A UNIFIED HESSIAN-GUIDED FRAMEWORK FOR POST-TRAINING QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-bit quantization enables efficient deployment of large language models by reducing memory footprint and computational cost. Achieving high accuracy at low precision requires selecting quantization ranges, linear transformations, and weight rounding decisions. Existing approaches typically optimize these components independently using local reconstruction objectives, which fail to account for the varying impact of quantization errors on downstream predictions. Direct optimization of task-level metrics would provide a more faithful objective, but is challenging due to the discontinuous nature of quantization operations and the high computational cost of evaluating their downstream effects.

In this work, we derive a Hessian-guided approximation of the output change induced by joint weight and activation quantization. Based on a tractable Fisher approximation, we formulate a unified post-training quantization framework that enables the joint optimization of quantization ranges, linear transformations, and weight rounding under a common output-sensitivity objective. The resulting method remains practical for modern large language models and consistently outperforms state-of-the-art transformation and rounding methods across both channel-wise and block-wise quantization settings.

1 INTRODUCTION

Large language models achieve strong performance across a wide range of language tasks, but their memory and computational requirements make deployment expensive. Quantization reduces these costs by representing weights and activations at lower precision, improving memory efficiency and enabling faster integer computation. The central challenge is to obtain these benefits while preserving the predictions of the full-precision model.

We focus on integer post-training quantization (PTQ) of both weights and activations. A small calibration set informs the quantization parameters. At a fixed bit width, accuracy depends on the quantization ranges, rounding choices, and representation basis. Quantization ranges set the scales: narrower ranges give finer grids but can increase clipping error. Rounding then selects weights on those grids. Both choices operate on a given representation of the layer. An invertible, function-preserving transform (van Breugel et al., 2026), such as a rotation or Hadamard transform (Ashkboos et al., 2024b), changes that representation by redistributing weight and activation magnitudes while preserving the full-precision layer output. Range selection and transforms can address both weights and activations, whereas adaptive rounding primarily addresses weights.

Local reconstruction evaluates these choices by the error they introduce at a layer's output. But, as shown in Figure 1, the magnitude of local perturbations is not necessarily a good proxy for the changes in the model output. Existing methods (Tseng et al., 2026) capture these downstream effects through an approximation to the Hessian of the full-model output KL divergence with respect to a layer's weights. However, methods based on the Weight Hessian fail to capture the error introduced by quantizing the activations (Figure 1, second column).

In this work, we use a quadratic expansion of the KL divergence between the original and quantized model predictions around the layer output with a tractable approximation (Figure 1, third and last column). This lets us derive a unified algorithm that optimizes transformations, range setting, and

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.

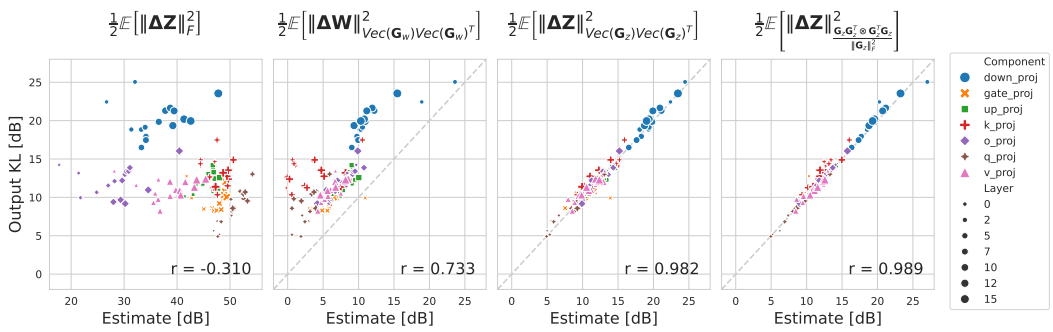

Figure 1: Relationship between the KL divergence between the original and quantized model outputs and several layer-local error proxies across linear layers of Llama 3.2 1B Instruct under W4A4 quantization. Conventional reconstruction metrics, such as the output Frobenius norm, exhibit weak correlation with downstream output degradation. Fisher-weighted weight error fails to capture the effect of activation quantization. Output Fisher-weighted error correlates more strongly with the KL divergence, and our proposed Kronecker factorization of the Output Fisher retains this correlation.

weight rounding under the same curvature metric, accounting for the downstream effects of both weight *and* activation quantization.

Our contributions are:

- We introduce a layer Output Kronecker-factorized approximation to the layer-output Fisher matrix that defines a quadratic proxy for the KL divergence between original and quantized outputs, accounting for both weight *and* activation quantization.
- Using the output Kronecker factorization, we derive $\boldsymbol{H_Z}$-PTQ, a unified procedure for transformation design, weight range selection, adaptive rounding, and activation clipping under a common downstream-sensitivity objective.
- We demonstrate improvements over existing PTQ baselines across multiple model families and quantization granularities.

2 Background

We first introduce the quantization choices considered in this work. Let $\boldsymbol{X} \in \mathbb{R}^{s\times i}$ represent the activation matrix for s tokens and i input features, and let $\boldsymbol{W} \in \mathbb{R}^{i\times o}$ be the weights of a linear layer with o output features. Its output is $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{W}$.

Quantization scales and clipping. For signed integer quantization with $b \geq 2$ bits, $k_{\min} = -2^{b-1}$ and $k_{\max} = 2^{b-1} - 1$. Given a grid step $s > 0$ and shift μ, round-to-nearest quantization is

$$Q(a; s, \mu) = \mu + s \operatorname{clip}\left(\operatorname{round}\left(\frac{a-\mu}{s}\right), k_{\min}, k_{\max}\right). \tag{1}$$

The grid step s determines the tradeoff between clipping and rounding resolution. Increasing s expands the representable range and reduces clipping, but produces a coarser quantization grid. Decreasing s improves resolution within the representable range, but may clip large-magnitude values.

Unlike weights, whose grid steps $\boldsymbol{s}_w$ are fixed after calibration, activation scales depend on the current input. We therefore optimize a shared clipping factor $\gamma_x \in (0, 1]$ (Sun et al., 2025) and compute the corresponding input-dependent activation steps $\boldsymbol{s}_x(\boldsymbol{X}, \gamma_x)$ at runtime. The activations $\boldsymbol{x}_i$ are clipped to $[-\gamma_x \max_i |\boldsymbol{x}_i|, \gamma_x \max_i |\boldsymbol{x}_i|]$ under symmetric quantization and to $[\gamma_x(\min_i \boldsymbol{x}_i - \mu), \gamma_x(\max \boldsymbol{x}_i - \mu)]$ under asymmetric quantization. We then apply Eq. 1 using a grid fitted to these bounds, with a zero-centered grid in the symmetric case and a shifted grid in the asymmetric case.

Weight rounding. Once the representation and quantization grid are fixed, rounding assigns each weight to a representable value. Round-to-nearest treats weights independently, whereas adaptive

rounding methods coordinate these decisions so that later choices can compensate for errors introduced earlier (Nagel et al., 2020; Frantar et al., 2023). Existing methods generally optimize either a layer-reconstruction objective or a weight-space approximation to the downstream model loss.

Transformations. For any invertible transformation $\boldsymbol{T} \in \mathbb{R}^{i\times i}$, transforming the activations and weights preserves the full-precision output, $(\boldsymbol{X}\boldsymbol{T})(\boldsymbol{T}^{-1}\boldsymbol{W}) = \boldsymbol{Z}$, but generally changes the quantized output, $\widetilde{\boldsymbol{Z}} = Q(\boldsymbol{X}\boldsymbol{T})Q(\boldsymbol{T}^{-1}\boldsymbol{W})$. We write $\Delta\boldsymbol{Z} = \boldsymbol{Z} - \widetilde{\boldsymbol{Z}}$ for the resulting layer-output perturbation. If $\boldsymbol{T}$ reduces a weight group's range under the same quantizer, $Q(\boldsymbol{T}^{-1}\boldsymbol{W})$ can use a finer grid and have lower weight error than $Q(\boldsymbol{W})$ in their respective representations (Sun et al., 2025). Diagonal transforms scale channels (Nagel et al., 2019; Xiao et al., 2023; Lin et al., 2023). Orthogonal transforms mix channels through randomized Hadamard transforms (Tseng et al., 2024; Ashkboos et al., 2024a), learned rotations (Liu et al., 2024), or blockwise rotations with permutations (Lin et al., 2024). More general transforms combine rotations and scaling (Hu et al., 2025). FlatQuant jointly learns invertible Kronecker factors, channel scaling, and clipping through block reconstruction (Sun et al., 2025). CAT (Federici et al., 2026) and WUSH (Chen et al., 2026a) design transforms to minimize a local error proxy.

3 METHOD

Our derivation proceeds in four steps. We first express PTQ as the minimization of the prediction divergence induced by replacing one full-precision layer with its quantized counterpart. We then approximate this divergence by a quadratic form in the layer-output perturbation. Because the resulting output Hessian is intractably large, we introduce a tractable approximation. Finally, we use this common quadratic proxy to derive practical optimization procedures for transformations, weight scales and rounding, and activation clipping.

3.1 JOINT OBJECTIVE

Ideally, post-training quantization would jointly select the transforms, weight and activation grids, and weight rounding decisions to minimize the downstream discrepancy between the full-precision and quantized models. For a linear layer, this idealized objective can be written as:

$$\min_{\gamma_x, \boldsymbol{s}_w, \boldsymbol{T}, \boldsymbol{W}^*} \mathbb{E}\left[\mathcal{L}\left(\underbrace{\overbrace{Q(\boldsymbol{X}\boldsymbol{T}; \boldsymbol{s}_x(\boldsymbol{X}\boldsymbol{T}, \gamma_x))}^{\text{quantized activations } \widetilde{\boldsymbol{X}}}\overbrace{Q(\boldsymbol{T}^{-1}\boldsymbol{W}^*; \boldsymbol{s}_w)}^{\text{quantized weights } \widetilde{\boldsymbol{W}}}}_{\text{quantized output } \widetilde{\boldsymbol{Z}}}\right)\right], \tag{2}$$

in which $\mathcal{L}\,()$ refers to a measure of change of the model output.

