

OPERA-MoE: Output-aware Routing Perturbation Alignment via Observable Discrepancy for MoE Quantization

Geonho Lee^{1†} Hancheol Park² Tae-Ho Kim² Jungwook Choi^{1*}

¹Hanyang University ²Nota Inc.

¹{thisisho, choij}@hanyang.ac.kr

²{hancheol.park, thkim}@nota.ai

Abstract

Mixture-of-Experts (MoE) architectures have become a dominant paradigm for scaling large language models (LLMs), but their large memory footprint still makes post-training quantization (PTQ) essential for practical deployment. Unlike dense LLMs, quantized MoE models exhibit routing perturbations, where quantization alters expert routing behavior. Existing router alignment methods mitigate this issue by uniformly minimizing routing discrepancies across all tokens. However, we observe that routing perturbations exhibit highly heterogeneous relationships with observable output discrepancy: many tokens preserve nearly identical output distributions despite substantial routing changes. Motivated by this observation, we propose OPERA, an output-aware router alignment strategy for quantized MoE models that selectively prioritizes tokens exhibiting observable output discrepancy after quantization. We further analyze output-preserving routing perturbations and show that they frequently occur on highly predictable tokens, where substantial routing perturbations and hidden-state deviations may not strongly affect projected output distributions. Experiments across multiple MoE LLMs demonstrate that OPERA generally improves over conventional router alignment, with particularly noticeable gains under aggressive low-bit quantization settings.

1 Introduction

Mixture-of-Experts (MoE) architectures (Jacobs et al., 1991; Jordan and Jacobs, 1993; Fedus et al., 2022; Dai et al., 2024) have become a central paradigm for scaling large language models (LLMs) by activating only a subset of experts for each token. This sparse activation substantially increases model capacity while maintaining relatively low computation cost, leading many recent fron-

tier LLMs to adopt MoE architectures (DeepSeek-AI, 2026; Qwen Team, 2026; GLM-5-Team et al., 2026; Park et al., 2025; LG AI Research, 2025; DeepMind, 2026; NVIDIA, 2025b).

Despite their computational efficiency, MoE models still incur substantial inference-time memory overhead because expert parameters typically remain resident in memory to support dynamic expert routing during generation. Quantization is therefore essential for practical deployment of large-scale MoE LLMs. Recent post-training quantization (PTQ) methods substantially reduce memory usage while largely preserving model quality in dense LLMs (Frantar et al., 2023; Lin et al., 2024b,a; Cheng et al., 2025). However, quantized MoE models introduce an additional challenge: quantization perturbs hidden representations and consequently alters routing behavior.

Prior work has shown that preserving routing consistency can improve quantized MoE performance and has proposed router alignment methods that minimize routing discrepancies between baseline and quantized models (Chen et al., 2025a; Fu et al., 2025; Park et al., 2026). Existing approaches, however, uniformly optimize routing consistency across all tokens, implicitly assuming that all routing perturbations are equally important.

In this work, we investigate how routing perturbations relate to observable output discrepancy after quantization. While some routing perturbations coincide with noticeable output deviations, many others preserve nearly identical output distributions despite substantial structural routing changes. This distinction is particularly relevant in autoregressive decoding, where output deviations may propagate through subsequent decoding steps (Lee et al., 2024; Li et al., 2026).

Based on this analysis, we propose OPERA (Output-aware Routing Perturbation Alignment), an output-aware router alignment strategy for MoE quantization. Rather than uniformly enforcing rout-

[†]Work done during an internship at Nota Inc.

^{*}Corresponding author.

