7	51.4	23.7	50.4	44.4	22.3	41.5	47.2	15.4	30.1	51.8	12.6	23.2	59.7
	ResComp	13.0	33.2	48.8	14.1	34.2	50.1	24.3	50.9	43.6	22.3	40.9	47.8	15.4	30.1	51.6	12.5	23.4	58.8
	CARE-Q	**12.1**	**26.5**	**49.6**	**13.5**	**28.5**	**51.8**	**22.5**	**41.0**	**45.0**	**20.2**	**37.1**	**49.6**	**14.2**	**26.2**	**54.7**	**11.9**	**21.6**	**61.2**

Table 3: Ablation study of CARE-Q. CARE-Q w/o DiCR removes DiCR while retaining ODPP.

Config	Method	Llama3-8B Wiki	Llama3-8B C4	Llama3-8B Avg	Llama3.1-8B Wiki	Llama3.1-8B C4	Llama3.1-8B Avg	Llama3.2-3B Wiki	Llama3.2-3B C4	Llama3.2-3B Avg	Qwen3-4B Wiki	Qwen3-4B C4	Qwen3-4B Avg	Qwen3-8B Wiki	Qwen3-8B C4	Qwen3-8B Avg	Qwen3-14B Wiki	Qwen3-14B C4	Qwen3-14B Avg
PC	GPTQ	21.5	26.9	50.6	24.8	33.0	53.5	48.5	53.5	46.1	28.4	34.6	52.0	19.3	26.3	56.0	15.1	19.9	62.7
	CARE-Q w/o DiCR	17.3	21.6	51.9	19.0	26.2	54.2	37.7	38.2	47.1	26.7	32.2	52.1	17.0	23.9	56.8	14.4	19.4	62.6
	CARE-Q	**14.7**	**19.9**	**52.8**	**16.4**	**23.3**	**54.8**	**26.6**	**32.2**	**47.6**	**26.6**	**31.4**	**52.2**	**16.7**	**23.4**	**57.2**	**14.1**	**19.2**	**62.7**
G128	GPTQ	18.7	24.5	51.1	21.1	29.7	53.1	46.5	43.7	46.9	25.7	31.9	51.8	16.6	23.3	57.1	13.0	18.9	63.5
	CARE-Q w/o DiCR	15.6	20.7	53.2	17.1	23.7	54.5	30.4	34.5	47.4	23.5	29.9	52.3	15.9	23.0	57.0	13.1	18.5	62.9
	CARE-Q	**13.8**	**19.5**	**54.0**	**15.4**	**22.3**	**54.9**	**25.8**	**30.0**	**48.3**	**23.9**	**30.0**	**52.5**	**15.3**	**22.7**	**57.5**	**12.7**	**18.4**	**63.5**

The gains of CARE-Q persist at larger model scales: under QuaRot W2A16KV16-PC, it achieves the lowest WikiText and C4 perplexities and the highest downstream average on Qwen3-32B and Llama3-70B (Appendix C.5). CARE-Q also remains effective under 3-bit weight-only quantization without transformations (Appendix C.2), showing that its improvements extend across model scales and diverse quantization configurations under challenging low-bit settings.

4.2 Full Quantization Results

We further evaluate CARE-Q under the challenging W2A4KV4 full quantization setting, where weights, activations, and KV cache are jointly quantized with the same FlatQuant preprocessing. As shown in Table 2, CARE-Q achieves the best WikiText perplexity, C4 perplexity, and average downstream accuracy in all 12 evaluated model-granularity configurations. Since all methods share identical preprocessing and quantization configurations, these improvements directly demonstrate the effectiveness of CARE-Q under aggressive low-bit constraints. For example, under PC quantization on Llama3-8B, CARE-Q reduces C4 perplexity from the best baseline result of 38.2 to 29.3, while improving average downstream accuracy from the best baseline result of 48.2 to 49.4. These results show that CARE-Q remains effective when quantization errors from weights, activations, and KV cache are introduced simultaneously.

4.3 Ablation Study

We conduct ablation studies to assess the contributions of DiCR and ODPP to CARE-Q. As shown in Table 3, CARE-Q w/o DiCR removes DiCR while retaining ODPP and lowers both WikiText and C4 perplexity relative to GPTQ in 11 of the 12 settings. Adding DiCR further improves average downstream accuracy in all 12 settings and lowers both perplexities in 11 settings. The progressive gains from ODPP alone to full CARE-Q show that improving error compensation and compensation-aware error formation provides complementary benefits within the quantization error lifecycle. Additional ablations further support the complementary contributions of DiCR and ODPP (Appendices C.6 and C.9), while CARE-Q remains robust across calibration budgets, broad coefficient ranges, and calibration datasets (Appendices C.7, C.8, and C.12).

Table 4: Quantization efficiency comparison on a single NVIDIA H20 GPU. Time reports mean±std over 5 runs; Peak denotes peak GPU memory usage (GB) measured by nvitop.

Model	Method	PC					G128				
		Wiki↓	C4↓	Avg↑	Time(s)	Peak	Wiki↓	C4↓	Avg↑	Time(s)	Peak
Llama3-8B	GPTQ	21.5	26.9	50.6	689±7	9.1	18.7	24.5	51.1	949±8	9.1
	GPTAQ	18.3	22.8	51.4	910±3	27.6	16.1	21.1	53.0	1169±7	27.6
	ResComp	18.7	22.9	51.4	1007±6	30.2	16.8	21.2	51.8	1269±16	30.2
	CARE-Q	**14.7**	**19.9**	**52.8**	989±8	29.9	**13.8**	**19.5**	**54.0**	1224±6	29.9
Qwen3-4B	GPTQ	28.4	34.6	52.0	498±3	5.4	25.7	31.9	51.8	683±18	5.4
	GPTAQ	31.2	34.5	50.6	636±7	17.6	25.5	30.9	51.2	830±17	17.6
	ResComp	30.9	33.8	50.4	695±6	19.0	24.8	30.8	50.6	886±7	19.0
	CARE-Q	**26.6**	**31.4**	**52.2**	678±4	18.7	**23.9**	**30.0**	**52.5**	870±7	18.7

Table 5: Complementarity study with transformations. QuaRot+RTN and FlatQuant+RTN denote transformation-only baselines, while Raw denotes the model without transformation.

Method	Llama3-8B			Llama3.1-8B			Qwen3-8B			Qwen3-14B		
	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg
QuaRot + RTN	1.1e6	1.0e6	36.1	1.6e6	9.3e5	36.8	2.4e7	2.8e7	36.0	2.4e6	3.1e6	37.8
FlatQuant + RTN	6.2e5	5.9e5	36.3	7.5e5	7.2e5	36.6	1.8e7	1.9e7	36.2	5.1e6	5.5e6	35.6
Raw + CARE-Q	245.3	73.7	41.2	104.3	74.1	41.1	72.7	56.6	45.2	21.7	27.0	48.9
QuaRot + CARE-Q	21.3	25.2	49.7	18.5	24.7	51.8	19.9	26.6	52.8	14.2	20.3	60.8
FlatQuant + CARE-Q	14.7	19.9	52.8	16.4	23.3	54.8	16.7	23.4	57.2	14.1	19.2	62.7

4.4 Efficiency Analysis

We further evaluate the quantization efficiency of CARE-Q under FlatQuant W2A16KV16 on a single NVIDIA H20 GPU. As shown in Table 4, CARE-Q achieves substantial accuracy improvements with offline quantization cost comparable to GPTAQ and ResComp. Reported time includes quantization only. On Llama3-8B under PC quantization, CARE-Q improves WikiText/C4/Avg from 18.7/22.9/51.4 to 14.7/19.9/52.8 over ResComp while maintaining similar quantization time and memory usage. CARE-Q therefore improves low-bit quantization quality without introducing inference-time overhead.

4.5 Complementarity with Transformation-Based Quantization

We investigate the complementarity between CARE-Q and transformation-based quantization under W2A16KV16 PC quantization. As shown in Table 5, the transformation-only baselines, QuaRot+RTN and FlatQuant+RTN, degrade severely in this extreme low-bit regime, whereas CARE-Q remains effective even without transformation preprocessing. Combining CARE-Q with either QuaRot or FlatQuant further reduces perplexity and improves average downstream accuracy over Raw+CARE-Q across all evaluated models, demonstrating that transformation-based methods and CARE-Q provide complementary mechanisms for improving low-bit quantization.

5 Conclusion

In this work, we revisit compensation-based LLM quantization from the perspective of the quantization error lifecycle, revealing that error formation and error compensation are coupled processes rather than isolated stages. Based on this observation, we derive a shared compensation representation and develop CARE-Q, which jointly improves error formation and error compensation through DiCR and ODPP. Extensive experiments across diverse LLMs and low-bit quantization settings demonstrate that CARE-Q consistently achieves substantial improvements over existing compensation-based quantization methods, while complementing transformation-based approaches to address the challenges of low-bit LLM quantization. This work provides a new perspective for understanding and improving LLM quantization through the quantization error lifecycle, while preserving the standard low-bit quantization format and incurring zero additional inference overhead.

AI USE STATEMENT

Generative AI tools were used during the preparation of this manuscript to assist with writing refinement, drafting selected sections, and verifying mathematical derivations and proofs. The theoretical formulation, algorithm design, mathematical claims, and proof development were conducted by the authors. AI tools were only used as auxiliary support for checking derivation consistency and identifying potential issues during the verification process. All AI-assisted content was carefully reviewed and verified by the authors. The authors take full responsibility for the final manuscript, including the correctness of the mathematical formulations, experimental results, and conclusions.

REPRODUCIBILITY STATEMENT

We provide the complete implementation of CARE-Q through an anonymous repository and supplementary materials. The released resources include the source code, environment setup, model preparation instructions, calibration and evaluation protocols, and ready-to-run scripts for the main experiments. The supplementary experiments in the appendix reuse the same implementations, coefficient settings, and evaluation protocols as the main experiments. The anonymous repository is available at: `https://anonymous.4open.science/r/CARE-Q/`. Detailed algorithm descriptions, theoretical derivations, implementation details, and additional experimental analyses are provided in the main paper and appendix.

REFERENCES

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-Free 4-Bit Inference in Rotated LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 100213–100240, 2024. doi: 10.52202/079017-3180.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M. De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4396–4429, 2023. doi: 10.52202/075280-0196.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, 2019. doi: 10.18653/v1/N19-1300.

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, 2018. URL `https://arxiv.org/abs/1803.05457`. arXiv:1803.05457.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers, 2023. URL `https://arxiv.org/abs/2210.17323v2`. arXiv:2210.17323, version 2, revised 22 March 2023.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB dataset of diverse text for language modeling, 2020. URL `https://arxiv.org/abs/2101.00027`. arXiv:2101.00027.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, et al. The Llama 3 herd of models, 2024. arXiv:2407.21783.

Shuaiting Li, Juncan Deng, Kedong Xu, Rongtao Deng, Hong Gu, Minghan Jiang, Haibin Shen, and Kejie Huang. Rethinking residual errors in compensation-based LLM quantization. In *The Fourteenth International Conference on Learning Representations*, 2026.

Yuhang Li, Ruokai Yin, Donghyun Lee, Shiting Xiao, and Priyadarshini Panda. GPTAQ: Efficient finetuning-free quantization for asymmetric calibration. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 36690–36706. PMLR, 2025.

Haokun Lin, Haobo Xu, Yichen Wu, Jingzhi Cui, Yingtao Zhang, Linzhan Mou, Linqi Song, Zhenan Sun, and Ying Wei. DuQuant: Distributing outliers via dual transformation makes stronger quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 87766–87800, 2024a. doi: 10.52202/079017-2786.

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. In *Proceedings of Machine Learning and Systems*, volume 6, pp. 87–100. MLSys, 2024b.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *The Thirteenth International Conference on Learning Representations*, 2025.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, 2018. doi: 10.18653/v1/D18-1260.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 7197–7206. PMLR, 2020. URL https://proceedings.mlr.press/v119/nagel20a.html.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial Winograd schema challenge at scale. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4463–4473, 2019. doi: 10.18653/v1/D19-1454.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=8Wuvhh0LYW.

Yuxuan Sun, Ruikang Liu, Haoli Bai, Han Bao, Kang Zhao, Yuening Li, Jiaxin Hu, Xianzhi Yu, Lu Hou, Chun Yuan, Xin Jiang, Wulong Liu, and Jun Yao. FlatQuant: Flatness matters for LLM quantization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 57587–57613. PMLR, 2025.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even Better LLM Quantization with Hadamard Incoherence and Lattice Codebooks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 48630–48656. PMLR, 2024.

Maurice Weber, Daniel Y. Fu, Quentin Anthony, Yonatan Oren, Shane Adams, Anton Alexandrov, Xiaozhong Lyu, Huu Nguyen, Xiaozhe Yao, Virginia Adams, Ben Athiwaratkun, Rahul Chalamala, Kezhen Chen, Max Ryabinin, Tri Dao, Percy Liang, Christopher Ré, Irina Rish, and Ce Zhang. RedPajama: an open dataset for training large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 116462–116492, 2024. doi: 10.52202/079017-3697.

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 *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 38087–38099. PMLR, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019. doi: 10.18653/v1/P19-1472.

Yuli Zhou, Qingxuan Chen, Luca Benini, Guolei Sun, and Yawei Li. Revisiting adaptive rounding with vectorized reparameterization for LLM quantization, 2026. URL https://arxiv.org/abs/2602.02151. arXiv:2602.02151.

