BEYOND MODALITY IMPORTANCE: TOKEN ROUTING ATTENTION FOR MULTIMODAL LLM QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-training quantization (PTQ) has become a key technique for reducing the storage and inference costs of multimodal large language models (MLLMs). Existing modality-aware PTQ methods learn independent activation smoothing scales for text and vision modalities, yet still rely on modality labels as the unit of quantization decisions, overlooking fine-grained variations in activation magnitudes across tokens within the same modality. In this paper, we propose Token Routing Attention Quantization (TRAQ), a PTQ framework that treats modality-specific activation smoothing scales as anchors and performs token-level scale routing through compatibility-based attention. Our core insight is that the optimal activation smoothing scale depends not only on global modality activation distributions but also on token-specific activation magnitudes. To this end, TRAQ treats modality-specific activation smoothing scales as prior anchors and employs a token-level attention mechanism that evaluates the compatibility between each token and each anchor, thereby assigning each token an activation smoothing scale aligned with its own activation magnitudes. On this basis, TRAQ computes layer-wise text and vision reconstruction errors and converts them into dynamic weights for the modality-specific reconstruction losses. The resulting weighted objective assigns greater emphasis to the modality with larger quantization error and is minimized to update the scale anchors. Extensive experiments show that TRAQ consistently outperforms competing PTQ methods, with gains of up to 7.70 points, while introducing no additional learnable modules and maintaining comparable inference efficiency to existing modality-aware PTQ methods.

1 INTRODUCTION

Multimodal large language models (MLLMs), which integrate vision encoders, cross-modal projectors, and language decoders, have demonstrated remarkable capabilities across a wide range of tasks, including visual question answering, document understanding and chart analysis (Bai et al., 2025b; Wang et al., 2025). However, the combination of massive model parameters, visual tokens, and long-sequence decoding introduces substantial storage and computational overhead, limiting their deployment on resource-constrained devices.

Post-training quantization (PTQ), which maps weights and activations to low-bit representations using only a small calibration set, has emerged as a crucial technique for reducing deployment costs. Methods such as SmoothQuant (Xiao et al., 2023), GuidedQuant (Kim et al., 2025), OSTQuant (Hu et al., 2025), and SpinQuant (Liu et al., 2025) have significantly improved low-bit performance for text-only large language models (Girija et al., 2025; Cheng et al., 2025). Among these, channel-wise smoothing suppresses activation outliers by redistributing numerical ranges between activations and weights, demonstrating particular effectiveness in weight-activation quantization. However, the language decoder of MLLM simultaneously processes text and vision tokens, whose activation magnitudes and outlier distributions differ markedly. Directly sharing a single activation smoothing scale across both modalities inevitably leads to cross-modal mismatch.

Existing studies primarily exploit modality information to address cross-modal quantization discrepancies. MBQ (Li et al., 2025) incorporates modality-dependent sensitivity into the calibration objective to balance reconstruction errors between text and vision tokens, while MASQuant (Hu et al., 2026) learns separate modality-specific smoothing factors with cross-modal compensation.

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.

SplitQ (Zhong et al., 2026) further decouples modality-specific outlier channels and applies modality-aware transformations and adaptive compensation. Despite their different mechanisms, these methods organize quantization adaptation around modality-level statistics or predefined modality-specific groups, rather than allowing individual tokens to select smoothing scales according to their own activation magnitudes.

Inner-modality token variation has recently gained increasing attention. ImpQuant (Zhou et al., 2026) estimates the importance of text and vision tokens through aggregated attention and contextual redundancy, respectively, and assigns higher calibration weights to critical tokens. QuantExperts (Jia et al., 2026) observes that important channels occur with different frequencies across tokens, and handles them via shared and routed experts, respectively. These works reveal that tokens within the same modality exhibit fine-grained differences in sensitivity, rendering modality labels insufficient as the sole basis for quantization decisions.

However, existing token-aware PTQ methods primarily focus on **token-aware error redistribution**: they answer the question of "which token errors should be prioritized during calibration", yet once the optimization focus is determined, the actual smoothing transformation is still governed by modality-level or channel-level smoothing scales. This leads to a granularity gap—fine-grained token sensitivity information is filtered out before entering the actual quantization decision, and tokens with different activation magnitudes within the same modality are ultimately mapped to the same class of smoothing transformation. This granularity gap motivates our Token Routing Attention Quantization (TRAQ), whose key idea is that token-specific activation characteristics should influence not only calibration loss weighting but also the selection of activation smoothing scales themselves.

To address this granularity gap, we formulate the problem as preserving modality-specific distribution priors while allowing individual tokens to adapt their quantization transformations according to their activation magnitudes. Accordingly, TRAQ treats modality-specific smoothing scales as learnable anchors and enables each token to construct a token-specific smoothing scale by adaptively aggregating these anchors through compatibility-based attention. Furthermore, we introduce a token-scale routing-aware calibration strategy that evaluates modality reconstruction errors on quantized outputs and dynamically adjusts the modality-wise optimization focus at each layer, allowing reconstruction feedback to jointly refine the scale anchors and token-scale routing.

The contributions of this paper are summarized as follows:

First, we identify a granularity gap in modality-aware quantization: token sensitivity influences loss coefficients but does not directly affect the smoothing transformation. We reframe the problem from "quantization error redistribution" to "scale-adaptive token routing" and propose TRAQ, a framework that jointly optimizes scale anchors, token-level scale aggregation, and layer-wise calibration.

Second, we propose a token-level scale-routing mechanism based on compatibility scores. By evaluating each token against the modality-specific scale anchors and converting the resulting scores into normalized routing coefficients, the mechanism constructs token-specific activation smoothing scales that match individual activation magnitudes.

Third, we introduce a token-scale routing-aware end-to-end calibration mechanism that dynamically adjusts the modality-wise optimization focus at each layer according to routing-conditioned reconstruction errors. By propagating these errors back to the scale anchors, it jointly refines token-anchor routing and quantization reconstruction, coupling token-level adaptation with layer-wise error feedback.

2 Related Work

2.1 Post-Training Quantization

Post-training quantization (PTQ) determines quantization parameters using a small calibration set and maps full-precision weights or activations to low-bit representations without retraining the full model. Weight-only PTQ has evolved from second-order reconstruction methods such as GPTQ (Frantar et al., 2023) to recent loss-aware and mixed-precision approaches such as LeanQuant and SKIM (Zhang & Shrivastava, 2025; Bai et al., 2025a). Weight-activation quantization must additionally handle activation outliers and distribution mismatch. Recent methods such as GuidedQuant, OSTQuant, FlatQuant,

and ASER improve calibration objectives, equivalent transformations, activation smoothing, or error reconstruction (Kim et al., 2025; Hu et al., 2025; Sun et al., 2025; Zhao et al., 2025), while SpinQuant suppresses quantization-sensitive outliers through learned rotations (Liu et al., 2025). Recent serving-oriented low-bit systems such as QServe and ABQ-LLM further push the accuracy–efficiency frontier through W4A8KV4 system co-design and arbitrary-bit GPU acceleration (Lin et al., 2025; Zeng et al., 2025). These methods are primarily designed for unimodal textual distributions and cannot capture the fine-grained variation in multimodal inputs where shared layers process text and vision tokens jointly.

2.2 Modality-Aware Multimodal Quantization

MLLM decoders process text and vision tokens within the same linear layers, making quantization errors sensitive to cross-modal activation differences. Early studies address this challenge by introducing modality-aware calibration strategies. MBQ balances text and vision reconstruction losses using modality-dependent gradients (Li et al., 2025), while MQuant designs fully static quantization for modality-specific activation ranges (Yu et al., 2025). More recently, MASQuant identifies "smoothing misalignment" caused by shared smoothing parameters across modalities, learning independent scales for each modality with cross-modal compensation (Hu et al., 2026). QuantExperts further explores token-level variations by observing that important channels appear with different frequencies across tokens, and adopts shared and routed experts to compensate token-independent and token-dependent channels, respectively (Jia et al., 2026). However, existing methods still rely on predefined modality or channel groups. TRAQ extends this line by introducing token-scale routing attention, enabling each token to adaptively determine its smoothing transformation.

2.3 Token Importance and Fine-Grained Quantization Sensitivity

Recent studies have advanced MLLM quantization from modality to token granularity. VLMQ incorporates token importance factors into a Hessian reconstruction objective to reduce bias from uniform token weighting (Xue et al., 2025). ImpQuant estimates text and vision token importance through aggregated attention and contextual redundancy, upweighting critical tokens in calibration (Zhou et al., 2026). QIG uses Quantization-aware Integrated Gradients to directly measure each token's contribution to quantization error, revealing sensitivity variations across modalities, tokens, and layers (Xiang et al., 2026). These methods primarily redistribute calibration errors according to importance, but the actual smoothing transformation remains modality- or channel-level. TRAQ addresses the complementary question of how tokens with different activation magnitudes receive different smoothing treatments, integrating modality differences, token variation, and layer-wise error evolution into a unified process.