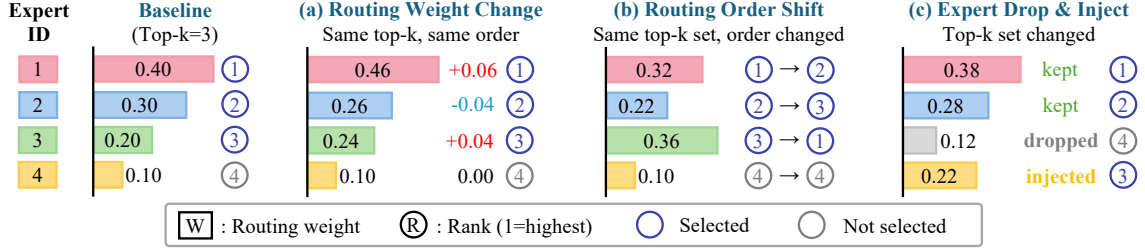


Figure 1: Examples of routing perturbations in quantized MoE models. Routing weights determine expert ranking and expert selection for each token. Quantization can alter routing weights, expert ordering, or selected experts.

ing consistency across all tokens, OPERA selectively prioritizes perturbations associated with observable output discrepancy after quantization.

Our contributions are summarized as follows:

- We analyze routing perturbations in quantized MoE models and show that routing perturbations exhibit highly heterogeneous relationships with observable output discrepancy.
- We investigate output-preserving routing perturbations and show that they frequently occur on highly predictable tokens.
- We propose OPERA, an output-aware router alignment strategy that reallocates alignment emphasis toward routing perturbations associated with observable output discrepancy, complementary to the underlying router alignment objective.
- We show that OPERA generally improves over conventional router alignment across multiple MoE LLMs and quantization settings.

2 Backgrounds

2.1 Mixture-of-Experts Architecture

MoE models employ sparse conditional computation, where only a subset of experts is activated for each token. Given an input representation x , a router selects the top- k experts $\mathcal{E}_k(x)$ along with their routing weights $w_i(x)$. Let $E_i(x)$ denote the output of the i -th expert. The final MoE output is computed as:

$$y = \sum_{i \in \mathcal{E}_k(x)} w_i(x) \cdot E_i(x). \quad (1)$$

Sparse expert activation enables MoE models to substantially increase model capacity while maintaining relatively low computation cost.

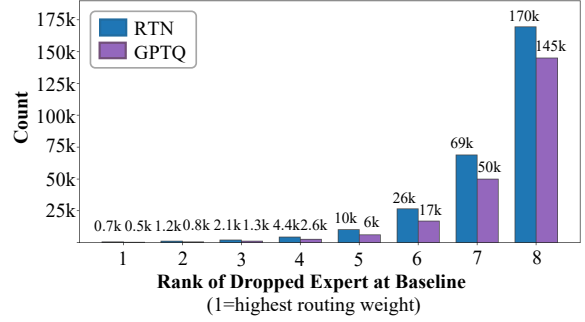


Figure 2: Distribution of dropped expert ranks under RTN and GPTQ quantization on Qwen3-30B-A3B using the Pile dataset. Lower-ranked experts are more frequently affected by routing perturbation. Although GPTQ reduces routing perturbations compared to RTN, residual routing differences remain after quantization.

2.2 Quantization in LLMs

Quantization reduces the memory footprint of LLMs by representing model parameters with lower numerical precision. Among various compression approaches, PTQ is widely adopted because it substantially reduces memory usage and inference cost without requiring expensive retraining. Recent PTQ methods improve quantized model quality by minimizing reconstruction error (Frantar et al., 2023; Shao et al., 2024; Cheng et al., 2025) or introducing quantization-friendly parameterizations (Lin et al., 2024b,a), enabling low-bit deployment of large-scale LLMs.

2.3 MoE-specific Quantization Strategies

Unlike dense LLMs, quantized MoE models introduce additional challenges due to sparse expert activation and expert heterogeneity: Since experts differ in activation frequency and functional importance (Su et al., 2025), prior work has explored MoE-specific quantization strategies such as expert-wise mixed-precision quantization (Li et al., 2024; Huang et al., 2025; Chowdhury et al., 2026) and expert-balanced calibration (Chen et al., 2025b). In addition to expert heterogeneity, quantization can also perturb routing behavior. Although routers are

typically maintained in full precision, quantization errors from preceding layers can still alter routing decisions through perturbed hidden representations. As illustrated in Figure 1, quantization can affect routing behavior at multiple levels, including routing weight changes, expert reordering, and top-k expert replacement. Figure 2 further shows that routing perturbations persist after applying advanced PTQ methods such as GPTQ (Frantar et al., 2023), although their frequency is reduced compared to naive quantization. These observations motivate router alignment methods for quantized MoE models.