A Proofs of Main Propositions

A.1 Proof of Proposition 3.1

Proof. We prove Proposition 3.1 by analytically eliminating the future compensation variables from the ideal joint objective in Eq. (7).

At quantization step t, the current working state $\mathbf{W}^{(t)}$, the residual $\mathbf{R}^{(t)}$, the calibration inputs, and the candidate set $\mathcal{Q}_t$ are fixed before choosing $\mathbf{q}_t$. Let

$$\mathbf{x}_t = \hat{\mathbf{X}}_{t,:}, \qquad \mathbf{X}_f = \hat{\mathbf{X}}_{t+1:,:}.$$

We assume that the active quantized inputs $\hat{\mathbf{X}}_{t:,:}$ have full row rank. For a fixed candidate $\mathbf{q}_t$, the corresponding quantization error $\mathbf{e}_t = \mathbf{w}_t - \mathbf{q}_t$ is fixed, and the inner problem in Eq. (7) becomes

$$\min_{\Delta \mathbf{W}_{:,t+1:}} \left\| \mathbf{R}^{(t)} + \mathbf{e}_t \mathbf{x}_t - \Delta \mathbf{W}_{:,t+1:} \mathbf{X}_f \right\|_F^2. \tag{24}$$

Define

$$\mathbf{Z}_t = \mathbf{R}^{(t)} + \mathbf{e}_t \mathbf{x}_t. \tag{25}$$

When future columns remain, $\mathbf{X}_f \mathbf{X}_f^\top$ is nonsingular under the full-row-rank assumption, and the unique least-squares solution is

$$\Delta \mathbf{W}^\star_{:,t+1:} = \mathbf{Z}_t \mathbf{X}_f^\top (\mathbf{X}_f \mathbf{X}_f^\top)^{-1}. \tag{26}$$

For nonempty $\mathbf{X}_f$, let

$$\mathbf{P}_f = \mathbf{X}_f^\top (\mathbf{X}_f \mathbf{X}_f^\top)^{-1} \mathbf{X}_f \tag{27}$$

be the orthogonal projector onto the row space of the future quantized inputs. When no future columns remain, the compensation variable is absent and we use the convention $\mathbf{P}_f = \mathbf{0}$. Define

$$\mathbf{M}_t = \mathbf{I} - \mathbf{P}_f. \tag{28}$$

Substituting the optimal compensation into Eq. (24) gives

$$J_t(\mathbf{q}_t) = \left\| \left(\mathbf{R}^{(t)} + \mathbf{e}_t \mathbf{x}_t \right) \mathbf{M}_t \right\|_F^2. \tag{29}$$

Define the component of the current quantized input orthogonal to the future input row space as

$$\mathbf{u}_t = \mathbf{M}_t \mathbf{x}_t^\top. \tag{30}$$

Since $\mathbf{M}_t$ is symmetric and idempotent, Eq. (29) expands as

$$J_t(\mathbf{q}_t) = \left\| \mathbf{R}^{(t)} \mathbf{M}_t \right\|_F^2 + 2 \mathbf{e}_t^\top \mathbf{R}^{(t)} \mathbf{u}_t + \|\mathbf{u}_t\|_2^2 \|\mathbf{e}_t\|_2^2. \tag{31}$$

We therefore define

$$\omega_t = \|\mathbf{u}_t\|_2^2 = \mathbf{x}_t \mathbf{M}_t \mathbf{x}_t^\top. \tag{32}$$

Because $\hat{\mathbf{X}}_{t:,:}$ has full row rank, $\mathbf{x}_t$ does not lie in the row space of $\mathbf{X}_f$. Hence $\mathbf{u}_t \neq \mathbf{0}$ and

$$\omega_t > 0. \tag{33}$$

The compensation-induced shift is

$$\mathbf{s}_t^\star = -\frac{\mathbf{R}^{(t)} \mathbf{u}_t}{\omega_t} = -\frac{\mathbf{R}^{(t)} \mathbf{M}_t \mathbf{x}_t^\top}{\mathbf{x}_t \mathbf{M}_t \mathbf{x}_t^\top}. \tag{34}$$

Completing the square in Eq. (31) yields

$$J_t(\mathbf{q}_t) = c_t + \omega_t \left\| \mathbf{e}_t - \mathbf{s}_t^\star \right\|_2^2, \tag{35}$$

where

$$c_t = \left\| \mathbf{R}^{(t)} \mathbf{M}_t \right\|_F^2 - \frac{\left\| \mathbf{R}^{(t)} \mathbf{u}_t \right\|_2^2}{\omega_t} \tag{36}$$

is independent of the current quantization candidate.

Since $\mathbf{e}_t = \mathbf{w}_t - \mathbf{q}_t$, and c_t and ω_t are candidate-independent with $\omega_t > 0$, minimizing Eq. (35) over $\mathbf{q}_t \in \mathcal{Q}_t$ is equivalent to

$$\mathbf{q}_t^\star = \arg\min_{\mathbf{q}\in\mathcal{Q}_t} \|\mathbf{w}_t - \mathbf{q} - \mathbf{s}_t^\star\|_2^2 . \tag{37}$$

Thus, after optimal future compensation is eliminated, its effect on the current discrete decision appears as a shift of the nearest-code criterion. When $\mathbf{s}_t^\star = \mathbf{0}$, the criterion reduces to nearest-code selection and therefore to conventional RTN for the standard scalar quantization grids considered in this work.

□

A.2 Proof of Proposition 3.2

Proof. Recall that

$$\mathbf{\Gamma} = \Delta\mathbf{X}\hat{\mathbf{X}}^\top\mathbf{L}, \qquad \mathbf{H}^{-1} = \mathbf{L}\mathbf{L}^\top, \tag{38}$$

where $\mathbf{L}$ is lower triangular and hence $\mathbf{L}^\top$ is upper triangular.

We first decompose the compensation representation into its diagonal, strictly upper-triangular, and strictly lower-triangular parts:

$$\mathbf{\Gamma} = \mathbf{\Gamma}_D + \mathbf{\Gamma}_U + \mathbf{\Gamma}_L, \tag{39}$$

where

$$\mathbf{\Gamma}_D = \operatorname{diag}\left(\operatorname{diag}(\mathbf{\Gamma})\right), \qquad \mathbf{\Gamma}_U = \operatorname{triu}(\mathbf{\Gamma}, 1), \qquad \mathbf{\Gamma}_L = \operatorname{tril}(\mathbf{\Gamma}, -1). \tag{40}$$

For the diagonal component, its interaction with the corresponding Cholesky diagonal gives

$$\operatorname{diag}\left(\mathbf{\Gamma}_D\mathbf{L}^\top\right) = \operatorname{diag}(\mathbf{\Gamma}) \odot \operatorname{diag}(\mathbf{L}), \tag{41}$$

and therefore the current-coordinate signal at step t is

$$g_t = \Gamma_{tt}L_{tt}. \tag{42}$$

We next consider the strictly upper-triangular component. Since both $\mathbf{\Gamma}_U$ and $\mathbf{L}^\top$ are upper triangular and $\mathbf{\Gamma}_U$ has zero diagonal, their product remains strictly upper triangular:

$$\left[\mathbf{\Gamma}_U\mathbf{L}^\top\right]_{ij} = 0, \qquad j \le i. \tag{43}$$

Hence, at quantization step t, this component can affect only future columns, with

$$\left[\mathbf{\Gamma}_U\mathbf{L}^\top\right]_{t,t+1:} = \mathbf{\Gamma}_{t,t+1:}\mathbf{L}^\top_{t+1:,t+1:}. \tag{44}$$

For the strictly lower-triangular component, $\mathbf{\Gamma}_L\mathbf{L}^\top$ is not generally triangular. Under column-wise sequential quantization, compensation at step t can only modify unprocessed columns. We therefore retain the strictly upper-triangular part

$$\mathbf{C}_L = \operatorname{triu}\left(\mathbf{\Gamma}_L\mathbf{L}^\top, 1\right), \tag{45}$$

which satisfies

$$[\mathbf{C}_L]_{ij} = 0, \qquad j \le i. \tag{46}$$

For row t, its future-column entries are

$$[\mathbf{C}_L]_{t,t+1:} = \mathbf{\Gamma}_{t,1:t-1}\mathbf{L}^\top_{1:t-1,t+1:}. \tag{47}$$

Finally, partitioning the t-th row of $\mathbf{\Gamma}$ into the processed, current, and future coordinates gives

$$\begin{aligned}\mathcal{F}_t = \Gamma_{tt}\mathbf{L}^\top_{t,t+1:} &+ \mathbf{\Gamma}_{t,t+1:}\mathbf{L}^\top_{t+1:,t+1:} \\ &+ \mathbf{\Gamma}_{t,1:t-1}\mathbf{L}^\top_{1:t-1,t+1:},\end{aligned} \tag{48}$$

which is exactly the decomposition in Eq. (17).

Therefore, the diagonal interaction yields the current-coordinate scalar g_t. The upper off-diagonal component is intrinsically causal under the upper-triangular structure of $\mathbf{L}^\top$, while the lower off-diagonal component contributes to future columns through its strictly upper-triangular projection. Since $\mathbf{\Gamma}$ and $\mathbf{L}$ are determined by the fixed calibration inputs and their second-order statistics before candidate evaluation, these quantities are independent of the current quantization candidate and can be precomputed before sequential quantization.

□

A.3 Connection Between the Lifecycle Representation and the Ideal Shift

Proof. We establish the structural connection between the lifecycle representation $\mathbf{\Gamma}$ and the input-path contribution to the ideal compensation-induced shift in Proposition 3.1.

Recall from Eq. (11) that

$$\mathbf{R}^{(t)} = \mathbf{R}^{(t)}_{\text{state}} + \mathbf{R}^{(t)}_{\text{path}}, \tag{49}$$

where

$$\mathbf{R}^{(t)}_{\text{state}} = (\mathbf{W} - \mathbf{W}^{(t)})\mathbf{X}, \qquad \mathbf{R}^{(t)}_{\text{path}} = \mathbf{W}^{(t)}\Delta\mathbf{X}. \tag{50}$$

Since the closed-form ideal shift in Eq. (34) is linear in $\mathbf{R}^{(t)}$, we can write

$$\mathbf{s}^\star_t = \mathbf{s}^\star_{t,\text{state}} + \mathbf{s}^\star_{t,\text{path}}, \tag{51}$$

where the input-path contribution is

$$\mathbf{s}^\star_{t,\text{path}} = -\mathbf{W}^{(t)}\Delta\mathbf{X}\frac{\mathbf{M}_t\mathbf{x}_t^\top}{\omega_t}. \tag{52}$$

Let $r_t = n - t + 1$ and define

$$\hat{\mathbf{X}}_{t:,:} = \begin{bmatrix}\mathbf{x}_t \\ \mathbf{X}_f\end{bmatrix} \in \mathbb{R}^{r_t \times N}, \qquad \mathbf{H}^{\text{act}}_t = \hat{\mathbf{X}}_{t:,:}\hat{\mathbf{X}}^\top_{t:,:}. \tag{53}$$

Let $\mathbf{b}_1 \in \mathbb{R}^{r_t}$ denote the first canonical basis vector. Under the full-row-rank assumption of Proposition 3.1, $\mathbf{H}^{\text{act}}_t$ is nonsingular.

For nonempty $\mathbf{X}_f$,

$$\mathbf{H}^{\text{act}}_t = \begin{bmatrix}\mathbf{x}_t\mathbf{x}_t^\top & \mathbf{x}_t\mathbf{X}_f^\top \\ \mathbf{X}_f\mathbf{x}_t^\top & \mathbf{X}_f\mathbf{X}_f^\top\end{bmatrix}. \tag{54}$$

The Schur complement of $\mathbf{X}_f\mathbf{X}_f^\top$ is

$$\begin{aligned}\omega_t &= \mathbf{x}_t\mathbf{x}_t^\top - \mathbf{x}_t\mathbf{X}_f^\top(\mathbf{X}_f\mathbf{X}_f^\top)^{-1}\mathbf{X}_f\mathbf{x}_t^\top \\ &= \mathbf{x}_t\mathbf{M}_t\mathbf{x}_t^\top.\end{aligned} \tag{55}$$

Hence, the first column of $(\mathbf{H}^{\text{act}}_t)^{-1}$ is

$$(\mathbf{H}^{\text{act}}_t)^{-1}\mathbf{b}_1 = \frac{1}{\omega_t}\begin{bmatrix}1 \\ -(\mathbf{X}_f\mathbf{X}_f^\top)^{-1}\mathbf{X}_f\mathbf{x}_t^\top\end{bmatrix}. \tag{56}$$

Multiplying by $\hat{\mathbf{X}}^\top_{t:,:}$ gives

$$\begin{aligned}\hat{\mathbf{X}}^\top_{t:,:}(\mathbf{H}^{\text{act}}_t)^{-1}\mathbf{b}_1 &= \frac{\mathbf{x}_t^\top - \mathbf{X}_f^\top(\mathbf{X}_f\mathbf{X}_f^\top)^{-1}\mathbf{X}_f\mathbf{x}_t^\top}{\omega_t} \\ &= \frac{\mathbf{M}_t\mathbf{x}_t^\top}{\omega_t}.\end{aligned} \tag{57}$$

When $t = n$, $\mathbf{X}_f$ is empty and $\mathbf{M}_t = \mathbf{I}$, and Eq. (57) follows directly.