To model the downstream effect of weight and activation quantization, we use a second-order Taylor expansion around the full-precision layer output $\boldsymbol{Z}$.

$$\mathcal{L}\left(\widetilde{\boldsymbol{Z}}\right) \approx \mathcal{L}\left(\boldsymbol{Z}\right) + \text{vec}\left(\Delta\boldsymbol{Z}\right)^T \text{vec}\left(\boldsymbol{G}_{\boldsymbol{Z}}\right) + \frac{1}{2}\text{vec}\left(\Delta\boldsymbol{Z}\right)^T \boldsymbol{H}_{\boldsymbol{Z}} \text{vec}\left(\Delta\boldsymbol{Z}\right), \tag{3}$$

in which vec () refers to a vectorization operation that stacks the rows of a matrix into a single column vector, $\boldsymbol{G}_{\boldsymbol{Z}}$ is the loss gradient with respect to $\boldsymbol{Z}$, and $\boldsymbol{H}_{\boldsymbol{Z}}$ is the (flattened) Hessian matrix with shape [$s\times$o, s$\times$o].

Common choices of loss $\mathcal{L}\,()$ include measures of divergence of the model output distribution with respect to the original (unperturbed model). For these choices of loss, the zero and first order terms vanish since $\boldsymbol{Z} = \widetilde{\boldsymbol{Z}}$ is a global minimum for which the divergence is zero:

$$D\left(p_{\boldsymbol{Z}} || p_{\widetilde{\boldsymbol{Z}}}\right) \approx \frac{1}{2}\text{vec}\left(\Delta\boldsymbol{Z}\right)^T \boldsymbol{H}_{\boldsymbol{Z}} \text{vec}\left(\Delta\boldsymbol{Z}\right) \overset{def}{=} \frac{1}{2}\|\Delta\boldsymbol{Z}\|^2_{\boldsymbol{H}_{\boldsymbol{Z}}}, \tag{4}$$

where $p_{\boldsymbol{Z}}$ and $p_{\widetilde{\boldsymbol{Z}}}$ refer to the model output distribution induced by the original and quantized layer, respectively. We use the shorter expression $\|\Delta\boldsymbol{Z}\|^2_{\boldsymbol{H}_{\boldsymbol{Z}}}$ to indicate that the norm of the layer output perturbation is weighted by a quadratic proxy of the output loss sensitivity $\boldsymbol{H}_{\boldsymbol{Z}}$.

3.2 MODELING THE HESSIAN MATRICES

Empirically estimating a full rank Hessian matrix is intractable for modern-sized deep learning architectures due to the computation and memory requirements. However, when selecting the Kullback-Leibler divergence for Equation 4, the Hessian becomes equivalent to the Fisher Information Matrix (FIM). The (empirical) Fisher matrix can be estimated by considering the outer product of the flattened Output gradients with respect to sampled labels from the output distribution:

$$\boldsymbol{H}_{\boldsymbol{Z}} = \boldsymbol{F}_{\boldsymbol{Z}} \overset{def}{=} \mathbb{E}\left[\text{vec}\left(\boldsymbol{G}_{\boldsymbol{Z}}\right)\text{vec}\left(\boldsymbol{G}_{\boldsymbol{Z}}\right)^T\right]. \tag{5}$$

Estimating a full Fisher matrix, which is quadratic in the product of sequence length and output feature dimension, is clearly still not practical for full-scale deep neural networks. For this reason, following a similar procedure as (Tseng et al., 2026), we simplify the computation by factorizing the effect of the FIM on the sequence dimension $\boldsymbol{F}_{s(\boldsymbol{Z})}$ and feature output dimension $\boldsymbol{F}_{f(\boldsymbol{Z})}$:

$$\boldsymbol{F}_{\boldsymbol{Z}} \approx \boldsymbol{F}_{\otimes \boldsymbol{Z}} \overset{def}{=} \frac{\overbrace{\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T}^{\boldsymbol{F}_{s(\boldsymbol{Z})}} \otimes \overbrace{\boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}}^{\boldsymbol{F}_{f(\boldsymbol{Z})}}}{\|\boldsymbol{G}_{\boldsymbol{Z}}\|_F^2}, \tag{6}$$

in which $\boldsymbol{F}_{s(\boldsymbol{Z})}$ and $\boldsymbol{F}_{f(\boldsymbol{Z})}$ represent proxies for the sequence and feature curvatures, respectively.

We refer to this approximation as the **Output Kronecker** Fisher factorization, which exactly reproduces each sampled gradient outer product when $\boldsymbol{G}_{\boldsymbol{Z}}$ is rank 1. A summary and comparison with the literature is included in Table 1, while the correlation between the Output Kronecker Fisher weighted error and the KL divergence is visualized in Figure 1.

In order to perform the optimization, we estimate and store the following quantities on a small calibration set: the expected output feature curvature $\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}\right]$, the weight input curvature $\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}}\boldsymbol{G}_{\boldsymbol{W}}^T\right]$ and the token gradient magnitude vector $\mathbb{E}\left[\text{Diag}\left(\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T\right)\right]$. The input feature curvature can be obtained at runtime by left and right multiplying the output feature curvature by the weight matrices $\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}\right] = \boldsymbol{W}\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}\right]\boldsymbol{W}^T$.

Table 1: Comparison of Fisher Information Metric estimates for linear layer output, weights and input implied by different approaches in literature. Normalization factors are omitted for brevity.

Estimate	Output Fisher: $\boldsymbol{F}_{\otimes \boldsymbol{Z}}$ $\overbrace{\boldsymbol{F}_{s(\boldsymbol{Z})} \otimes \boldsymbol{F}_{f(\boldsymbol{Z})}}$	Weight Fisher: $\boldsymbol{F}_{\otimes \boldsymbol{W}}$ $\overbrace{\boldsymbol{F}_{i(\boldsymbol{W})} \otimes \boldsymbol{F}_{o(\boldsymbol{W})}}$	Input Fisher: $\boldsymbol{F}_{\otimes \boldsymbol{X}}$ $\overbrace{\boldsymbol{F}_{s(\boldsymbol{X})} \otimes \boldsymbol{F}_{f(\boldsymbol{X})}}$	Assumptions
Isotropic	$\boldsymbol{I}_s \otimes \boldsymbol{I}_o$	$\boldsymbol{X}^T\boldsymbol{X} \otimes \boldsymbol{I}_o$	$\boldsymbol{I}_s \otimes \boldsymbol{W}\boldsymbol{W}^T$	Isotropic $\boldsymbol{F}_{\boldsymbol{Z}}$
K-FAC	$\boldsymbol{I}_s \otimes \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}$	$\boldsymbol{X}^T\boldsymbol{X} \otimes \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}$	$\boldsymbol{I}_s \otimes \boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}$	Isotropic $\boldsymbol{F}_{s(\boldsymbol{Z})}$
YAQA	$\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T \otimes \boldsymbol{G}_{\boldsymbol{W}}^T\boldsymbol{G}_{\boldsymbol{W}}$	$\boldsymbol{G}_{\boldsymbol{W}}\boldsymbol{G}_{\boldsymbol{W}}^T \otimes \boldsymbol{G}_{\boldsymbol{W}}^T\boldsymbol{G}_{\boldsymbol{W}}$	$\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T \otimes \boldsymbol{W}\boldsymbol{G}_{\boldsymbol{W}}^T\boldsymbol{G}_{\boldsymbol{W}}\boldsymbol{W}^T$	Rank-1 $\boldsymbol{G}_{\boldsymbol{W}}$
Ours	$\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T \otimes \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}$	$\boldsymbol{G}_{\boldsymbol{W}}\boldsymbol{G}_{\boldsymbol{W}}^T \otimes \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}$	$\boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T \otimes \boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}$	Rank-1 $\boldsymbol{G}_{\boldsymbol{Z}}$

3.3 $\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ: OPTIMIZATION PROCEDURE

We start optimizing the output-Fisher-weighted proxy by considering the linear transformation $\boldsymbol{T}$. This is because the transformation $\boldsymbol{T}$ affects model weights, activations, and intermediate gradients, making a joint optimization very challenging:

$$\min_{\gamma_x, \boldsymbol{s}_w, \boldsymbol{T}, \boldsymbol{W}_{\boldsymbol{T}}^*} D_{\text{KL}}\left(p_{\boldsymbol{Z}} || p_{\widetilde{\boldsymbol{Z}}}\right) \approx \frac{1}{2} \min_{\gamma_x, \boldsymbol{s}_w, \boldsymbol{T}, \boldsymbol{W}^*} \mathbb{E}\left[\|\Delta \boldsymbol{Z}\|_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}^2\right] \approx \frac{1}{2} \min_{\gamma_x, \boldsymbol{s}_w, \boldsymbol{W}^*} \mathbb{E}\left[\|\Delta \boldsymbol{Z}_{\hat{\boldsymbol{T}}}\|_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}^2\right], \tag{7}$$

in which $\hat{\boldsymbol{T}}$ represents the optimal transformation $\arg\min_{\boldsymbol{T}} \mathbb{E}\left[\|\Delta \boldsymbol{Z}\|_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}^2\right]$, and $\Delta \boldsymbol{Z}_{\hat{\boldsymbol{T}}}$ is the output discrepancy with the transformation $\hat{\boldsymbol{T}}$ applied. Once $\hat{\boldsymbol{T}}$ is fixed, we denote the transformed activations and weights by $\boldsymbol{X}_{\hat{\boldsymbol{T}}} = \boldsymbol{X}\hat{\boldsymbol{T}}$ and $\boldsymbol{W}_{\hat{\boldsymbol{T}}} = \hat{\boldsymbol{T}}^{-1}\boldsymbol{W}$, respectively. We express the corresponding activation and weight Fisher estimates in these coordinates and use them to determine the remaining quantization parameters.