2.4 Router Alignment and Observable Output Discrepancy

Prior work proposes router alignment methods that encourage quantized MoE models to mimic the routing behavior of baseline models. Existing approaches primarily focus on designing alignment objectives that minimize routing discrepancies across calibration tokens, including value-based, distribution-based, and structural alignment losses (Chen et al., 2025a; Fu et al., 2025; Fang and Huang, 2025; Park et al., 2026). While preserving routing consistency can improve quantized performance, our work instead focuses on a complementary question: which routing perturbations are more strongly associated with observable output discrepancy after quantization. Rather than introducing a new alignment objective, we investigate how router alignment can selectively prioritize routing perturbations associated with larger observable output discrepancy. This distinction is particularly relevant in autoregressive decoding, where output deviations may propagate through subsequent decoding steps.

3 Analysis

In this section, we analyze the relationship between routing perturbations and observable output discrepancy in quantized MoE models.

3.1 Observable Output Discrepancy

To measure observable output discrepancy between baseline and quantized MoE models, we use Jensen-Shannon Divergence (JSD) over token-level output distributions. Since autoregressive decoding is primarily determined by a small set of competing candidate tokens, we compute JSD over the union

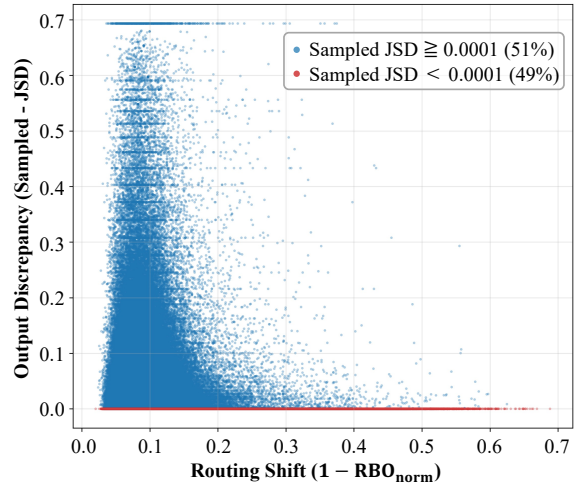


Figure 3: Routing shift and observable output discrepancy measured on Qwen3-30B-A3B under W4A16. Each point represents a token. Similar patterns are observed across datasets and bit-width settings (Appendix B).

of candidate token sets considered by both models.

$$\text{JSD}_{\text{Sampled}} = \text{JSD}(p_U^*, \hat{p}_U), \quad (2)$$

where p_U^* and $\hat{p}_U$ denote distributions restricted to the union candidate set U . By focusing on candidate tokens relevant to generation, Sampled-JSD reduces sensitivity to low-probability noise. Appendix A provides additional examples and comparisons between standard JSD and Sampled-JSD.

Throughout our analysis, we use a small discrepancy threshold $\tau = 0.0001$ to separate near-zero discrepancy tokens from tokens exhibiting observable output deviation. As shown in Figure 3, this threshold approximately separates a substantial fraction of tokens whose output distributions remain nearly unchanged after quantization.

3.2 Structural Routing Perturbation

To quantify structural routing perturbation, we compare the top- k expert rankings of the baseline and quantized MoE models. Since routing perturbations involve both ranked ordering and set overlap, we use Rank-Biased Overlap (RBO) (Webber et al., 2010), a ranking-aware similarity metric that emphasizes agreement at higher-ranked positions:

$$\text{RBO}(S, T, p) = (1 - p) \sum_{d=1}^k p^{d-1} \cdot A_d. \quad (3)$$

Where A_d denotes the overlap between the top- d prefixes of the ranked expert lists S and T ,