Substituting Eq. (57) into Eq. (52) yields

$$\mathbf{s}^\star_{t,\text{path}} = -\mathbf{W}^{(t)}\Delta\mathbf{X}\hat{\mathbf{X}}^\top_{t:,:}(\mathbf{H}^{\text{act}}_t)^{-1}\mathbf{b}_1. \tag{58}$$

We next connect the active inverse Gram matrix to the Cholesky factor used by CARE-Q. Recall that

$$\mathbf{H}^{-1} = \mathbf{L}\mathbf{L}^\top, \qquad \mathbf{H} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top, \tag{59}$$

where $\mathbf{L}$ is lower triangular. Define $\mathbf{A} = \mathbf{L}^{-1}$, so that

$$\mathbf{H} = \mathbf{A}^\top \mathbf{A}. \tag{60}$$

Since $\mathbf{A}$ is lower triangular, the columns indexed by $t:$ have zero entries in rows $1:t-1$. Therefore,

$$\mathbf{H}_{t:,t:} = \mathbf{A}_{t:,t:}^\top \mathbf{A}_{t:,t:}. \tag{61}$$

Since

$$\mathbf{H}_{t:,t:} = \hat{\mathbf{X}}_{t:,:}\hat{\mathbf{X}}_{t:,:}^\top = \mathbf{H}_t^{\mathrm{act}}, \tag{62}$$

we obtain

$$(\mathbf{H}_t^{\mathrm{act}})^{-1} = \mathbf{A}_{t:,t:}^{-1}\mathbf{A}_{t:,t:}^{-\top}. \tag{63}$$

To make the trailing-block relation explicit, partition the lower-triangular matrix $\mathbf{A}$ as

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{0} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}, \qquad \mathbf{A}_{22} = \mathbf{A}_{t:,t:}. \tag{64}$$

Its inverse is also lower triangular and has trailing block $\mathbf{A}_{22}^{-1}$. Since $\mathbf{L} = \mathbf{A}^{-1}$,

$$\mathbf{A}_{t:,t:}^{-1} = \mathbf{L}_{t:,t:}. \tag{65}$$

Consequently,

$$(\mathbf{H}_t^{\mathrm{act}})^{-1} = \mathbf{L}_{t:,t:}\mathbf{L}_{t:,t:}^\top. \tag{66}$$

Because $\mathbf{L}_{t:,t:}^\top$ is upper triangular,

$$\mathbf{L}_{t:,t:}^\top \mathbf{b}_1 = L_{tt}\mathbf{b}_1, \tag{67}$$

and therefore

$$(\mathbf{H}_t^{\mathrm{act}})^{-1}\mathbf{b}_1 = L_{tt}\mathbf{L}_{t:,t}. \tag{68}$$

Substituting Eq. (68) into Eq. (58) gives

$$\begin{aligned} \mathbf{s}_{t,\mathrm{path}}^\star &= -\mathbf{W}^{(t)}\Delta\mathbf{X}\hat{\mathbf{X}}_{t:,:}^\top \mathbf{L}_{t:,t} L_{tt} \\ &= -\mathbf{W}^{(t)}\Delta\mathbf{X}\hat{\mathbf{X}}^\top \mathbf{L}_{:,t} L_{tt}, \end{aligned} \tag{69}$$

where the second equality follows from the lower-triangular structure of $\mathbf{L}$, which gives $\mathbf{L}_{1:t-1,t} = \mathbf{0}$.

Using the lifecycle representation

$$\mathbf{\Gamma} = \Delta\mathbf{X}\hat{\mathbf{X}}^\top\mathbf{L}, \tag{70}$$

Eq. (69) becomes

$$\boxed{\mathbf{s}_{t,\mathrm{path}}^\star = -\mathbf{W}^{(t)}\mathbf{\Gamma}_{:,t}L_{tt}.} \tag{71}$$

Expanding over the columns of $\mathbf{W}^{(t)}$ yields

$$\mathbf{s}_{t,\mathrm{path}}^\star = -\Gamma_{tt}L_{tt}\mathbf{w}_t - L_{tt}\sum_{k\neq t}\Gamma_{kt}\mathbf{w}_k^{(t)}. \tag{72}$$

Since $g_t = \Gamma_{tt}L_{tt}$, the current-column self term in Eq. (72) is exactly $-g_t\mathbf{w}_t$.

Remark. The derivation above does not imply that the current-column self term alone recovers the complete ideal shift $\mathbf{s}_t^\star$. The complete ideal shift also contains the state-dependent contribution $\mathbf{s}_{t,\mathrm{state}}^\star$ in Eq. (51) and the cross-column terms in Eq. (72). Rather, the result establishes that the scalar g_t used by DiCR is directly identified through an exact structural decomposition of the input-path contribution to the ideal compensation-induced shift.

Importantly, the practical DiCR shift does not directly approximate the isolated self term $-g_t\mathbf{w}_t$. DiCR retains the structurally derived scalar g_t and constructs a calibrated surrogate $\mathbf{s}_t^{\mathrm{DiCR}} = \beta g_t \mathbf{w}_t^{\mathrm{ref}}$, where the signed coefficient β calibrates its effective direction and magnitude. The use of the fixed reference $\mathbf{w}_t^{\mathrm{ref}}$ instead of the evolving $\mathbf{w}_t$ is evaluated separately in Appendix C.11.

B SURROGATE-SHIFT STABILITY OF DICR

We further characterize how the surrogate shift used by DiCR affects the discrete candidate decision induced by the compensation-conditioned objective in Proposition 3.1. The analysis is performed at a fixed sequential quantization state. If clipping collapses the two adjacent candidates of an element into a single legal value, the decision is trivially invariant and incurs zero excess objective; we therefore consider distinct adjacent candidates below.

Proposition B.1 (Surrogate-Shift Stability for DiCR Candidate Selection)**.** *Let $\mathbf{s}_t^{\mathrm{DiCR}}$ and $\mathbf{q}_t^{\mathrm{DiCR}}$ be the DiCR surrogate shift and candidate decision defined in Eqs. (18) and (19), respectively. Consider the compensation-conditioned objective*

$$J_t(\mathbf{q}_t) = c_t + \omega_t \left\|\mathbf{w}_t - \mathbf{q}_t - \mathbf{s}_t^\star\right\|_2^2, \qquad \omega_t > 0. \tag{73}$$

For each element j, let $\mathcal{C}_{t,j} = \{a_{t,j}, b_{t,j}\}$ with $a_{t,j} < b_{t,j}$ denote the adjacent candidates used by DiCR, and let $\mathcal{C}_t = \prod_j \mathcal{C}_{t,j}$. Let $\mathbf{q}_t^{\star,\mathcal{C}}$ denote any restricted ideal minimizer:

$$\mathbf{q}_t^{\star,\mathcal{C}} \in \operatorname*{arg\,min}_{\mathbf{q}\in\mathcal{C}_t} \left\|\mathbf{w}_t - \mathbf{q} - \mathbf{s}_t^\star\right\|_2^2. \tag{74}$$

Define

$$r_{t,j} = \frac{a_{t,j} + b_{t,j}}{2}, \qquad \Delta_{t,j} = b_{t,j} - a_{t,j}, \tag{75}$$

and the surrogate-shift discrepancy and ideal decision margin as

$$\varepsilon_{t,j} = \left|s_{t,j}^{\mathrm{DiCR}} - s_{t,j}^\star\right| = \left|\beta g_t w_{t,j}^{\mathrm{ref}} - s_{t,j}^\star\right|, \qquad \mu_{t,j} = \left|w_{t,j} - s_{t,j}^\star - r_{t,j}\right|. \tag{76}$$

Then, for every element j,

$$\varepsilon_{t,j} < \mu_{t,j} \quad \Longrightarrow \quad q_{t,j}^{\mathrm{DiCR}} = q_{t,j}^{\star,\mathcal{C}}. \tag{77}$$

Moreover, for $\mathcal{M}_t = \{j : q_{t,j}^{\mathrm{DiCR}} \neq q_{t,j}^{\star,\mathcal{C}}\}$, the excess compensation-conditioned objective is exactly

$$J_t(\mathbf{q}_t^{\mathrm{DiCR}}) - J_t(\mathbf{q}_t^{\star,\mathcal{C}}) = 2\omega_t \sum_{j\in\mathcal{M}_t} \Delta_{t,j}\mu_{t,j}, \tag{78}$$

and consequently

$$0 \le J_t(\mathbf{q}_t^{\mathrm{DiCR}}) - J_t(\mathbf{q}_t^{\star,\mathcal{C}}) \le 2\omega_t \sum_j \Delta_{t,j}\varepsilon_{t,j}. \tag{79}$$

Proof. The objective in Eq. (73) is separable across the elements of the current weight column. We therefore consider an arbitrary element and omit the subscripts t, j for clarity. Write the two candidates as $\mathcal{C} = \{a, b\}$ with $a < b$, midpoint $r = (a + b)/2$, and spacing $\Delta = b - a > 0$. Define the ideal and DiCR-shifted targets as $y^\star = w - s^\star$ and $\widetilde{y} = w - s^{\mathrm{DiCR}}$. Their distance is exactly the surrogate-shift discrepancy:

$$|\widetilde{y} - y^\star| = \left|s^{\mathrm{DiCR}} - s^\star\right| = \varepsilon. \tag{80}$$

For an arbitrary scalar target y, the difference between its squared distances to the two candidates is

$$\begin{aligned}(y-a)^2 - (y-b)^2 &= 2(b-a)\left(y - \frac{a+b}{2}\right) \\ &= 2\Delta(y - r).\end{aligned} \tag{81}$$

Hence, the candidate decision is determined by which side of the boundary r the shifted target lies on. In particular, the restricted ideal decision selects a when $y^\star < r$ and b when $y^\star > r$.

Since $\mu = |y^\star - r|$, if $\varepsilon < \mu$, then Eq. (80) implies that $\widetilde{y}$ cannot cross the boundary r relative to $y^\star$. Therefore, $y^\star$ and $\widetilde{y}$ lie on the same side of r, yielding

$$\varepsilon < \mu \quad \Longrightarrow \quad q^{\mathrm{DiCR}} = q^{\star,\mathcal{C}}. \tag{82}$$

This proves Eq. (77).

We next consider the case in which the DiCR decision differs from the restricted ideal decision. A disagreement can occur only if the displacement from $y^\star$ to $\widetilde{y}$ reaches or crosses the candidate boundary. Consequently,

$$\mu = |y^\star - r| \le |\widetilde{y} - y^\star| = \varepsilon. \tag{83}$$

Suppose first that $y^\star < r$, so that the restricted ideal decision is a while DiCR selects b. Using Eq. (81), the resulting increase in the compensation-conditioned objective is

$$\begin{aligned}\omega_t \left[(y^\star - b)^2 - (y^\star - a)^2\right] &= 2\omega_t \Delta (r - y^\star) \\ &= 2\omega_t \Delta \mu.\end{aligned} \tag{84}$$

The case $y^\star > r$ is symmetric. If $y^\star = r$, both candidates attain the same restricted ideal objective, so $\mu = 0$ and the excess objective is zero regardless of tie breaking.

Restoring the indices and summing over the mismatch set $\mathcal{M}_t$ gives

$$J_t(\mathbf{q}_t^{\mathrm{DiCR}}) - J_t(\mathbf{q}_t^{\star,\mathcal{C}}) = 2\omega_t \sum_{j \in \mathcal{M}_t} \Delta_{t,j} \mu_{t,j}, \tag{85}$$

which proves Eq. (78).

Finally, Eq. (83) implies $\mu_{t,j} \le \varepsilon_{t,j}$ for every $j \in \mathcal{M}_t$. Hence,

$$\begin{aligned}0 &\le J_t(\mathbf{q}_t^{\mathrm{DiCR}}) - J_t(\mathbf{q}_t^{\star,\mathcal{C}}) \\ &\le 2\omega_t \sum_{j \in \mathcal{M}_t} \Delta_{t,j} \varepsilon_{t,j} \\ &\le 2\omega_t \sum_{j} \Delta_{t,j} \varepsilon_{t,j},\end{aligned} \tag{86}$$

which establishes Eq. (79).

□

The margin condition in Eq. (77) is sufficient, rather than an assumption required to hold for every coordinate. When it holds, DiCR preserves the restricted ideal decision; otherwise, Eq. (78) still gives the exact excess objective and Eq. (79) a discrepancy-dependent upper bound. Moreover, a smaller $\mu_{t,j}$ makes a boundary crossing easier but proportionally reduces its objective penalty. Thus, Proposition B.1 characterizes margin-based sensitivity rather than universal recovery of the ideal decision.

Any unrestricted ideal minimizer $\mathbf{q}_t^\star$ in Proposition 3.1 that belongs to $\mathcal{C}_t$ is also a restricted minimizer; if it is unique, then $\mathbf{q}_t^{\star,\mathcal{C}} = \mathbf{q}_t^\star$.

Comparison with RTN. Conventional RTN corresponds to the zero-shift rule $\mathbf{s}_t^{\mathrm{RTN}} = \mathbf{0}$. Hence, the RTN and DiCR discrepancies from the compensation-conditioned ideal shift are

$$\varepsilon_{t,j}^{\mathrm{RTN}} = |s_{t,j}^\star|, \qquad \varepsilon_{t,j}^{\mathrm{DiCR}} = \left|s_{t,j}^{\mathrm{DiCR}} - s_{t,j}^\star\right|. \tag{87}$$

For the scalar quantization grids considered in this work, the RTN solution over the full legal grid is attained within the adjacent candidate pair $\mathcal{C}_{t,j}$, except for the clipping-collapsed case handled above. Thus, RTN and DiCR can be characterized under the same candidate-stability criterion.