Once the transformations are fixed, using mild independence assumptions, we can factorize the effect of activation and weight quantization:

$$\mathbb{E}\left[\|\Delta \boldsymbol{Z}_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right] \approx \underbrace{\mathbb{E}\left[\|\boldsymbol{X}_{\hat{\boldsymbol{T}}} \overbrace{(\boldsymbol{W}_{\hat{\boldsymbol{T}}} - \widetilde{\boldsymbol{W}}_{\hat{\boldsymbol{T}}})}^{\Delta \boldsymbol{W}_{\hat{\boldsymbol{T}}}} \|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right]}_{\text{Weight quantization error}} + \underbrace{\mathbb{E}\left[\|\overbrace{(\boldsymbol{X}_{\hat{\boldsymbol{T}}} - \widetilde{\boldsymbol{X}_{\hat{\boldsymbol{T}}}})}^{\Delta \boldsymbol{X}_{\hat{\boldsymbol{T}}}} \boldsymbol{W}_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right]}_{\text{Activation quantization error}}. \tag{8}$$

The approximate split in Eq. 8 permits disentangling weight- and activation-specific parameters. First, we find the optimal weight scale and rounding:

$$\boldsymbol{s}_{\hat{\boldsymbol{W}}}, \hat{\boldsymbol{W}}^*_{\hat{\boldsymbol{T}}} = \arg\min_{\boldsymbol{s}_w, \boldsymbol{W}^*} \mathbb{E}\left[\|\boldsymbol{X}_{\hat{\boldsymbol{T}}} \Delta \boldsymbol{W}_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right]. \tag{9}$$

Lastly, we can determine the optimal clipping based on the updated weights $\hat{\boldsymbol{W}}^*_{\hat{\boldsymbol{T}}}$ by determining:

$$\hat{\gamma_x} = \arg\min_{\gamma_x} \mathbb{E}\left[\|\Delta \boldsymbol{X}_{\hat{\boldsymbol{T}}} \hat{\boldsymbol{W}}^*_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right]. \tag{10}$$

We refer to the resulting framework as $\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ because all optimization stages are derived from the same layer-output curvature proxy $\boldsymbol{F}_{\otimes \boldsymbol{Z}}$. Sections 3.4–3.6 describe each stage.

3.4 Optimal Transformations

In order to find the optimal transformation $\hat{\boldsymbol{T}}$, we first use standard rounding-noise assumptions to explicitly express our quadratic loss proxy as a function of the transformed quantization ranges for activation and weights, which determine the stretching of the quantization grid, and the norm of the transformed gradients:

$$\mathbb{E}\left[\|\Delta \boldsymbol{Z}_{\boldsymbol{T}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right] \approx \frac{\mathbb{E}\left[\sum_t \alpha_t r(\boldsymbol{x}_t \boldsymbol{T})^2\right]}{12(2^{b_x}-1)^2} \mathbb{E}\left[\|\boldsymbol{G}_{\boldsymbol{X}} \boldsymbol{T}^{-T}\|^2_F\right] + \frac{\mathbb{E}\left[\sum_j \beta_j r\left(\boldsymbol{T}^{-1} \boldsymbol{w}_j\right)^2\right]}{12(2^{b_w}-1)^2} \mathbb{E}\left[\|\boldsymbol{T}^T \boldsymbol{G}_{\boldsymbol{W}}\|^2_F\right] \tag{11}$$

The activation and weight bit widths are b_x, b_w, and $r(\boldsymbol{v})$ denotes the range of the quantization grid. The weights $\alpha_t = \|(\boldsymbol{G}_{\boldsymbol{Z}})_{t,:}\|_2^2 / \|\boldsymbol{G}_{\boldsymbol{Z}}\|_F^2$ and $\beta_j = \|(\boldsymbol{G}_{\boldsymbol{Z}})_{:,j}\|_2^2 / \|\boldsymbol{G}_{\boldsymbol{Z}}\|_F^2$ measure the relative gradient energy of token t and output feature j. They remain unchanged by the transforms because $\boldsymbol{X}_{\boldsymbol{T}} \boldsymbol{W}_{\boldsymbol{T}} = \boldsymbol{Z}$. A full derivation is reported in Appendix A.1. Overall, as expected, the expression in Eq. 11 is invariant to scaling of $\boldsymbol{T}$. Furthermore, the norm of the transformed gradients $\|\boldsymbol{G}_{\boldsymbol{X}} \boldsymbol{T}^{-T}\|_F^2$ and $\|\boldsymbol{T}^T \boldsymbol{G}_{\boldsymbol{W}}\|_F^2$ are invariant to orthogonal transformations.

Following Federici et al. (2026), we approximately minimize Eq. 11 with a two-step procedure. First, we choose the symmetric positive-definite component $\boldsymbol{M}$ to minimize the sum of squared transformed gradient norms:

$$\hat{\boldsymbol{M}} = \arg\min_{\boldsymbol{M} \succ 0} \mathbb{E}\left[\|\boldsymbol{G}_{\boldsymbol{X}} \boldsymbol{M}^{-T}\|_F^2\right] + \mathbb{E}\left[\|\boldsymbol{M}^T \boldsymbol{G}_{\boldsymbol{W}}\|_F^2\right] = \left(\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}} \boldsymbol{G}_{\boldsymbol{W}}^T\right]^{-1} \# \, \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{X}}^T \boldsymbol{G}_{\boldsymbol{X}}\right]\right)^{1/2}. \tag{12}$$

Here, $\#$ denotes the matrix geometric mean, see Appendix A.2 for the derivation. In practice, we use a block-wise approximation $\hat{\boldsymbol{M}}_{\text{block}} = \text{BlockDiag}(\hat{\boldsymbol{M}}, \text{block})$ to reduce the memory and computation cost (Federici et al., 2026; Chen et al., 2026a).

Secondly, we apply a Hadamard rotation to reduce the quantization ranges (Federici et al., 2026). The final transformation is obtained by combining the two components $\hat{\boldsymbol{T}} = \hat{\boldsymbol{M}}_{\text{block}} \boldsymbol{H}$.

Note that by making the additional assumption that the output gradients are isotropic ($\boldsymbol{G}_{\boldsymbol{Z}} \propto \boldsymbol{I}$), the expression in Eq. 11 simplifies to the one reported in Federici et al. (2026), and the solution in Eq. 12 is equivalent to the transforms proposed in Federici et al. (2026); Chen et al. (2026b), for which $\hat{\boldsymbol{M}}$ aligns the directions of variations of activations and weights instead of their gradients.

3.5 Weight scales and rounding

The optimal quantization scales and rounding can be determined by considering:

$$\min_{\boldsymbol{s}_w, \boldsymbol{W}^*_{\hat{\boldsymbol{T}}}} \mathbb{E}\left[\|\boldsymbol{X}_{\hat{\boldsymbol{T}}} \Delta \boldsymbol{W}_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right] \approx \min_{\boldsymbol{s}_w, \boldsymbol{W}^*_{\hat{\boldsymbol{T}}}} \left\|\boldsymbol{W}_{\hat{\boldsymbol{T}}} - Q\left(\boldsymbol{W}^*_{\hat{\boldsymbol{T}}}; \boldsymbol{s}_w\right)\right\|^2_{\frac{\hat{\boldsymbol{T}}^T \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}} \boldsymbol{G}_{\boldsymbol{W}}^T\right] \hat{\boldsymbol{T}} \otimes \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T \boldsymbol{G}_{\boldsymbol{Z}}\right]}{\mathbb{E}\left[\|\boldsymbol{G}_{\boldsymbol{Z}}\|_2^2\right]}}. \tag{13}$$

In which $\hat{\boldsymbol{T}}^T \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}} \boldsymbol{G}_{\boldsymbol{W}}^T\right] \hat{\boldsymbol{T}}$ determines the (transformed) weight input-side curvature while $\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T \boldsymbol{G}_{\boldsymbol{Z}}\right]$ the output-side curvature. In order to disentangle the optimization of scales and rounding, we follow a procedure similar to the one introduced in Tseng et al. (2026): first we perform a block-LDL decomposition of the input and output curvature based on the size of the quantization tile $[k_i, k_o]$. Here k_i represents the number of input channels that share the same scale, while k_o represents the number of output channels that share the same scale (usually 1):

$$\boldsymbol{L}_i, \boldsymbol{D}_i = \text{BlockLDL}\left(\hat{\boldsymbol{T}}^T \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}} \boldsymbol{G}_{\boldsymbol{W}}^T\right] \hat{\boldsymbol{T}}; k_i\right), \boldsymbol{L}_o, \boldsymbol{D}_o = \text{BlockLDL}\left(\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T \boldsymbol{G}_{\boldsymbol{Z}}\right]; k_o\right). \quad (14)$$

Then, we use an iterative procedure in which we alternately set a quantization scale and adjust the weight rounding, starting from the original (transformed) weights $\boldsymbol{W}^{(0)} = \boldsymbol{W}_{\hat{\boldsymbol{T}}}$.

The curvature for the quantization tile in coordinate (j, k) is fully determined by the block diagonal components of $\boldsymbol{D}_i[j]$ and $\boldsymbol{D}_o[k]$. Since $\boldsymbol{D}_o[k]$ is a scalar, the optimal scale s_{jk} can be determined by selecting:

$$\hat{s}_{jk} = \arg\min_{s_{jk}} \left\| \boldsymbol{W}_{\hat{\boldsymbol{T}}, B(j,k)} - Q\left(\boldsymbol{W}^{(0)}_{B(j,k)}; s_{jk}\right) \right\|^2_{\boldsymbol{D}_i[j]}, \quad (15)$$

in which $B(j, k)$ is used to indicate the entries of the block (j, k). The expression in Eq. 15 can be easily optimized using a grid search over a few candidates to pick the best scale. Fixing the scale $\hat{s}_{jk}$, we can determine the quantization error given by the quantization function in the block (j, k): $\Delta \boldsymbol{W}^{(0)}_{B(j,k)} = \boldsymbol{W}_{\hat{\boldsymbol{T}}, B(j,k)} - Q\left(\boldsymbol{W}^{(0)}_{B(j,k)}; \hat{s}_{jk}\right)$ and zero everywhere else. Lastly, we can use the $\boldsymbol{L}_i$ and $\boldsymbol{L}_o$ matrices to determine the correction and update the rounded weight for the remaining blocks

$$\boldsymbol{W}^{(1)} = \boldsymbol{W}^{(0)} + \boldsymbol{L}_i^T \Delta \boldsymbol{W}^{(0)} \boldsymbol{L}_o \quad (16)$$

Proceeding in anti-diagonal order starting from the bottom-right, the quantization scales can be set one at a time, and the final $\boldsymbol{W}^{(i/k_i \times o/k_o)}$ represents the adjusted weights $\hat{\boldsymbol{W}}^*_{\hat{\boldsymbol{T}}}$. The key difference from Tseng et al. (2026) lies in using $\boldsymbol{D}_i$ to set the scale of each block and in the different Fisher proxy. Similar to Frantar et al. (2022), before running the algorithm, we sort the block rows and columns so that blocks corresponding to larger Fisher traces are quantized first.

