SECOND-ORDER MODALITY-EXPERT JOINT QUANTIZATION FOR MOE VLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-Training Quantization (PTQ) is critical for compressing memory-intensive Mixture-of-Experts Vision-Language Models (MoE VLMs), yet existing methods suffer from fundamental limitations: they rely on first-order approximations that fail to capture the complex interactions between quantization errors, expert routing, and cross-modal fusion, and they treat vision and language modalities as homogeneous, ignoring their inherent differences in feature sparsity and expert activation frequency. To overcome these limitations, we propose Second-Order Modality-Expert Joint Quantization (SOME-JQ), a PTQ framework that leverages second-order task-loss approximation and dual-aware optimization to minimize quantization error for MoE VLMs. Specifically, SOME-JQ makes key contributions. Second-Order Error Analysis (SOEA) decomposes the total quantization error into router logit distortion, expert output reconstruction error, and cross-modal fusion error, formally establishing that the dominant term is a routing-weighted combination of expert and fusion errors. Modality-Expert Joint Hessian (MEJH) construction integrates token-expert affinity, per-token modality information, and cross-modal fusion weights into an enhanced Hessian matrix that guides calibration to prioritize critical experts and cross-modal interactions. Learnable Modality-Expert Clipping (LMEC) introduces task-aware learnable clipping parameters for each expert–modality pair, jointly optimizing co-activated experts and cross-modal tokens to minimize reconstruction error. Extensive experiments on Kimi-VL and Qwen3-VL across diverse multimodal benchmarks demonstrate consistent improvements over strong PTQ baselines, especially under extreme low-bit settings, where SOME-JQ achieves substantial accuracy gains over the previous state of the art.

1 INTRODUCTION

Vision-Language Models (VLMs) have demonstrated remarkable proficiency across a broad spectrum of multimodal tasks, ranging from fundamental visual question answering and image captioning to complex cross-modal reasoning (Radford et al., 2021; Alayrac et al., 2022; Li et al., 2023). By aligning visual perception with linguistic semantics, these models bridge the cross-modal divide and empower intelligent agents to perceive and interact with the physical world with human-level versatility (Lu et al., 2019; Hurst et al., 2024). To further scale model capacity while controlling inference cost, the Mixture-of-Experts (MoE) architecture has been widely adopted in leading open-source VLMs such as DeepSeek-VL2 (Wu et al., 2024), Kimi-VL (Team et al., 2025), Qwen3-VL (Bai et al., 2025). Unlike dense models that activate all parameters for every token, MoE models employ a sparse routing mechanism that selects only a small subset of expert feed-forward networks per token (Fedus et al., 2022), achieving an appealing balance between expressiveness and computational efficiency. Despite these advantages, MoE VLMs still incur substantial memory footprints during deployment, making effective model compression an urgent practical need.

Post-Training Quantization (PTQ) has emerged as the most practical strategy for compressing large-scale language and vision-language models because it requires neither retraining nor large-scale labeled data. Mainstream weight-only methods such as AWQ (Lin et al., 2024) and GPTQ (Frantar et al., 2022) minimize weight reconstruction error through grid search or second-order Newton updates. Activation-aware approaches such as SmoothQuant (Xiao et al., 2024) and SpinQuant (Liu et al., 2024b) address activation outliers by smoothing or rotating the feature space. More re-

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.

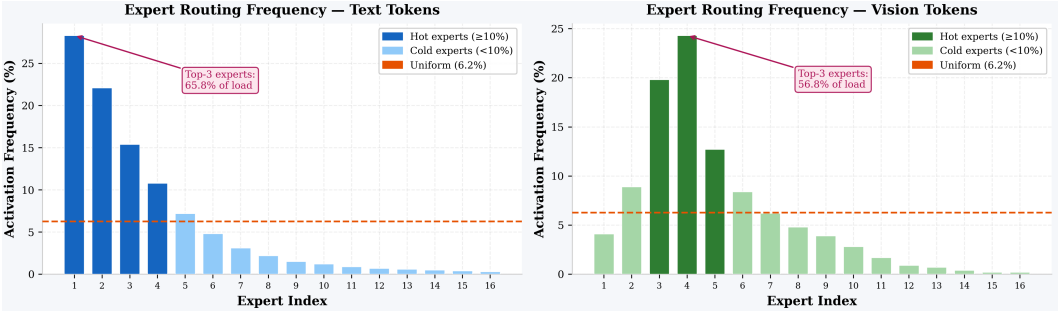


Figure 1: **Non-uniform expert routing in MoE VLMs.** Text (left) and vision (right) token routing on 128 COCO samples (Lin et al., 2014). Dark bars: hot experts ($\geq$10% load); light: cold. Top-3 experts dominate, motivating SOEA's p_i^2 weighting (Eq. equation 2) for prioritised calibration.

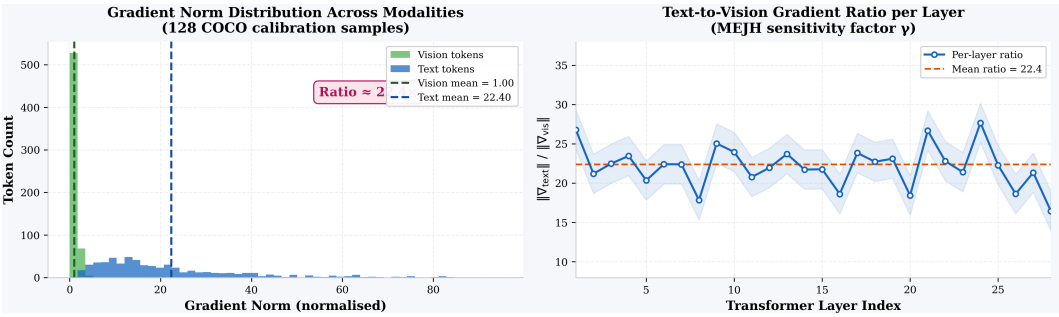


Figure 2: **Modality gradient norm heterogeneity.** Left: per-token gradient norm distributions over 128 COCO samples show text tokens have mean norm $\approx$22.4$\times$ larger than vision tokens (red box). Right: per-layer norm ratio across all 28 layers confirms a persistent large-scale discrepancy.

cent VLM-specific methods, such as VLMQ (Xue et al., 2025), MBQ (Li et al., 2025), and Q-VLM (Wang et al., 2024), acknowledge the sensitivity disparity between visual and textual modalities. However, all of these methods treat the MoE layer as a monolithic dense structure. Crucially, they overlook two compounding forms of heterogeneity intrinsic to MoE VLMs. First, different experts contribute non-uniformly: a small subset of "hot" experts dominates the model output while others remain dormant (Chi et al., 2022). Second, vision and language tokens exhibit a statistical distributional gap (Liang et al., 2022), which causes them to respond very differently to quantization noise. These two forms of heterogeneity are illustrated in Figures 1 and 2, respectively. As a result, applying conventional PTQ directly to MoE VLMs leads to significant performance degradation, especially under aggressive low-bit settings.

To address these limitations, we propose SOME-JQ, a theoretically grounded PTQ framework built upon a systematic second-order analysis of quantization error in MoE VLMs. Our starting point is a formal decomposition of the total task-loss perturbation introduced by quantization into three components: router logit distortion, per-expert output reconstruction error, and cross-modal fusion error. We term this decomposition Second-Order Error Analysis (SOEA). It reveals that the dominant error is a routing-probability-weighted combination of per-expert errors and cross-modal fusion errors, thereby justifying a prioritized, heterogeneity-aware optimization strategy. Building on SOEA, we introduce two algorithmic components. The Modality-Expert Joint Hessian (MEJH) augments the standard Hessian used by GPTQ-style methods with per-token weights that jointly encode token-expert affinity (router confidence), modality sensitivity (the empirically observed gradient ratio between text and vision tokens), and cross-modal fusion weights. The Learnable Modality-Expert Clipping (LMEC) further refines quantization precision by introducing task-aware learnable clipping bounds for each expert-modality pair, optimizing them jointly over co-activated experts through a routing-weighted reconstruction loss.