3 Methodology

3.1 Problem Formulation: From Modality Mismatch to Token-Level Scale Mismatch

Consider the linear transformation at layer l of an MLLM language decoder. Let $\mathbf{h}_i \in \mathbb{R}^d$ denote the activation vector of the i-th input token, where d is the hidden dimension; $\mathbf{W} \in \mathbb{R}^{d_o \times d}$ denotes the weight matrix of VLM, where d_o is the output dimension; and $\mathbf{y}_i = \mathbf{h}_i \mathbf{W}^\top \in \mathbb{R}^{d_o}$ is the full-precision output.

Channel-wise activation smoothing leverages scale equivariance (Xiao et al., 2023) by introducing a positive scale vector $\mathbf{s} \in \mathbb{R}^d_+$:

$$\mathbf{y}_i = \mathbf{h}_i \mathbf{W}^\top = (\mathbf{h}_i \oslash \mathbf{s})(\mathbf{W} \odot \mathbf{s})^\top, \tag{1}$$

where $\oslash$ and $\odot$ denote channel-wise division and multiplication, respectively, i.e., $(\mathbf{h}_i \oslash \mathbf{s})_j = h_{i,j}/s_j$ and $(\mathbf{W} \odot \mathbf{s})_{r,j} = W_{r,j} \cdot s_j$.

Under weight-activation quantization, the quantized output is:

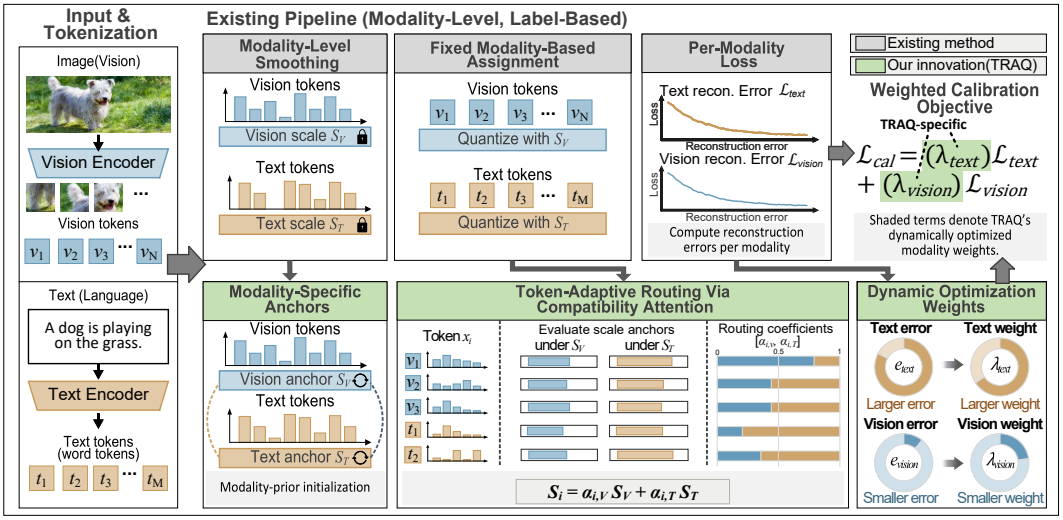


Figure 1: Overview of Token Routing Attention for Multimodal LLM Quantization (TRAQ). Existing modality-aware PTQ assigns fixed smoothing scales based on modality labels, while TRAQ introduces modality-specific anchors, token-adaptive scale aggregation via compatibility-based routing, and error-driven dynamic calibration weights for adaptive optimization.

$$\hat{\mathbf{y}}_i(\mathbf{s}) = Q_A(\mathbf{h}_i \oslash \mathbf{s}) \cdot Q_W(\mathbf{W} \odot \mathbf{s})^\top, \tag{2}$$

where $Q_A(\cdot)$ and $Q_W(\cdot)$ denote the activation and weight quantizers, respectively. We adopt the standard uniform quantizer defined as:

$$Q(x) = \Delta \cdot \text{clip}\left(\lfloor x/\Delta \rceil, -2^{b-1}, 2^{b-1}-1\right), \tag{3}$$

where b denotes the target bit width, Δ is the quantization step size, and $\lfloor \cdot \rceil$ denotes the rounding-to-nearest operator.

Let Ω_t and Ω_v denote the sets of text and vision tokens, respectively. For modality $m \in \{t, v\}$, the average reconstruction error is:

$$\mathcal{L}_m(\mathbf{s}) = \frac{1}{|\Omega_m|} \sum_{i \in \Omega_m} \frac{1}{d_o} \left\| \mathbf{y}_i - \hat{\mathbf{y}}_i(\mathbf{s}) \right\|_1. \tag{4}$$

The global reconstruction error $\mathcal{L}(\mathbf{s})$ is the weighted average over all tokens:

$$\mathcal{L}(\mathbf{s}) = \frac{N_t}{N_t + N_v} \mathcal{L}_t(\mathbf{s}) + \frac{N_v}{N_t + N_v} \mathcal{L}_v(\mathbf{s}). \tag{5}$$

Let $\mathbf{s}^* = \arg\min_{\mathbf{s}} \mathcal{L}(\mathbf{s})$ be the optimal unified scale. To analyze the inherent limitation of unified scaling, consider the inner-product relationship between the two modality gradients. At the optimal unified scale $\mathbf{s}^*$, the global gradient satisfies:

$$\nabla_{\mathbf{s}} \mathcal{L}(\mathbf{s}^*) = \frac{N_t}{N_t + N_v} \nabla_{\mathbf{s}} \mathcal{L}_t(\mathbf{s}^*) + \frac{N_v}{N_t + N_v} \nabla_{\mathbf{s}} \mathcal{L}_v(\mathbf{s}^*) = \mathbf{0}, \tag{6}$$

where $\nabla_{\mathbf{s}} \mathcal{L}_t(\mathbf{s}^*)$ and $\nabla_{\mathbf{s}} \mathcal{L}_v(\mathbf{s}^*)$ define the error gradient of modality t and modality v at $\mathbf{s}^*$. This yields $\mathbf{g}_t = -\frac{N_v}{N_t} \mathbf{g}_v$, indicating that the two modality gradients point in opposite directions at $\mathbf{s}^*$. This reveals the inherent limitation of unified scaling: at the optimal $\mathbf{s}^*$, reducing one modality's error increases the other's.

3.2 Modality-Specific Smoothing Scale Anchors: Preserving Cross-Modal Distribution Priors

Eq. 6 shows that the text and vision reconstruction gradients at the optimal unified scale are oppositely directed, indicating the two modalities should not be forced to share a single activation smoothing scale. TRAQ therefore first constructs an independent activation smoothing anchor for each modality.

Importantly, these anchors are not treated as the final token-wise smoothing decisions. Instead, they provide a compact set of modality-informed numerical priors over which token-level routing is subsequently performed.

From modality k, we estimate the channel-wise activation magnitudes from the calibration data and summarize the weight magnitudes along the corresponding input channels as:

$$\mathbf{a}_k = \max_{i \in \Omega_k} |\mathbf{h}_i| \in \mathbb{R}^d, \qquad \mathbf{w} = \max_{r=1}^{d_o} |\mathbf{W}_{r,:}| \in \mathbb{R}^d. \tag{7}$$

Each element of $\mathbf{a}_k$ characterizes the peak activation magnitudes of modality k on one hidden channel, while $\mathbf{w}$ captures the peak magnitudes of the corresponding weight channel. The initial scale anchor for modality k is constructed as:

$$\mathbf{s}_{k,0} = \mathbf{a}_k^{\beta_k} \oslash \mathbf{w}^{1-\beta_k}, \quad \beta_k \in [0, 1]. \tag{8}$$

The exponent β_k controls the relative contribution of activation and weight magnitudes. Instead of sharing a single coefficient across modalities, TRAQ learns an independent β_k for each modality, allowing the initial smoothing prior to reflect modality-specific numerical statistics.

We refer to $s_{k,0}$ as an anchor rather than a fixed modality scale. This distinction is essential. A fixed scale estimated from the original calibration distribution may become suboptimal after preceding layers have been quantized, since layer-wise quantization perturbations alter the activation distribution received by subsequent layers. Considering modality k in isolation, a modality-specific scale s_k^* satisfies:

$$\nabla_{\mathbf{s}} \mathcal{L}_k(\mathbf{s}_k^*) = \mathbf{0}. \tag{9}$$

The modality-specific scaling scheme avoids the constraint where the gradients of two modalities cancel each other. This theoretically explains why modality-specific scaling breaks the optimization dilemma of unified scaling.

3.3 Token-Adaptive Scale Routing via Compatibility Attention

The modality scale anchors $\{\mathbf{s}_k^*\}_{k=1}^K$ preserve cross-modal distribution priors, yet activation magnitudes of tokens within the same modality may differ by orders of magnitude. Given token $\mathbf{h}_i$ from any modality, TRAQ defines its compatibility score with each candidate scale $\mathbf{s}_k^*$ $(k = 1, \ldots, K)$ — regardless of whether the anchor belongs to the same modality as the token:

$$g_{i,k} = -\frac{1}{d} \left\| \mathbf{h}_i \oslash \mathbf{s}_k^* \right\|_1. \tag{10}$$

The scale s_k^* transforms the activation into $h_i \oslash s_k^*$. The quantization step is determined by the range of the normalized activations. A scale that produces smaller normalized activation magnitudes leads to a smaller activation range and hence a finer quantization step, reducing activation-side quantization error. The score $g_{i,k}$ thus directly reflects the expected quantization fidelity under each scale.