3.6 Activation clipping.

For the activation clipping it is non-trivial to determine a good $\gamma_x \in (0, 1]$: a heuristic constant value may not be optimal, but training it end-to-end is expensive and may yield sub-optimal solutions. We propose to use the Hessian proxy as a good middle ground that takes into account the downstream sensitivity.

$$\mathbb{E}\left[\|\Delta \boldsymbol{X}_{\hat{\boldsymbol{T}}} \boldsymbol{W}_{\hat{\boldsymbol{T}}}\|^2_{\boldsymbol{F} \otimes \boldsymbol{Z}}\right] \approx \mathbb{E}\left[\|\boldsymbol{X}_{\hat{\boldsymbol{T}}} - Q(\boldsymbol{X}_{\hat{\boldsymbol{T}}}; \boldsymbol{s}_x(\boldsymbol{X}_{\hat{\boldsymbol{T}}}, \gamma_x))\|^2_{\mathbb{E}\left[\text{Diag}\left(\boldsymbol{G}_{\boldsymbol{Z}} \boldsymbol{G}_{\boldsymbol{Z}}^T\right)\right] \otimes \hat{\boldsymbol{T}}^{-1} \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{X}}^T \boldsymbol{G}_{\boldsymbol{X}}\right] \hat{\boldsymbol{T}}^{-T}}\right], \quad (17)$$

The optimal γ_x can then be determined on a small calibration set by picking the minimum error over a small set of candidates. Note that instead of using the dense sequence curvature $\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}} \boldsymbol{G}_{\boldsymbol{Z}}^T\right]$, we consider only a diagonal component $\mathbb{E}\left[\text{Diag}\left(\boldsymbol{G}_{\boldsymbol{Z}} \boldsymbol{G}_{\boldsymbol{Z}}^T\right)\right]$. This is to drastically reduce the memory requirements since only one vector needs to be stored. This approximation results in little error since we empirically validated that the sequence curvature component is mostly diagonal.

4 Experimental Results

We evaluate our downstream-output proxy, compare quantization performance across models and quantization granularities, and ablate the contributions of individual components. Appendix B experimental details and Appendix C includes additional results.

Quantization and Calibration We consider two quantization granularities and evaluate three configurations: (i) **W4A4KV4**, a channel-wise setting in which weights, activations, and the KV cache are quantized to 4-bit integers (Sun et al., 2025; Ashkboos et al., 2024b; Federici et al., 2026); weights use one scale per output channel, while activations use one scale per token; (ii) **W4-g32**, a

Table 2: W4A4KV4 channel-wise quantization results. The effectiveness of $\boldsymbol{H_Z}$-PTQ demonstrates the advantage of the joint optimization under a unified Hessian-guided objective.

Method	*Llama v3.2 1B inst.* wiki (↓)	csr (↑)	*Llama v3 8B* wiki (↓)	csr (↑)	*Ministral 8B inst.* wiki (↓)	csr (↑)	*Qwen v3 8B* wiki (↓)	csr (↑)
FP	13.16	57.61	6.14	70.12	6.97	71.29	9.73	67.46
RTN	878.89	30.02	317.60	32.40	39.73	40.62	10441.69	29.23
QuaRot^{+GPTQ}	$20.03^{\pm 0.33}$	$48.93^{\pm 0.56}$	$8.34^{\pm 0.04}$	$63.39^{\pm 0.15}$	$8.18^{\pm 0.00}$	$67.77^{\pm 0.09}$	$34.43^{\pm 9.39}$	$51.64^{\pm 3.05}$
CAT^{+GPTQ}	$17.68^{\pm 0.09}$	$52.19^{\pm 0.46}$	$7.53^{\pm 0.02}$	$65.38^{\pm 0.13}$	$7.90^{\pm 0.02}$	$68.06^{\pm 0.21}$	$10.67^{\pm 0.02}$	$64.60^{\pm 0.34}$
FlatQuant^{+GPTQ}	$17.23^{\pm 0.11}$	$52.95^{\pm 0.40}$	$7.19^{\pm 0.01}$	$\mathbf{67.32^{\pm 0.55}}$	$7.82^{\pm 0.01}$	$68.88^{\pm 0.33}$	$10.60^{\pm 0.03}$	$65.26^{\pm 0.11}$
$\boldsymbol{H_Z}$-PTQ	$\mathbf{16.05^{\pm 0.02}}$	$\mathbf{54.21^{\pm 0.11}}$	$\mathbf{7.06^{\pm 0.01}}$	$\mathbf{67.25^{\pm 0.23}}$	$\mathbf{7.69^{\pm 0.01}}$	$\mathbf{69.32^{\pm 0.37}}$	$\mathbf{10.43^{\pm 0.03}}$	$\mathbf{65.56^{\pm 0.46}}$

Table 3: For block-wise quantization, $\boldsymbol{H_Z}$-PTQ consistently achieves the lowest perplexity across both W4-g32 and W4A4-g32 settings, demonstrating that the proposed Hessian-guided objective remains effective even at finer quantization granularities.

	Method	*Llama v3.2 1B inst.* wiki (↓)	csr (↑)	*Llama v3 8B* wiki (↓)	csr (↑)	*Ministral 8B inst.* wiki (↓)	csr (↑)	*Qwen v3 8B* wiki (↓)	csr (↑)
	FP	13.16	57.61	6.14	70.12	6.97	71.29	9.73	67.46
W4-g32	RTN	15.47	55.27	6.65	68.67	7.28	70.88	9.99	65.95
W4-g32	Had.$^{+\text{GPTQ}}$	$13.73^{\pm 0.02}$	$56.83^{\pm 0.18}$	$6.43^{\pm 0.01}$	$68.97^{\pm 0.19}$	$7.13^{\pm 0.01}$	$71.01^{\pm 0.12}$	$9.99^{\pm 0.05}$	$66.80^{\pm 0.15}$
W4-g32	CAT$^{+\text{GPTQ}}$	$13.77^{\pm 0.08}$	$56.77^{\pm 0.17}$	$6.44^{\pm 0.01}$	$69.64^{\pm 0.18}$	$7.14^{\pm 0.00}$	$\mathbf{70.99^{\pm 0.03}}$	$9.92^{\pm 0.04}$	$67.14^{\pm 0.28}$
W4-g32	WUSH$^{+\text{GPTQ}}$	$13.85^{\pm 0.08}$	$56.81^{\pm 0.14}$	$6.44^{\pm 0.01}$	$69.43^{\pm 0.33}$	$7.13^{\pm 0.01}$	$70.83^{\pm 0.05}$	$9.88^{\pm 0.08}$	$67.06^{\pm 0.32}$
W4-g32	Had.$^{+\text{YAQA}}$	$13.74^{\pm 0.02}$	$\mathbf{57.02^{\pm 0.18}}$	$6.39^{\pm 0.00}$	$69.61^{\pm 0.03}$	$7.22^{\pm 0.00}$	$\mathbf{70.96^{\pm 0.12}}$	$9.93^{\pm 0.06}$	$67.29^{\pm 0.22}$
W4-g32	$\boldsymbol{H_Z}$-PTQ	$\mathbf{13.53^{\pm 0.04}}$	$\mathbf{57.04^{\pm 0.19}}$	$\mathbf{6.32^{\pm 0.00}}$	$\mathbf{69.93^{\pm 0.17}}$	$\mathbf{7.09^{\pm 0.01}}$	$\mathbf{70.97^{\pm 0.04}}$	$\mathbf{9.84^{\pm 0.01}}$	$\mathbf{67.42^{\pm 0.25}}$
W4A4-g32	RTN	17.99	51.75	7.25	66.68	7.72	69.21	10.39	63.89
W4A4-g32	Had.$^{+\text{GPTQ}}$	$14.57^{\pm 0.02}$	$55.59^{\pm 0.29}$	$6.72^{\pm 0.01}$	$67.66^{\pm 0.37}$	$7.35^{\pm 0.02}$	$70.19^{\pm 0.23}$	$10.19^{\pm 0.06}$	$66.20^{\pm 0.27}$
W4A4-g32	CAT$^{+\text{GPTQ}}$	$14.54^{\pm 0.09}$	$55.64^{\pm 0.45}$	$6.71^{\pm 0.01}$	$68.66^{\pm 0.20}$	$7.40^{\pm 0.01}$	$70.02^{\pm 0.14}$	$10.11^{\pm 0.04}$	$66.38^{\pm 0.11}$
W4A4-g32	WUSH$^{+\text{GPTQ}}$	$14.65^{\pm 0.04}$	$55.28^{\pm 0.45}$	$6.74^{\pm 0.02}$	$67.98^{\pm 0.54}$	$7.42^{\pm 0.01}$	$69.83^{\pm 0.21}$	$10.11^{\pm 0.09}$	$66.30^{\pm 0.31}$
W4A4-g32	Had.$^{+\text{YAQA}}$	$14.56^{\pm 0.02}$	$55.53^{\pm 0.17}$	$6.68^{\pm 0.01}$	$68.51^{\pm 0.15}$	$7.46^{\pm 0.01}$	$70.10^{\pm 0.11}$	$10.13^{\pm 0.07}$	$66.22^{\pm 0.33}$
W4A4-g32	$\boldsymbol{H_Z}$-PTQ	$\mathbf{14.22^{\pm 0.05}}$	$\mathbf{55.75^{\pm 0.16}}$	$\mathbf{6.56^{\pm 0.00}}$	$\mathbf{68.82^{\pm 0.15}}$	$\mathbf{7.29^{\pm 0.00}}$	$\mathbf{70.38^{\pm 0.13}}$	$\mathbf{9.99^{\pm 0.01}}$	$\mathbf{66.77^{\pm 0.38}}$

block-wise weight-only setting in which groups of 32 weight values share a floating-point scale, similar to the one used in Tseng et al. (2026); and (iii) **W4A4-g32**, a block-wise weight-and-activation setting in which groups of 32 values share a floating-point scale (Chen et al., 2026b). For all methods, quantization parameters and Hessian estimates are computed using 2048 randomly sampled sequences of 2048 tokens from the DCLM-Edu dataset (Allal et al., 2025).