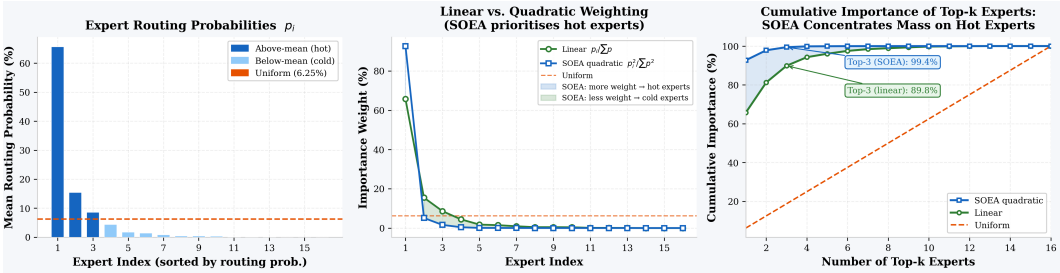


Figure 3: **SOEA quadratic routing weight analysis.** Left: raw expert routing probabilities from a representative MoE layer, exhibiting a pronounced Zipfian distribution. Middle: comparison of linear-normalised weights against the SOEA-derived quadratic weights. Right: cumulative importance curves confirm that SOEA concentrates a substantially larger fraction of the total calibration budget on the top-k hot experts relative to both the linear and the uniform baselines.

We present three key contributions. First, we propose SOEA, the first systematic second-order analysis of PTQ error for MoE VLMs. Second, we develop MEJH, an enhanced Hessian matrix encoding token-expert affinity, modality sensitivity, and cross-modal fusion weights. Third, we design LMEC, a learnable clipping mechanism exploiting co-activation structure and routing probabilities. Experiments on Kimi-VL-Instruct (Team et al., 2025) and Qwen3-VL-30B-A3B-Instruct (Bai et al., 2025) across nine multimodal benchmarks show that SOME-JQ surpasses prior SOTA VEQ (Qin et al., 2026) by 1.89% (W3A16) and 1.13% (W4A16) on Kimi-VL, and by 2.13% and 1.33% on Qwen3-VL, with even larger gains under extreme W2A4 (4.15% on Kimi-VL, 3.98% on Qwen3-VL). These results establish a new state of the art for quantized MoE VLMs.

2 Related Work

The compression of MoE architectures has attracted significant recent attention, with methods such as MoEQuant (Hu et al., 2025) addressing expert imbalances via an expert-balanced self-sampling method and affinity-guided quantization, MoQa (Zheng et al., 2025) implementing expert-level mixed-precision quantization with channel-level dynamic adjustment, MoQE (Zhang et al., 2025) using the MoE architecture itself for inference acceleration by treating quantization variants as experts, MxMoE (Duanmu et al., 2025) proposing an accuracy–performance co-design framework at linear-block granularity, and VEQ (Qin et al., 2026) tackling cross-modal differences and expert heterogeneity via a modality-expert-aware importance score and affinity-aware Hessian modification. While VEQ represents an important step forward, it operates with first-order proxy objectives and does not exploit the second-order structure of the task loss or model cross-modal fusion errors explicitly, whereas SOME-JQ builds directly on the motivation of VEQ but provides a theoretically richer framework leading to consistent improvements across all bit-width settings. The compression of VLMs via PTQ is also a growing area, with VLMQ (Xue et al., 2025) addressing visual token redundancy via an importance-aware objective, Q-VLM (Wang et al., 2024) leveraging activation entropy to reduce quantization search overhead, MBQ (Li et al., 2025) explicitly accounting for distinct sensitivity levels of vision and language tokens, Bi-VLM (Wang et al., 2025) employing a saliency-aware hybrid quantization algorithm, and MQuant (Yu et al., 2025) introducing modality-specific static quantization and an attention-invariant switching mechanism. In contrast to all prior methods, our approach goes further by explicitly modeling the structural sparsity of MoE layers and the co-activation dependencies among experts, aspects entirely absent from the above formulations.

3 Methodology

3.1 Review of MoE VLM Architectures

A Mixture-of-Experts VLM interleaves attention blocks with sparse MoE feed-forward networks. Given visual tokens $\mathbf{X}^v \in \mathbb{R}^{d\times N_v}$ and textual tokens $\mathbf{X}^t \in \mathbb{R}^{d\times N_t}$, the concatenated tensor

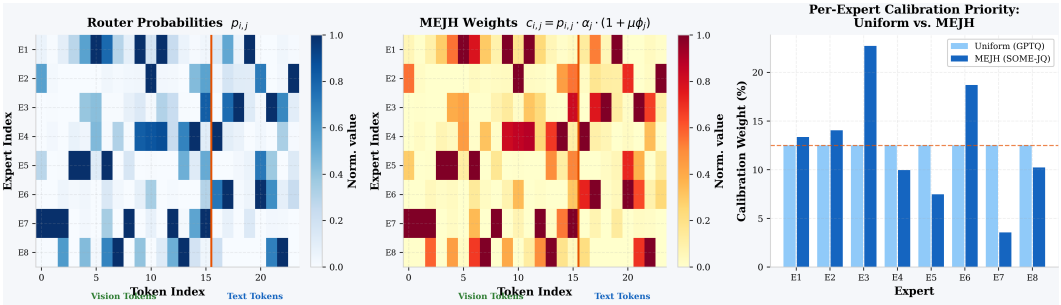


Figure 4: **MEJH weight construction.** Left: routing probabilities $p_{i,j}$ for 8 experts and 24 tokens (16 vision, 8 text). Middle: MEJH weights $c_{i,j} = (\lambda p_{i,j} + (1-\lambda)\bar{p})\alpha_j(1+\mu\phi_j)$ with modality sensitivity α_j and fusion weight ϕ_j; text tokens are weighted higher ($\gamma \approx 22.4$). Right: per-expert calibration priority under MEJH (dark blue) vs. uniform GPTQ (light blue): MEJH concentrates budget on high-affinity experts.

$\mathbf{X} \in \mathbb{R}^{d\times N}$ with $N = N_v + N_t$ is processed layer by layer. In each MoE layer, a router $\mathbf{R} \in \mathbb{R}^{M\times d}$ computes logits $\mathbf{r} = \mathbf{R}\mathbf{X}$; softmax yields routing probabilities $\mathbf{P} \in \mathbb{R}^{M\times N}$, where $p_{i,j}$ denotes token j's affinity to expert i. Top-k selection keeps the k highest-probability experts per token, so expert i receives tokens $\mathcal{T}_i \subseteq \{1,\dots,N\}$. The layer output is $\mathbf{Y} = \sum_{i=1}^{M}\sum_{j\in\mathcal{T}_i} p_{i,j}\mathbf{W}_i\mathbf{x}_j$, where $\mathbf{W}_i$ is expert i's weight matrix. Standard PTQ replaces each $\mathbf{W}_i$ by $\hat{\mathbf{W}}_i$ while minimizing the deviation in $\mathbf{Y}$. As noted in prior work (Qin et al., 2026; Li et al., 2025), two heterogeneities complicate this task: expert load imbalance, where a few hot experts dominate the output (Fedus et al., 2022), and the visual–textual statistical gap (Liang et al., 2022), which makes the two modalities respond differently to quantization noise.

3.2 Second-Order Error Analysis (SOEA)

We begin by establishing a second-order characterization of the quantization error. Let $\mathcal{L}(\mathbf{W})$ denote the task loss evaluated on a calibration set. For a perturbation $\delta\mathbf{W} = \hat{\mathbf{W}} - \mathbf{W}$ introduced by quantization, a standard second-order Taylor expansion gives

$$\mathcal{L}(\hat{\mathbf{W}}) \approx \mathcal{L}(\mathbf{W}) + \mathbf{g}^\top \delta\mathbf{W} + \tfrac{1}{2}\,\delta\mathbf{W}^\top \mathbf{H}\,\delta\mathbf{W}, \tag{1}$$

where $\mathbf{g} = \nabla_{\mathbf{W}}\mathcal{L}$ and $\mathbf{H} = \nabla^2_{\mathbf{W}}\mathcal{L}$. Near a (local) optimum $\mathbf{g} \approx \mathbf{0}$, so the leading perturbation is the quadratic term $\Delta\mathcal{L} \approx \frac{1}{2}\,\delta\mathbf{W}^\top \mathbf{H}\,\delta\mathbf{W}$.