$$A_d = \frac{|\{s_1, \dots, s_d\} \cap \{t_1, \dots, t_d\}|}{d}. \quad (4)$$

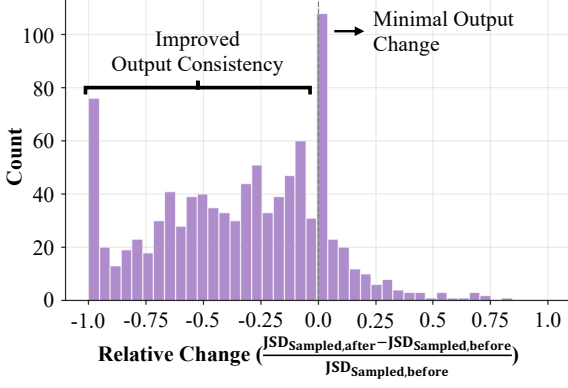


Figure 4: Relative change in Sampled-JSD after replacing quantized routing decisions with baseline routing for tokens exhibiting both high routing shift and high output discrepancy. Negative values indicate reduced output discrepancy after restoring baseline routing, suggesting that routing perturbations can partially contribute to observable output deviations after quantization.

The weighting factor $p \in (0, 1)$ ($p = 0.9$ in our experiments) places greater emphasis on higher-ranked experts, making RBO suitable for characterizing structural routing perturbations in MoE routing behavior. We then define routing shift as:

$$\text{Routing Shift} = 1 - \text{RBO}_{\text{norm}}, \quad (5)$$

where RBO_{norm} denotes normalized RBO in the range $[0, 1]$. Routing shift ranges from 0 (identical routing) to 1 (completely different expert ordering and selection).

3.3 Heterogeneous Routing Perturbations

We measure routing shift and observable output discrepancy for each token under quantization. Figure 3 shows that routing perturbations exhibit highly heterogeneous relationships with observable output discrepancy. While some tokens exhibit both large routing shift and output discrepancy, many others preserve nearly identical output distributions despite substantial routing perturbations.

These observations suggest that routing perturbations may differ substantially in their relationships with observable output discrepancy, motivating alignment strategies that place greater emphasis on output-sensitive perturbations.

3.3.1 Routing Perturbations Still Contribute

Although many routing perturbations remain output-preserving, routing perturbations can still contribute to observable output discrepancy in certain cases. This distinction is important because our observations do not suggest that routing perturbations should be ignored entirely, but rather that

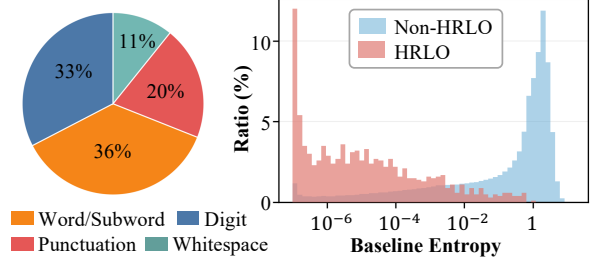


Figure 5: Characteristics of output-preserving routing perturbations (HRLO). Left: token category distribution. Right: baseline entropy distribution. Output-preserving perturbations are frequently associated with low entropy, highly predictable tokens, including whitespace, punctuation, digits, and common subwords. Examples of HRLO tokens are represented in Appendix C

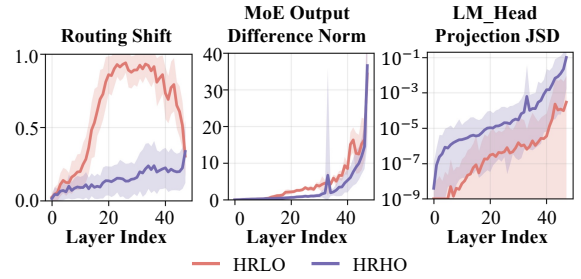


Figure 6: Layer-wise analysis using routing injection from the quantized model into the baseline model on Qwen3-30B-A3B with Pile. The plots show routing shift, MoE output difference norm, and LM-head projected output discrepancy (JSD) across layers. Despite substantial routing perturbations and large intermediate MoE output deviations, HRLO tokens maintain relatively stable LM-head projected output distributions across layers.

their observable effects may differ substantially across tokens. In particular, when both routing shift and output discrepancy are large, it remains unclear whether the observed discrepancy primarily originates from routing perturbations or from quantization errors in other model components.