Evaluation Protocol Following standard evaluation practice for LLM quantization (Sun et al., 2025; Federici et al., 2026), we report perplexity on WikiText (Merity et al., 2017) using a context length of 2048 tokens and the mean normalized accuracy across six common-sense reasoning tasks (csr): PIQA (Bisk et al., 2020), WinoGrande (Sakaguchi et al., 2021), HellaSwag (Zellers et al., 2019), ARC-e and ARC-c (Clark et al., 2018), and LAMBADA (Paperno et al., 2016). For methods involving calibration-dependent optimization, we report the mean and standard deviation over three random calibration seeds.

4.1 Channel-wise W4A4KV4 Quantization

We first evaluate the W4A4KV4 setting, where only a single scale is available for each weight output channel and activation token. Results are reported in Table 2. $\boldsymbol{H_Z}$-PTQ consistently achieves the lowest perplexity across all evaluated architectures while matching or improving reasoning accuracy. Importantly, the gains are observed for every model family considered, ranging from the 1B-parameter Llama model to larger 8B-parameter architectures. This consistency indicates that the benefits of the proposed objective are not tied to a specific model architecture or size. Compared to FlatQuant, which learns quantization-aware transformations through layer-wise optimization, $\boldsymbol{H_Z}$-PTQ achieves lower perplexity on all four models. Likewise, $\boldsymbol{H_Z}$-PTQ consistently outperforms CAT, despite both methods relying on the same transformation with different parameters.

Table 4: Ablation for the components for $\boldsymbol{H_Z}$-PTQ on W4A4KV4. −Round refers to a version with round-to-nearest and a standard L_p weight range-setting. −Trans. refers to the use of Hadamard transform instead of our gradient alignment transforms. −Hess. indicates the use of a GPTQ-style hessian approximation instead of our output Kronecker approximation.

Method	*Llama v3.2 1B inst.* wiki (↓)	csr (↑)	*Llama v3 8B* wiki (↓)	csr (↑)	*Ministral 8B inst.* wiki (↓)	csr (↑)	*Qwen v3 8B* wiki (↓)	csr (↑)
$\boldsymbol{H_Z}$-PTQ	$16.05^{\pm 0.02}$	$54.11^{\pm 0.19}$	$7.06^{\pm 0.01}$	$67.25^{\pm 0.23}$	$7.69^{\pm 0.01}$	$69.30^{\pm 0.26}$	$10.43^{\pm 0.03}$	$65.56^{\pm 0.46}$
−Round	$16.52^{\pm 0.07}$	$53.13^{\pm 0.57}$	$7.13^{\pm 0.00}$	$66.54^{\pm 0.04}$	$7.74^{\pm 0.02}$	$69.10^{\pm 0.15}$	$10.56^{\pm 0.04}$	$65.14^{\pm 0.42}$
−A.clip	$17.05^{\pm 0.03}$	$52.35^{\pm 0.36}$	$7.41^{\pm 0.01}$	$65.72^{\pm 0.14}$	$7.89^{\pm 0.01}$	$68.35^{\pm 0.62}$	$10.63^{\pm 0.05}$	$64.08^{\pm 0.05}$
−Tran.	$18.64^{\pm 0.13}$	$51.68^{\pm 0.86}$	$7.61^{\pm 0.01}$	$66.06^{\pm 0.39}$	$8.13^{\pm 0.01}$	$67.98^{\pm 0.30}$	$25.19^{\pm 0.90}$	$55.80^{\pm 0.48}$
−Hess.	$16.83^{\pm 0.10}$	$52.80^{\pm 0.64}$	$7.37^{\pm 0.02}$	$65.75^{\pm 0.11}$	$7.83^{\pm 0.03}$	$68.90^{\pm 0.34}$	$10.42^{\pm 0.04}$	$65.77^{\pm 0.03}$

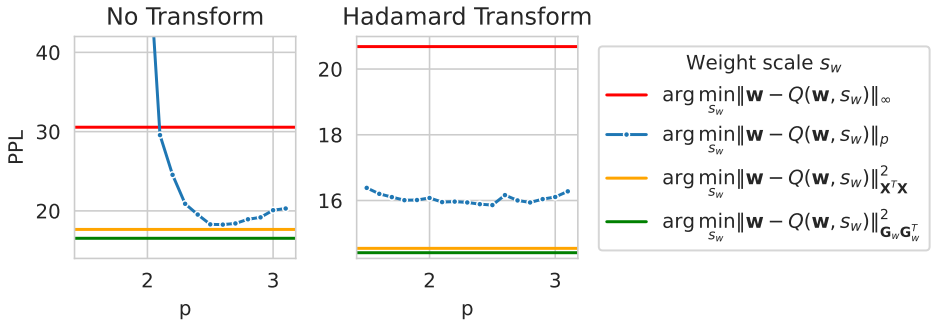

Figure 2: WikiText perplexity obtained by several weight scale-setting objectives on Llama 3.2 1B inst. under W4 per-channel quantization, with and without Hadamard transformations. Weighting the quantization error using the proposed Fisher estimate consistently exceeds the best reconstruction objective without requiring hyperparameter tuning.

4.2 Block-wise Quantization

We next consider block-wise quantization using groups of size 32. This setting is particularly challenging for transformation-based methods since smaller quantization groups already reduce range variability, leaving less room for improvement through range equalization alone. Following Chen et al. (2026b), we apply independent block-wise transforms before quantization and fuse the corresponding inverses into the weights. We evaluate both weight-only (W4-g32) and weight-and-activation (W4A4-g32) quantization scenarios. Results are reported in Table 3.

Despite the reduced optimization headroom available in the block-wise setting, $\boldsymbol{H_Z}$-PTQ consistently achieves the lowest perplexity and highest accuracy across all evaluated architectures and quantization formats. The improvement persists in both W4 and W4A4 settings, indicating that the proposed objective benefits not only transformation design, but also downstream range selection and rounding decisions. Notably, the relative gains remain stable across model families, further supporting the robustness of the proposed framework.

4.3 Ablation Studies

Contribution of Individual Components Table 4 evaluates the contribution of each component within $\boldsymbol{H_Z}$-PTQ in the W4A4KV4 setting. Removing any component results in a measurable degradation in performance, demonstrating that the final gains arise from the interaction of all optimization stages rather than from a single dominant factor. Among the examined components, the transformation stage provides the largest individual contribution, followed by activation clipping and Hessian modeling. Replacing the proposed output Kronecker Hessian approximation with a simpler gradient-free Hessian estimate also degrades performance, providing evidence that the improved results are not solely due to the optimization procedure but also depend on the quality of the underlying curvature model.

Table 5: Comparison of rounding algorithms for different hessian estimates on the *Llama v3.2 1B instruct* architecture in two quantization settings. The bottom rows report the effect of applying a Hadamard transform prior to quantization. $\mathbf{G}_W\mathbf{G}_W^T \otimes \mathbf{G}_Z^T\mathbf{G}_Z$ refers to our approximation.

Tran.	Hessian Estimate	W4 Per-channel GPTQ	W4 Per-channel YAQA	W4 Per-channel **ours**	W4 Per-block (32) GPTQ	W4 Per-block (32) YAQA	W4 Per-block (32) **ours**
Id.	$\mathbf{X}^T\mathbf{X} \otimes \mathbf{I}$	$30.96^{\pm 4.58}$	$30.56^{\pm 0.00}$	$17.65^{\pm 0.09}$	$14.35^{\pm 0.04}$	$69.59^{\pm 10.14}$	$14.11^{\pm 0.07}$
	$\mathbf{G}_W\mathbf{G}_W^T \otimes \mathbf{G}_W^T\mathbf{G}_W$	$18.42^{\pm 0.48}$	$18.81^{\pm 0.10}$	$15.34^{\pm 0.03}$	$14.28^{\pm 0.11}$	$13.75^{\pm 0.05}$	$\mathbf{13.64}^{\pm 0.05}$
	$\mathbf{G}_W\mathbf{G}_W^T \otimes \mathbf{G}_Z^T\mathbf{G}_Z$	$19.66^{\pm 3.89}$	$19.49^{\pm 0.42}$	$\mathbf{15.23}^{\pm 0.06}$	$14.26^{\pm 0.25}$	$13.92^{\pm 0.08}$	$\mathbf{13.64}^{\pm 0.02}$
Had.	$\mathbf{X}^T\mathbf{X} \otimes \mathbf{I}$	$14.96^{\pm 0.08}$	$20.68^{\pm 0.00}$	$14.50^{\pm 0.01}$	$13.68^{\pm 0.03}$	$13.90^{\pm 0.07}$	$13.69^{\pm 0.05}$
	$\mathbf{G}_W\mathbf{G}_W^T \otimes \mathbf{G}_W^T\mathbf{G}_W$	$14.50^{\pm 0.06}$	$19.12^{\pm 0.29}$	$14.26^{\pm 0.05}$	$13.61^{\pm 0.01}$	$13.70^{\pm 0.05}$	$\mathbf{13.56}^{\pm 0.02}$
	$\mathbf{G}_W\mathbf{G}_W^T \otimes \mathbf{G}_Z^T\mathbf{G}_Z$	$14.49^{\pm 0.04}$	$19.29^{\pm 0.24}$	$\mathbf{14.25}^{\pm 0.08}$	$13.60^{\pm 0.02}$	$13.66^{\pm 0.01}$	$\mathbf{13.56}^{\pm 0.01}$

Hessian-Guided Weight Range Selection Existing range-setting methods typically minimize an L_p reconstruction objective, requiring the practitioner to select an appropriate value of p. Figure 2 evaluates the sensitivity of this choice. The optimal value of p varies substantially across quantization settings and transformation choices, making fixed reconstruction objectives difficult to tune. In contrast, the proposed Hessian-weighted objective proposed in eq. (13) directly incorporates downstream sensitivity information and consistently matches or outperforms the best manually selected L_p objective without requiring any hyperparameter tuning. These results suggest that incorporating curvature information into scale selection provides a more reliable objective than purely reconstruction-based metrics.