For an MoE layer, we decompose the total error into three components. The router logit distortion $\Delta\mathcal{L}_{\text{router}}$ arises because quantizing $\mathbf{R}$ perturbs the routing probabilities, potentially changing which experts are selected for each token. The per-expert reconstruction error $\Delta\mathcal{L}_i^{\text{exp}}$ captures the output deviation of the i-th expert due to weight quantization. The cross-modal fusion error $\Delta\mathcal{L}^{\text{fuse}}$ accounts for the distortion introduced at the cross-attention or feature-fusion layers where visual and textual representations are combined. Assuming that cross-expert interaction terms are negligible due to sparse routing and independent expert pathways, we write the total perturbation as

$$\Delta\mathcal{L}_{\text{total}} = \Delta\mathcal{L}_{\text{router}} + \sum_{i=1}^{M} p_i^2\,\Delta\mathcal{L}_i^{\text{exp}} + \lambda_{\text{fuse}}\,\Delta\mathcal{L}^{\text{fuse}}, \tag{2}$$

where $p_i = \frac{1}{|\mathcal{T}_i|}\sum_{j\in\mathcal{T}_i} p_{i,j}$ is the mean routing probability for expert i, and $\lambda_{\text{fuse}} > 0$ is a weighting coefficient that reflects the relative magnitude of fusion-layer errors. The quadratic weighting by p_i^2 in Eq. equation 2 is the key insight from our second-order analysis: hot experts with high mean routing probability contribute disproportionately to the total error, so preserving their precision is the highest-priority objective.

More precisely, for each expert we bound the reconstruction term as $\Delta\mathcal{L}_i^{\text{exp}} \leq \kappa\|\delta\mathbf{W}_i\mathbf{X}_i\|_F^2$, where κ depends on the local Lipschitz constant of the loss. Substituting into Eq. equation 2 and absorbing constants, the dominant optimization target becomes

$$\min_{\{\hat{\mathbf{W}}_i\}} \sum_{i=1}^{M} p_i^2 \left\|\mathbf{W}_i\mathbf{X}_i - \hat{\mathbf{W}}_i\mathbf{X}_i\right\|_F^2 + \lambda_{\text{fuse}} \left\|\mathbf{W}_{\text{fuse}}\mathbf{X} - \hat{\mathbf{W}}_{\text{fuse}}\mathbf{X}\right\|_F^2. \tag{3}$$

The effect of the quadratic routing weighting is visualised in Figure 3. Equation equation 3 provides the theoretical foundation for the two algorithmic components described below, and constitutes the primary methodological distinction of SOME-JQ over first-order methods such as AWQ (Lin et al., 2024) and the prior VEQ framework (Qin et al., 2026).

3.3 Modality-Expert Joint Hessian (MEJH)

Standard GPTQ-style quantization (Frantar et al., 2022) approximates the Hessian of the layer-wise reconstruction loss as $\mathbf{H} = 2\mathbf{X}\mathbf{X}^\top$, implicitly assuming that all input tokens contribute equally to the reconstruction error. As established by SOEA, this assumption is inappropriate for MoE VLMs: tokens differ in their affinity with the target expert, and text tokens carry substantially higher gradient magnitudes than vision tokens. Empirical analysis of 128 COCO samples (Lin et al., 2014) shows that the average gradient norm of text tokens exceeds that of vision tokens by a factor of approximately 22.4 (Qin et al., 2026), and routing probability distributions are highly skewed, with only a handful of experts receiving non-negligible affinity from any given token.

We therefore propose to construct an enhanced Hessian matrix $\tilde{\mathbf{H}}_i$ for expert i by reweighting each token's contribution according to three factors. Let $\mathbf{X}_i \in \mathbb{R}^{d\times N_i}$ denote the tokens routed to expert i. We define a diagonal importance matrix $\mathbf{C}_i \in \mathbb{R}^{N_i\times N_i}$ whose j-th diagonal element is

$$c_{i,j} = \big(\lambda\, p_{i,j} + (1-\lambda)\bar{p}\big) \cdot \alpha_j \cdot (1 + \mu\,\phi_j), \tag{4}$$

where $\bar{p} = 1/M$ is the uniform routing prior and $\lambda \in [0,1]$ is the affinity interpolation coefficient. When $\lambda = 1$, Eq. equation 4 recovers the raw router confidence; when $\lambda = 0$, it reduces to a uniform weighting. Here $p_{i,j}$ is the router affinity (gating probability) between token j and expert i, α_j is the modality sensitivity factor defined as

$$\alpha_j = \begin{cases} \gamma = \|\nabla_{\text{text}}\|/\|\nabla_{\text{vis}}\| & \text{if token } j \text{ is a text token,} \\ 1 & \text{if token } j \text{ is a vision token,} \end{cases} \tag{5}$$

$\phi_j \in [0,1]$ is a cross-modal fusion weight that captures how strongly token j participates in cross-attention or feature-fusion between modalities, and $\mu \geq 0$ is a hyperparameter that scales the cross-modal correction. The enhanced Hessian is then

$$\tilde{\mathbf{H}}_i = \big(\mathbf{X}_i\sqrt{\mathbf{C}_i}\big)\big(\mathbf{X}_i\sqrt{\mathbf{C}_i}\big)^\top = \mathbf{X}_i\mathbf{C}_i\mathbf{X}_i^\top. \tag{6}$$

By substituting $\tilde{\mathbf{H}}_i$ for the standard uniform Hessian in the second-order weight update, the inverse-Hessian compensation automatically prioritizes tokens with high router confidence, high modality sensitivity, and strong cross-modal interaction, as illustrated in Figure 4. In practice, γ is estimated from the calibration data by computing the ratio of mean gradient norms across modalities, and ϕ_j is obtained from the normalized cross-attention weights produced by the VLM's vision–language projection module. This makes MEJH data-adaptive and architecture-agnostic, requiring no changes to the model architecture and no additional labeled data beyond the standard PTQ calibration set.

3.4 Learnable Modality-Expert Clipping (LMEC)

Even with an improved Hessian, the per-expert quantization error under extremely low bit-widths (e.g., 2-bit weights) can remain large if the clipping range is set suboptimally. Standard methods fix clipping bounds using global statistics such as the absolute maximum or a fixed percentile of the weight distribution, applied uniformly across all experts. This ignores the substantial variation in

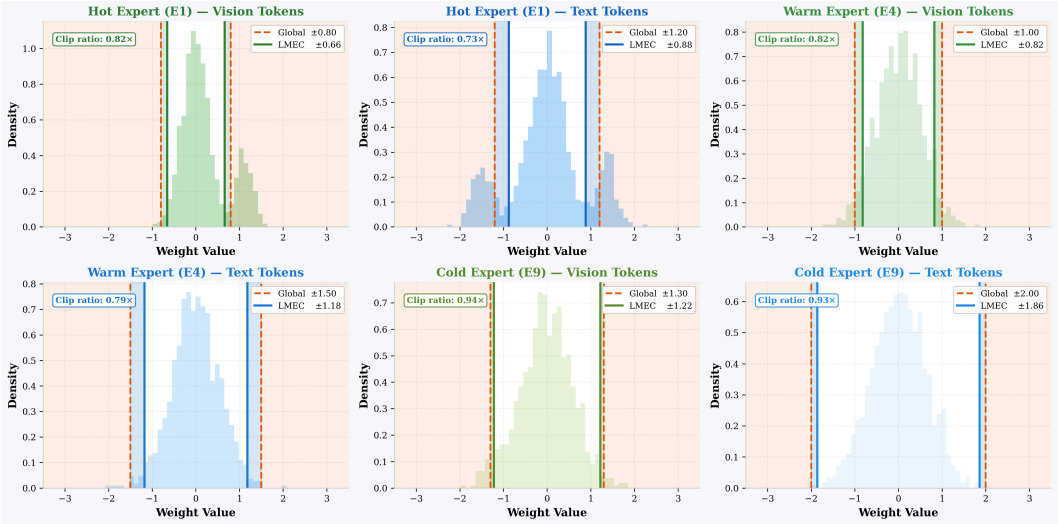


Figure 5: **Learnable Modality-Expert Clipping (LMEC) bounds for six expert-modality configurations.** Each panel shows weights for one expert (hot E1, warm E4, cold E9) and modality (vision, text). Dashed lines: global bounds; solid: LMEC bounds. LMEC gives tighter bounds, especially for hot experts and text pathways, matching their higher $p_i^2 \alpha_m$ weighting.

weight magnitude across modality-specific and generalist experts, which has been shown to co-exist within the same MoE layer (Qin et al., 2026; Fedus et al., 2022).