Softmax normalization converts these scores into routing coefficients, assigning larger coefficients to anchors that are more compatible with the activation magnitudes of the given token:

$$\alpha_{i,k} = \frac{\exp(g_{i,k})}{\sum_{j=1}^K \exp(g_{i,j})}, \qquad \sum_{k=1}^K \alpha_{i,k} = 1. \tag{11}$$

The modality-specific anchors are aggregated according to the token-specific routing distribution to construct a token-adaptive smoothing scale:

$$s_i = \sum_{k=1}^{K} \alpha_{i,k} s_k^*. \tag{12}$$

Here, $\alpha_{i,k}$ is computed on-the-fly from the compatibility between token activation and scale anchors. The resulting s_i is used to adapt the activation range of token i before quantization. In contrast to the token-adaptive activation path, weight quantization is performed once offline after calibration. We denote the resulting fixed quantized weight by $\widetilde{W}_q = Q_W(W)$, which is shared by all tokens during inference. After activation quantization, the original activation scale is restored by multiplying s_i:

$$\hat{y}_i = [Q_A(h_i \oslash s_i) \odot s_i]\, \widetilde{W}_q^\top, \qquad \widetilde{W}_q = Q_W(W). \tag{13}$$

In practice, the token-specific rescaling can be fused into activation dequantization: the quantized activation is dequantized using an effective scale that incorporates s_i and is directly consumed by the subsequent linear kernel, without materializing an additional full-precision activation tensor. Since the quantized weight $\widetilde{W}_q$ is fixed and shared across tokens and $K = 2$ in our setting, the additional routing overhead remains lightweight. A CUDA-style implementation sketch of this fusion is provided in Appendix I.

3.4 Token-Scale Routing-Aware End-To-End Error Calibration

Rather than treating routing and calibration as independent stages, TRAQ evaluates reconstruction errors on the resulting token-adaptive quantized outputs, so that the calibration objective jointly reflects token-scale routing and quantization fidelity.

We denote by $E_m^{(l)}$ the average reconstruction error of modality m evaluated on these token-adaptive quantized outputs. Unlike conventional modality-wise reconstruction errors under a fixed smoothing transformation, $E_m^{(l)}$ is conditioned on the token-specific routed scales $s_i^{(l)}$, which are in turn determined by the routing distribution $\alpha_{i,k}^{(l)}$. Therefore, it directly reflects how well the modality-specific scale anchors are combined for tokens of each modality.

Since routing-induced quantization errors may differ across modalities and evolve across layers, we dynamically adjust the modality-wise calibration focus at each layer according to the current reconstruction errors:

$$w_m^{(l)} = \frac{\exp\left(\mathrm{sg}\left(E_m^{(l)}\right)\right)}{\sum_{j\in\{t,v\}} \exp\left(\mathrm{sg}\left(E_j^{(l)}\right)\right)}, \qquad \sum_{m\in\{t,v\}} w_m^{(l)} = 1, \tag{14}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operation. Modalities with larger routing-conditioned errors receive larger calibration weights, steering the optimization toward the modality that is more affected by quantization at the current layer.

The TRAQ calibration objective at layer l is then defined as:

$$\mathcal{L}_{\text{TRAQ}}^{(l)} = \sum_{m\in\{t,v\}} w_m^{(l)} \frac{1}{|\Omega_m^{(l)}|} \sum_{i\in\Omega_m^{(l)}} \frac{1}{d_o} \left\| y_i^{(l)} - \left[Q_A\left(h_i^{(l)} \oslash s_i^{(l)}\right) \odot s_i^{(l)}\right] \left(\widetilde{W}_q^{(l)}\right)^\top \right\|_1. \tag{15}$$

By incorporating the token-specific routed scale into Eq. 15, the objective couples token-scale routing with reconstruction rather than merely reweighting modality-wise losses. In particular, the reconstruction error of each token jointly depends on its routing coefficients $\alpha_{i,k}^{(l)}$ and the resulting activation scale $\mathbf{s}_i^{(l)}$.

During calibration, gradients through the rounding operations in $Q_A(\cdot)$ are approximated using the straight-through estimator. Under this formulation, the gradient with respect to the k-th optimized scale anchor $s_k^{*(l)}$ further illustrates the coupling between scale routing and reconstruction:

$$\frac{\partial \mathcal{L}_{\text{TRAQ}}^{(l)}}{\partial s_k^{*(l)}} = \sum_{m\in\{t,v\}} w_m^{(l)} \sum_{i\in\Omega_m^{(l)}} \frac{\partial E_m^{(l)}}{\partial \hat{y}_i^{(l)}} \frac{\partial \hat{y}_i^{(l)}}{\partial s_i^{(l)}} \left[\alpha_{i,k}^{(l)} I + \alpha_{i,k}^{(l)} \left(s_k^{*(l)} - s_i^{(l)} \right) \left(\frac{\partial g_{i,k}^{(l)}}{\partial s_k^{*(l)}} \right)^{\top} \right]. \tag{16}$$

The first term captures the direct scale aggregation effect: updating $s_k^{*(l)}$ changes the token-specific activation scale $s_i^{(l)}$ according to its routing coefficient $\alpha_{i,k}^{(l)}$. The second term captures the routing effect: updating $s_k^{*(l)}$ changes its compatibility with token i, thereby modifying the routing coefficients and consequently the aggregated scale.

This forms a routing-aware calibration loop within each layer: token-scale routing determines the smoothing scales and quantized outputs, while reconstruction errors update the calibration focus and refine the scale anchors. Quantization effects further propagate across layers, as the outputs at layer l influence the activations and reconstruction errors at layer $l+1$:

$$E_m^{(l+1)} = E_m^{(l+1)} \left(\{s_k^{*(l)}\}_{k=1}^{K}, \{\hat{y}_i^{(l)}\}_{i=1}^{N_l} \right), \tag{17}$$

where N_l denotes the number of tokens at layer l. Therefore, TRAQ couples token-level scale routing with per-layer reconstruction-aware calibration to adapt scale anchors to evolving quantization errors. Only the quantization parameters are optimized, while the original model weights remain frozen, preserving the post-training quantization setting.

4 Experiments

4.1 Experimental Setups

Datasets and Models. To activate both visual and linguistic capabilities during calibration, we construct a set of 128 image-text pairs randomly sampled from MS COCO 2017 (Lin et al., 2014) and ShareGPT4V (Chen et al., 2024a). We evaluate on two series of multimodal large language models: Qwen2.5-VL (Bai et al., 2025b) (7B) and InternVL2 (Chen et al., 2024b) (8B). Performance is measured via the LMMs-Eval (Zhang et al., 2025) on four diverse benchmarks: MMMU (Yue et al., 2024) (multi-discipline reasoning), RealWorldQA (xAI, 2024) (question answering), OCRBench (Liu et al., 2024) (text recognition), and AI2D (Kembhavi et al., 2016) (diagram reasoning).

Baselines and Configuration. We compare TRAQ against the following PTQ methods: (1) MASQuant (Hu et al., 2026), a modality-aware smoothing quantization method that learns independent smoothing scales for different modalities; (2) MBQ (Li et al., 2025), which balances text and vision reconstruction losses using modality-dependent gradients; (3) ImpQuant (Zhou et al., 2026), an importance-aware quantization method that assigns higher calibration weights to critical tokens; and (4) QuantExperts (Jia et al., 2026), which compensates token-independent and token-dependent channels via shared and routed experts, respectively.

4.2 Main Results on Multimodal Benchmarks

We compare TRAQ with representative PTQ methods under various quantization settings. As shown in Table 1, TRAQ achieves the strongest overall performance across the evaluated models and quantization settings. Appendix A further extends the evaluation to InternVL2-1B, 2B, 26B and LLaVA-OneVision-7B, covering broader model scales and an additional multimodal model family.

Under the challenging W4A8 setting, TRAQ achieves the best performance on all four benchmarks for both Qwen2.5-VL-7B and InternVL2-8B. This consistent gain under aggressive weight-activation quantization indicates that token-adaptive scale aggregation better captures fine-grained activation variations beyond modality-level smoothing.

Table 1: Performance comparison of different PTQ methods under various quantization settings.