To isolate their contribution, we perform a controlled routing replacement analysis. Specifically, for selected tokens, we replace the router-selected experts and corresponding routing weights of the quantized model with those from the baseline model across all MoE layers, while keeping all other quantized model components unchanged. Figure 4 shows that restoring baseline routing behavior frequently reduces observable output discrepancy, suggesting that routing perturbations may partially contribute to output deviations after quantization. These observations further support distinguishing between routing perturbations that are more strongly associated with observable output

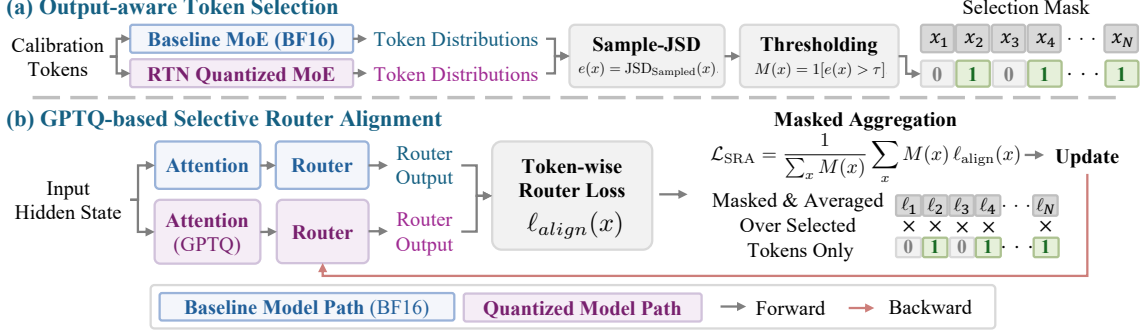


Figure 7: Overview of OPERA. Tokens exhibiting observable output discrepancy are selected for router alignment, while low-discrepancy tokens are excluded from alignment loss aggregation.

discrepancy and those that remain largely output-preserving after quantization.

3.3.2 Output-Preserving Routing Perturbations

To better understand why some routing perturbations remain output-preserving despite large routing shifts, we analyze tokens exhibiting high routing shift but low output discrepancy (HRLO). Figure 5 shows that HRLO tokens are predominantly highly predictable tokens, including whitespace, punctuation, digits, and common subwords. HRLO tokens also exhibit substantially lower entropy distributions compared to other tokens.

To further investigate this behavior, we compare HRLO tokens with HRHO (High Routing shift, High Output discrepancy) tokens through a controlled routing injection analysis. Specifically, for each MoE layer, we replace the baseline router-selected experts with those selected by the quantized model while keeping all other model components to isolate the effect of routing perturbations without introducing additional quantization errors.

Figure 6 presents layer-wise analyses of routing shift, MoE output difference norm, and intermediate LM-head projected output discrepancy, computed by projecting intermediate hidden states through the LM-head and measuring the resulting token distribution divergence (JSD). Compared to HRHO tokens, HRLO tokens often exhibit larger routing perturbations and hidden-state deviations across many middle and later layers, while maintaining consistently smaller projected output discrepancy throughout decoding. These observations suggest that large routing-induced hidden-state perturbations are not always reflected in substantial observable output changes after LM-head projection, and that hidden-state perturbation magnitude alone may not reliably predict observable output

discrepancy.

Prior work on intermediate decoding and logit-space analysis has shown that many token predictions become recoverable from intermediate representations well before the final decoding layers (Belrose et al., 2023; Nguyen, 2024). Combined with the low-entropy patterns in Figure 5, our observations further suggest that highly predictable tokens may be less sensitive to routing-induced representation changes during later decoding stages. This interpretation is broadly consistent with recent early-exit studies suggesting that computational requirements vary substantially across token types, where predictable or repetitive tokens can often be reliably generated without requiring full-depth computation (Zarch et al., 2025; Luo et al., 2025).