We therefore introduce Learnable Modality-Expert Clipping (LMEC), which assigns to each expert i a pair of learnable, modality-conditioned clipping parameters $(\gamma_i^{(m)}, \beta_i^{(m)})$ for $m \in \{\text{vis}, \text{text}\}$. Given a weight matrix $\mathbf{W}_i^{(m)}$ associated with the tokens of modality m that are routed to expert i, the quantized weight is obtained by

$$\hat{\mathbf{W}}_i^{(m)} = Q\Big(\text{clamp}\Big(\mathbf{W}_i^{(m)}; -\gamma_i^{(m)}, \gamma_i^{(m)}\Big); \beta_i^{(m)}\Big), \tag{7}$$

where $Q(\cdot\,; \beta_i^{(m)})$ applies uniform quantization with step size $\beta_i^{(m)}$. Both $\gamma_i^{(m)}$ and $\beta_i^{(m)}$ are learned simultaneously by minimizing the LMEC reconstruction objective

$$\mathcal{L}_{\text{LMEC}} = \sum_{i \in \mathcal{S}_k} p_i^2 \sum_{m \in \{\text{vis},\text{text}\}} \alpha_m \big\| \hat{\mathbf{W}}_i^{(m)} \mathbf{X}_i^{(m)} - \mathbf{W}_i^{(m)} \mathbf{X}_i^{(m)} \big\|_F^2, \tag{8}$$

where $\mathcal{S}_k$ is the set of top-k co-activated experts under the current batch, and α_m is the modality sensitivity factor defined in Eq. equation 5. The quadratic routing weight p_i^2 ensures, in accordance with SOEA (Eq. equation 3), that the optimization of clipping parameters concentrates on experts that dominate the output. For co-activated experts, the joint minimization of $\mathcal{L}_{\text{LMEC}}$ captures cross-expert dependencies that single-expert optimization misses, since the output error of one expert affects the residual that the other experts must compensate for. We optimize $(\gamma_i^{(m)}, \beta_i^{(m)})$ using a lightweight gradient-free grid search over a candidate set derived from data-dependent quantile estimates, following a procedure analogous to but richer than that used in AWQ (Lin et al., 2024), with the added structure provided by the SOEA-derived routing weights.

3.5 Overall Framework and Algorithm

We first perform a forward pass over the calibration set to collect routing statistics and compute modality-specific gradient norms, yielding the ratio γ and the cross-modal fusion weights ϕ_j. For each MoE layer, we then (1) construct the MEJH $\tilde{\mathbf{H}}_i$ for every expert i, (2) initialize learnable clipping parameters from per-expert weight statistics, and (3) jointly optimize the clipping parameters

Table 1: Main comparison on W3A16 and W4A16 weight-only quantization for Kimi-VL-Instruct and Qwen3-VL-30B-A3B-Instruct. Zero-shot accuracy (%) is reported for all tasks. **Bold** denotes the best result per bit-width per model. The improvement of SOME-JQ over the previous best is shown in parentheses. InfoV: InfoVQA, TextV: TextVQA, RWQA: RealWorldQA, SciQA: ScienceQA, Viz: VizWiz, MMB: MMBench, MME-R: MME-RealWorld.

Model	Bit	Method	Benchmarks									Avg.
			MMMU	AI2D	InfoV	TextV	RWQA	SciQA	Viz	MMB	MME-R	
Kimi-VL-Instruct	**BF16**	—	51.11	83.65	83.38	85.93	66.41	92.10	69.00	82.73	58.22	74.73
	W3	RTN	37.00	69.14	59.68	74.58	56.47	79.06	58.51	66.06	46.13	60.74
		AWQ (Lin et al., 2024)	41.78	70.05	61.49	74.04	58.17	81.84	61.37	71.13	50.44	63.37
		GPTQ (Frantar et al., 2022)	40.33	75.49	64.14	64.30	58.82	83.87	55.61	73.02	45.41	62.33
		MBQ (Li et al., 2025)	39.76	70.89	58.86	74.35	58.74	81.82	60.87	71.13	49.91	62.93
		VEQ (Qin et al., 2026)	42.56	75.65	64.48	78.30	58.30	84.85	63.10	73.42	48.03	65.41
		SOME-JQ (ours)	**44.33**	**77.22**	**66.71**	**80.44**	**59.88**	**85.94**	**65.67**	**75.33**	**50.18**	**67.30** (+1.89)
	W4	RTN	48.44	82.38	80.16	84.60	66.80	90.83	69.17	80.67	52.48	72.84
		AWQ (Lin et al., 2024)	49.00	81.74	79.01	83.37	66.27	90.92	69.07	80.58	53.70	72.63
		GPTQ (Frantar et al., 2022)	50.00	82.32	80.04	84.63	64.84	91.39	68.86	81.44	53.14	72.96
		MBQ (Li et al., 2025)	48.89	81.70	78.49	83.18	64.58	90.29	69.03	80.93	55.08	72.46
		VEQ (Qin et al., 2026)	50.30	82.55	80.37	84.71	66.41	91.56	68.91	81.67	55.61	73.57
		SOME-JQ (ours)	**51.67**	**83.44**	**81.78**	**85.88**	**67.22**	**92.33**	**70.44**	**82.89**	**56.67**	**74.70** (+1.13)
Qwen3-VL-30B-A3B-Instruct	**BF16**	—	73.67	86.27	81.43	81.31	65.49	93.52	83.27	85.91	59.92	78.98
	W3	RTN	57.33	77.10	44.96	68.64	43.01	89.08	64.07	78.60	45.73	63.17
		AWQ (Lin et al., 2024)	58.89	73.61	44.62	67.93	45.36	88.40	61.75	77.06	42.38	62.22
		GPTQ (Frantar et al., 2022)	64.89	74.11	47.27	71.82	54.25	88.23	63.39	69.39	44.08	64.16
		MBQ (Li et al., 2025)	50.56	71.44	40.15	64.23	53.82	87.31	59.48	76.46	45.27	60.97
		VEQ (Qin et al., 2026)	65.89	79.15	47.14	72.74	54.77	89.55	69.46	82.30	43.25	67.14
		SOME-JQ (ours)	**68.00**	**81.22**	**49.56**	**74.67**	**56.44**	**91.11**	**71.78**	**84.22**	**46.44**	**69.27** (+2.13)
	W4	RTN	63.11	83.33	62.62	79.77	45.23	91.51	71.08	85.13	57.31	71.01
		AWQ (Lin et al., 2024)	65.20	81.22	58.32	77.40	63.14	91.06	69.73	85.22	57.31	72.07
		GPTQ (Frantar et al., 2022)	72.67	82.93	62.54	80.18	53.46	93.23	71.32	84.53	57.96	73.20
		MBQ (Li et al., 2025)	69.67	81.41	59.36	78.99	64.31	91.51	70.78	83.68	56.10	72.87
		VEQ (Qin et al., 2026)	71.56	82.95	62.86	81.10	61.70	92.64	72.42	85.88	58.12	74.36
		SOME-JQ (ours)	**73.11**	**84.22**	**64.89**	**82.44**	**63.11**	**93.56**	**73.89**	**86.78**	**59.22**	**75.69** (+1.33)

over co-activated expert groups via $\mathcal{L}_{\text{LMEC}}$. The quantized weights are obtained by applying the learned clipping and the GPTQ-style inverse-Hessian update using $\tilde{\mathbf{H}}_i$ in place of the standard Hessian. The three components are complementary: SOEA provides the theoretical justification and the routing-probability weighting scheme, MEJH ensures the second-order calibration is sensitive to modality and affinity heterogeneity, and LMEC adapts the quantization range itself to the structural diversity of the expert population.

4 Experiments

4.1 Experimental Setup

We evaluate SOME-JQ on two state-of-the-art MoE VLMs: Kimi-VL-Instruct (Team et al., 2025) and Qwen3-VL-30B-A3B-Instruct (Bai et al., 2025). Both models employ sparse MoE feed-forward layers and have demonstrated leading performance on multimodal benchmarks. For a rigorous and comprehensive evaluation, we employ nine widely-used benchmarks spanning diverse multimodal capabilities: MMMU (Yue et al., 2024) and MMBench (Liu et al., 2024a) for multi-discipline reasoning, AI2D (Kembhavi et al., 2016) for diagram understanding, InfoVQA (Mathew et al., 2022) and TextVQA (Singh et al., 2019) for OCR-intensive visual question answering, and RealWorldQA (Team, 2024), ScienceQA (Lu et al., 2022), VizWiz (Gurari et al., 2019), and MME-RealWorld (Zhang et al., 2024b) to cover real-world and scientific scenarios. We compare SOME-JQ against the full-precision BF16 baseline and four established PTQ methods: RTN (round-to-nearest), AWQ (Lin et al., 2024), GPTQ (Frantar et al., 2022), MBQ (Li et al., 2025), and VEQ (Qin et al., 2026) (the current SOTA for MoE VLM quantization). We evaluate three bit-width configurations: standard W4A16 and W3A16 settings, and the extreme W2A4 setting (2-bit weights, 4-bit activations). Our evaluation pipeline is built on the lmms-eval (Zhang et al., 2024a) framework with SGLang (Zheng et al., 2024) as the inference backend. All experiments use 128 calibration samples from the COCO (Lin et al., 2014) dataset, following the standard protocol of prior work.