Hessian estimation and Rounding Strategies Table 5 compares different rounding algorithms under several Hessian approximations. This experiment disentangles the effect of the curvature estimate from that of the rounding procedure itself. First, for a fixed rounding algorithm, our proposed output Kronecker Hessian proxy performs on par with or outperforms the YAQA-style estimate, outperforming the gradient-free GPTQ-style hessian sketches.

Second, for a fixed Hessian estimate, incorporating our Hessian-aware rounding and range-setting procedure consistently improves performance over conventional GPTQ/LDLQ and YAQA optimization. This observation suggests that using hessian information to set the weight ranges while performing the rounding correction can better incorporate curvature information into the local decisions. This is especially relevant for coarse grained quantization granularities. Note that when using GPTQ rounding, the output Kronecker and YAQA-style hessian are equivalent since only the first component of the Kronecker product determines the weight corrections.

5 Discussion and Conclusion

We introduced a unified Hessian-guided framework for post-training quantization. Unlike existing approaches that optimize transformations, range selection, activation clipping, and rounding independently and using separate reconstruction objectives, our framework derives all quantization decisions from a common approximation of downstream prediction loss. Central to this framework is the proposed Output Kronecker approximation, a tractable output-space Fisher factorization that better correlates with downstream KL divergence than conventional local reconstruction metrics. Across diverse architectures and quantization granularities, jointly optimizing these components consistently improved performance over existing state-of-the-art methods. These results suggest that much of the PTQ design space can be viewed as a single downstream optimization problem rather than a collection of independent heuristics.

The method nevertheless remains an approximation to the ideal downstream objective. Its derivation relies on a second-order expansion, a Kronecker-rank-one Fisher approximation, a decomposition of activation and weight errors, and rounding-noise and covariance assumptions. Moreover, estimating output gradients adds calibration time and memory relative to gradient-free reconstruction methods. Future work could investigate richer output-curvature models, adaptive approximations that trade accuracy for calibration cost, and extensions to non-uniform, mixed-precision, and hardware-aware quantization.

AI USE STATEMENT

In this work, we used generative AI tools for [formulating mathematical expressions, implementing methods, assisting in the writing of proofs, proposing or refining hypotheses, provide critical ingredients for proving mathematical claims]. We have not used generative AI tools for: [help develop theoretical models or conceptual frameworks, interpret results], and [Generate synthetic data sets, design or provide feedback on research methodology or experiments, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis] are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model, 2025. URL https://arxiv.org/abs/2502.02737.

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. *arXiv preprint arXiv:2404.00456*, 2024a. URL https://arxiv.org/abs/2404.00456.

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-Free 4-Bit Inference in Rotated LLMs, March 2024b. URL https://arxiv.org/abs/2404.00456v1.

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, April 2020. ISSN 2374-3468. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239. Number: 05.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-Bit Quantization of Large Language Models With Guarantees, July 2023. URL https://arxiv.org/abs/2307.13304v2.

Jiale Chen, Vage Egiazarian, Roberto L. Castro, Torsten Hoefler, and Dan Alistarh. WUSH: Near-optimal adaptive transforms for LLM quantization. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=ZsECxUkbKB.

Jiale Chen, Vage Egiazarian, Roberto L. Castro, Torsten Hoefler, and Dan Alistarh. Wush: Near-optimal adaptive transforms for llm quantization, 2026b. URL https://arxiv.org/abs/2512.00956.

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, March 2018. URL http://arxiv.org/abs/1803.05457. arXiv:1803.05457 [cs].

Marco Federici, Boris van Breugel, Paul N. Whatmough, and Markus Nagel. Dissecting quantization error: A concentration-alignment perspective. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=7BoFyTPqMr.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.17323.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Xing Hu, Yuan Cheng, Dawei Yang, Zukang Xu, Zhihang Yuan, Jiangyong Yu, Chen Xu, Zhe Jiang, and Sifan Zhou. OSTQuant: Refining large language model quantization with orthogonal and scaling transformations for better distribution fitting. *arXiv preprint arXiv:2501.13987*, 2025. URL https://arxiv.org/abs/2501.13987.

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. *arXiv preprint arXiv:2406.01721*, 2024. URL https://arxiv.org/abs/2406.01721.

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 LLM compression and acceleration. *arXiv preprint arXiv:2306.00978*, 2023. URL https://arxiv.org/abs/2306.00978.

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. *arXiv preprint arXiv:2405.16406*, 2024. URL https://arxiv.org/abs/2405.16406.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Mistral.ai. Un ministral, des ministraux, 2025. URL https://mistral.ai/news/ministraux.

Markus Nagel, Mart van Baalen, Tijmen Blankevoort, and Max Welling. Data-free quantization through weight equalization and bias correction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

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*, pp. 7197–7206, 2020. URL https://proceedings.mlr.press/v119/nagel20a.html.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc-Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, 2016.

W. Pusz and S.L. Woronowicz. Functional calculus for sesquilinear forms and the purification map. *Reports on Mathematical Physics*, 8(2):159–170, 1975. ISSN 0034-4877. doi: https://doi.org/10.1016/0034-4877(75)90061-0. URL https://www.sciencedirect.com/science/article/pii/0034487775900610.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL https://dl.acm.org/doi/10.1145/3474381.

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 *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=uTz2Utym5n.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even better LLM quantization with hadamard incoherence and lattice codebooks. *arXiv preprint arXiv:2402.04396*, 2024. URL https://arxiv.org/abs/2402.04396.

Albert Tseng, Zhaofeng Sun, and Christopher De Sa. Model-preserving adaptive rounding, 2026. URL https://openreview.net/forum?id=oGlgHjYKBi.

Boris van Breugel, Yelysei Bondarenko, Paul Whatmough, and Markus Nagel. FPTQuant: Function-preserving transforms for LLM quantization. In *Forty-third International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2506.04985.

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*, pp. 38087–38099, 2023. URL https://proceedings.mlr.press/v202/xiao23c.html.

Pourya Zadeh, Reshad Hosseini, and Suvrit Sra. Geometric mean metric learning. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2464–2471. PMLR, 2016. URL https://proceedings.mlr.press/v48/zadeh16.html.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a Machine Really Finish Your Sentence?, May 2019. URL http://arxiv.org/abs/1905.07830. arXiv:1905.07830 [cs].

A Quantization loss decomposition

A.1 Derivation of the transformation objective

We derive Eq. 11 from the output KL divergence, first separating activation and weight errors to make the derivation of the optimal transform tractable.

From output KL to a quadratic proxy. Using the second-order KL approximation in Eq. 4 and the Output Kronecker approximation in Eq. 6, we obtain

$$\mathbb{E}\left[D_{\mathrm{KL}}\left(p_{\boldsymbol{Z}}||p_{\widetilde{\boldsymbol{Z}}}\right)\right] \approx \frac{1}{2}\mathbb{E}\left[\|\Delta \boldsymbol{Z}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}\right]. \tag{18}$$

For a label $y \sim p_{\boldsymbol{Z}}$, define $\boldsymbol{G}_{\boldsymbol{Z}} = \nabla_{\boldsymbol{Z}}[-\log p_{\boldsymbol{Z}}(y)]$. Here and below, expectations average calibration inputs, sampled labels, and the rounding-noise model defined next.

For the trace calculations, write $\boldsymbol{A} = \boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{G}_{\boldsymbol{Z}}^T$, $\boldsymbol{B} = \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}$, and $c = \|\boldsymbol{G}_{\boldsymbol{Z}}\|_F^2$. For $c > 0$ and any output perturbation $\boldsymbol{U}$, row vectorization gives $\|\boldsymbol{U}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}} = \mathrm{Tr}\left(\boldsymbol{A}\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^T\right)/c$.

Rounding model and error decomposition. For this derivation, activation row t uses grid step $d_{x,t}$ and weight column j uses grid step $d_{w,j}$. We model scalar rounding errors as conditionally independent and uniform on their quantization bins, with negligible clipping. Conditioning on the full-precision activations, weights, sampled gradients, and grid steps, the errors have zero mean and variances $d_{x,t}^2/12$ and $d_{w,j}^2/12$.

The output perturbation satisfies $\Delta \boldsymbol{Z} = \boldsymbol{X}\Delta \boldsymbol{W} + \Delta \boldsymbol{X}\boldsymbol{W} - \Delta \boldsymbol{X}\Delta \boldsymbol{W}$. The product $\Delta \boldsymbol{X}\Delta \boldsymbol{W}$ contributes terms of third and fourth order in the quadratic proxy, so we omit it in the second-order approximation. Conditional independence and zero means eliminate the mixed quadratic term between $\boldsymbol{X}\Delta \boldsymbol{W}$ and $\Delta \boldsymbol{X}\boldsymbol{W}$. This gives the activation–weight decomposition in Eq. 8, following Federici et al. (2026).

Activation contribution. The chain rule gives $\boldsymbol{G}_{\boldsymbol{X}} = \boldsymbol{G}_{\boldsymbol{Z}}\boldsymbol{W}^T$. Using the trace form of the proxy,

$$\|\Delta \boldsymbol{X}\boldsymbol{W}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}} = \frac{\mathrm{Tr}\left(\Delta \boldsymbol{X}^T\boldsymbol{A}\Delta \boldsymbol{X}\boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}\right)}{c}. \tag{19}$$

Only matching scalar error indices survive the conditional expectation. Thus

$$\begin{aligned}
\mathbb{E}[\Delta \boldsymbol{X}^T\boldsymbol{A}\Delta \boldsymbol{X}] &= \frac{1}{12}\left(\sum_t A_{tt}d_{x,t}^2\right)\boldsymbol{I}_i, \\
\mathbb{E}[\|\Delta \boldsymbol{X}\boldsymbol{W}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}] &= \frac{1}{12}\left(\sum_t \alpha_t d_{x,t}^2\right)\|\boldsymbol{G}_{\boldsymbol{X}}\|_F^2,
\end{aligned} \tag{20}$$

using $\alpha_t = A_{tt}/c$.

Write $\boldsymbol{x}_t = \boldsymbol{X}_{t,:}$ and $r(\boldsymbol{x}_t) = (2^{b_x} - 1)d_{x,t}$. For our asymmetric per-token activation grid, $r(\boldsymbol{x}_t) = \max_a X_{ta} - \min_a X_{ta}$.