Consider the same adjacent candidate pair $\mathcal{C}_{t,j}$ and suppose that the unshifted RTN target $w_{t,j}$ and the ideal shifted target $w_{t,j} - s_{t,j}^\star$ lie strictly on opposite sides of the candidate boundary $r_{t,j}$. RTN then selects a different candidate from the restricted ideal decision. If simultaneously

$$\varepsilon_{t,j}^{\mathrm{DiCR}} < \mu_{t,j}, \tag{88}$$

DiCR preserves the restricted ideal decision, yielding

$$q_{t,j}^{\mathrm{DiCR}} = q_{t,j}^{\star,\mathcal{C}} \neq q_{t,j}^{\mathrm{RTN}}. \tag{89}$$

For such a coordinate, the corresponding contribution to the compensation-conditioned objective is reduced relative to RTN by exactly

$$2\omega_t \Delta_{t,j} \mu_{t,j}. \tag{90}$$

C Additional Experimental Details

C.1 Implementation Details

Unless otherwise specified, GPTQ, GPTAQ, ResComp, and CARE-Q share identical calibration protocols, quantization configurations, and evaluation settings for fair comparison.

Calibration Protocol and Common Configurations. Following previous compensation-based quantization methods, we use 128 calibration samples with a sequence length of 2048, drawn from C4 for weight-only quantization and WikiText for full quantization. All methods use the same calibration samples with `seed`=0. We enable `act_order` and `gptq_mse` for all methods and set `percdamp`=0.01, with the standard GPTQ diagonal damping applied before inverse-Cholesky factorization. All methods use a block size of $B = 128$ for blockwise sequential quantization. The calibration and evaluation batch sizes are 4 and 16, respectively.

Transformation-based Preprocessing. CARE-Q is evaluated with both raw quantization and transformation-based preprocessing. Besides W2A16KV16 with QuaRot and FlatQuant, we additionally evaluate raw W3A16KV16 weight-only quantization without transformation preprocessing (Appendix C.2).

For experiments with FlatQuant preprocessing, we follow its official training pipeline. Since directly training transformations under W2 causes severe performance degradation, we use the official W4 training configuration and reuse the resulting transformations for W2: W4A16KV16 transformations for W2A16KV16 and W4A4KV4 transformations for W2A4KV4. As the original FlatQuant implementation does not support Qwen3, we adapt it to all evaluated models. We consistently use `--lwc --lac --cali_trans --add_diag`. QuaRot requires no transformation training and directly applies its predefined rotations.

Quantization Configurations. Weights and activations use symmetric quantization, while KV cache uses asymmetric quantization. We evaluate both per-channel (PC) and group-wise (G128) weight quantization, with group size 128 for G128. Activation quantization is per-token and enabled only for W2A4KV4; in the same setting, both key and value caches use group size 128. Thus, W2A4KV4 jointly quantizes weights, activations, and KV cache, whereas weight-only settings quantize weights only.

Baseline Configurations. We follow the official open-source implementations and configurations of all compensation-based baselines. GPTQ and GPTAQ use the settings provided in the GPTAQ implementation. For ResComp, we follow its official configuration, using `allw` for W3 and `org` for W2. The official ResComp implementation uses $r_a = 0.25$ by default. To strengthen the comparison, we additionally calibrate this residual compensation coefficient and use $r_a = 0.1$ with FlatQuant and $r_a = 0.2$ with QuaRot, which provide better overall performance than the default value in our evaluated settings. Detailed search results are provided in Appendix C.10.

CARE-Q Implementation. For DiCR, we use two rounding candidates, consistent with RTN, and the original full-precision weight as reference. The coefficient β controls the DiCR adjustment during error formation. ODPP follows Eq. 22, where ρ controls the accumulated propagation path $\mathbf{P}_L$. For each model under a given preprocessing setting, we select a single shared coefficient pair (ρ, β) using a calibration pool consisting of 128 WikiText samples and 128 C4 samples, based on the overall calibration-set perplexity trends across the corresponding quantization configurations. Once selected, the same pair is fixed across all corresponding quantization configurations, including W2A16KV16 and W2A4KV4 with both PC and G128, and is consistently reused throughout the corresponding main and supplementary experiments. Thus, ρ and β are not tuned separately for individual bit-width or quantization-granularity configurations. The coefficient settings are specified in the released main experimental scripts and reused by the corresponding supplementary experiments. Sensitivity and component analyses are provided in Appendices C.8 and C.9.

All experiments are conducted with PyTorch on NVIDIA H20 GPUs, and each quantization experiment can be executed on a single GPU. Details of the software environment and ready-to-run scripts for the main experiments are provided in the released repository. The supplementary experiments in the appendix reuse the same implementations, coefficient settings, and evaluation protocols.

Table 6: Detailed W3A16KV16 weight-only quantization results on raw models without transformation under the G128 setting following ResComp (Li et al., 2026). Avg denotes the mean accuracy over eight zero-shot tasks; FP denotes full precision. Best quantized results are in bold.

Model	Method	Wiki↓	C4↓	ARC-E	ARC-C	PIQA	SIQA	HS	OBQA	WG	BoolQ	Avg↑
Llama3-8B	FP	6.1	9.4	77.9	53.2	80.6	47.0	79.2	45.0	73.1	81.1	67.1
	GPTQ	8.5	13.4	71.7	43.0	77.5	44.6	67.3	40.0	71.5	76.3	61.5
	GPTAQ	8.3	12.7	71.1	44.4	78.5	43.8	71.5	43.2	70.6	79.1	62.8
	ResComp	7.9	**12.3**	71.3	44.6	77.9	45.9	75.0	43.8	71.9	77.6	63.5
	CARE-Q	**7.9**	12.6	73.9	46.3	79.3	45.5	72.7	42.4	72.7	78.6	**63.9**
Llama3.1-8B	FP	7.2	11.4	79.7	55.0	81.1	49.4	79.2	43.0	74.1	84.2	68.2
	GPTQ	9.1	14.2	62.8	39.9	74.7	45.1	74.2	42.6	71.7	82.4	61.7
	GPTAQ	8.8	13.9	75.9	49.9	77.9	45.6	74.7	42.8	73.5	83.2	65.4
	ResComp	8.9	13.8	71.8	47.7	78.3	46.2	75.0	41.2	72.1	82.5	64.4
	CARE-Q	**8.8**	**13.8**	77.9	51.3	78.7	45.8	75.2	41.0	71.0	82.5	**65.4**
Llama3.2-3B	FP	11.0	16.5	68.1	45.7	75.6	45.1	70.5	36.0	67.7	78.4	60.9
	GPTQ	14.5	20.5	65.2	40.5	73.6	44.0	65.6	33.8	64.7	75.0	57.8
	GPTAQ	14.2	19.8	65.2	40.8	73.0	44.9	66.0	35.6	63.1	76.9	58.2
	ResComp	13.7	**19.6**	64.3	40.8	72.9	44.6	66.7	36.0	64.3	76.6	58.3
	CARE-Q	**13.5**	19.7	66.4	42.2	74.2	44.3	66.2	35.2	64.0	77.8	**58.8**
Qwen3-4B	FP	13.7	19.9	78.3	53.8	74.9	49.6	68.6	40.2	66.2	85.0	64.6
	GPTQ	15.5	21.9	65.5	43.5	73.9	45.3	64.0	39.0	64.8	82.6	59.8
	GPTAQ	15.9	22.2	68.5	44.5	74.2	46.1	63.5	40.4	63.6	83.5	60.5
	ResComp	15.8	22.1	64.7	42.7	72.7	46.1	63.4	37.8	63.3	83.3	59.2
	CARE-Q	**14.3**	**21.6**	71.9	47.1	73.6	45.7	63.8	38.0	64.0	81.9	**60.7**
Qwen3-8B	FP	9.7	15.4	80.9	57.0	77.9	51.5	74.9	41.6	67.6	86.5	67.2
	GPTQ	11.3	17.0	76.1	52.0	77.0	47.8	71.6	42.4	68.0	84.9	65.0
	GPTAQ	11.3	17.2	74.4	52.1	76.7	48.5	70.9	41.0	68.8	83.5	64.5
	ResComp	11.6	17.5	73.7	49.3	75.7	48.5	70.6	40.4	68.8	85.3	64.1
	CARE-Q	**11.1**	**17.0**	76.6	52.0	77.9	51.2	71.4	41.2	68.4	84.4	**65.4**
Qwen3-14B	FP	8.6	13.8	82.7	60.3	80.0	51.8	78.9	46.4	72.9	89.3	70.3
	GPTQ	9.6	15.0	80.3	58.9	79.4	50.4	76.4	43.0	71.9	87.8	68.5
	GPTAQ	9.6	14.9	79.8	57.3	79.4	50.4	76.3	46.6	72.7	86.5	68.6
	ResComp	9.5	14.9	79.4	57.0	78.9	49.5	75.8	43.4	72.1	88.3	68.1
	CARE-Q	**9.4**	**14.7**	80.1	60.6	78.9	50.1	76.8	45.2	71.3	88.6	**69.0**

C.2 W3A16KV16 Weight-Only Quantization

We further evaluate CARE-Q under W3A16KV16 weight-only quantization to examine whether its effectiveness extends beyond the more aggressive W2 setting studied in the main text. Following the experimental configuration of ResComp (Li et al., 2026), we perform group-wise quantization with group size 128 (G128) on the raw models, without applying any transformation-based preprocessing. All compared methods use the same quantization and calibration settings to ensure a controlled comparison. Table 6 reports WikiText and C4 perplexities together with the average and individual accuracies over the eight zero-shot downstream tasks.

CARE-Q remains consistently effective under this higher-bit setting. It achieves the best or tied-best WikiText perplexity and average downstream accuracy on all six evaluated models, while obtaining the best or tied-best C4 perplexity on four of the six models. For example, on Llama3-8B, CARE-Q improves the WikiText/C4/Avg results from 8.5/13.4/61.5 of GPTQ to 7.9/12.6/63.9, and on Qwen3-14B from 9.6/15.0/68.5 to 9.4/14.7/69.0. Although the gains are naturally smaller than those observed under W2 quantization, where quantization errors are more severe, the consistent improvements under the raw G128 setting show that CARE-Q is not restricted to extremely low-bit regimes or transformation-assisted quantization.

Table 7: Detailed results of W2A16KV16 weight-only quantization with QuaRot. Both per-channel (PC) and group-wise quantization with group size 128 (G128) are evaluated. WikiText and C4 report perplexity, while Avg denotes the average accuracy over 8 zero-shot tasks.