Method	Bits	Qwen2.5-VL-7B				InternVL2-8B			
		MMMU	RWQA	OCR	AI2D	MMMU	RWQA	OCR	AI2D
Dense	FP16	0.5000	0.6876	0.839	0.8248	0.4811	0.6078	0.621	0.8187
MBQ	W4A16	0.4298	0.6415	0.702	0.7714	0.4279	0.6155	0.715	0.7832
MASQuant	W4A16	0.4433	0.6431	0.714	0.7835	0.4343	0.6379	0.737	0.7947
ImpQuant	W4A16	0.4493	0.6511	0.728	0.7906	0.4451	0.6403	0.739	0.7938
QuantExperts	W4A16	0.4693	0.6504	0.735	0.8017	0.4502	0.6438	0.751	**0.8032**
TRAQ (Ours)	W4A16	**0.4711**	**0.6797**	**0.812**	**0.8212**	**0.4756**	**0.6444**	**0.781**	0.8027
MBQ	W4A8	0.4063	0.6304	0.698	0.7438	0.4289	0.6034	0.716	0.7805
MASQuant	W4A8	0.4089	0.6235	0.680	0.7785	0.4378	0.6261	0.740	0.7902
ImpQuant	W4A8	0.4263	0.6315	0.742	0.7656	0.4438	0.6311	0.741	0.7915
QuantExperts	W4A8	0.4366	0.6203	0.754	0.7655	0.4517	0.6276	0.745	0.7706
TRAQ (Ours)	W4A8	**0.4512**	**0.6667**	**0.802**	**0.7863**	**0.4744**	**0.6442**	**0.777**	**0.8147**
MBQ	W6A16	0.4606	0.6689	0.801	0.8034	0.4608	0.6289	0.737	0.8023
MASQuant	W6A16	0.4732	0.6704	0.836	**0.8225**	0.4729	0.6471	0.757	0.8199
ImpQuant	W6A16	0.4744	0.6832	0.834	0.8198	0.4614	0.6511	0.778	0.8027
QuantExperts	W6A16	0.4839	0.6722	0.828	0.8205	0.4715	0.6522	0.783	0.8139
TRAQ (Ours)	W6A16	**0.4989**	**0.6993**	**0.836**	0.8212	**0.4767**	**0.6523**	**0.783**	**0.8229**
MBQ	W6A8	0.4533	0.6572	0.783	0.7987	0.4519	0.6311	0.729	0.7983
MASQuant	W6A8	0.4587	0.6704	0.824	0.8011	0.4744	0.6431	0.778	0.8180
ImpQuant	W6A8	0.4599	0.6738	**0.828**	0.8107	0.4618	0.6459	0.778	0.8145
QuantExperts	W6A8	0.4635	0.6647	0.820	0.8088	0.4724	0.6447	0.764	0.8087
TRAQ (Ours)	W6A8	**0.4774**	**0.6780**	0.827	**0.8170**	**0.4822**	**0.6471**	**0.778**	**0.8209**

Table 2: Ablation study of TRAQ under W4A8 and W6A8. Each entry reports Qwen2.5-VL-7B / InternVL2-8B. R and C denote token-adaptive routing and error-driven calibration.

Variant	W4A8				W6A8			
	MMMU	RWQA	OCR	AI2D	MMMU	RWQA	OCR	AI2D
Unified	0.3862/0.4034	0.5915/0.5874	0.664/0.701	0.7366/0.7653	0.4463/0.4389	0.6398/0.6214	0.759/0.711	0.7763/0.7805
Anchor	0.4035/0.4356	0.6153/0.6041	0.686/0.722	0.7442/0.7785	0.4497/0.4526	0.6535/0.6347	0.767/0.765	0.7977/0.8032
Anchor + R	0.4475/0.4702	0.6429/0.6307	0.775/0.747	0.7639/0.7917	0.4601/0.4732	0.6666/0.6438	0.793/0.771	0.8056/0.8117
Anchor + C	0.4381/0.4677	0.6388/0.6296	0.769/0.746	0.7599/0.7904	0.4633/0.4710	0.6645/0.6442	0.786/0.770	0.8047/0.8122
TRAQ	**0.4512/0.4744**	**0.6667/0.6442**	**0.802/0.777**	**0.7863/0.8147**	**0.4774/0.4822**	**0.6780/0.6471**	**0.827/0.778**	**0.8170/0.8209**

4.3 Ablation Study

We conduct ablations under W4A8 and W6A8 across all four benchmarks to evaluate the contribution of each component in TRAQ. As shown in Table 2, modality-specific anchors consistently improve over unified smoothing on both models. Token-adaptive routing further yields clear gains across different tasks and quantization settings, demonstrating the benefit of fine-grained token-level scale adaptation. Error-driven calibration also consistently improves the anchor-only baseline by emphasizing modalities with larger reconstruction errors. Combining both components achieves the strongest overall performance, confirming their complementary effects across models, benchmarks, and bit-width configurations.

4.4 Analysis of Token-Adaptive Scale Routing

We analyze the token-adaptive routing behavior of TRAQ on a representative Qwen2.5-VL-7B input containing 324 vision tokens and 30 text tokens. In Fig. 2, we randomly sample 8 vision tokens and 8 text tokens to visualize their routing weights over the two scale anchors S_V and S_T. In Fig. 3, we visualize the relationship between normalized token activation magnitudes $\frac{1}{d}\|h_i\|_1$ and the routing weight toward the vision anchor $\alpha_{i,V}$.

As shown in Fig. 2, tokens from the same modality exhibit diverse preferences over the two scale anchors, indicating that modality labels alone cannot fully capture token-level variations. Fig. 3

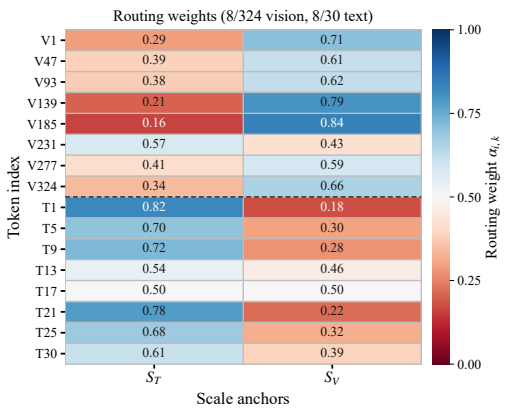


Figure 2: Tokens within the same modality exhibit diverse routing behaviors.

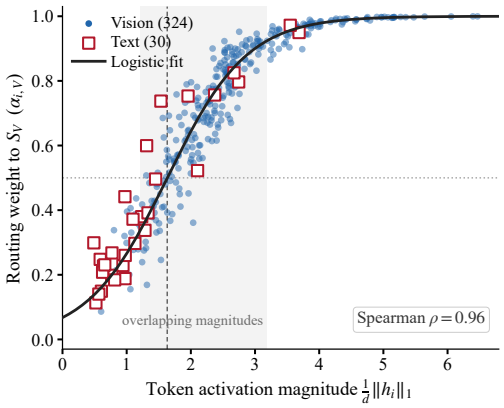


Figure 3: Relationship between token activation magnitudes and routing preference.

Table 3: End-to-end efficiency comparison on Qwen2.5-VL-7B under W4A8 quantization.

Method	Calibration Time (min)	Prefill		Decode	
		Latency (ms)	Speedup	TPOT (ms/token)	Memory (GB)
FP16	–	33.2	–	21.40	13.73
MASQuant	13.73	16.2	2.05×	9.73	5.15
QuantExperts	19.20	14.4	2.31×	10.60	5.03
TRAQ (Ours)	15.65	14.2	2.34×	9.68	4.97

shows that routing weights are strongly correlated with token activation magnitudes: tokens with larger activation magnitudes tend to assign higher weights to S_V. These experiments verify that TRAQ enables each token to select an appropriate combination of smoothing scales according to its own activation magnitudes, overcoming the limitations of modality-level quantization strategies.

4.5 Efficiency Analysis

We evaluate efficiency on Qwen2.5-VL-7B under W4A8 quantization using a single RTX 4090 with batch size 1. Prefill latency and decode TPOT are measured with 354-token multimodal inputs and 128 generated tokens, respectively. As shown in Table 3, TRAQ achieves comparable inference efficiency with low deployment overhead.

TRAQ achieves a prefill latency of 14.2 ms, corresponding to a 2.34× speedup over FP16 and slightly surpassing QuantExperts (2.31×). During decoding, TRAQ obtains a TPOT of 9.68 ms, matching MASQuant and outperforming QuantExperts. The calibration time of TRAQ is 15.65 minutes, which remains substantially lower than QuantExperts while being comparable to MASQuant. Moreover, TRAQ achieves the lowest memory consumption (4.97 GB) among all compared methods, demonstrating efficient deployment without additional learnable modules.

5 Conclusion

In this work, we identify the granularity gap in multimodal LLM quantization and propose TRAQ, which constructs token-specific smoothing scales by adaptively aggregating modality-specific scale anchors according to token activation magnitudes. This design preserves modality-level distribution priors while enabling fine-grained token adaptation through a single quantization path. TRAQ further couples token-scale routing with reconstruction-aware calibration, consistently improving low-bit quantization performance on Qwen2.5-VL, InternVL2 and LLaVA without introducing auxiliary learnable modules. These results underscore the value of token-level quantization adaptation.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, language refinement, literature checking, reference formatting, and improving the clarity of mathematical exposition. No experimental results were generated by AI tools. All technical content, methodological decisions, experimental results, and conclusions were independently verified by the authors, who take full responsibility for the final manuscript.

ETHICS STATEMENT

This work focuses on post-training quantization of existing multimodal large language models and does not involve new human-subject data collection or sensitive personal data. Our experiments use publicly available models, datasets, and benchmarks. Since quantization preserves the underlying capabilities and limitations of the original models, the resulting quantized models may inherit their existing biases and potential misuse risks. We therefore encourage responsible use in accordance with the licenses and usage policies of the corresponding models and datasets.

REPRODUCIBILITY STATEMENT

We provide the complete formulation, optimization objective, and calibration procedure of TRAQ in Section 3, including modality-specific scale anchor initialization, token-level scale routing, routing-aware calibration, and the inference formulation with fixed quantized weights.