Weight contribution. For the weights, $\boldsymbol{G}_{\boldsymbol{W}} = \boldsymbol{X}^T\boldsymbol{G}_{\boldsymbol{Z}}$. The same calculation contracts $\Delta \boldsymbol{W}\boldsymbol{B}\Delta \boldsymbol{W}^T$, whose conditional expectation is $\frac{1}{12}(\sum_j B_{jj}d_{w,j}^2)\boldsymbol{I}_i$. Thus

$$\mathbb{E}[\|\boldsymbol{X}\Delta \boldsymbol{W}\|^2_{\boldsymbol{F}_{\otimes \boldsymbol{Z}}}] = \frac{1}{12}\left(\sum_j \beta_j d_{w,j}^2\right)\|\boldsymbol{G}_{\boldsymbol{W}}\|_F^2, \tag{21}$$

using $\beta_j = B_{jj}/c$. For column $\boldsymbol{w}_j = \boldsymbol{W}_{:,j}$, define $r(\boldsymbol{w}_j) = (2^{b_w} - 1)d_{w,j}$ using the step of the configured symmetric weight grid. This range is the grid span, with the signed endpoints defined in Section 2.

Separating ranges and gradient norms. Let $R_x = \sum_t \alpha_t r(\boldsymbol{x}_t)^2$ and $R_w = \sum_j \beta_j r(\boldsymbol{w}_j)^2$. Averaging the conditional results gives $\mathbb{E}\left[R_x \|\boldsymbol{G_X}\|_F^2\right] / [12(2^{b_x}-1)^2]$ and $\mathbb{E}\left[R_w \|\boldsymbol{G_W}\|_F^2\right] / [12(2^{b_w}-1)^2]$ for the activation and weight contributions, respectively. To split the range and gradient terms we assume $\mathrm{Cov}(R_x, \|\boldsymbol{G_X}\|_F^2) \approx 0$ and $\mathrm{Cov}(R_w, \|\boldsymbol{G_W}\|_F^2) \approx 0$ over calibration inputs and sampled labels. Applying $\mathbb{E}[UV] = \mathbb{E}[U]\,\mathbb{E}[V] + \mathrm{Cov}(U,V)$ to the two contributions yields

$$\mathbb{E}\left[\|\Delta \boldsymbol{X}\boldsymbol{W}\|_{\boldsymbol{F}\otimes \boldsymbol{z}}^2\right] \approx \frac{\mathbb{E}\left[\sum_t \alpha_t r(\boldsymbol{x}_t)^2\right]}{12(2^{b_x}-1)^2} \mathbb{E}\left[\|\boldsymbol{G_X}\|_F^2\right], \tag{22}$$

$$\mathbb{E}\left[\|\boldsymbol{X}\Delta \boldsymbol{W}\|_{\boldsymbol{F}\otimes \boldsymbol{z}}^2\right] \approx \frac{\mathbb{E}\left[\sum_j \beta_j r(\boldsymbol{w}_j)^2\right]}{12(2^{b_w}-1)^2} \mathbb{E}\left[\|\boldsymbol{G_W}\|_F^2\right]. \tag{23}$$

Combining these with Eq. 8 gives Eq. 11.

A.2 Geometric-mean solution for gradient balancing

We derive the solution of the gradient-norm objective in Eq. 12. Define the gradient second-moment matrices

$$\boldsymbol{A} = \mathbb{E}\left[\boldsymbol{G_W}\boldsymbol{G_W}^T\right], \qquad \boldsymbol{B} = \mathbb{E}\left[\boldsymbol{G_X}^T\boldsymbol{G_X}\right], \tag{24}$$

and assume $\boldsymbol{A}, \boldsymbol{B} \succ 0$. For any invertible $\boldsymbol{M}$, set $\boldsymbol{P} = \boldsymbol{M}\boldsymbol{M}^T \succ 0$. Using $\|\boldsymbol{U}\|_F^2 = \mathrm{Tr}\left(\boldsymbol{U}^T\boldsymbol{U}\right)$ and cyclicity of the trace, the two terms become

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{G_X}\boldsymbol{M}^{-T}\|_F^2\right] &= \mathrm{Tr}\left(\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{M}^{-T}\right) = \mathrm{Tr}\left(\boldsymbol{B}\boldsymbol{P}^{-1}\right), \\ \mathbb{E}\left[\|\boldsymbol{M}^T\boldsymbol{G_W}\|_F^2\right] &= \mathrm{Tr}\left(\boldsymbol{M}^T\boldsymbol{A}\boldsymbol{M}\right) = \mathrm{Tr}\left(\boldsymbol{A}\boldsymbol{P}\right). \end{aligned} \tag{25}$$

The objective therefore depends on $\boldsymbol{M}$ only through $\boldsymbol{P}$. Every positive-definite $\boldsymbol{P}$ has a unique symmetric positive-definite square root $\boldsymbol{M} = \boldsymbol{P}^{1/2}$, so Eq. 12 is equivalent to

$$\min_{\boldsymbol{P}\succ 0} \ \mathrm{Tr}\left(\boldsymbol{A}\boldsymbol{P}\right) + \mathrm{Tr}\left(\boldsymbol{B}\boldsymbol{P}^{-1}\right). \tag{26}$$

This is exactly the trace minimization in Zadeh et al. (2016, Section 2.2, Eq. (8)). Their solution following Eq. (12) gives the unique minimizer

$$\boldsymbol{P}^* = \boldsymbol{A}^{-1}\#\boldsymbol{B} = \boldsymbol{A}^{-1/2}\left(\boldsymbol{A}^{1/2}\boldsymbol{B}\boldsymbol{A}^{1/2}\right)^{1/2}\boldsymbol{A}^{-1/2}, \tag{27}$$

where $\#$ is the Pusz–Woronowicz geometric mean (Pusz & Woronowicz, 1975). Taking the symmetric positive-definite square root yields

$$\boldsymbol{M}^* = (\boldsymbol{A}^{-1}\#\boldsymbol{B})^{1/2}, \tag{28}$$

which is the solution stated in Eq. 12. Over all invertible matrices, the minimizers are $\boldsymbol{M} = (\boldsymbol{P}^*)^{1/2}\boldsymbol{R}$ with $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}_i$. Thus the gradient-norm objective fixes the positive-definite component while leaving the rotation free for the subsequent range-reduction step.

A.3 Range clipping for weights derivation

Our algorithm applies the selected transformation before setting quantization ranges and rounding the weights. These steps require gradients and Fisher estimates in the transformed coordinates. The chain rule lets us compute these quantities from the original gradient estimates, avoiding an additional backward pass through the model. For the transformed coordinates defined in Section 2, the chain rule gives

$$\boldsymbol{G_{X_T}} = \boldsymbol{G_Z}\boldsymbol{W_T}^T = \boldsymbol{G_X}\boldsymbol{T}^{-T}, \qquad \boldsymbol{G_{W_T}} = \boldsymbol{X_T}^T\boldsymbol{G_Z} = \boldsymbol{T}^T\boldsymbol{G_W}. \tag{29}$$

This shows that, instead of estimating the Fisher approximation directly from $\boldsymbol{G_{W_T}}$, we can transform the existing approximation computed from the untransformed gradient $\boldsymbol{G_W}$, using the identity $\boldsymbol{G_{W_T}} = \boldsymbol{T}^T\boldsymbol{G_W}$. This relation is used in Eq. 13 and in the weight contribution to Eq. 11.

Table 6: Details on the setting used to produce Table 2. The block size for the transforms used for CAT and our method is 128.

Method	W. Scale	W. Round	Act. Clip	Transforms
RTN	$2\max_i \lvert\boldsymbol{w}_i\rvert$	None	$\gamma = 1$	None
QuaRot^{+GPTQ}	$\min_{s_w} \lVert\boldsymbol{w} - Q(\boldsymbol{w}; s_w)\rVert^{2.4}$	GPTQ	$\gamma = 1$	(Fused) $\boldsymbol{H}$
CAT^{+GPTQ}	$\min_{s_w} \lVert\boldsymbol{w} - Q(\boldsymbol{w}; s_w)\rVert^{2.4}$	GPTQ	$\gamma = 1$	$B(\boldsymbol{\Sigma}_{\boldsymbol{X}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{W}})^{1/2}\boldsymbol{H}$
FlatQuant^{+GPTQ}	Learned s_w	GPTQ	Learned γ	Learned $\boldsymbol{T}_i \otimes \boldsymbol{T}_o$
$\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ	Algorithm in Sec. 3.5		Sec. 3.6	$B(\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{W}}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{X}}})^{1/2}\boldsymbol{H}$

For fixed $\boldsymbol{T}$ and $\Delta\boldsymbol{W}_{\boldsymbol{T}}$, write $\boldsymbol{P} = \boldsymbol{G}_{\boldsymbol{W}_{\boldsymbol{T}}}\boldsymbol{G}_{\boldsymbol{W}_{\boldsymbol{T}}}^T$ and $\boldsymbol{Q} = \boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}/\lVert\boldsymbol{G}_{\boldsymbol{Z}}\rVert_F^2$. We assume that $\lVert\boldsymbol{G}_{\boldsymbol{Z}}\rVert_F^2$ is constant, and that the correlations between the input- and output-side curvature factors contribute negligibly to the expected weight-error proxy:

$$\mathbb{E}\left[\mathrm{Tr}\left((\boldsymbol{P} - \mathbb{E}[\boldsymbol{P}])\Delta\boldsymbol{W}_{\boldsymbol{T}}(\boldsymbol{Q} - \mathbb{E}[\boldsymbol{Q}])\Delta\boldsymbol{W}_{\boldsymbol{T}}^T\right)\right] \approx 0. \tag{30}$$