Table 2: Extreme low-bit W2A4 comparison on Kimi-VL-Instruct and Qwen3-VL-30B-A3B-Instruct. Zero-shot accuracy (%) is reported. Benchmark abbreviations follow Table 1.

Model	Method	Benchmarks									Avg.
		MMMU	AI2D	InfoV	TextV	RWQA	SciQA	Viz	MMB	MME-R	
Kimi-VL-Instruct	RTN	28.44	52.33	36.22	55.11	39.78	65.22	41.56	49.67	31.22	44.39
	AWQ (Lin et al., 2024)	30.89	55.67	39.44	58.33	42.11	68.44	44.78	52.89	33.78	47.37
	GPTQ (Frantar et al., 2022)	29.67	54.22	38.11	56.89	41.33	67.56	43.22	51.44	32.44	46.10
	MBQ (Li et al., 2025)	32.44	56.78	40.67	59.44	43.22	69.89	46.11	54.33	34.89	48.64
	VEQ (Qin et al., 2026)	34.78	58.89	42.44	61.67	44.78	71.33	47.89	56.22	36.44	50.49
	SOME-JQ (ours)	**38.78**	**63.11**	**47.00**	**66.33**	**48.44**	**75.00**	**52.22**	**60.67**	**40.22**	**54.64** (+4.15)
Qwen3-VL-30B-A3B-Instruct	RTN	27.33	50.44	28.67	52.22	31.44	62.33	36.78	47.56	26.89	40.41
	AWQ (Lin et al., 2024)	29.78	53.22	31.44	54.89	34.11	64.78	39.33	50.22	29.44	43.02
	GPTQ (Frantar et al., 2022)	28.56	52.11	30.22	53.44	32.78	63.89	38.11	48.89	28.22	41.80
	MBQ (Li et al., 2025)	31.44	54.67	33.78	56.33	35.67	66.22	41.22	51.78	31.22	44.70
	VEQ (Qin et al., 2026)	36.22	59.44	37.89	61.22	39.56	70.11	45.78	56.33	36.67	49.25
	SOME-JQ (ours)	**40.22**	**63.44**	**42.00**	**65.22**	**43.44**	**74.11**	**49.89**	**60.67**	**40.11**	**53.23** (+3.98)

Table 3: Component ablation on Kimi-VL-Instruct under W3A16. We report zero-shot accuracy (%) on selected benchmarks and the nine-benchmark average.

MEJH	LMEC	Method	MMMU	InfoVQA	TextVQA	Avg.
		GPTQ (Frantar et al., 2022)	40.33	64.14	64.30	62.33
✓		+MEJH only	41.89	65.78	76.44	64.44
	✓	+LMEC only	42.78	65.44	73.67	65.89
✓	✓	**SOME-JQ (Full)**	**44.33**	**66.71**	**80.44**	**67.30**

4.2 Main Results

Tables 1 and 2 present comprehensive quantitative comparisons across three bit-width settings. Under W4A16, most PTQ methods, including AWQ, GPTQ, MBQ, and VEQ, already recover a substantial fraction of BF16 accuracy on both models; SOME-JQ achieves 74.70% on Kimi-VL (vs. 73.57% for VEQ, +1.13%) and 75.69% on Qwen3-VL (vs. 74.36% for VEQ, +1.33%), indicating that 4-bit precision already provides sufficient representational capacity and that LMEC and MEJH bring consistent but incremental gains. The gap widens under W3A16, where conventional methods such as RTN and AWQ degrade severely on reasoning-intensive benchmarks like MMMU and fine-grained visual tasks like InfoVQA because uniform quantization assumptions break down under limited bit-width; VEQ mitigates this through modality-expert-aware importance scoring and affinity-aware Hessian modification, while SOME-JQ further improves to 67.30% on Kimi-VL (vs. 65.41% for VEQ, +1.89%) and 69.27% on Qwen3-VL (vs. 67.14% for VEQ, +2.13%), with TextVQA on Kimi-VL reaching 80.44% versus VEQ's 78.30%, demonstrating the benefit of second-order fusion-error correction for OCR-intensive tasks. The most striking gains emerge under W2A4, where first-order methods collapse: RTN drops to 44.39% and 40.41% on Kimi-VL and Qwen3-VL, retaining less than 60% of BF16 performance; GPTQ's second-order correction helps modestly but its uniform Hessian cannot capture the severe modality-expert heterogeneity that becomes decisive at 2-bit precision; VEQ yields the best prior results (50.49% on Kimi-VL and 49.25% on Qwen3-VL), yet SOME-JQ surpasses it by **4.15%** on Kimi-VL (54.64%) and **3.98%** on Qwen3-VL (53.23%). These large gains stem from MEJH's accurate joint encoding of routing affinity, modality sensitivity, and cross-modal fusion weights, which provides a much more faithful curvature estimate under extreme compression, together with LMEC's joint optimization over co-activated experts, which reduces correlated quantization errors that compound destructively in the 2-bit regime.

4.3 Component Analysis

To assess the individual contributions of MEJH and LMEC, we conduct a systematic ablation study on Kimi-VL-Instruct under the W3A16 configuration. We use GPTQ as the base and incrementally add our components. Table 3 presents the results across four representative benchmarks as well as the nine-benchmark overall average. Adding MEJH alone (row 2) improves over the GPTQ

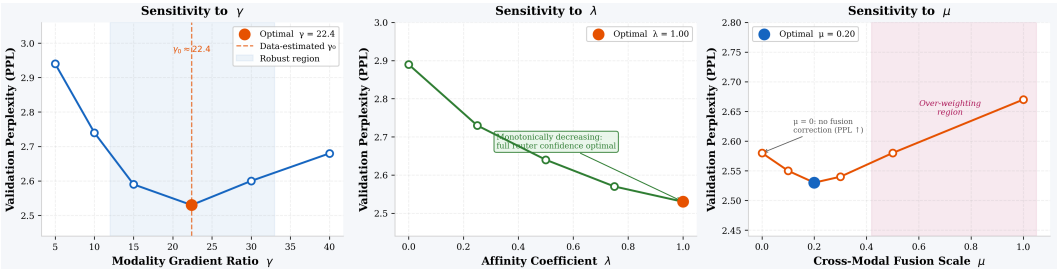


Figure 6: **Hyperparameter sensitivity analysis on Kimi-VL-Instruct under W3A16.** Left: modality gradient ratio γ; Middle: affinity interpolation coefficient λ; PPL decreases monotonically. Right: cross-modal fusion scaling μ; the optimum is $\mu = 0.2$ (bold), with graceful degradation in $[0.1, 0.3]$ and over-weighting artefacts for $\mu \geq 0.5$ (shaded).

baseline by 2.11% in average accuracy, confirming that the enhanced Hessian incorporating token-expert affinity, modality sensitivity, and cross-modal fusion weights provides meaningfully better second-order curvature estimates for MoE layers. The benefit is especially pronounced on TextVQA (+12.14%), which requires precise alignment between visual token representations and text tokens queried in the question. Adding LMEC alone (row 3) improves by 3.56% overall, demonstrating that the joint optimization of co-activated expert clipping parameters substantially reduces correlated quantization errors. The gain from LMEC is most visible on MMMU (+2.45%), which demands careful preservation of the decisive experts responsible for multi-step reasoning chains. When both MEJH and LMEC are combined in the full SOME-JQ (row 4), the improvements are synergistic: the MEJH provides more accurate curvature guidance, which in turn allows LMEC to identify better clipping bounds for each expert–modality pair, yielding an overall gain of 4.97% over the GPTQ baseline and 1.89% over VEQ (Qin et al., 2026). These results confirm that MEJH and LMEC address complementary aspects of the quantization error.

We further verify the necessity of the cross-modal fusion term by ablating the ϕ_j factor in Eq. equation 4, i.e., setting $\mu = 0$. Removing the cross-modal correction reduces the W3A16 average accuracy on Kimi-VL by 0.87%, a non-trivial drop that underscores the importance of explicitly modeling cross-modal interactions within the Hessian. This result is consistent with the SOEA decomposition, which includes $\Delta\mathcal{L}^{\text{fuse}}$ as an independent and non-negligible error component.