Section 4 reports the experimental details required for reproduction, including calibration data, evaluated models (Qwen2.5-VL-7B, InternVL2-1B,2B,8B,26B and LLaVA-OneVision-7B), quantization configurations, evaluation benchmarks, baseline methods, and efficiency measurement settings. Specifically, we provide the calibration datasets, model configurations, evaluation protocols, and hardware settings for reproducing both quantization accuracy and inference efficiency results.

REFERENCES

Runsheng Bai, Bo Liu, and Qiang Liu. SKIM: Any-bit quantization pushing the limits of post-training quantization. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 2520–2534. PMLR, 2025a. URL https://proceedings.mlr.press/v267/bai25c.html.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Siyuan Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. ShareGPT4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision (ECCV)*, 2024a.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24185–24198, 2024b.

Jian Cheng, Haidong Kang, Yongbo Shao, Nan Li, Pengjun Chen, Rui Wang, Saiqin Long, Xiaochun Yang, and Lianbo Ma. Survey on efficient large language models: Principles, algorithms, applications, and open issues. *IEEE Transactions on Neural Networks and Learning Systems*, 2025. doi: 10.1109/TNNLS.2025.3628671. Online ahead of print.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tcbBPnfwxS.

Sanjay Surendranath Girija, Shashank Kapoor, Lakshit Arora, Dipen Pradhan, Aman Raj, and Ankit Shetgaonkar. Optimizing LLMs for resource-constrained environments: A survey of model compression techniques. In *2025 IEEE 49th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1657–1664, Toronto, ON, Canada, 2025. IEEE. doi: 10.1109/COMPSAC61105.2025.00221.

Lulu Hu, Wenhu Xiao, Xin Chen, Xinhua Xu, Bowen Xu, Kun Li, and Yongliang Tao. MASQuant: Modality-aware smoothing quantization for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2603.04800.

Xing Hu, Yuan Cheng, Dawei Yang, Zhixuan Chen, Zukang Xu, Jiangyong Yu, Chen Xu, Zhihang Yuan, Zhe Jiang, and Sifan Zhou. OSTQuant: Refining large language model quantization with orthogonal and scaling transformations for better distribution fitting. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=rAcgDBdKnP.

Chenwei Jia, Baoting Li, Xuchong Zhang, Mingzhuo Wei, Bochen Lin, and Hongbin Sun. Quant experts: Token-aware adaptive error reconstruction with mixture of experts for large vision-language models quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2602.24059.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision (ECCV)*, pp. 235–251. Springer, 2016.

Jinuk Kim, Marwa El Halabi, Wonpyo Park, Clemens Js Schaefer, Deokjae Lee, Yeonhong Park, Jae W. Lee, and Hyun Oh Song. Guidedquant: Large language model quantization via exploiting end loss guidance. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30011–30037. PMLR, 2025. URL https://proceedings.mlr.press/v267/kim25d.html.

Shiyao Li, Yingchun Hu, Xuefei Ning, Xihui Liu, Ke Hong, Xiaotao Jia, Xiuhong Li, Yaqi Yan, Pei Ran, Guohao Dai, Shengen Yan, Huazhong Yang, and Yu Wang. MBQ: Modality-balanced quantization for large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2412.19509.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pp. 740–755. Springer, 2014.

Yujun Lin, Haotian Tang, Shang Yang, Zhekai Zhang, Guangxuan Xiao, Chuang Gan, and Song Han. Qserve: W4a8kv4 quantization and system co-design for efficient llm serving. In *Proceedings of Machine Learning and Systems*, volume 7. MLSys, 2025. URL https://proceedings.mlsys.org/paper_files/paper/2025/hash/fbe2b2f74a2ece8070d8fb073717bda6-Abstract-Conference.html.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: On the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024.

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 *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/e5b1c0d4866f72393c522c8a00eed4eb-Abstract-Conference.html.

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 Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 57587–57613. PMLR, 2025. URL https://proceedings.mlr.press/v267/sun25l.html.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Bowen Zhou, Weijie Su, Kai Chen, Yu Qiao, Wenhai Wang, and Gen Luo. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025. URL https://arxiv.org/abs/2508.18265.

xAI. Grok-1.5 vision preview, April 2024. URL https://x.ai/news/grok-1.5v. Introduces the RealWorldQA benchmark.

Ziwei Xiang, Fanhu Zeng, Hongjian Fang, Rui-Qi Wang, Renxing Chen, Yanan Zhu, Yi Chen, Peipei Yang, and Xu-Yao Zhang. Fine-grained post-training quantization for large vision language models with quantization-aware integrated gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2603.17809.

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 *International Conference on Machine Learning*, pp. 38087–38099, 2023. URL https://proceedings.mlr.press/v202/xiao23c.html.

Yufei Xue, Yushi Huang, Jiawei Shao, Lunjie Zhu, Chi Zhang, Xuelong Li, and Jun Zhang. VLMQ: Token saliency-driven post-training quantization for vision-language models. *arXiv preprint arXiv:2508.03351*, 2025.

JiangYong Yu, Sifan Zhou, Dawei Yang, Shuo Wang, Shuoyu Li, Xing Hu, Chen Xu, Zukang Xu, Changyong Shu, and Zhihang Yuan. MQuant: Unleashing the inference potential of multimodal large language models via full static quantization. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025. doi: 10.1145/3746027.3755433. URL https://arxiv.org/abs/2502.00425.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9556–9567, 2024.

Chao Zeng, Songwei Liu, Yusheng Xie, Hong Liu, Xiaojian Wang, Miao Wei, Shu Yang, Fangmin Chen, and Xing Mei. Abq-llm: Arbitrary-bit quantized inference acceleration for large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22299–22307, 2025. doi: 10.1609/aaai.v39i21.34385. URL https://ojs.aaai.org/index.php/AAAI/article/view/34385.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Association for Computational Linguistics (ACL)*, pp. 881–916, 2025.

Tianyi Zhang and Anshumali Shrivastava. Leanquant: Accurate and scalable large language model quantization with loss-error-aware grid. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ISqx8giekS.

Weibo Zhao, Yubin Shi, Xinyu Lyu, Wanchen Sui, Shen Li, and Yong Li. Aser: Activation smoothing and error reconstruction for large language model quantization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22822–22830, 2025. doi: 10.1609/aaai.v39i21.34443. URL https://ojs.aaai.org/index.php/AAAI/article/view/34443.

Yi Zhong, Haotong Qin, Xindong Zhang, Lei Zhang, and Guolei Sun. Breaking modality heterogeneity in low-bit quantization for large vision-language models. *arXiv preprint arXiv:2605.19929*, 2026.

Jundong Zhou, Tianao Cai, Yujie Huang, Xinbing Wang, Guang-Zhong Yang, and Nanyang Ye. ImpQuant: Fine-grained importance-aware quantization for large vision-language models. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/64367.

A GENERALIZATION ACROSS MODEL SCALES AND ARCHITECTURES

To further evaluate the generalization of TRAQ beyond the two model configurations used in the main paper, we extend the experiments to InternVL2-1B, InternVL2-2B, InternVL2-26B, and LLaVA-OneVision-7B. Together with Qwen2.5-VL-7B and InternVL2-8B evaluated in the main paper, these experiments broaden the evaluated model scale from 1B to 26B parameters and introduce an additional multimodal model family and vision-language architecture.

All additional experiments follow the same calibration protocol, quantization configurations, evaluation benchmarks, and evaluation pipeline as those in Sec. 4 of the main paper. Specifically, we use the same calibration data construction and evaluate on MMMU, RealWorldQA, OCRBench, and AI2D under W4A16, W4A8, W6A16, and W6A8. Tables 4– 7 report the corresponding results.

Discussion. These additional experiments substantially broaden the evaluation beyond the two 7B–8B models considered in the main paper. The InternVL2 series provides a controlled comparison across model scales from 1B to 26B, while LLaVA-OneVision-7B evaluates transfer to a different multimodal model family. Together, these settings allow us to examine whether the benefits of token-adaptive scale routing depend on a particular parameter regime or model family.

Table 4: Quantization results on InternVL2-1B under different weight-activation configurations.

Method	Bits	MMMU	RWQA	OCR	AI2D
Dense	FP16	0.3500	0.5053	0.742	0.6900
MBQ	W4A16	0.3052	0.4678	0.685	0.6488
MASQuant	W4A16	0.3147	0.4802	0.701	0.6552
ImpQuant	W4A16	0.3233	0.4783	0.706	0.6621
QuantExperts	W4A16	0.3258	0.4844	0.712	0.6570
TRAQ (Ours)	W4A16	**0.3368**	**0.4935**	**0.728**	**0.6659**
MBQ	W4A8	0.2854	0.4499	0.668	0.6177
MASQuant	W4A8	0.2896	0.4621	0.686	0.6418
ImpQuant	W4A8	0.3009	0.4660	0.692	0.6420
QuantExperts	W4A8	**0.3188**	0.4678	0.694	0.6403
TRAQ (Ours)	W4A8	0.3173	**0.4711**	**0.716**	**0.6505**
MBQ	W6A16	0.3317	0.5032	0.721	0.6721
MASQuant	W6A16	0.3380	0.4968	0.723	0.6820
ImpQuant	W6A16	0.3365	0.5011	0.736	0.6800
QuantExperts	W6A16	0.3402	0.5049	0.739	**0.6856**
TRAQ (Ours)	W6A16	**0.3451**	**0.5074**	**0.740**	0.6847
MBQ	W6A8	0.3223	0.4856	0.711	0.6643
MASQuant	W6A8	0.3300	0.4971	0.729	0.6760
ImpQuant	W6A8	0.3287	0.4962	0.725	0.6741
QuantExperts	W6A8	0.3341	0.5011	0.732	0.6781
TRAQ (Ours)	W6A8	**0.3422**	**0.5042**	**0.738**	**0.6805**

Table 5: Quantization results on InternVL2-2B under different weight-activation configurations.