Model	Config	Method	Wiki↓	C4↓	ARC-E	ARC-C	PIQA	SIQA	HellaS	OBQA	WinoG	BoolQ	Avg↑
Llama3-8B	PC	GPTQ	40.8	52.0	40.4	23.6	63.1	37.0	39.5	29.2	53.4	63.5	43.7
		GPTAQ	23.1	27.5	47.7	25.5	68.4	40.9	50.3	31.8	61.2	67.1	49.1
		ResComp	23.3	27.7	49.6	26.7	69.1	40.6	50.0	30.4	59.4	64.5	48.8
		CARE-Q	**21.3**	**25.2**	50.2	28.2	67.7	39.7	54.0	32.0	59.0	66.7	**49.7**
	G128	GPTQ	35.4	47.3	44.0	25.8	62.5	38.4	42.1	29.0	56.6	63.0	45.2
		GPTAQ	19.5	25.0	50.5	29.3	68.8	40.6	52.3	34.0	61.8	67.5	50.6
		ResComp	21.1	26.4	49.3	28.0	67.8	40.1	50.7	30.4	61.0	67.5	49.4
		CARE-Q	**17.0**	**21.5**	54.9	31.5	71.0	41.1	58.4	32.0	60.3	66.4	**51.9**
Llama3.1-8B	PC	GPTQ	38.3	52.3	44.6	27.1	65.0	39.0	45.8	29.2	55.2	64.3	46.3
		GPTAQ	22.2	29.2	52.5	31.1	67.9	42.7	50.8	28.8	58.8	67.7	50.0
		ResComp	23.8	30.9	52.4	28.5	68.3	40.8	49.8	32.2	58.6	68.3	49.9
		CARE-Q	**18.5**	**24.7**	56.0	33.3	69.2	42.8	55.9	30.4	59.8	67.0	**51.8**
	G128	GPTQ	29.6	41.9	47.9	26.8	66.4	39.6	47.5	29.2	58.2	67.5	47.9
		GPTAQ	19.9	27.2	57.0	31.7	70.7	41.6	53.4	32.4	61.2	69.5	52.2
		ResComp	19.3	25.7	55.4	31.5	70.2	42.3	54.8	32.0	60.8	72.1	52.4
		CARE-Q	**17.3**	**23.8**	55.3	33.8	70.7	41.5	57.2	32.8	60.0	69.6	**52.6**
Llama3.2-3B	PC	GPTQ	93.0	111.6	31.8	23.6	56.0	35.3	31.9	28.2	52.7	59.3	39.9
		GPTAQ	42.7	46.0	35.6	25.6	58.9	37.5	38.1	28.4	52.0	64.7	42.6
		ResComp	44.5	46.8	37.1	26.1	59.9	36.3	38.6	29.0	53.8	64.3	43.1
		CARE-Q	**39.0**	**42.1**	39.4	27.0	61.3	37.3	42.5	28.0	54.7	65.7	**44.5**
	G128	GPTQ	75.0	84.3	34.5	25.8	55.5	36.0	34.6	26.0	53.7	47.9	39.2
		GPTAQ	33.7	38.6	38.1	27.0	60.1	38.4	42.7	29.0	54.1	63.1	44.1
		ResComp	37.3	40.9	38.0	27.1	59.2	36.4	41.1	28.8	55.0	64.3	43.7
		CARE-Q	**33.0**	**36.6**	37.9	27.3	60.8	36.8	45.9	29.6	54.7	66.3	**44.9**
Qwen3-4B	PC	GPTQ	61.1	73.1	31.3	28.3	56.8	34.4	33.8	28.6	52.6	62.2	41.0
		GPTAQ	42.6	49.9	34.0	28.6	61.1	34.2	36.4	29.4	54.6	62.2	42.6
		ResComp	42.1	50.3	32.0	27.7	59.5	34.8	37.0	29.6	53.9	62.3	42.1
		CARE-Q	**40.3**	**47.8**	33.8	29.7	60.5	34.4	36.7	29.2	55.2	62.3	**42.7**
	G128	GPTQ	39.1	52.0	31.4	28.8	58.8	34.1	34.9	28.2	54.9	62.2	41.6
		GPTAQ	32.5	40.8	33.8	30.7	61.5	33.6	38.8	28.6	54.8	62.3	43.0
		ResComp	32.0	39.1	33.3	29.7	60.8	35.2	38.1	29.8	56.4	63.1	43.3
		CARE-Q	**30.5**	**39.0**	37.0	30.3	60.6	34.3	38.3	30.6	56.2	62.7	**43.8**
Qwen3-8B	PC	GPTQ	22.0	29.3	59.8	37.1	69.4	40.4	52.6	31.8	60.7	76.8	53.6
		GPTAQ	21.9	27.9	55.0	32.8	70.7	42.6	51.0	31.4	60.9	71.0	51.9
		ResComp	21.3	27.3	58.2	34.0	70.0	40.7	51.4	31.6	60.8	73.4	52.5
		CARE-Q	**19.9**	**26.6**	56.4	32.9	69.9	41.7	52.6	32.2	61.6	75.2	52.8
	G128	GPTQ	18.2	25.7	59.9	35.0	70.6	42.4	53.9	32.2	60.3	78.0	54.0
		GPTAQ	18.3	25.2	56.7	35.1	70.7	42.0	53.2	34.0	62.8	69.4	53.0
		ResComp	17.8	24.6	55.7	33.7	71.3	40.8	54.3	33.6	61.8	67.4	52.3
		CARE-Q	**17.4**	**24.1**	57.4	35.8	71.2	42.8	55.2	34.0	63.5	79.3	**54.9**
Qwen3-14B	PC	GPTQ	15.0	21.3	70.8	43.3	74.0	44.2	61.3	37.8	67.3	85.6	60.5
		GPTAQ	15.5	21.1	65.3	40.8	73.0	44.7	60.5	38.4	66.9	85.5	59.4
		ResComp	16.2	21.3	67.3	40.9	73.9	43.7	59.8	36.4	65.7	82.5	58.8
		CARE-Q	**14.2**	**20.3**	69.1	43.5	74.9	43.9	62.8	40.2	67.3	84.8	**60.8**
	G128	GPTQ	13.1	19.6	74.0	47.1	74.7	45.2	63.4	38.8	67.1	85.7	62.0
		GPTAQ	13.7	19.6	68.8	43.7	74.8	44.7	62.9	38.0	66.9	84.3	60.5
		ResComp	13.7	19.3	70.1	44.0	74.6	44.5	63.0	39.6	67.2	85.3	61.0
		CARE-Q	**13.1**	**19.0**	73.9	47.1	74.8	46.2	65.2	39.2	66.1	86.0	**62.3**

C.3 Detailed Results under QuaRot W2A16KV16 Quantization

Table 7 provides detailed downstream evaluation results for W2A16KV16 weight-only quantization with QuaRot, including the individual accuracies of all eight zero-shot tasks. Across all 12 model-granularity configurations, CARE-Q achieves the lowest or tied-lowest WikiText and C4 perplexity in every configuration, together with the best or tied-best average downstream accuracy in 11 of them. These improvements hold across all six evaluated models under both PC and G128 quantiza-

Table 8: Detailed results of W2A16KV16 weight-only quantization with FlatQuant. Both per-channel (PC) and group-wise quantization with a group size of 128 (G128) are reported. WikiText and C4 are evaluated using perplexity, while Avg denotes the average accuracy over eight zero-shot tasks. Best quantized results are highlighted in bold.

Model	Config	Method	Wiki↓	C4↓	ARC-E	ARC-C	PIQA	SIQA	HellaS	OBQA	WinoG	BoolQ	Avg↑
Llama3-8B	PC	GPTQ	21.5	26.9	51.4	29.4	66.5	40.2	52.6	32.6	62.4	69.5	50.6
		GPTAQ	18.3	22.8	51.0	28.8	68.3	41.8	54.9	34.0	62.5	69.7	51.4
		ResComp	18.7	22.9	53.7	30.1	70.2	40.5	54.4	34.0	63.8	64.7	51.4
		CARE-Q	**14.7**	**19.9**	53.9	31.8	71.8	42.3	57.8	33.0	63.5	68.2	**52.8**
	G128	GPTQ	18.7	24.5	50.9	29.4	67.8	40.9	52.5	31.2	64.4	71.5	51.1
		GPTAQ	16.1	21.1	55.4	31.3	70.8	42.3	56.6	33.2	63.5	71.1	53.0
		ResComp	16.8	21.2	49.6	30.0	68.9	42.1	57.0	33.4	62.7	70.6	51.8
		CARE-Q	**13.8**	**19.5**	57.9	33.5	71.1	42.1	59.1	34.4	63.9	70.2	**54.0**
Llama3.1-8B	PC	GPTQ	24.8	33.0	59.5	34.0	69.2	42.1	53.5	32.2	62.1	75.0	53.5
		GPTAQ	19.8	25.7	54.8	31.7	71.1	42.1	55.7	32.8	63.1	74.2	53.2
		ResComp	20.7	26.3	51.5	28.6	67.9	41.9	55.0	34.2	61.9	73.1	51.8
		CARE-Q	**16.4**	**23.3**	56.3	34.3	73.0	43.3	59.2	34.6	63.1	74.2	**54.8**
	G128	GPTQ	21.1	29.7	55.4	32.5	67.2	41.8	55.8	33.2	63.9	75.2	53.1
		GPTAQ	17.8	23.7	59.9	35.0	71.3	41.4	57.2	33.6	64.9	71.5	54.3
		ResComp	17.8	23.7	55.8	32.3	70.7	42.2	57.1	35.0	63.6	75.2	54.0
		CARE-Q	**15.4**	**22.3**	56.6	35.0	71.3	41.2	61.0	33.6	64.5	75.9	**54.9**
Llama3.2-3B	PC	GPTQ	48.5	53.5	43.4	27.5	61.8	38.2	44.8	27.8	56.8	68.9	46.1
		GPTAQ	33.5	36.5	47.0	28.8	63.3	39.6	46.6	31.4	54.5	67.2	47.3
		ResComp	35.0	37.9	45.6	26.9	64.2	39.8	45.1	29.0	55.9	67.5	46.8
		CARE-Q	**26.6**	**32.2**	45.6	29.0	64.7	39.7	50.6	30.2	55.5	65.7	**47.6**
	G128	GPTQ	46.5	43.7	44.1	27.2	61.9	39.9	47.0	28.6	56.9	69.2	46.9
		GPTAQ	36.1	33.8	45.8	28.2	63.2	40.2	47.0	29.8	58.2	66.9	47.4
		ResComp	31.6	34.5	42.6	27.2	62.4	40.5	46.4	29.0	56.9	67.7	46.6
		CARE-Q	**25.8**	**30.0**	46.8	29.4	65.1	39.9	52.1	30.6	56.8	65.7	**48.3**
Qwen3-4B	PC	GPTQ	28.4	34.6	55.1	32.6	68.7	41.3	49.9	31.8	59.9	77.1	52.0
		GPTAQ	31.2	34.5	54.1	31.3	67.3	41.7	48.0	29.6	59.1	73.7	50.6
		ResComp	30.9	33.8	50.2	30.3	67.2	40.7	47.5	30.6	58.6	77.9	50.4
		CARE-Q	**26.6**	**31.4**	55.7	34.9	68.1	41.6	49.5	31.0	58.3	78.2	**52.2**
	G128	GPTQ	25.7	31.9	56.5	32.9	68.1	40.2	50.4	30.0	59.1	77.3	51.8
		GPTAQ	25.5	30.9	54.8	34.2	67.0	40.4	49.3	31.4	58.6	73.5	51.2
		ResComp	24.8	30.8	50.0	31.5	68.2	40.4	49.4	30.2	58.2	77.0	50.6
		CARE-Q	**23.9**	**30.0**	57.6	35.3	68.5	42.0	50.5	32.6	57.9	76.0	**52.5**
Qwen3-8B	PC	GPTQ	19.3	26.3	63.2	37.3	72.1	42.5	56.9	33.6	64.7	77.9	56.0
		GPTAQ	19.6	24.9	62.8	37.7	72.7	42.4	56.1	34.2	63.6	74.2	55.5
		ResComp	20.3	25.0	61.7	36.4	71.3	42.4	55.6	34.2	65.8	72.8	55.0
		CARE-Q	**16.7**	**23.4**	63.1	38.7	72.8	43.0	59.3	36.2	64.3	80.2	**57.2**
	G128	GPTQ	16.6	23.3	66.6	38.8	72.0	42.5	59.0	36.0	63.9	77.7	57.1
		GPTAQ	17.6	23.8	63.9	38.5	70.9	43.2	55.8	34.2	62.5	76.0	55.6
		ResComp	18.3	23.8	62.2	37.5	71.7	42.8	56.7	35.0	63.6	77.6	55.9
		CARE-Q	**15.3**	**22.7**	67.2	39.2	72.0	42.7	58.8	36.4	63.5	80.3	**57.5**
Qwen3-14B	PC	GPTQ	15.1	19.9	73.3	47.4	75.0	46.5	65.4	41.8	69.1	83.3	62.7
		GPTAQ	15.7	19.9	73.8	44.3	75.3	46.0	63.5	40.8	68.2	85.3	62.2
		ResComp	15.9	20.0	72.3	44.2	74.9	47.6	62.9	40.0	69.1	83.6	61.8
		CARE-Q	**14.1**	**19.2**	73.0	45.6	74.9	47.3	65.5	41.2	69.0	85.5	**62.7**
	G128	GPTQ	13.0	18.9	74.5	47.4	75.5	47.3	66.3	41.6	69.1	86.2	63.5
		GPTAQ	14.0	18.9	73.2	46.6	75.6	47.3	65.1	40.0	68.3	86.1	62.8
		ResComp	13.8	19.0	72.6	45.2	75.6	46.5	65.5	38.4	68.7	86.5	62.4
		CARE-Q	**12.7**	**18.4**	73.9	48.6	75.2	47.0	67.3	40.2	70.2	85.4	**63.5**

tion, showing consistent benefits across model families and weight granularities. At the individual-task level, the results exhibit heterogeneous behavior across benchmarks, with gains varying across tasks and occasional regressions on specific benchmarks. Nevertheless, the consistently stronger aggregate performance across nearly all configurations demonstrates that CARE-Q provides robust improvements under QuaRot-based low-bit quantization.

Table 9: Detailed downstream evaluation results under FlatQuant-based W2A4KV4 full quantization. Both per-channel (PC) and group-wise quantization with group size 128 (G128) are evaluated. WikiText and C4 report perplexity, while Avg denotes the average accuracy over 8 zero-shot tasks.