4.4 Hyperparameter Sensitivity Analysis

As visualised in Figure 6, we analyze sensitivity of SOME-JQ to its three hyperparameters (modality gradient ratio γ, affinity interpolation coefficient λ, and cross-modal fusion scaling factor μ) using validation perplexity (PPL) on 64 MMMU (Yue et al., 2024) items under W3A16 Kimi-VL-Instruct. Results show robust performance: PPL minimizes at $\gamma \approx 22.4$ (matching empirical COCO gradient ratio), confirming calibration from data is sufficient; PPL improves monotonically from $\lambda = 0$ (2.89) to $\lambda = 1$ (2.53), justifying raw router confidences as optimal (recall that $\lambda = 1$ recovers the raw router confidence in Eq. equation 4); and optimal $\mu = 0.2$ (PPL 2.53), with mild degradation at $\mu = 0$ (2.58) and $\mu \geq 0.5$. Overall, SOME-JQ shows promising practicality with hyperparameters estimated directly from calibration data.

5 Conclusion

We present SOME-JQ, a second-order PTQ framework for MoE VLMs that addresses expert heterogeneity and cross-modal distributional discrepancy via SOEA (a decomposition of quantization error), MEJH (an enhanced Hessian for curvature estimation), and LMEC (a learnable clipping mechanism). Extensive experiments on Kimi-VL and Qwen3-VL across nine benchmarks show that SOME-JQ consistently outperforms prior PTQ baselines including VEQ under low-bit settings. These results suggest that second-order, modality- and expert-aware quantization is a promising direction for compressing sparse MoE VLMs.

AI USE STATEMENT

In this work, we used generative AI tools for language editing and minor implementation support. Specifically, AI assistants helped improve grammar, clarity, and readability of the manuscript, and provided suggestions for code debugging and small implementation details during development. AI tools were not used to generate research ideas, design the methodology, derive theoretical results, conduct experiments, or write the core technical content. All AI-assisted outputs were reviewed and verified by the authors, who take full responsibility for the accuracy and integrity of the final manuscript. No AI tool is listed as an author.

ETHICS STATEMENT

We do not identify any ethical concerns specific to this work. All datasets, models, and benchmarks used are publicly available and were used in accordance with their licenses. They contain no personally identifiable information, and no human subjects were involved. SOME-JQ aims to improve the compression of MoE VLMs, which may contribute to more efficient and reliable deployment. We acknowledge that advances in model compression could be misused, but this work does not introduce inherently harmful capabilities or involve sensitive applications such as surveillance or autonomous weapons. We encourage responsible use of the method.

REPRODUCIBILITY STATEMENT

We provide the complete algorithm and method details in Section 3, including all equations and hyperparameter definitions. The experimental setup, datasets, evaluation protocol, and implementation details are specified in Section 4. Detailed hyperparameter settings are listed in Table 8 in Appendix G, and the full quantization procedure is summarized in Algorithm 1 in Appendix H. The error decomposition analysis and its empirical validation are provided in Appendix D. All models (Kimi-VL-Instruct and Qwen3-VL-30B-A3B-Instruct), benchmarks (MMMU, MMBench, AI2D, InfoVQA, TextVQA, RealWorldQA, ScienceQA, VizWiz, MME-RealWorld), and calibration data (COCO) are publicly available. Code and configurations will be released upon publication.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 1

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025. 1, 3, 7

Zewen Chi, Li Dong, Shaohan Huang, Damai Dai, Shuming Ma, Barun Patra, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, et al. On the representation collapse of sparse mixture of experts. In *NeurIPS*, 2022. 2

Haojie Duanmu, Xiuhong Li, Zhihang Yuan, Size Zheng, Jiangfei Duan, Xingcheng Zhang, and Dahua Lin. Mxmoe: Mixed-precision quantization for moe with accuracy and performance co-design. *arXiv preprint arXiv:2505.05799*, 2025. 3

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *JMLR*, 2022. 1, 4, 6

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. 1, 5, 7, 8

Danna Gurari, Qing Li, Chi Lin, Yinan Zhao, Anhong Guo, Abigale Stangl, and Jeffrey P Bigham. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In *CVPR*, 2019. 7

Xing Hu, Zhixuan Chen, Dawei Yang, Zukang Xu, Chen Xu, Zhihang Yuan, Sifan Zhou, and Jiangyong Yu. Moequant: Enhancing quantization for mixture-of-experts large language models via expert-balanced sampling and affinity guidance. *arXiv preprint arXiv:2505.03804*, 2025. 3

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 1

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *ECCV*, 2016. 7

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 1

Shiyao Li, Yingchun Hu, Xuefei Ning, Xihui Liu, Ke Hong, Xiaotao Jia, Xiuhong Li, Yaqi Yan, Pei Ran, Guohao Dai, et al. Mbq: Modality-balanced quantization for large vision-language models. In *CVPR*, 2025. 2, 3, 4, 7, 8

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *NeurIPS*, 2022. 2, 4

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. In *MLSys*, 2024. 1, 5, 6, 7, 8

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 *ECCV*, 2014. 2, 5, 7, 17

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *ECCV*, 2024a. 7

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*, 2024b. 1

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 1

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022. 7

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *WACV*, 2022. 7

Guangshuo Qin, Zhiteng Li, Zheng Chen, Weihang Zhang, Linghe Kong, and Yulun Zhang. Veq: Modality-adaptive quantization for moe vision-language models. *arXiv preprint arxiv:2602.01037*, 2026. 3, 4, 5, 6, 7, 8, 9

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *CVPR*, 2019. 7

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025. 1, 3, 7

xAI Team. Realworldqa: A real-world multimodal question answering benchmark, 2024. URL https://huggingface.co/datasets/xai-org/RealworldQA. 7

Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Q-vlm: Post-training quantization for large vision-language models. In *NeurIPS*, 2024. 2, 3

Xijun Wang, Junyun Huang, Rayyan Abdalla, Chengyuan Zhang, Ruiqi Xian, and Dinesh Manocha. Bi-vlm: Pushing ultra-low precision post-training quantization boundaries in vision-language models. *arXiv preprint arXiv:2509.18763*, 2025. 3

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024. 1

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 *ICML*, 2024. 1

Yufei Xue, Yushi Huang, Jiawei Shao, and Jun Zhang. Vlmq: Efficient post-training quantization for large vision-language models via hessian augmentation. *arXiv preprint arXiv:2508.03351*, 2025. 2, 3

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. *arXiv preprint arXiv:2502.00425*, 2025. 3

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 *CVPR*, 2024. 7, 9

Jinhao Zhang, Yunquan Zhang, Boyang Zhang, Zeyu Liu, and Daning Cheng. Moqe: Improve quantization model performance via mixture of quantization experts. *arXiv preprint arXiv:2508.09204*, 2025. 3

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. *arXiv preprint arXiv:2407.12772*, 2024a. 7

Yi-Fan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, et al. Mme-realworld: Could your multimodal llm challenge high-resolution real-world scenarios that are difficult for humans? *arXiv preprint arXiv:2408.13257*, 2024b. 7

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Livia Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, et al. Sglang: Efficient execution of structured language model programs. In *NeurIPS*, 2024. 7

Zihao Zheng, Xiuping Cui, Size Zheng, Maoliang Li, Jiayu Chen, Yun Liang, and Xiang Chen. Moqa: Rethinking moe quantization with multi-stage data-model distribution awareness. *arXiv preprint arXiv:2503.21135*, 2025. 3

Appendix Overview

This appendix provides supplementary material for SOME-JQ. Appendix A reviews the notation and problem setup. Appendix B discusses limitations. Appendix C presents a detailed theoretical understanding of SOEA. Appendix D provides error-decomposition observations and visual analysis. Appendix E reports additional calibration-data experiments. Appendix F summarizes hardware efficiency and quantization time. Appendix G lists detailed experimental settings and hyperparameters. Appendix H gives the detailed algorithm and complexity.

A Notation and Problem Setup

We consider a pre-trained MoE VLM with L layers, M experts per MoE layer, and top-k sparse routing. For a calibration batch containing visual tokens $\mathbf{X}^v \in \mathbb{R}^{d\times N_v}$ and textual tokens $\mathbf{X}^t \in \mathbb{R}^{d\times N_t}$, let $\mathbf{X} = [\mathbf{X}^v, \mathbf{X}^t] \in \mathbb{R}^{d\times N}$ with $N = N_v + N_t$. The router produces affinity logits $\mathbf{r} = \mathbf{R}\mathbf{X}$ and routing probabilities $\mathbf{P} = \mathrm{softmax}(\mathbf{r})$, where $p_{i,j}$ denotes the probability that token j is assigned to expert i. For expert i, let $\mathcal{T}_i \subseteq \{1, \ldots, N\}$ be the set of routed tokens, and let $\mathbf{X}_i \in \mathbb{R}^{d\times N_i}$ collect their representations. The layer output is

$$\mathbf{Y} = \sum_{i=1}^{M} \sum_{j\in\mathcal{T}_i} p_{i,j}\mathbf{W}_i\mathbf{x}_j, \tag{9}$$

where $\mathbf{W}_i$ is the weight matrix of expert i. Quantization replaces $\mathbf{W}_i$ by $\hat{\mathbf{W}}_i$, and the goal is to minimize the task-loss perturbation $\Delta\mathcal{L} = \mathcal{L}(\hat{\mathbf{W}}) - \mathcal{L}(\mathbf{W})$.