Method	Bits	MMMU	RWQA	OCR	AI2D
Dense	FP16	0.4000	0.5452	0.763	0.7411
MBQ	W4A16	0.3540	0.5144	0.696	0.7006
MASQuant	W4A16	0.3588	0.5250	0.718	0.7122
ImpQuant	W4A16	0.3770	0.5155	0.714	0.7100
QuantExperts	W4A16	0.3704	0.5307	0.721	0.7155
TRAQ (Ours)	W4A16	**0.3822**	**0.5325**	**0.726**	**0.7244**
MBQ	W4A8	0.3374	0.5011	0.688	0.6854
MASQuant	W4A8	0.3480	0.5236	0.702	**0.7111**
ImpQuant	W4A8	0.3499	0.5140	0.707	0.7023
QuantExperts	W4A8	0.3511	0.5144	0.712	0.6960
TRAQ (Ours)	W4A8	**0.3783**	**0.5325**	**0.722**	0.7106
MBQ	W6A16	0.3856	0.5327	0.728	0.7233
MASQuant	W6A16	0.3902	0.5419	0.740	0.7381
ImpQuant	W6A16	0.3880	0.5433	0.741	0.7360
QuantExperts	W6A16	0.3941	0.5467	**0.746**	0.7320
TRAQ (Ours)	W6A16	**0.3986**	**0.5508**	0.743	**0.7415**
MBQ	W6A8	0.3714	0.5255	0.719	0.7165
MASQuant	W6A8	0.3820	0.5314	0.734	0.7311
ImpQuant	W6A8	0.3807	0.5319	0.737	0.7282
QuantExperts	W6A8	0.3877	0.5390	0.740	0.7293
TRAQ (Ours)	W6A8	**0.3953**	**0.5401**	**0.740**	**0.7346**

Table 6: Quantization results on InternVL2-26B under different weight-activation configurations.

Method	Bits	MMMU	RWQA	OCR	AI2D
Dense	FP16	0.5833	0.6874	0.780	0.8805
MBQ	W4A16	0.5169	0.6650	0.721	0.8453
MASQuant	W4A16	0.5402	0.6762	0.742	0.8522
ImpQuant	W4A16	0.5367	0.6780	0.746	0.8520
QuantExperts	W4A16	0.5504	0.6811	0.752	0.8584
TRAQ (Ours)	W4A16	**0.5621**	**0.6833**	**0.762**	**0.8644**
MBQ	W4A8	0.5077	0.6511	0.714	0.8330
MASQuant	W4A8	0.5206	0.6625	0.739	0.8435
ImpQuant	W4A8	0.5271	0.6666	0.736	0.8480
QuantExperts	W4A8	0.5355	0.6660	0.745	0.8429
TRAQ (Ours)	W4A8	**0.5550**	**0.6800**	**0.760**	**0.8558**
MBQ	W6A16	0.5574	0.6720	0.752	0.8766
MASQuant	W6A16	0.5701	0.6736	0.767	0.8640
ImpQuant	W6A16	0.5682	0.6805	0.763	0.8731
QuantExperts	W6A16	0.5703	0.6793	0.765	0.8755
TRAQ (Ours)	W6A16	**0.5784**	**0.6865**	**0.778**	**0.8820**
MBQ	W6A8	0.5498	0.6709	0.744	0.8560
MASQuant	W6A8	0.5600	0.6740	0.761	0.8669
ImpQuant	W6A8	0.5584	**0.6799**	0.757	0.8682
QuantExperts	W6A8	0.5666	0.6726	0.765	0.8690
TRAQ (Ours)	W6A8	**0.5740**	0.6785	**0.773**	**0.8788**

Table 7: Quantization results on LLaVA-OneVision-7B under different weight-activation configurations.

Method	Bits	MMMU	RWQA	OCR	AI2D
Dense	FP16	0.4960	0.6678	0.623	0.8165
MBQ	W4A16	0.4542	0.6223	0.602	0.7926
MASQuant	W4A16	0.4612	0.6240	0.610	0.8011
ImpQuant	W4A16	0.4678	0.6281	0.613	0.8100
QuantExperts	W4A16	0.4750	0.6322	0.620	0.8045
TRAQ (Ours)	W4A16	**0.4880**	**0.6411**	**0.621**	**0.8134**
MBQ	W4A8	0.4305	0.5944	0.588	0.7775
MASQuant	W4A8	0.4468	0.6100	0.600	0.7902
ImpQuant	W4A8	0.4485	0.6068	0.604	0.7920
QuantExperts	W4A8	0.4552	0.6111	0.608	0.7884
TRAQ (Ours)	W4A8	**0.4733**	**0.6309**	**0.614**	**0.8100**
MBQ	W6A16	0.4611	0.6300	0.614	0.7979
MASQuant	W6A16	0.4667	0.6321	0.621	0.8066
ImpQuant	W6A16	0.4785	0.6466	0.619	**0.8143**
QuantExperts	W6A16	0.4766	0.6431	**0.625**	0.8099
TRAQ (Ours)	W6A16	**0.4793**	**0.6660**	0.624	0.8140
MBQ	W6A8	0.4554	0.6208	0.607	0.7956
MASQuant	W6A8	0.4565	0.6330	0.615	0.8011
ImpQuant	W6A8	0.4603	0.6351	0.617	0.8069
QuantExperts	W6A8	**0.4756**	0.6368	0.619	0.8001
TRAQ (Ours)	W6A8	0.4732	**0.6449**	**0.622**	**0.8122**

B WHY MODALITY-SPECIFIC SCALE ANCHORS ARE PREFERABLE TO UNIFIED SCALING

We first provide a complementary view of the limitation of unified activation smoothing discussed in Sec. 3.1 of the main paper. Let $L_m(s)$ denote the reconstruction loss of modality $m \in \{t, v\}$ under smoothing scale s, and let $p_m > 0$ denote the corresponding modality weight with $p_t + p_v = 1$. Unified smoothing solves

$$L^*_{\text{uni}} = \min_s \big[p_t L_t(s) + p_v L_v(s)\big]. \tag{1}$$

In contrast, modality-specific smoothing allows the two modalities to use independent scales:

$$L^*_{\text{sep}} = p_t \min_{s_t} L_t(s_t) + p_v \min_{s_v} L_v(s_v). \tag{2}$$

Proposition 1. Modality-specific smoothing has a feasible set no smaller than that of unified smoothing, and therefore

$$L^*_{\text{sep}} \le L^*_{\text{uni}}. \tag{3}$$

Proof. Let s^* be any feasible unified scale. Setting $s_t = s_v = s^*$ gives a feasible solution to the modality-specific problem. Therefore,

$$p_t \min_{s_t} L_t(s_t) + p_v \min_{s_v} L_v(s_v) \le p_t L_t(s^*) + p_v L_v(s^*). \tag{4}$$

Taking s^* to be the minimizer of the unified objective in Eq. 1 above, the inequality becomes strict when the two modalities prefer incompatible smoothing scales. Let

$$\mathcal{S}^*_m = \arg\min_s \mathcal{L}_m(s) \tag{5}$$

denote the set of optimal scales for modality m. If

$$\mathcal{S}_t^* \cap \mathcal{S}_v^* = \emptyset, \tag{6}$$

then no single scale can simultaneously attain both modality-wise minima, and the unified objective must incur a compromise.

This result complements the first-order gradient analysis in Sec. 3.1 of the main paper. At a stationary point of the unified objective,

$$p_t \nabla L_t(s^*) + p_v \nabla L_v(s^*) = 0, \tag{7}$$

so that, unless both modality gradients vanish independently, the two gradients are oppositely aligned. The modality-specific anchors introduced in Sec. 3.2 of the main paper remove this requirement that different modalities share the same stationary scale.

Importantly, TRAQ does not directly use these modality-wise optima as fixed token decisions. Instead, they serve as modality-informed anchors over which token-level interpolation is subsequently performed.

C EXPRESSIVENESS OF TOKEN-ADAPTIVE SCALE AGGREGATION

Given K modality-specific scale anchors

$$\mathcal{S} = \{s_1^*, \ldots, s_K^*\}, \tag{8}$$

TRAQ constructs the smoothing scale of token i as

$$s_i = \sum_{k=1}^{K} \alpha_{i,k} s_k^*, \quad \alpha_{i,k} \geq 0, \quad \sum_{k=1}^{K} \alpha_{i,k} = 1. \tag{9}$$

This immediately yields the following property.

Proposition 2. Every token-specific smoothing scale generated by TRAQ lies in the convex hull of the modality-specific anchors:

$$s_i \in \mathrm{conv}\{s_1^*, \ldots, s_K^*\}. \tag{10}$$

Thus, TRAQ provides continuous token-level adaptation without discarding the modality-level priors encoded by the anchors. Rather than learning an unconstrained scale independently for every token, it interpolates within a compact scale space determined by multimodal activation statistics.