Model	Config	Method	Wiki↓	C4↓	ARC-E	ARC-C	PIQA	SIQA	HellaS	OBQA	WinoG	BoolQ	Avg↑
Llama3-8B	PC	GPTQ	16.5	43.6	52.0	29.2	63.7	37.5	47.1	32.6	56.3	67.1	48.2
		GPTAQ	13.9	38.2	48.9	28.5	63.8	37.8	47.9	32.2	57.4	65.3	47.7
		ResComp	14.1	38.9	50.3	30.0	65.1	37.7	47.1	32.2	58.6	62.9	48.0
		CARE-Q	**12.5**	**29.3**	50.1	31.6	66.7	38.4	51.3	32.6	59.2	65.1	**49.4**
	G128	GPTQ	15.3	37.2	50.9	28.4	62.6	37.8	49.1	32.0	58.0	66.2	48.1
		GPTAQ	13.0	32.2	51.1	29.2	64.0	39.2	51.1	33.2	59.9	66.3	49.2
		ResComp	13.0	33.2	49.7	29.5	63.8	38.3	49.8	31.2	60.6	67.1	48.8
		CARE-Q	**12.1**	**26.5**	48.6	30.1	65.2	38.6	53.6	33.8	60.4	66.4	**49.6**
Llama3.1-8B	PC	GPTQ	18.8	48.1	51.5	30.3	63.5	39.0	49.8	32.6	60.5	63.7	48.9
		GPTAQ	15.5	37.9	51.8	31.1	65.1	39.8	49.6	31.8	57.9	65.0	49.0
		ResComp	15.4	37.9	56.6	30.7	65.7	39.3	49.0	30.8	61.0	64.4	49.7
		CARE-Q	**14.1**	**31.9**	53.9	32.0	67.8	40.4	55.3	32.8	58.5	68.5	**51.1**
	G128	GPTQ	17.7	43.9	56.7	32.9	66.2	39.4	51.0	30.4	59.0	63.6	49.9
		GPTAQ	14.1	33.7	55.3	31.7	66.4	41.1	51.0	31.8	62.4	71.1	51.4
		ResComp	14.1	34.2	53.0	29.9	66.0	39.8	50.7	32.4	61.2	67.9	50.1
		CARE-Q	**13.5**	**28.5**	54.9	32.9	67.0	39.8	55.6	32.4	61.5	70.0	**51.8**
Llama3.2-3B	PC	GPTQ	36.6	77.9	42.9	25.5	59.1	36.9	40.0	30.6	53.8	65.1	44.2
		GPTAQ	27.2	60.0	41.0	25.6	59.7	37.0	39.0	28.2	54.5	64.9	43.8
		ResComp	27.0	63.8	42.5	25.7	58.5	37.5	38.4	27.6	54.4	62.5	43.4
		CARE-Q	**22.8**	**43.6**	42.1	26.9	60.7	38.2	44.6	29.2	52.7	64.2	**44.8**
	G128	GPTQ	33.7	73.3	43.6	25.7	59.6	37.9	41.4	29.6	54.1	64.4	44.5
		GPTAQ	23.7	50.4	42.1	25.1	59.7	37.3	42.1	28.0	55.2	65.5	44.4
		ResComp	24.3	50.9	40.5	25.4	59.9	36.8	41.8	27.6	53.4	63.3	43.6
		CARE-Q	**22.5**	**41.0**	41.1	27.1	59.0	36.2	46.1	29.2	55.8	65.8	**45.0**
Qwen3-4B	PC	GPTQ	22.1	43.8	51.4	31.7	63.9	39.8	46.0	29.4	56.5	71.1	48.7
		GPTAQ	24.6	48.0	49.1	28.9	62.3	39.2	43.6	29.2	56.6	72.2	47.6
		ResComp	24.7	47.8	44.8	28.9	63.1	39.0	43.7	27.6	54.9	63.7	45.7
		CARE-Q	**21.8**	**41.2**	49.7	31.3	63.4	40.0	45.5	31.2	56.8	72.3	**48.8**
	G128	GPTQ	20.8	38.9	50.6	30.7	64.6	39.3	47.1	31.2	55.4	60.6	47.5
		GPTAQ	22.3	41.5	46.0	29.9	62.2	38.6	43.9	28.4	56.8	72.0	47.2
		ResComp	22.3	40.9	49.6	29.6	63.1	38.3	43.9	28.4	56.8	72.8	47.8
		CARE-Q	**20.2**	**37.1**	52.1	32.1	64.8	40.1	46.4	30.4	58.0	73.1	**49.6**
Qwen3-8B	PC	GPTQ	15.6	31.0	58.7	38.0	68.5	40.2	53.8	32.4	62.9	71.7	53.3
		GPTAQ	16.3	32.6	58.8	36.3	67.4	40.2	51.4	34.6	59.7	67.3	51.9
		ResComp	17.1	34.4	55.1	34.6	66.6	41.4	51.0	34.6	61.3	67.7	51.5
		CARE-Q	**14.7**	**27.3**	62.2	38.6	69.3	41.8	56.1	35.8	59.1	75.8	**54.9**
	G128	GPTQ	14.4	28.2	61.1	37.6	68.5	40.8	54.9	34.4	62.9	69.7	53.8
		GPTAQ	15.4	30.1	59.0	34.7	67.8	41.2	51.2	32.8	62.5	65.0	51.8
		ResComp	15.4	30.1	57.5	34.9	66.8	40.8	50.8	32.8	60.2	68.7	51.6
		CARE-Q	**14.2**	**26.2**	62.0	35.8	69.1	41.7	55.4	35.0	60.7	77.9	**54.7**
Qwen3-14B	PC	GPTQ	13.4	25.2	68.6	43.2	70.6	43.5	60.1	38.4	64.9	83.0	59.0
		GPTAQ	13.5	27.1	64.4	42.1	71.1	43.8	58.9	39.2	65.4	83.8	58.6
		ResComp	13.4	26.3	66.6	42.1	71.1	44.6	58.9	38.8	66.0	83.8	59.0
		CARE-Q	**12.3**	**23.1**	67.9	43.4	72.8	44.8	62.6	39.2	65.1	82.5	**59.8**
	G128	GPTQ	12.4	22.6	69.2	43.3	72.1	43.4	62.1	39.8	65.2	82.3	59.7
		GPTAQ	12.6	23.2	72.2	44.5	72.0	45.5	61.3	39.0	67.1	76.2	59.7
		ResComp	12.5	23.4	70.0	44.9	70.9	43.3	59.9	36.8	65.5	79.3	58.8
		CARE-Q	**11.9**	**21.6**	72.3	45.8	72.9	45.6	63.3	40.0	66.1	83.6	**61.2**

C.4 Detailed Results under FlatQuant-based Low-bit Quantization

Tables 8 and 9 provide detailed evaluation results under FlatQuant-based low-bit quantization. Table 8 reports W2A16KV16 weight-only quantization results, while Table 9 presents W2A4KV4 full quantization results. Both per-channel (PC) and group-wise quantization with a group size of 128 (G128) are evaluated. The tables include WikiText and C4 perplexities, average zero-shot accu-

Table 10: Additional W2A16KV16 weight-only quantization results on larger models with QuaRot preprocessing and per-channel (PC) weight quantization. Each entry reports WikiText PPL / C4 PPL / average zero-shot accuracy over eight downstream tasks.

Model	Method	Wiki↓	C4↓	ARC-E	ARC-C	PIQA	SIQA	HellaS	OBQA	WinoG	BoolQ	Avg↑
Qwen3-32B	FP	7.61	12.45	83.3	60.9	82.1	52.3	82.6	46.6	72.5	86.4	70.8
	GPTQ	11.79	17.48	75.2	50.2	76.4	48.4	69.6	38.0	68.4	84.8	63.9
	GPTAQ	11.70	17.05	76.1	50.0	77.5	46.6	69.4	40.4	71.0	84.4	64.4
	ResComp	11.74	17.13	72.8	48.5	76.2	45.9	69.0	38.2	72.5	83.1	63.3
	CARE-Q	**11.38**	**16.79**	77.2	52.1	76.3	47.7	71.7	41.8	69.2	85.1	**65.1**
Llama3-70B	FP	2.86	7.17	85.9	64.1	84.5	50.6	84.9	48.8	80.6	85.1	73.1
	GPTQ	36.53	57.31	40.1	23.6	63.9	35.7	36.2	27.2	52.7	61.7	42.6
	GPTAQ	21.22	28.24	56.6	32.7	71.7	40.6	50.0	33.6	62.8	65.4	51.7
	ResComp	20.06	27.90	53.2	30.0	69.3	40.0	43.7	32.6	64.9	62.7	49.5
	CARE-Q	**18.34**	**25.51**	56.1	33.1	73.2	41.4	51.0	36.0	64.6	67.3	**52.9**

Table 11: Additional ablation results under QuaRot W2A16KV16 and FlatQuant W2A4KV4 quantization settings. Each entry reports WikiText perplexity, C4 perplexity, and average zero-shot accuracy.

Config	Method	Llama3-8B			Llama3.1-8B			Llama3.2-3B			Qwen3-4B			Qwen3-8B			Qwen3-14B		
		Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg
QuaRot (PC)	GPTQ	40.8	52.0	43.7	38.3	52.3	46.3	93.0	111.6	39.9	61.1	73.1	41.0	22.0	29.3	53.6	15.0	21.3	60.5
	CARE-Q w/o DiCR	24.9	30.1	48.9	23.4	29.4	50.6	42.7	46.2	43.0	39.5	48.2	42.8	20.6	26.8	52.9	14.5	20.4	60.5
	CARE-Q	**21.3**	**25.2**	**49.7**	**18.5**	**24.7**	**51.8**	**39.0**	**42.1**	**44.5**	**40.3**	**47.8**	**42.7**	**19.9**	**26.6**	**52.8**	**14.2**	**20.3**	**60.8**
QuaRot (G128)	GPTQ	35.4	47.3	45.2	29.6	41.9	47.9	75.0	84.3	39.2	39.1	52.0	41.6	18.2	25.7	54.0	13.1	19.6	62.0
	CARE-Q w/o DiCR	21.3	26.1	50.4	19.7	26.3	52.9	34.0	39.5	44.4	33.5	41.3	43.3	17.1	23.9	53.8	13.3	19.0	62.3
	CARE-Q	**17.0**	**21.5**	**51.9**	**17.3**	**23.8**	**52.6**	**33.0**	**36.6**	**44.9**	**30.5**	**39.0**	**43.8**	**17.4**	**24.1**	**54.9**	**13.1**	**19.0**	**62.3**
FlatQuant (PC)	GPTQ	16.5	43.6	48.2	18.8	48.1	48.9	36.6	77.9	44.2	22.1	43.8	48.7	15.6	31.0	53.3	13.4	25.2	59.0
	CARE-Q w/o DiCR	13.4	33.4	48.5	15.1	36.8	50.0	27.5	60.1	43.9	22.6	42.9	48.7	15.0	27.9	55.8	12.5	23.6	59.7
	CARE-Q	**12.5**	**29.3**	**49.4**	**14.1**	**31.9**	**51.1**	**22.8**	**43.6**	**44.8**	**21.8**	**41.2**	**48.8**	**14.7**	**27.3**	**54.9**	**12.3**	**23.1**	**59.8**
FlatQuant (G128)	GPTQ	15.3	37.2	48.1	17.7	43.9	49.9	33.7	73.3	44.5	20.8	38.9	47.5	14.4	28.2	53.8	12.4	22.6	59.7
	CARE-Q w/o DiCR	12.6	30.6	48.7	14.2	32.9	50.8	24.6	51.3	44.6	21.1	38.4	48.9	14.2	26.7	55.9	12.2	21.8	59.2
	CARE-Q	**12.1**	**26.5**	**49.6**	**13.5**	**28.5**	**51.8**	**22.5**	**41.0**	**45.0**	**20.2**	**37.1**	**49.6**	**14.2**	**26.2**	**54.7**	**11.9**	**21.6**	**61.2**

racy, and individual accuracies on the eight downstream tasks. CARE-Q consistently improves over existing compensation-based baselines across different models and quantization settings.

C.5 Scaling CARE-Q to Larger Models

We further investigate whether the improvements of CARE-Q persist at larger model scales by evaluating Qwen3-32B and Llama3-70B under QuaRot W2A16KV16 per-channel quantization. As shown in Table 10, CARE-Q achieves the best WikiText perplexity, C4 perplexity, and average downstream accuracy on both models. On Qwen3-32B, CARE-Q improves WikiText/C4 perplexity from 11.70/17.05 to 11.38/16.79 and average accuracy from 64.4 to 65.1 over GPTAQ. The improvements become more pronounced on Llama3-70B: CARE-Q reduces WikiText/C4 perplexity from the best baseline results of 20.06/27.90 to 18.34/25.51, while increasing average accuracy from 51.7 to 52.9. These results show that the gains of CARE-Q persist at substantially larger model scales, extending its effectiveness to models with up to 70B parameters.

C.6 Additional Ablation Results

Table 11 provides additional ablation results under QuaRot W2A16KV16 and FlatQuant W2A4KV4 settings, complementing the main ablation study. Across these settings, CARE-Q w/o DiCR generally improves quantization quality over GPTQ, supporting the effectiveness of the proposed compensation mechanism. Adding DiCR further improves perplexity and average downstream accuracy in most configurations. Although gains are not uniform across all metrics, the overall results support the complementary benefits of compensation-aware rounding and enhanced error compensation across different transformation frameworks and quantization granularities.

Table 12: Calibration set size analysis under W2 quantization with FlatQuant. Results are reported with different numbers of calibration samples ($n_s \in \{64, 96, 128, 192\}$) under per-channel quantization. For W2A16KV16, calibration samples are collected from C4, while W2A4KV4 uses WikiText following previous compensation-based PTQ methods. Each entry reports WikiText PPL / C4 PPL / average zero-shot accuracy.

n_s	Method	W2A16KV16						W2A4KV4					
		Llama3-8B			Qwen3-8B			Llama3-8B			Qwen3-8B		
		Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg
64	GPTQ	24.4	28.5	49.7	20.5	26.6	54.0	16.8	45.2	47.2	16.1	31.2	54.1
	GPTAQ	21.4	23.9	51.3	20.0	26.4	55.1	14.6	38.0	47.8	17.0	33.2	52.5
	ResComp	22.6	24.4	51.0	21.8	26.7	54.6	15.2	43.7	47.2	16.7	35.3	50.6
	CARE-Q	**16.5**	**21.2**	**52.1**	**19.0**	**25.8**	**55.9**	**13.3**	**32.3**	**48.8**	**15.1**	**27.2**	**54.5**
96	GPTQ	22.3	28.2	50.9	19.7	26.0	55.2	16.8	46.7	47.4	15.4	31.3	53.5
	GPTAQ	19.0	23.3	51.2	19.9	25.2	55.7	14.2	37.7	47.9	16.8	31.9	52.9
	ResComp	20.0	23.8	51.4	21.5	25.6	55.5	14.2	38.7	47.8	17.3	34.7	52.3
	CARE-Q	**15.6**	**20.7**	**52.9**	**16.7**	**23.8**	**56.6**	**12.7**	**30.2**	**49.2**	**14.8**	**27.9**	**54.9**
128	GPTQ	21.5	26.9	50.6	19.3	26.3	56.0	16.5	43.6	48.2	15.6	31.0	53.3
	GPTAQ	18.3	22.8	51.4	19.6	24.9	55.5	13.9	38.2	47.7	16.3	32.6	51.9
	ResComp	18.7	22.9	51.4	20.3	25.0	55.0	14.1	38.9	48.0	17.1	34.4	51.5
	CARE-Q	**14.7**	**19.9**	**52.8**	**16.7**	**23.4**	**57.2**	**12.5**	**29.3**	**49.4**	**14.7**	**27.3**	**54.9**
192	GPTQ	20.6	26.4	51.8	20.2	25.6	55.9	16.3	42.0	47.5	15.5	31.1	52.5
	GPTAQ	17.6	22.2	51.9	18.9	24.4	54.9	14.0	37.0	48.7	16.9	32.4	51.2
	ResComp	18.7	22.7	51.4	19.8	24.5	55.7	13.8	38.5	48.5	16.6	33.8	51.5
	CARE-Q	**14.2**	**19.7**	**53.6**	**16.3**	**23.0**	**57.6**	**12.4**	**29.1**	**49.2**	**14.4**	**26.1**	**54.6**

C.7 Calibration Budget Analysis

We investigate the sensitivity of CARE-Q to the calibration budget under W2 quantization with FlatQuant. Following GPTAQ (Li et al., 2025) and ResComp (Li et al., 2026), we use C4 as the calibration dataset for W2A16KV16 and WikiText as the calibration dataset for W2A4KV4. Table 12 reports results with different numbers of calibration samples ($n_s \in \{64, 96, 128, 192\}$) under per-channel quantization.