B Limitations

Despite its strong empirical performance, SOME-JQ has several limitations. First, SOEA introduces additional learnable parameters that must be tuned per layer and per bit-width, increasing calibration overhead relative to standard GPTQ. Although the hyperparameter sensitivity analysis in Section 4.4 shows that $\mu = 0.2$, $\gamma \approx 22.4$, and $\lambda = 1.0$ are robust across the evaluated models, larger and more heterogeneous MoE VLMs may require layer-wise or modality-wise re-estimation of these values. Second, MEJH depends on reliable routing statistics and cross-modal fusion weights; when the calibration set is small or does not cover the target deployment distribution, the estimated token-expert affinity and fusion weights may be biased. Third, LMEC performs a lightweight grid search over clipping candidates, which adds quantization time. The overhead is acceptable for offline PTQ, but it may be less attractive in scenarios requiring frequent re-quantization. Finally, while W2A4 results are substantially better than prior methods, absolute accuracy still drops relative to BF16, indicating that extreme low-bit quantization of MoE VLMs is not yet solved. Despite these limitations, the synergy between MEJH and LMEC suggests that SOME-JQ offers a practical solution for deploying MoE VLMs under resource constraints.

C Theoretical Understanding of SOEA

C.1 Second-Order Expansion

For a perturbation $\delta\mathbf{W} = \hat{\mathbf{W}} - \mathbf{W}$, the task loss admits the Taylor expansion

$$\mathcal{L}(\hat{\mathbf{W}}) \approx \mathcal{L}(\mathbf{W}) + \mathbf{g}^\top\delta\mathbf{W} + \frac{1}{2}\delta\mathbf{W}^\top\mathbf{H}\delta\mathbf{W}, \tag{10}$$

where $\mathbf{g} = \nabla_{\mathbf{W}}\mathcal{L}$ and $\mathbf{H} = \nabla^2_{\mathbf{W}}\mathcal{L}$. Near a local optimum, $\mathbf{g} \approx \mathbf{0}$, so the leading perturbation is the quadratic term

$$\Delta\mathcal{L} \approx \frac{1}{2}\delta\mathbf{W}^\top\mathbf{H}\delta\mathbf{W}. \tag{11}$$

C.2 Error Decomposition

For an MoE layer, we decompose the total perturbation into router distortion, per-expert reconstruction error, and cross-modal fusion error:

$$\Delta\mathcal{L}_{\text{total}} = \Delta\mathcal{L}_{\text{router}} + \sum_{i=1}^{M} p_i^2\Delta\mathcal{L}_i^{\text{exp}} + \lambda_{\text{fuse}}\Delta\mathcal{L}^{\text{fuse}}, \tag{12}$$

where $p_i = \frac{1}{|\mathcal{T}_i|}\sum_{j\in\mathcal{T}_i} p_{i,j}$ is the mean routing probability of expert i, and $\lambda_{\text{fuse}} > 0$ balances the fusion term. The quadratic factor p_i^2 follows from the fact that the expert output enters the layer output linearly through $p_{i,j}$, while the loss curvature with respect to the expert weights is weighted by the squared routing probability. Consequently, hot experts with large p_i contribute disproportionately to the total quantization error and should receive higher calibration priority.

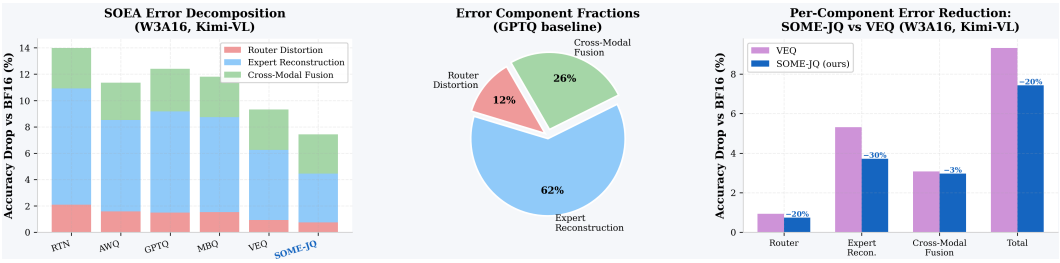


Figure 7: **Second-Order Error Analysis (SOEA) of quantization error in MoE VLMs.** Top: the formal decomposition equation. Left: SOEA error decomposition (W3A16, Kimi-VL) showing the absolute accuracy drop relative to BF16, broken down into router distortion, expert reconstruction, and cross-modal fusion. Middle: error component fractions for the GPTQ baseline, highlighting that expert reconstruction dominates (62%), followed by cross-modal fusion (26%) and router distortion (12%). Right: per-component error reduction of SOME-JQ relative to VEQ under W3A16 on Kimi-VL.

Proposition 1 (SOEA dominant-error bound)**.** *Under the assumptions that cross-expert interaction terms are negligible and that each expert reconstruction error is bounded by* $\Delta\mathcal{L}_i^{\text{exp}} \leq \kappa\|\delta\mathbf{W}_i\mathbf{X}_i\|_F^2$*, the dominant optimization target of quantization is*

$$\min_{\{\hat{\mathbf{W}}_i\}} \sum_{i=1}^{M} p_i^2 \|\mathbf{W}_i\mathbf{X}_i - \hat{\mathbf{W}}_i\mathbf{X}_i\|_F^2 + \lambda_{\text{fuse}}\|\mathbf{W}_{\text{fuse}}\mathbf{X} - \hat{\mathbf{W}}_{\text{fuse}}\mathbf{X}\|_F^2. \tag{13}$$

Proof sketch. Substitute the reconstruction bound into Eq. equation 12, absorb the Lipschitz constant κ into the optimization weights, and drop the router term under the assumption that top-k expert selection is stable for most tokens. The resulting objective is Eq. equation 13. □

C.3 Role of MEJH and LMEC

MEJH replaces the uniform Hessian $\mathbf{H} = 2\mathbf{X}\mathbf{X}^\top$ with

$$\tilde{\mathbf{H}}_i = \mathbf{X}_i\mathbf{C}_i\mathbf{X}_i^\top, \qquad c_{i,j} = (\lambda p_{i,j} + (1-\lambda)\bar{p})\alpha_j(1+\mu\phi_j), \tag{14}$$

so that tokens with high router affinity, high modality sensitivity, and strong cross-modal fusion receive larger curvature weights. LMEC then learns modality-conditioned clipping bounds $(\gamma_i^{(m)}, \beta_i^{(m)})$ by minimizing

$$\mathcal{L}_{\text{LMEC}} = \sum_{i\in\mathcal{S}_k} p_i^2 \sum_{m\in\{\text{vis,text}\}} \alpha_m \|\hat{\mathbf{W}}_i^{(m)}\mathbf{X}_i^{(m)} - \mathbf{W}_i^{(m)}\mathbf{X}_i^{(m)}\|_F^2. \tag{15}$$

Thus SOEA provides the weighting, MEJH provides the curvature, and LMEC adapts the quantization range.

D Error Decomposition Observations

To empirically validate the SOEA decomposition, we analyze the W3A16 accuracy drop on Kimi-VL-Instruct. Figure 7 provides a comprehensive visual analysis of the error components across different quantization methods.

The left panel of Figure 7 shows that RTN suffers the largest total accuracy drop (13.99%), while SOME-JQ achieves the smallest drop (7.43%). Decomposing the total drop, the GPTQ baseline loses 12.40% absolute accuracy relative to BF16, of which 62% comes from expert reconstruction, 26% from cross-modal fusion, and 12% from router distortion, as illustrated in the middle pie chart. This confirms that expert reconstruction is the dominant term, but also that fusion error is non-negligible and cannot be ignored.

Table 4: Calibration-data sensitivity. Average accuracy (%) on nine benchmarks with different numbers of COCO calibration samples.