We further compare token-adaptive routing with modality-level routing. Define

$$s(\boldsymbol{\alpha}_i) = \sum_{k=1}^{K} \alpha_{i,k} s_k^*. \tag{11}$$

and the token reconstruction loss

$$\ell_i(\alpha_i) = \frac{1}{d_o} \left\| y_i - \left[Q_A\big(h_i \oslash s(\alpha_i)\big) \odot s(\alpha_i) \right] \widetilde{W}_q^\top \right\|_1. \tag{12}$$

A modality-level routing strategy constrains all tokens from the same modality to share a common routing vector:

$$\alpha_i = \pi_m, \quad \forall i \in \Omega_m. \tag{13}$$

To compare the expressive capacity of token-level routing with modality-level routing, we consider the routing vectors $\boldsymbol{\alpha}_i$ as token-specific variables on the simplex. This comparison characterizes the routing space and does not imply that $\boldsymbol{\alpha}_i$ are independently optimized in TRAQ; in the actual routing mechanism, they are determined by compatibility attention.

Proposition 3. For a fixed set of anchors, the minimum reconstruction loss attainable by token-level routing cannot be worse than that attainable by modality-level routing:

$$\min_{\{\alpha_i \in \Delta^{K-1}\}} \sum_m \sum_{i \in \Omega_m} \ell_i(\alpha_i) \leq \min_{\{\pi_m \in \Delta^{K-1}\}} \sum_m \sum_{i \in \Omega_m} \ell_i(\pi_m). \tag{14}$$

Proof. Modality-level routing is a constrained special case of token-level routing obtained by enforcing Eq. 13. Hence its feasible set is contained in that of token-level routing, which directly gives Eq. 14.

The inequality can be strict when tokens within the same modality have different preferred routing distributions. This formalizes the granularity gap motivating TRAQ: a modality label specifies a useful distributional prior, but it does not uniquely determine the most suitable smoothing transformation for every token.

D COMPATIBILITY ATTENTION AS ENTROPY-REGULARIZED ROUTING

TRAQ defines the token-anchor compatibility score as

$$g_{i,k} = -c_{i,k}, \quad c_{i,k} = \frac{1}{d} \left\| h_i \oslash s_k^* \right\|_1 . \tag{15}$$

Here, $c_{i,k}$ should be interpreted as a lightweight activation-side compatibility cost rather than an exact estimate of the total quantization error.

The softmax routing rule in Eq. 11 of the main paper admits an optimization interpretation. Consider the entropy-regularized assignment problem

$$\min_{\boldsymbol{\alpha}_i \in \Delta^{K-1}} \left[\sum_{k=1}^{K} \alpha_{i,k} c_{i,k} + \sum_{k=1}^{K} \alpha_{i,k} \log \alpha_{i,k} \right], \tag{16}$$

where

$$\Delta^{K-1} = \left\{ \boldsymbol{\alpha} : \alpha_k \geq 0,\ \sum_k \alpha_k = 1 \right\}. \tag{17}$$

Introducing a Lagrange multiplier λ for the simplex constraint gives

$$\mathcal{J} = \sum_k \alpha_{i,k} c_{i,k} + \sum_k \alpha_{i,k} \log \alpha_{i,k} + \lambda \left(\sum_k \alpha_{i,k} - 1 \right). \tag{18}$$

Setting its derivative with respect to $\alpha_{i,k}$ to zero gives

$$c_{i,k} + 1 + \log \alpha_{i,k} + \lambda = 0, \tag{19}$$

which yields

$$\alpha_{i,k} = \frac{\exp(-c_{i,k})}{\sum_j \exp(-c_{i,j})}. \tag{20}$$

With $g_{i,k} = -c_{i,k}$, Eq. 20 exactly recovers the compatibility attention in Eq. 11 of the main paper. This view provides a useful interpretation of the routing mechanism. The first term in Eq. 16 favors anchors with lower compatibility cost, whereas the entropy term prevents premature hard assignment and allows information from multiple modality priors to contribute to the token-specific scale. TRAQ therefore realizes a soft token-adaptive interpolation over modality-informed anchors rather than a hard modality assignment.

E DERIVATION OF THE ROUTING-AWARE SCALE GRADIENT

We provide the detailed derivation of Eq. 16 in the main paper. For clarity, the layer superscript is omitted.

The routed scale is

$$s_i = \sum_{j=1}^{K} \alpha_{i,j} s_j^*, \quad \alpha_{i,j} = \frac{\exp(g_{i,j})}{\sum_r \exp(g_{i,r})}. \tag{21}$$

Consider the derivative with respect to anchor s_k^*. Since $g_{i,j}$ depends only on s_j^*, only $g_{i,k}$ varies with s_k^*. The softmax differential is

$$d\alpha_{i,j} = \alpha_{i,j}\big(\delta_{jk} - \alpha_{i,k}\big) dg_{i,k}. \tag{22}$$

Differentiating Eq. 21 gives

$$ds_i = \alpha_{i,k} ds_k^* + \sum_j s_j^* d\alpha_{i,j}. \tag{23}$$

Substituting Eq. 22,

$$\sum_j s_j^* d\alpha_{i,j} = \alpha_{i,k}\big(s_k^* - s_i\big) dg_{i,k}. \tag{24}$$

Because

$$dg_{i,k} = \big(\nabla_{s_k^*} g_{i,k}\big)^\top ds_k^*,$$

we obtain

$$\frac{\partial s_i}{\partial s_k^*} = \alpha_{i,k} I + \alpha_{i,k}\big(s_k^* - s_i\big)\big(\nabla_{s_k^*} g_{i,k}\big)^\top. \tag{25}$$

The two terms have complementary interpretations. The first term describes the direct contribution of anchor s_k^* to the aggregated scale, while the second captures its indirect effect through the routing distribution.

For positive scale elements,

$$g_{i,k} = -\frac{1}{d} \sum_{r=1}^{d} \frac{|h_{i,r}|}{s_{k,r}^*},$$

and hence

$$\nabla_{s_k^*} g_{i,k} = \frac{1}{d} \frac{|h_i|}{(s_k^*)^2}, \tag{26}$$

where division and squaring are element-wise.

Let $\hat{y}_i = Q(h_i, W; s_i)$ denote the quantized transformation in Eq. 13 of the main paper. Since rounding is non-differentiable, TRAQ uses the straight-through estimator (STE) during calibration to obtain surrogate gradients through the quantizers. The vector-Jacobian form of the anchor gradient is then

$$\nabla_{s_k^*} \mathcal{L}_{\text{TRAQ}} = \sum_m \frac{w_m}{|\Omega_m| d_o} \sum_{i \in \Omega_m} J_{s_i, s_k^*}^\top J_{\hat{y}_i, s_i}^\top \operatorname{sign}\big(\hat{y}_i - y_i\big), \tag{27}$$

where J_{s_i, s_k^*} is given by Eq. 25, and gradients through w_m are stopped as specified in Eq. 14 of the main paper.

Eq. 27 makes explicit that the optimized anchors receive reconstruction feedback through both scale interpolation and routing adaptation.

F DYNAMIC MODALITY REWEIGHTING AS SMOOTH WORST-MODALITY OPTIMIZATION

TRAQ assigns the modality calibration weight

$$w_m = \frac{\exp(E_m)}{\sum_j \exp(E_j)}. \tag{28}$$

When computing w_m, stop-gradient is applied to E_m.

This weighting rule can be interpreted through the log-sum-exp objective

$$F(E_t, E_v) = \log \left[\sum_{m \in \{t,v\}} \exp(E_m) \right]. \tag{29}$$

Its gradient with respect to model or quantization parameters θ is

$$\nabla_\theta F = \sum_m \frac{\exp(E_m)}{\sum_j \exp(E_j)} \nabla_\theta E_m = \sum_m w_m \nabla_\theta E_m. \tag{30}$$

Because w_m is treated as constant with respect to the model or quantization parameters during back-propagation, the gradient of the weighted calibration objective has exactly the form in Eq. 30. Thus, the proposed dynamic weighting can be viewed as following the gradient of a smooth approximation to the maximum modality error.

Indeed,

$$\max_m E_m \le F(E_t, E_v) \le \max_m E_m + \log 2. \tag{31}$$

Therefore, assigning larger weight to the currently higher-error modality is not merely heuristic: it approximates minimizing the worst modality reconstruction error while maintaining a smooth optimization objective.

The stop-gradient operation is important. Without it, differentiating

$$\sum_m w_m E_m$$

would introduce additional terms involving ∇w_m, changing the optimization dynamics. Stop-gradient preserves the intended error-dependent weighting while avoiding these second-order interactions between modality weights and reconstruction errors.

G CROSS-LAYER PROPAGATION OF QUANTIZATION ERROR

The layer-wise calibration strategy in Eq. 17 of the main paper is motivated by the fact that quantization perturbations introduced at an earlier layer alter the activation distributions observed by subsequent layers.