CARE-Q consistently achieves the best or tied-best results across all calibration budgets, outperforming existing compensation-based baselines regardless of the number of calibration samples. The consistent gains across different calibration sizes and bit-width settings demonstrate that the effectiveness of CARE-Q is not sensitive to a specific calibration budget.

C.8 Sensitivity Analysis of CARE-Q Coefficients

We further analyze the sensitivity of CARE-Q to the two coefficients β and ρ on Llama-3-8B with FlatQuant preprocessing. We evaluate different coefficient combinations under four W2 quantization settings: W2A16KV16 and W2A4KV4 with both per-channel and group-wise quantization. The results are reported in Tables 13–16.

Across all four settings, CARE-Q exhibits a broad high-performing region rather than relying on a narrowly tuned coefficient pair. Performance remains consistently strong over a wide range of ρ and β, with noticeable degradation occurring mainly under more aggressive coefficient choices. This consistent pattern across activation precisions and weight granularities demonstrates that CARE-Q is robust to coefficient variation.

Following the protocol in Appendix C.1, for this model and preprocessing setting we select $(\rho, \beta) = (0.2, 0.3)$ using the same 128 calibration samples and keep it fixed across all four quantization configurations. Without per-configuration retuning, this single pair remains consistently competitive under W2A16KV16 and W2A4KV4 with both PC and G128 quantization, further demonstrating the stability of CARE-Q across different activation precisions and weight granularities.

Table 13: Sensitivity analysis of β and ρ under W2A16KV16-PC quantization on Llama-3-8B. Each entry reports WikiText / C4 perplexity (lower is better). For reference, GPTQ, GPTAQ, and ResComp achieve 21.5 / 26.9, 18.3 / 22.8, and 18.7 / 22.9, respectively.

$\rho\backslash\beta$	0.1	0.2	0.25	0.3	0.4	0.5	0.6	0.75
0.1	16.6 / 21.5	15.8 / 20.6	15.2 / 20.4	15.2 / 20.4	14.6 / 19.7	14.2 / 19.5	13.9 / 19.3	15.0 / 20.3
0.2	15.9 / 21.2	15.9 / 20.5	14.6 / 20.1	14.7 / 19.9	14.4 / 19.6	14.3 / 19.5	14.4 / 20.0	16.3 / 22.2
0.25	15.5 / 20.9	14.8 / 20.4	15.1 / 20.2	14.4 / 19.7	14.2 / 19.6	14.4 / 19.7	14.7 / 20.4	18.0 / 24.2
0.3	15.8 / 21.1	15.3 / 20.7	14.6 / 20.1	14.6 / 20.2	14.6 / 19.8	14.7 / 20.0	15.3 / 21.4	23.1 / 27.4
0.4	16.1 / 21.5	15.7 / 21.1	15.3 / 20.8	15.3 / 20.6	15.5 / 20.8	16.6 / 22.1	21.3 / 25.8	56.5 / 57.7

Table 14: Sensitivity analysis of β and ρ under W2A16KV16-G128 quantization on Llama-3-8B. Each entry reports WikiText / C4 perplexity (lower is better). GPTQ, GPTAQ, and ResComp obtain 18.7 / 24.5, 16.1 / 21.1, and 16.8 / 21.2, respectively.

$\rho\backslash\beta$	0.1	0.2	0.25	0.3	0.4	0.5	0.6	0.75
0.1	15.5 / 20.3	14.7 / 19.8	14.2 / 19.6	14.4 / 19.4	13.7 / 19.2	13.5 / 19.0	13.7 / 19.2	15.3 / 21.2
0.2	15.0 / 19.9	14.3 / 19.4	14.0 / 19.3	13.8 / 19.5	13.8 / 19.2	13.7 / 19.3	13.7 / 20.0	17.3 / 24.4
0.25	14.9 / 20.0	14.0 / 19.5	14.2 / 19.5	13.8 / 19.5	13.6 / 19.3	13.9 / 19.5	14.7 / 20.8	21.4 / 27.5
0.3	14.6 / 20.1	14.1 / 19.6	14.1 / 19.6	13.8 / 19.5	13.9 / 19.5	14.2 / 20.1	15.8 / 22.2	23.7 / 32.2
0.4	14.9 / 20.5	14.6 / 20.3	14.6 / 20.0	14.6 / 20.5	14.8 / 20.5	16.9 / 23.1	22.7 / 28.6	134.2 / 113.5

Table 15: Sensitivity analysis of β and ρ under W2A4KV4-PC quantization on Llama-3-8B. Each entry reports WikiText / C4 perplexity (lower is better). GPTQ, GPTAQ, and ResComp obtain 16.5 / 43.6, 13.9 / 38.2, and 14.1 / 38.9, respectively.

$\rho\backslash\beta$	0.1	0.2	0.25	0.3	0.4	0.5	0.6	0.75
0.1	13.1 / 32.1	12.8 / 31.7	12.6 / 30.1	12.6 / 29.7	12.5 / 28.4	12.5 / 28.3	13.1 / 29.5	15.2 / 33.1
0.2	12.9 / 31.7	12.6 / 29.9	12.6 / 30.1	12.5 / 29.3	12.6 / 29.0	12.9 / 28.7	13.9 / 30.5	18.9 / 44.2
0.25	12.8 / 30.8	12.7 / 29.2	12.7 / 28.7	12.6 / 28.9	12.7 / 28.2	13.2 / 29.0	14.6 / 32.9	22.9 / 53.2
0.3	13.1 / 33.0	12.7 / 30.1	12.7 / 29.5	12.8 / 29.2	13.0 / 28.7	13.9 / 32.2	16.0 / 33.9	32.6 / 73.3
0.4	13.4 / 31.4	13.4 / 30.1	13.5 / 31.3	13.7 / 31.1	14.7 / 33.5	18.0 / 42.3	28.5 / 64.1	149.4 / 243.1

Table 16: Sensitivity analysis of β and ρ under W2A4KV4-G128 quantization on Llama-3-8B. Each entry reports WikiText / C4 perplexity (lower is better). GPTQ, GPTAQ, and ResComp obtain 15.3 / 37.2, 13.0 / 32.2, and 13.0 / 33.2, respectively.

$\rho\backslash\beta$	0.1	0.2	0.25	0.3	0.4	0.5	0.6	0.75
0.1	12.4 / 29.5	12.4 / 28.7	12.1 / 27.7	12.0 / 27.3	12.1 / 26.4	12.3 / 26.4	13.1 / 27.8	17.0 / 37.1
0.2	12.3 / 28.6	12.2 / 27.1	12.2 / 27.7	12.1 / 26.5	12.2 / 26.3	12.9 / 27.8	14.4 / 29.4	25.0 / 55.5
0.25	12.4 / 28.3	12.4 / 28.4	12.3 / 27.2	12.3 / 26.8	12.5 / 26.5	13.4 / 28.4	16.0 / 36.0	34.1 / 87.8
0.3	12.5 / 27.9	12.4 / 27.2	12.4 / 26.9	12.5 / 26.7	13.0 / 27.9	14.5 / 30.9	18.8 / 39.5	66.1 / 135.5
0.4	13.0 / 29.7	13.1 / 28.6	13.3 / 28.5	13.3 / 28.4	14.7 / 33.9	19.0 / 42.2	53.1 / 109.6	706.6 / 635.1

C.9 Analysis of Component Contributions

To further isolate the contributions controlled by β and ρ, we evaluate different coefficient combinations on Llama-3-8B with FlatQuant preprocessing under W2 quantization. Specifically, we compare $(\rho, \beta) \in \{(0,0), (0.2,0), (0,0.3), (0.2,0.3)\}$ across W2A16KV16 and W2A4KV4 with both per-channel and group-wise quantization. Setting $\beta = 0$ disables the DiCR adjustment, while setting $\rho = 0$ removes the accumulated lower-triangular propagation component in ODPP. In particular, $(0.2, 0)$ corresponds to CARE-Q w/o DiCR in Table 3.

As shown in Table 17, enabling either the DiCR adjustment or the ρ-controlled propagation component in ODPP individually reduces both WikiText and C4 perplexity over $(0, 0)$ across all four settings. Combining the two further reduces both perplexities relative to either individual configuration in every setting, while also achieving the highest average downstream accuracy across all four settings. These results provide direct evidence that DiCR and ODPP make complementary contri-

Table 17: Component contribution analysis of CARE-Q on Llama-3-8B with FlatQuant preprocessing under W2 quantization. The zero settings isolate the DiCR adjustment controlled by β and the additional propagation component controlled by ρ. Wiki and C4 denote perplexity (↓), and Avg denotes the 8-task zero-shot average accuracy (↑).

ρ	β	A16-PC			A16-G128			A4-PC			A4-G128		
		Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg	Wiki	C4	Avg
0.0	0.0	18.33	22.84	51.4	16.11	21.09	53.0	13.94	38.18	47.7	12.98	32.19	49.2
0.2	0.0	17.26	21.64	51.9	15.59	20.70	53.2	13.38	33.37	48.5	12.63	30.62	48.7
0.0	0.3	15.54	20.63	52.5	14.41	19.74	53.2	12.79	31.50	49.0	12.22	28.01	48.9
0.2	0.3	14.72	19.93	52.8	13.77	19.48	54.0	12.52	29.30	49.4	12.07	26.52	49.6

butions to CARE-Q, supporting our quantization error lifecycle view that error formation and error compensation should be jointly optimized rather than treated as isolated stages.

Table 18: Hyperparameter calibration of ResComp. Each entry reports WikiText / C4 perplexity (lower is better) with different residual compensation coefficients r_a. The setting $r_a = 0$ removes the additional residual compensation term and reduces to the GPTAQ formulation.

Framework	Model	Config	r_a					
			0	0.1	0.2	0.25	0.3	0.4
FlatQuant	Llama3-8B	W2A16KV16-PC	18.3 / 22.8	18.7 / 22.9	20.7 / 23.5	21.5 / 24.2	22.8 / 24.7	25.0 / 26.4
		W2A16KV16-G128	16.1 / 21.1	16.8 / 21.2	18.6 / 21.6	18.6 / 21.8	19.4 / 23.2	20.5 / 23.9
		W2A4KV4-PC	13.9 / 38.2	14.1 / 38.9	14.2 / 40.2	14.3 / 40.7	14.5 / 41.7	14.9 / 42.5
		W2A4KV4-G128	13.0 / 32.2	13.0 / 33.2	13.2 / 34.6	13.1 / 33.6	13.3 / 35.5	13.9 / 37.5
	Qwen3-4B	W2A16KV16-PC	31.2 / 34.5	30.9 / 33.8	32.8 / 33.9	34.2 / 34.4	39.9 / 37.2	36.5 / 37.1
		W2A16KV16-G128	25.5 / 30.9	24.8 / 30.8	25.7 / 31.1	25.5 / 31.1	26.6 / 30.9	27.4 / 32.2
		W2A4KV4-PC	24.6 / 48.0	24.7 / 47.8	26.0 / 49.5	25.8 / 54.4	28.1 / 52.6	26.9 / 54.6
		W2A4KV4-G128	22.3 / 41.5	22.3 / 40.9	22.0 / 41.1	21.9 / 42.4	22.4 / 43.3	22.6 / 44.3
QuaRot	Llama3-8B	W2A16KV16-PC	23.1 / 27.5	26.7 / 30.1	23.3 / 27.7	26.3 / 29.5	26.1 / 29.0	24.8 / 28.8
		W2A16KV16-G128	19.5 / 25.0	20.3 / 24.7	21.1 / 26.4	19.8 / 24.7	20.7 / 24.9	20.3 / 25.3
	Qwen3-4B	W2A16KV16-PC	42.6 / 49.9	41.6 / 49.7	42.1 / 50.3	44.5 / 52.9	44.6 / 53.2	46.9 / 53.9
		W2A16KV16-G128	32.5 / 40.8	33.1 / 40.5	32.0 / 39.1	33.5 / 41.1	31.7 / 39.8	35.8 / 41.5

C.10 HYPERPARAMETER CALIBRATION OF RESCOMP

To ensure a fair comparison, we perform a dedicated calibration of ResComp by searching its residual compensation coefficient r_a. The search is conducted on representative models (Llama-3-8B and Qwen3-4B) under both FlatQuant and QuaRot preprocessing. We evaluate $r_a \in \{0, 0.1, 0.2, 0.25, 0.3, 0.4\}$ while keeping the other ResComp settings fixed, including `gptaq_alpha`=0.25 and `rescomp_mode=org`. With all other settings kept identical, setting $r_a = 0$ removes the additional residual compensation term and recovers the GPTAQ update; this case is therefore represented by the GPTAQ baseline. WikiText and C4 perplexities are reported in Table 18.