These findings further support the motivation behind selective router alignment: routing perturbations may differ substantially in their observable sensitivity, such that perturbations exhibiting relatively small observable discrepancy despite large structural deviations may require less alignment emphasis than perturbations more strongly reflected in observable output discrepancy. From an optimization perspective, this suggests that uniformly enforcing routing consistency across all tokens may allocate alignment effort to perturbations with limited observable impact. This motivates investigating whether router alignment can benefit from allocating greater alignment emphasis toward output-sensitive routing perturbations.

4 Method: OPERA

Our analysis suggests that large structural routing perturbations do not necessarily translate into substantial observable output discrepancy after quantization. This observation motivates a different perspective of router alignment: rather than uniformly enforcing routing consistency across all to-

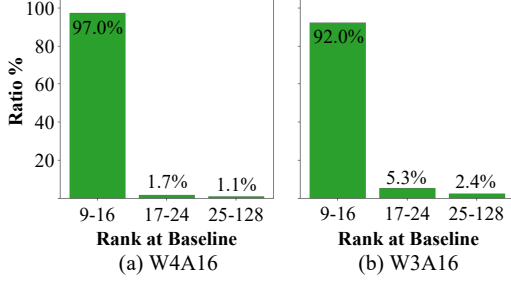


Figure 8: Distribution of injected expert ranks after quantization on Qwen3-30B-A3B (top- $k=8$) using the C4 dataset. Most injected experts originate near the original top- k experts.

kens, alignment should prioritize perturbations that are more strongly reflected in observable output discrepancy. Motivated by this observation, OPERA allocates greater alignment emphasis toward tokens exhibiting observable output discrepancy during router alignment. Figure 7 provides an overview of OPERA.

4.1 Output-aware Token Selection

We first estimate token-wise observable output discrepancy using Sampled-JSD between baseline and RTN-quantized MoE models. For a calibration token x , the discrepancy is defined as:

$$e(x) = \text{JSD}_{\text{Sampled}}(x). \quad (6)$$

We then construct a binary selection mask based on the discrepancy score:

$$M(x) = 1[e(x) > \tau], \quad (7)$$

where τ denotes the discrepancy threshold and $M(x) = 1$ indicates that token x is selected to contribute to router alignment loss aggregation. This selection mask is precomputed prior to router alignment and reused throughout PTQ optimization. We use RTN-based discrepancy as a practical, alignment-independent proxy that can be computed before GPTQ optimization. We empirically examine its consistency with GPTQ-derived discrepancy in Section 5.4. We adopt binary masking primarily as a simple mechanism for reducing the contribution of near-zero discrepancy tokens, rather than attempting to precisely estimate token-wise alignment importance. Appendix E further shows that the same principle can be extended to continuous weighting.

Using a small threshold τ reduces the alignment contribution from tokens whose observable output distributions remain nearly unchanged after quantization. As shown in Figure 3, a substantial fraction

of tokens remain in this low-discrepancy regime despite noticeable routing perturbations.

4.2 Selective Router Alignment

We integrate OPERA into GPTQ-based (Frantar et al., 2023) router alignment by selectively masking token contributions during alignment optimization. GPTQ performs layer-wise quantization by sequentially optimizing quantized layers while propagating intermediate hidden states through the quantized model. During this process, router alignment minimizes discrepancies between the router outputs of the baseline and quantized model paths.

Given a calibration token x , we define a token-wise router alignment loss $\ell_{\text{align}}(x)$ using the router outputs from the baseline and quantized model paths. Unlike conventional router alignment methods, which uniformly aggregate alignment losses across all calibration tokens, OPERA selectively prioritizes alignment optimization towards tokens exhibiting observable output discrepancy after quantization:

$$\mathcal{L}_{\text{SRA}} = \frac{1}{\sum_x M(x)} \sum_x M(x) \ell_{\text{align}}(x) \quad (8)$$

where $M(x)$ denotes the output-aware selection mask defined in Section 4.1.