Let f_l denote the full-precision transformation of layer l, and let $\hat{f}_l$ denote its quantized counterpart. Define

$$\delta_l = \left\| \hat{h}^{(l)} - h^{(l)} \right\|$$

as the accumulated activation perturbation entering layer l, and

$$\epsilon_l = \left\| \hat{f}_l\left(\hat{h}^{(l)}\right) - f_l\left(\hat{h}^{(l)}\right) \right\|$$

as the local quantization perturbation introduced by that layer at the actual input to the quantized layer.

Assume f_l is locally L_l-Lipschitz over the calibration region. Then

$$\begin{aligned}\delta_{l+1} &= \left\|\hat{f}_l(\hat{h}^{(l)}) - f_l(h^{(l)})\right\| \\ &\leq \left\|\hat{f}_l(\hat{h}^{(l)}) - f_l(\hat{h}^{(l)})\right\| + \left\|f_l(\hat{h}^{(l)}) - f_l(h^{(l)})\right\| \\ &\leq \epsilon_l + L_l\delta_l.\end{aligned} \tag{32}$$

Assuming that the full-precision and quantized networks receive the same input, we have $\delta_0 = 0$. Recursively applying Eq. 32 yields

$$\delta_L \leq \sum_{r=0}^{L-1}\left(\prod_{q=r+1}^{L-1} L_q\right)\epsilon_r. \tag{33}$$

Hence, reconstruction errors introduced in early layers can influence later activation distributions and may be amplified by subsequent transformations. This explains why a scale anchor calibrated solely from the original full-precision activation distribution can become suboptimal after preceding layers have been quantized.

TRAQ therefore performs reconstruction-aware calibration layer by layer, allowing each layer to optimize its anchors under the activation distribution actually produced by previously quantized layers.

H ROUTING COMPLEXITY AND CALIBRATION PROCEDURE

For each token, compatibility evaluation in Eq. 10 of the main paper requires K channel-wise comparisons over a hidden dimension d, giving a routing complexity of $O(Kd)$.

The weighted scale aggregation in Eq. 12 of the main paper also requires $O(Kd)$ operations. The routing module therefore adds $O(Kd)$ computation per token and stores $O(Kd)$ anchor values per layer. In the vision-language setting considered in this work, $K = 2$, corresponding to the text and vision anchors.

Unlike an output-space mixture that would require evaluating multiple complete quantization paths and combining their outputs, TRAQ first aggregates the scale anchors and subsequently performs only one quantized transformation for each token. The complexity above refers specifically to the routing module; the actual end-to-end inference cost additionally depends on the implementation of the quantized transformation and is therefore evaluated empirically in Sec. 4.5 of the main paper.

The layer-wise calibration procedure can be summarized as follows. Given the modality-specific initial anchors, TRAQ evaluates token-anchor compatibility scores, obtains soft routing weights, aggregates the anchors into token-specific scales, and reconstructs the quantized outputs. Text and vision reconstruction errors are then computed from these outputs and converted into dynamic modality weights. The resulting weighted reconstruction objective is optimized with respect to the activation scale anchors and quantization parameters while keeping the original model weights frozen. The calibrated outputs of the current layer are propagated to the next layer so that subsequent anchors are optimized under the activation distributions induced by preceding quantization.

A compact algorithmic description is given below.

Algorithm 1 TRAQ Layer-wise Calibration

Input: $H^{(l)}, W^{(l)}, \{\Omega_m^{(l)}\}, \{s_{k,0}^{(l)}\}_{k=1}^{K}$

Initialize: $s_k^{*(l)} \leftarrow s_{k,0}^{(l)},\ k = 1, \ldots, K$

for each calibration step **do**

$g_{i,k}^{(l)} \leftarrow -\frac{1}{d}\left\| h_i^{(l)} \oslash s_k^{*(l)} \right\|_1$

$\alpha_{i,k}^{(l)} \leftarrow \mathrm{softmax}_k\big(g_{i,k}^{(l)}\big)$

$s_i^{(l)} \leftarrow \sum_k \alpha_{i,k}^{(l)} s_k^{*(l)}$

$\hat{y}_i^{(l)} \leftarrow [Q_A(h_i^{(l)} \oslash s_i^{(l)}) \odot s_i^{(l)}](\widetilde{W}_q^{(l)})^\top$

$E_m^{(l)} \leftarrow \mathrm{Avg}_{i \in \Omega_m^{(l)}} \ell\big(y_i^{(l)}, \hat{y}_i^{(l)}\big)$

$w_m^{(l)} \leftarrow \mathrm{softmax}_m\big(\mathrm{sg}(E_m^{(l)})\big)$

$\mathcal{L}_{\text{TRAQ}} \leftarrow \sum_m w_m^{(l)} E_m^{(l)}$

Update quantization parameters using STE gradients.

end for

Output: calibrated scale anchors and quantized layer outputs.

This procedure preserves the standard PTQ setting because the original pretrained model weights remain frozen throughout calibration.

I CUDA-STYLE IMPLEMENTATION OF FUSED RESCALING

We provide a CUDA-style implementation sketch to clarify how the token-specific rescaling in Eq. 13 of the main paper can be fused into activation dequantization without materializing an additional full-precision activation tensor. The purpose of this discussion is to demonstrate a feasible implementation path; the exact thread mapping, MMA datatype, and quantized linear backend are hardware- and implementation-dependent.

Let

$$z_i = h_i \oslash s_i \tag{34}$$

denote the normalized activation in Eq. 13 of the main paper. For the symmetric uniform quantizer defined in Eq. 3 of the main paper, we separate the integer code from its dequantization scale as

$$q_i^A = \mathrm{clip}\left(\left\lfloor z_i \oslash \Delta_i^A \right\rceil, q_{\min}, q_{\max}\right), \qquad Q_A(z_i) = q_i^A \odot \Delta_i^A, \tag{35}$$

where Δ_i^A denotes the activation quantization step with broadcasting determined by the adopted quantization granularity. For example, it may be shared per tensor, token, or group; the fusion described below does not depend on a particular choice.

Substituting this representation into Eq. 13 of the main paper gives

$$\begin{aligned} Q_A(h_i \oslash s_i) \odot s_i &= q_i^A \odot \Delta_i^A \odot s_i \\ &= q_i^A \odot \Delta_{i,\text{eff}}^A, \end{aligned} \qquad \Delta_{i,\text{eff}}^A = \Delta_i^A \odot s_i. \tag{36}$$

Therefore, the routed scale s_i can be absorbed into an effective activation dequantization scale. Importantly, q_i^A remains the low-bit integer activation code; only its dequantization scale is modified. The fixed quantized weight $\widetilde{W}_q = Q_W(W)$ in Eq. 13 of the main paper remains unchanged and is shared by all tokens.

Moreover, the complete token-specific scale matrix does not need to be explicitly stored. According to Eq. 12 of the main paper, for a token tile I and an input-channel tile J, the required scale tile can be reconstructed on-the-fly as

$$S_{I,J} = \sum_{k=1}^{K} \alpha_{I,k} s_{k,J}^*, \tag{37}$$

where the routing coefficients $\alpha_{I,k}$ are obtained from Eq. 11 of the main paper. Since K is small ($K = 2$ in our experiments), this operation consists only of a small number of element-wise multiply-accumulate operations.

Algorithm 2 CUDA-style sketch of fused token-specific rescaling

Require: Activation tile H, routing coefficients α, anchors $\{s_k^*\}_{k=1}^K$, fixed quantized weight $\widetilde{W}_q$
Ensure: Output tile Y

```
Initialize accumulator Y ← 0
for each token tile I and input-channel tile J do
    S_{I,J} ← Σ_{k=1}^{K} α_{I,k} s*_{k,J}                ▷ Eq. 12 of the main paper
    Z_{I,J} ← H_{I,J} ⊘ S_{I,J}
    (q^A_{I,J}, Δ^A_{I,J}) ← ActivationQuantize(Z_{I,J})
    Δ^{A,eff}_{I,J} ← Δ^A_{I,J} ⊙ S_{I,J}
    A^{reg}_{I,J} ← cast(q^A_{I,J}) ⊙ Δ^{A,eff}_{I,J}      ▷ fused dequantization
    Load the corresponding tile of the fixed W̃_q
    Y ← Y + LinearTile(A^{reg}_{I,J}, W̃_{q,J})
end for
return Y
```

Algorithm 2 sketches one possible tiled implementation. ACTIVATIONQUANTIZE denotes the activation quantizer used in the main paper and returns both the integer code and its associated quantization step. The computation of the quantization step follows the granularity of the underlying quantizer and is orthogonal to the proposed fusion.

Here, $A_{I,J}^{\mathrm{reg}}$ denotes a transient dequantized activation tile held in registers or other on-chip storage and directly consumed by the subsequent linear computation. It is not written back as an additional full-precision activation tensor. Similarly, $S_{I,J}$ can be generated from the shared anchors and the token routing coefficients when each tile is processed, avoiding the need to materialize an $N \times d$ token-scale matrix.

This implementation preserves the single-path structure of TRAQ: the anchors are first aggregated into one routed scale for each token, after which only one linear transformation is evaluated. It therefore requires neither token-specific copies of the weight matrix nor multiple expert-style linear paths. The handling of the fixed quantized weight inside LINEARTILE can follow the dequantization or mixed-precision path of the underlying quantized linear backend and is independent of token routing. The exact end-to-end cost is backend-dependent and is therefore evaluated empirically in Sec. 4.5 of the main paper.