Using $\boldsymbol{G}_{\boldsymbol{W}_{\boldsymbol{T}}} = \boldsymbol{X}_{\boldsymbol{T}}^T\boldsymbol{G}_{\boldsymbol{Z}}$ in the trace form of the proxy gives

$$\begin{aligned}\mathbb{E}\left[\lVert\boldsymbol{X}_{\boldsymbol{T}}\Delta\boldsymbol{W}_{\boldsymbol{T}}\rVert_{\boldsymbol{F}_{\otimes\boldsymbol{Z}}}^2\right] &= \mathbb{E}\left[\mathrm{Tr}\left(\Delta\boldsymbol{W}_{\boldsymbol{T}}\boldsymbol{Q}\Delta\boldsymbol{W}_{\boldsymbol{T}}^T\boldsymbol{P}\right)\right] \\ &\approx \mathrm{Tr}\left(\Delta\boldsymbol{W}_{\boldsymbol{T}}\mathbb{E}[\boldsymbol{Q}]\,\Delta\boldsymbol{W}_{\boldsymbol{T}}^T\mathbb{E}[\boldsymbol{P}]\right).\end{aligned} \tag{31}$$

Under the constant-norm assumption, $\mathbb{E}[\boldsymbol{Q}] = \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}\right]/\mathbb{E}\left[\lVert\boldsymbol{G}_{\boldsymbol{Z}}\rVert_F^2\right]$. Substituting the definitions of $\boldsymbol{P}$ and $\boldsymbol{Q}$ and applying row vectorization therefore yields

$$\mathbb{E}\left[\lVert\boldsymbol{X}_{\boldsymbol{T}}\Delta\boldsymbol{W}_{\boldsymbol{T}}\rVert_{\boldsymbol{F}_{\otimes\boldsymbol{Z}}}^2\right] \approx \lVert\Delta\boldsymbol{W}_{\boldsymbol{T}}\rVert^2_{\frac{\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}_{\boldsymbol{T}}}\boldsymbol{G}_{\boldsymbol{W}_{\boldsymbol{T}}}^T\right]\otimes\mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{Z}}^T\boldsymbol{G}_{\boldsymbol{Z}}\right]}{\mathbb{E}\left[\lVert\boldsymbol{G}_{\boldsymbol{Z}}\rVert_F^2\right]}}. \tag{32}$$

B Experimental details

B.1 Models and Baselines

We evaluate across multiple model families and scales, including *Llama v3 8B* and *Llama v3.2 1B instruct* (Grattafiori et al., 2024), *Ministral 8B instruct* (Mistral.ai, 2025), and *Qwen v3 8B* (Team, 2025). We compare against representative state-of-the-art transformation methods, including Quarot (Ashkboos et al., 2024b), FlatQuant (Sun et al., 2025), CAT (Federici et al., 2026) and WUSH (Chen et al., 2026b). For rounding algorithms, we consider GPTQ/LDLQ (Frantar et al., 2022; Chee et al., 2023), and YAQA Sketch B (Tseng et al., 2026).

B.2 Detailed Setup

The experimental configuration for the results reported in Table 2 and Table 3 is reported in Table 6 and Table 7, respectively. In the tables $\boldsymbol{\Sigma}_{\boldsymbol{X}}$ refers to $\mathbb{E}\left[\boldsymbol{X}^T\boldsymbol{X}\right]$, $\boldsymbol{\Sigma}_{\boldsymbol{W}}$ to $\boldsymbol{W}\boldsymbol{W}^T$, and $\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{X}}} = \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}\right]$, $\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{W}}} = \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{W}}\boldsymbol{G}_{\boldsymbol{W}}^T\right]$. The matrix $\boldsymbol{U}$ refers to the eigen-decomposition of $\boldsymbol{\Sigma}_{\boldsymbol{X}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{W}} = \boldsymbol{U}\Lambda\boldsymbol{U}^T$. $B(\cdot)$ refers to the extraction of a block-diagonal component. Note that for block-wise quantization experiments, we omit the activation clipping strategy because of the limited impact on the small block size.

For all the experiments, we use 500 candidates sampled uniformly in $[\max_i \lvert\boldsymbol{w}_i\rvert, 2\max_i \lvert\boldsymbol{w}_i\rvert]$ to determine the minimizer for Hessian-weighted and L_p norms. For the minimization of the clipping parameter, we use only 100 candidates uniformly sampled in $[0.5, 1]$, determining the best by summing the error over 32 different sequences. In block-wise quantization settings, both YAQA and our algorithm process 32 output channels at a time during rounding. We select the quantization scale independently for each block, but batch the scale search across the 32 blocks. This reduces runtime without significantly affecting quantization performance.

Table 7: Details on the setting used to produce Table 3. The transformation in this table already refer to a single block of size 32.

Method	Weight Scale	Weight Round	Act. Clip	Block Transforms
RTN	$2\max_i \lvert\boldsymbol{w}_i\rvert$	None	$\gamma = 1$	None
Had.$^{+GPTQ}$	$2\max_i \lvert\boldsymbol{w}_i\rvert$	GPTQ	$\gamma = 1$	Hadamard
Had.$^{+YAQA}$	$2\max_i \lvert\boldsymbol{w}_i\rvert$	YAQA	$\gamma = 1$	Hadamard
CAT^{+GPTQ}	$2\max_i \lvert\boldsymbol{w}_i\rvert$	GPTQ	$\gamma = 1$	$(\boldsymbol{\Sigma}_{\boldsymbol{X}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{W}})^{1/2}\boldsymbol{H}$
WUSH^{+GPTQ}	$2\max_i \lvert\boldsymbol{w}_i\rvert$	GPTQ	$\gamma = 1$	$(\boldsymbol{\Sigma}_{\boldsymbol{X}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{W}})^{1/2}\boldsymbol{U}\boldsymbol{H}$
$\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ	Algorithm in Sec. 3.5		$\gamma = 1$	$(\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{W}}}^{-1}\#\boldsymbol{\Sigma}_{\boldsymbol{G}_{\boldsymbol{X}}})^{1/2}\boldsymbol{H}$

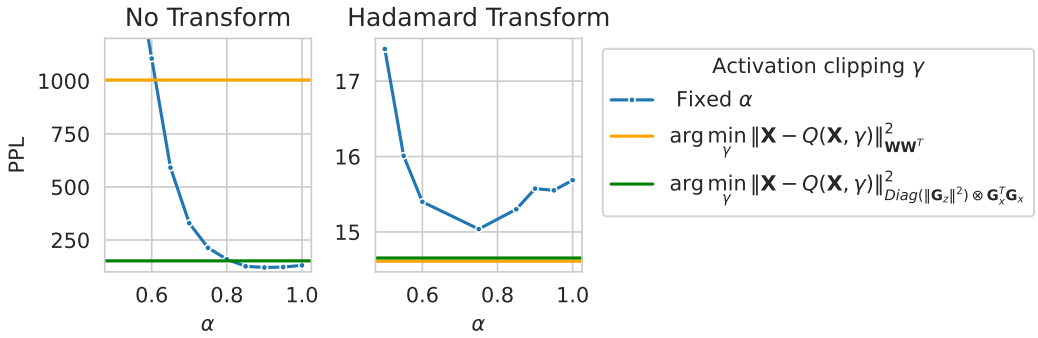

Figure 3: Wikitext perplexity obtained by different activation clipping objectives on Llama 3.2 1B Instruct under 4-bit per token quantization, with and without Hadamard transformations.

C Additional Results

Activation Clipping Figure 3 shows the effect of different activation clipping strategies. Overall, using the proposed Hessian proxy $\mathbb{E}\left[\text{Diag}\left(\left[\|\boldsymbol{g}_{\boldsymbol{Z},1}\|_2^2, \ldots, \|\boldsymbol{g}_{\boldsymbol{Z},s}\|_2^2\right]\right)\right] \otimes \mathbb{E}\left[\boldsymbol{G}_{\boldsymbol{X}}^T\boldsymbol{G}_{\boldsymbol{X}}\right]$, performs strongly against manual selection of the clipping percentage or clipping using simpler hessian proxies. However, the sub-optimality is likely due to the fact that the same "average" offline estimated curvature is used for all tokens instead of using a curvature estimation for each token independently. For $\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ we assume the covariances have been transformed by T according to Eq. 29.

Results with simple Round to Nearest Table 8 reports the results for the methods and quantization setting in Table 2 without any weight rounding adjustment. Our $\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ still outperforms the baselines even without the improved weight-rounding and scale setting algorithm, demonstrating the effectiveness of our transformations and activation clipping method.

Table 8: A4W4KV4 Quantization results without any weight rounding optimization and standard L_p-based ranges.

Method	*Llama v3.2 1B inst.* wiki (↓)	*Llama v3.2 1B inst.* csr (↑)	*Llama v3 8B* wiki (↓)	*Llama v3 8B* csr (↑)	*Ministral 8B inst.* wiki (↓)	*Ministral 8B inst.* csr (↑)	*Qwen v3 8B* wiki (↓)	*Qwen v3 8B* csr (↑)
FP	13.16	57.61	6.14	70.12	6.97	71.29	9.73	67.46
QuaRot	29.77	46.23	10.99	57.07	9.14	64.55	13.68	57.33
CAT	$18.20^{\pm 0.10}$	$51.41^{\pm 0.28}$	$7.67^{\pm 0.01}$	$63.90^{\pm 0.24}$	$7.99^{\pm 0.01}$	$67.72^{\pm 0.20}$	$11.01^{\pm 0.04}$	$63.66^{\pm 0.05}$
FlatQuant	$17.09^{\pm 0.04}$	$52.60^{\pm 0.42}$	$7.50^{\pm 0.52}$	$66.28^{\pm 2.04}$	$7.84^{\pm 0.05}$	$68.71^{\pm 0.14}$	$10.55^{\pm 0.13}$	$65.34^{\pm 0.32}$
$\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ$^{-a.clip}$	$17.78^{\pm 0.12}$	$51.42^{\pm 0.16}$	$7.51^{\pm 0.01}$	$64.55^{\pm 0.56}$	$7.95^{\pm 0.02}$	$68.23^{\pm 0.25}$	$10.78^{\pm 0.04}$	$63.21^{\pm 0.00}$
$\boldsymbol{H}_{\boldsymbol{Z}}$-PTQ	$16.52^{\pm 0.07}$	$53.13^{\pm 0.57}$	$7.13^{\pm 0.00}$	$66.54^{\pm 0.04}$	$7.74^{\pm 0.02}$	$69.10^{\pm 0.15}$	$10.56^{\pm 0.04}$	$65.14^{\pm 0.42}$