The results show that the appropriate compensation strength varies across different models and quantization configurations. Based on the overall trends in Table 18, we use $r_a = 0.1$ for experiments with FlatQuant and $r_a = 0.2$ for experiments with QuaRot. These values are shared across the evaluated models and configurations rather than being tuned separately for each individual setting. The official ResComp implementation uses $r_a = 0.25$ by default, whereas $r_a = 0.1$ and $r_a = 0.2$ provide better overall performance in our FlatQuant and QuaRot evaluations, respectively. Even with these calibrated ResComp configurations, CARE-Q consistently achieves stronger overall performance across the evaluated quantization settings.

Table 19: Ablation of the proxy reference used by DiCR on Llama3-8B with FlatQuant preprocessing. "Current" uses the evolving working column $\mathbf{w}_t$, whereas "Original" uses the fixed full-precision reference $\mathbf{w}_t^{\mathrm{ref}}$. Wiki and C4 denote perplexity; lower is better. Relative changes are computed for Original with respect to Current using the unrounded perplexity values.

Configuration	Current		Original		Relative change	
	Wiki	C4	Wiki	C4	Wiki	C4
W2A16KV16-PC	15.3	20.3	**14.7**	**19.9**	-3.63%	-1.79%
W2A16KV16-G128	13.9	19.8	**13.8**	**19.5**	-0.92%	-1.27%
W2A4KV4-PC	12.7	30.4	**12.5**	**29.3**	-1.40%	-3.56%
W2A4KV4-G128	**12.1**	**26.5**	**12.1**	**26.5**	$+0.06\%$	-0.02%

C.11 Ablation on the DiCR Proxy Reference

The derivation in Appendix A.3 shows that $-g_t\mathbf{w}_t$ is exactly the current-column self term in the input-path contribution to the ideal shift. In practical sequential quantization, however, $\mathbf{w}_t$ evolves with preceding quantization and compensation updates. DiCR instead uses the fixed full-precision reference $\mathbf{w}_t^{\mathrm{ref}}$ when constructing its surrogate shift.

To examine this design choice, we compare the evolving current working column ("Current") with the fixed original full-precision column ("Original") on Llama3-8B with FlatQuant preprocessing. We evaluate both per-channel (PC) and group-wise (G128) quantization under W2A16 and W2A4KV4 settings, reporting perplexity on WikiText2 and C4.

As shown in Table 19, the fixed reference improves perplexity in seven of the eight evaluated dataset–configuration pairs and is essentially unchanged in the remaining case. The gains are more pronounced for W2A16-PC and W2A4KV4-PC, while the two references are nearly indistinguishable under W2A4KV4-G128. These results support using $\mathbf{w}_t^{\mathrm{ref}}$ as the practical proxy reference in DiCR, removing additional dependence of the rounding surrogate on the evolving sequential working state.

C.12 Calibration Dataset Robustness

We further investigate whether CARE-Q remains effective under diverse calibration data distributions. Following the experimental settings of existing compensation-based methods, including GPTAQ and ResComp, our main weight-only quantization experiments use C4 as the calibration dataset for fair comparison. In this analysis, we additionally evaluate three alternative calibration datasets, including WikiText2, the RedPajama-Stack subset of RedPajama (Weber et al., 2024), and The Pile (Gao et al., 2020), while keeping the evaluation protocol unchanged. For CARE-Q, the hyperparameters are selected using C4 calibration and directly applied to other calibration datasets without dataset-specific re-tuning. All methods are evaluated on the same WikiText and C4 test sets, together with the average accuracy over eight zero-shot tasks. The results are summarized in Table 20.

CARE-Q consistently achieves better overall performance than existing compensation-based baselines across different calibration datasets. While different calibration distributions lead to variations in the absolute performance of all methods, CARE-Q maintains clear advantages in both perplexity and downstream accuracy under diverse calibration settings, including the challenging RedPajama-Stack and The Pile calibration cases. These results indicate that CARE-Q is not strongly dependent on a specific calibration dataset and can effectively generalize across diverse calibration distributions.

C.13 Additional Results on Complementarity with Transformation-based Quantization

We provide additional results on Llama3.2-3B and Qwen3-4B to complement the analysis in Table 5. Following the same W2A16KV16 per-channel quantization setting, we compare transformation-only baselines without compensation, CARE-Q applied directly to untransformed models, and the

Table 20: Calibration dataset robustness under FlatQuant W2A16KV16-PC. Different calibration datasets, including WikiText2, C4, RedPajama-Stack, and Pile, are evaluated without re-tuning CARE-Q hyperparameters. Each entry reports WikiText PPL / C4 PPL / average zero-shot accuracy. Lower perplexity and higher accuracy indicate better performance.

Calibration	Method	Llama3-8B	Llama3.1-8B	Qwen3-8B
WikiText2	GPTQ	14.33 / 31.60 / 49.95	16.76 / 40.05 / 51.55	15.12 / 29.03 / 55.07
	GPTAQ	12.24 / 28.13 / 49.63	13.53 / 31.57 / 51.83	15.70 / 29.84 / 54.73
	ResComp	12.58 / 29.56 / 49.98	13.66 / 32.42 / 51.20	16.00 / 30.48 / 54.42
	CARE-Q	**11.30 / 23.88 / 50.87**	**12.76 / 27.44 / 52.98**	**14.28 / 27.07 / 55.35**
C4	GPTQ	21.49 / 26.93 / 50.60	24.82 / 33.00 / 53.45	19.31 / 26.29 / 56.02
	GPTAQ	18.33 / 22.84 / 51.38	19.83 / 25.71 / 53.18	19.60 / 24.95 / 55.47
	ResComp	18.74 / 22.94 / 51.43	20.71 / 26.28 / 51.76	20.28 / 24.98 / 55.02
	CARE-Q	**14.72 / 19.93 / 52.79**	**16.39 / 23.28 / 54.76**	**16.67 / 23.45 / 57.22**
RedPajama-Stack	GPTQ	26.28 / 39.54 / 48.55	33.34 / 52.32 / 50.63	24.39 / 34.41 / 53.03
	GPTAQ	25.60 / 35.92 / 49.86	27.68 / 39.48 / 51.00	25.58 / 37.09 / 52.10
	ResComp	27.10 / 38.37 / 48.25	28.89 / 42.11 / 50.46	27.83 / 37.65 / 51.34
	CARE-Q	**19.67 / 28.04 / 50.99**	**20.85 / 31.10 / 52.02**	**20.54 / 30.12 / 54.08**
Pile	GPTQ	21.36 / 31.53 / 49.38	24.28 / 40.34 / 50.55	20.14 / 29.03 / 53.68
	GPTAQ	18.99 / 28.77 / 49.20	21.23 / 32.84 / 50.47	20.87 / 29.84 / 52.88
	ResComp	19.49 / 29.57 / 49.12	21.31 / 33.59 / 48.79	21.19 / 30.09 / 52.89
	CARE-Q	**14.95 / 23.34 / 51.43**	**16.44 / 26.59 / 52.08**	**16.95 / 25.88 / 54.93**

Table 21: Additional results on the complementarity between CARE-Q and transformation-based quantization under W2A16KV16-PC. QuaRot+RTN and FlatQuant+RTN denote transformation-only baselines without compensation, while Raw+CARE-Q applies CARE-Q directly to the untransformed model. Each entry reports WikiText PPL / C4 PPL / average zero-shot accuracy, where lower perplexity and higher accuracy indicate better performance.

Method	Llama3.2-3B			Qwen3-4B		
	Wiki	C4	Avg	Wiki	C4	Avg
QuaRot + RTN	6.2e5	5.3e5	36.4	1.6e7	1.1e7	37.1
FlatQuant + RTN	1.0e5	8.1e4	35.6	3.5e6	3.4e6	36.7
Raw + CARE-Q	257.1	135.3	38.5	137.8	113.6	40.4
QuaRot + CARE-Q	39.0	42.1	44.5	40.3	47.8	42.7
FlatQuant + CARE-Q	**26.6**	**32.2**	**47.6**	**26.6**	**31.4**	**52.2**

combinations of CARE-Q with QuaRot or FlatQuant. As shown in Table 21, the additional models exhibit the same trend as the main results. QuaRot+RTN and FlatQuant+RTN remain severely degraded in this extreme low-bit regime, whereas Raw+CARE-Q substantially recovers quantization quality. More importantly, combining CARE-Q with either QuaRot or FlatQuant further reduces perplexity and improves average downstream accuracy over Raw+CARE-Q on both models. These results further support the complementarity between transformation-based preprocessing and CARE-Q across different model families.

C.14 Distinction from Optimization-Based Adaptive Rounding

Adaptive rounding and DiCR both depart from conventional round-to-nearest (RTN), but they do so through fundamentally different mechanisms. AdaRound (Nagel et al., 2020) formulates rounding decisions as an optimization problem by introducing learnable rounding variables and optimizing a relaxed layer-wise reconstruction objective on calibration data. Such element-wise rounding variables scale with the number of quantized weights, making their optimization increasingly costly as model size grows. This scalability issue motivates recent LLM-oriented methods such as VQRound (Zhou et al., 2026), which reparameterizes dense rounding variables using compact vector-quantized codebooks. Nevertheless, VQRound remains optimization-based: following layer-

wise K-Means initialization, the codebook parameters are optimized end-to-end with Adam for 5,000 steps using 128 C4 calibration sequences of length 2048.

CARE-Q instead belongs to the family of compensation-based sequential quantization methods, alongside GPTQ, GPTAQ, and ResComp. Within this framework, DiCR modifies the code-selection step of sequential quantization without turning rounding into a separate optimization problem. It introduces neither learnable rounding variables nor a separate gradient-based rounding optimization stage. Instead, its rounding signal is derived directly from the compensation representation $\mathbf{\Gamma}$ already constructed by CARE-Q. Specifically, DiCR uses the diagonal signal g_t to select among adjacent legal quantization candidates, while ODPP exploits the off-diagonal propagation information from the same representation for error compensation. Therefore, code selection and compensation are coupled through a shared compensation representation within the same sequential reconstruction process.

This distinction is particularly important for large-scale LLM quantization. During the CARE-Q quantization stage, compensation-aware code selection is integrated directly into the compensation-based reconstruction process without introducing backward propagation, trainable rounding states, or an additional rounding-finetuning stage. As shown in Table 4, CARE-Q's offline quantization cost remains comparable to GPTAQ and ResComp: CARE-Q quantizes Llama3-8B in 989 ± 8 seconds under PC quantization on a single NVIDIA H20 GPU while achieving substantially stronger low-bit quantization quality. Thus, CARE-Q extends compensation-based sequential quantization by coupling compensation-aware code selection with error propagation through a shared compensation representation, rather than introducing a separate optimization-based adaptive-rounding stage.

C.15 Inference-Time Model Format and Overhead

CARE-Q operates exclusively during offline weight reconstruction and leaves the inference-time computation unchanged. The compensation representation $\mathbf{\Gamma}$, together with the DiCR and ODPP quantities derived from it, is used only to guide quantization code selection and sequential weight compensation. These quantities are transient and are discarded once the quantized weights have been reconstructed.

The final model therefore preserves the same low-bit representation as the corresponding baseline quantizer, consisting only of the quantized weights and their standard quantization parameters, such as scales and zero points. CARE-Q introduces no additional model parameters, persistent auxiliary states, runtime transformations, or inference-time compensation operators. For a fixed quantization format and inference backend, it consequently executes the same inference graph and low-bit linear operators as the corresponding baseline. Hence, the additional inference-time computation and storage introduced by CARE-Q are zero by construction; all CARE-Q-specific computation is confined to the offline quantization stage. This separation allows CARE-Q to exploit richer reconstruction-time information without altering the standard low-bit inference pipeline.

C.16 Summary of Additional Experiments

The additional experiments provide a comprehensive validation of CARE-Q from the perspectives of effectiveness, mechanism, robustness, and compatibility. Detailed evaluations across diverse LLMs, low-bit quantization settings, and quantization granularities demonstrate that CARE-Q consistently improves compensation-based quantization beyond the main experimental configurations. Ablation and component analyses further verify the complementary roles of error formation and error compensation, while coefficient sensitivity experiments show that CARE-Q maintains strong performance over a broad range of coefficient configurations. Moreover, robustness evaluations under different calibration budgets and calibration datasets demonstrate that CARE-Q remains effective across diverse calibration settings. Finally, the complementarity experiments with QuaRot and FlatQuant show that CARE-Q can be effectively combined with existing transformation-based quantization methods, further supporting the quantization error lifecycle view that error formation and error compensation should be jointly considered in low-bit LLM quantization.