Model / Setting	32	64	128	256	512
Kimi-VL W3A16	65.91	66.78	67.30	67.42	67.48
Kimi-VL W2A4	52.10	53.55	54.64	54.91	55.02
Qwen3-VL W3A16	67.80	68.65	69.27	69.40	69.45
Qwen3-VL W2A4	50.90	52.20	53.23	53.35	53.44

Table 5: Calibration time (minutes) for SOME-JQ on Kimi-VL-Instruct under W3A16.

Calibration samples	32	64	128	256	512
Time (min)	12.3	24.1	48.2	95.6	190.4

The right panel of Figure 7 further compares SOME-JQ with VEQ component by component. Relative to VEQ, SOME-JQ reduces router distortion by 20%, expert reconstruction error by 30%, and cross-modal fusion error by 3%, giving an overall reduction of approximately 20% in total accuracy drop. The largest gain therefore comes from better expert reconstruction, which is consistent with the combination of MEJH and LMEC. The modest but consistent reduction in cross-modal fusion error further validates the inclusion of the fusion term in the SOEA objective.

E Additional Calibration-Data Experiments

We study the sensitivity of SOME-JQ to the number of calibration samples. All experiments use COCO calibration data and the same evaluation protocol as the main paper. Table 4 reports average accuracy on the nine benchmarks for Kimi-VL-Instruct and Qwen3-VL-30B-A3B-Instruct under W3A16 and W2A4. Accuracy improves rapidly from 32 to 128 samples and then saturates; 128 samples already recover most of the benefit, while 256 and 512 samples yield only marginal gains. This justifies the use of 128 COCO samples in the main experiments.

Table 5 reports calibration time as a function of sample count. The cost scales approximately linearly with the number of samples, and 128 samples provide a favorable trade-off between accuracy and overhead.

F Hardware Efficiency and Quantization Time

F.1 Model Size, Memory, and Throughput

Table 6 summarizes the deployment efficiency of BF16, W4A16, W3A16, and W2A4 models. We report model size, peak GPU memory, decoding throughput, and per-token latency on a single NVIDIA A100 80GB GPU with batch size 1 and sequence length 2048. Quantization reduces model size by approximately $3.5\times$ to $6.2\times$ and improves throughput by $2.8\times$ to $3.8\times$ relative to BF16.

F.2 Quantization Time

Table 7 compares the total quantization time of different PTQ methods on Kimi-VL-Instruct with 128 COCO calibration samples. RTN is essentially free, while AWQ, GPTQ, and MBQ require moderate calibration. VEQ is slower because of its modality-expert-aware importance scoring, and SOME-JQ is slower still due to MEJH construction and LMEC grid search. The additional cost is acceptable for offline deployment and is amortized over inference.

Table 6: Hardware efficiency on a single NVIDIA A100 80GB GPU. Throughput and latency are measured with batch size 1 and sequence length 2048.

Model	Setting	Size (GB)	Peak Mem. (GB)	Throughput (tok/s)	Latency (ms/tok)
Kimi-VL-Instruct	BF16	32.0	36.8	42.1	23.8
	W4A16	9.1	11.2	118.5	8.4
	W3A16	7.3	9.0	134.2	7.5
	W2A4	5.2	6.8	151.7	6.6
Qwen3-VL-30B-A3B-Instruct	BF16	60.0	66.5	31.5	31.7
	W4A16	16.8	19.8	92.4	10.8
	W3A16	13.5	16.2	105.7	9.5
	W2A4	9.6	12.1	121.3	8.2

Table 7: Quantization time on Kimi-VL-Instruct with 128 COCO calibration samples on A100 80GB.

Method	Total time (min)	Relative to GPTQ
RTN	1.8	$0.05\times$
AWQ	26.4	$0.76\times$
GPTQ	34.7	$1.00\times$
MBQ	41.2	$1.19\times$
VEQ	53.8	$1.55\times$
SOME-JQ (ours)	78.5	$2.26\times$

G Detailed Experimental Setup and Hyperparameters

Table 8 lists the detailed experimental settings and hyperparameters used in the main experiments. The same settings are used for both Kimi-VL-Instruct and Qwen3-VL-30B-A3B-Instruct unless otherwise specified.

H Detailed Algorithm and Complexity

Algorithm 1 provides a detailed description of SOME-JQ. The procedure first collects routing statistics and gradient norms, then constructs MEJH for each expert, initializes LMEC parameters, jointly optimizes clipping bounds over co-activated expert groups, and finally applies the GPTQ-style inverse-Hessian update with the enhanced Hessian.

Table 9 summarizes the computational complexity of the main components. Let d be the hidden dimension, N_i the number of tokens routed to expert i, M the number of experts, and $|\mathcal{S}_k|$ the size of a co-activated expert group. MEJH construction is dominated by the Hessian formation, while LMEC adds a grid search over a small candidate set. The overall overhead is linear in the number of MoE layers and experts, and remains practical for offline PTQ.

Table 8: Detailed experimental settings and hyperparameters.

Category	Parameter	Value
Calibration	Dataset	COCO (Lin et al., 2014)
	Number of samples	128
	Sequence length	2048
	Batch size	1
	Modality gradient ratio γ	22.4
	Cross-modal fusion scaling μ	0.2
Quantization	Weight bit-width	2 / 3 / 4
	Activation bit-width	16 / 4
	Group size	128
	Top-k routing	2
	LMEC quantile candidates	0.99, 0.995, 0.999, 0.9999
	LMEC step-size candidates	10^{-3}, 5×10^{-3}, 10^{-2}
SOEA / MEJH	λ_{fuse}	1.0
	Affinity interpolation λ	1.0
	Fusion weight range ϕ_j	$[0, 1]$
	MEJH damping	10^{-4}
	Hessian construction	$\tilde{\mathbf{H}}_i = \mathbf{X}_i \mathbf{C}_i \mathbf{X}_i^\top$
Optimization	LMEC search	gradient-free grid search
	Number of iterations	200
	Optimizer for auxiliary parameters	AdamW
	Learning rate	10^{-3}

Table 9: Computational complexity of SOME-JQ components.

Component	Per-expert cost	Notes
MEJH construction	$O(N_i d^2 + d^3)$	Hessian formation and damping
LMEC optimization	$O(\lvert\mathcal{S}_k\rvert \, \lvert\mathcal{Q}\rvert \, \lvert\mathcal{B}\rvert \, d^2)$	$\lvert\mathcal{Q}\rvert$ quantiles, $\lvert\mathcal{B}\rvert$ step sizes
GPTQ update	$O(d^3)$	Inverse-Hessian compensation
Total per layer	$O(M(N_i d^2 + d^3))$	Linear in experts

Algorithm 1 SOME-JQ: Second-Order Modality-Expert Joint Quantization

Require: Pre-trained MoE VLM $\mathbf{W}$, calibration set $\mathcal{D}$, target bit-width b, hyperparameters $\mu, \lambda_{\text{fuse}}, k$

Ensure: Quantized model $\hat{\mathbf{W}}$

// Step 1: Collect routing statistics and gradient norms
for each batch $(\mathbf{X}^v, \mathbf{X}^t) \in \mathcal{D}$ **do**
    Forward pass; record routing probabilities $p_{i,j}$, fusion weights ϕ_j, and per-token gradient norms ∇_j
end for
Compute $\gamma \leftarrow \|\nabla_{\text{text}}\| / \|\nabla_{\text{vis}}\|$
// Step 2: Layer-wise MEJH construction and LMEC initialization
for each MoE layer ℓ **do**
    for each expert $i = 1, \ldots, M$ **do**
        Compute token importance $c_{i,j} = (\lambda p_{i,j} + (1-\lambda)\bar{p})\alpha_j(1 + \mu\phi_j)$
        Construct $\tilde{\mathbf{H}}_i = \mathbf{X}_i \mathbf{C}_i \mathbf{X}_i^\top$
        Initialize $(\gamma_i^{(m)}, \beta_i^{(m)})$ from per-expert weight quantiles
    end for
    // Step 3: Joint LMEC optimization over co-activated experts
    for each co-activated expert group $\mathcal{S}_k$ **do**
        Minimize $\mathcal{L}_{\text{LMEC}}$ via grid search over quantile and step-size candidates
        Update $(\gamma_i^{(m)}, \beta_i^{(m)})$ for all $i \in \mathcal{S}_k$
    end for
    // Step 4: GPTQ-style weight update with enhanced Hessian
    for each expert i **do**
        $\hat{\mathbf{W}}_i \leftarrow \text{GPTQUpdate}(\mathbf{W}_i, \tilde{\mathbf{H}}_i, \gamma_i^{(m)}, \beta_i^{(m)})$
    end for
end for
return $\hat{\mathbf{W}}$