Importantly, all calibration tokens still participate in forward propagation and routing computation during quantization. OPERA only modifies how alignment losses are aggregated. This can be interpreted as reallocating alignment emphasis toward routing perturbations more strongly associated with observable output discrepancy, rather than uniformly allocating alignment emphasis across all tokens.

Because OPERA modifies how token-wise alignment losses are aggregated rather than the underlying loss formulation itself, it can be combined with different router alignment objectives, as further examined in Section 5.4

4.3 Router Alignment Scope: Top-3 k

Prior work has observed that aligning the full routing distribution can introduce noise from lower-ranked experts with negligible routing weights, motivating alignment over an expanded subset of high-ranked experts beyond the original top- k selection (Chen et al., 2025a). To further examine how routing perturbations propagate beyond the routing boundary, we analyze the baseline ranks

Table 1: Main results on Qwen3-30B-A3B under W4A16 and W3A16 quantization. OPERA generally improves over conventional router alignment. Avg. denotes the average performance across all reported benchmarks. MMLU-Pro results in this table are averaged over two runs.

Bit	Method	AIME24	AIME25	MATH500	GPQA-D	MMLU-Pro	IFEval	HumanEval	BBH	Avg.
Baseline		80.42	78.75	96.80	61.30	72.81	82.67	93.90	72.34	79.87
W4A16	RTN	75.84	64.58	93.60	59.28	69.62	79.99	89.94	54.22	73.38
	GPTQ	79.58	67.50	96.20	60.54	71.11	81.93	91.46	57.21	75.69
	RA	79.58	67.50	95.80	60.73	71.05	81.93	91.31	57.34	75.66
	OPERA	79.58	70.00	97.00	61.87	71.16	82.26	92.07	62.10	77.01
W3A16	RTN	46.67	34.58	85.00	41.92	57.44	75.79	85.98	49.67	59.63
	GPTQ	66.67	48.75	92.40	53.66	64.47	76.16	83.54	51.49	67.14
	RA	69.58	52.50	92.00	55.50	64.70	73.80	85.98	53.77	68.48
	OPERA	72.08	55.42	93.60	54.05	66.92	76.43	86.13	64.59	71.15

of experts newly selected after quantization. As shown in Figure 8, most injected experts originate from ranks near the original top- k boundary.

Following this observation, we adopt the baseline Top-3 k experts as the default alignment scope. We use MSE over this expert set as the default router alignment objective. Let $\mathcal{E}_{3k}(x)$ denote the baseline top-3 k experts for token x . The alignment loss is defined as:

$$\ell_{align}(x) = \frac{1}{|\mathcal{E}_{3k}(x)|} \sum_{i \in \mathcal{E}_{3k}(x)} (r_{b,i}(x) - r_{q,i}(x))^2, \quad (9)$$

where $r_{b,i}(x)$ and $r_{q,i}(x)$ denote the router outputs from the baseline and quantized model paths, respectively. Using Top-3 k MSE allows the alignment objective to capture most meaningful injected experts while reducing sensitivity to substantially lower-ranked experts.

5 Experiments

5.1 Experimental Setup

Models and Benchmarks. We evaluate OPERA on recent MoE LLMs, including Qwen3-30B-A3B (Qwen Team, 2025), Solar-Open-100B (Park et al., 2025), and GLM-4.5-Air (GLM Team et al., 2025).

We evaluate across diverse math, instruction-following, coding, and knowledge benchmarks, including AIME24, AIME25 (Maxwell-Jia, 2025), MATH-500 (Lightman et al., 2024), GPQA-Diamond (GPQA-D) (Rein et al., 2024), MMLU-Pro (MMLU-P) (Wang et al., 2024), IFEval (Zhou et al., 2023), HumanEval (Chen et al., 2021), and BBH (Suzgun et al., 2023). For repeated evaluations, we report the average over 8 runs for AIME24/25 and GPQA-D, and 4 runs for IFEval, HumanEval, and BBH. MATH-500 and MMLU-

Pro are evaluated using a single run unless otherwise specified. Detailed evaluation configurations are provided in Appendix D

Quantization and Router Alignment. We adopt GPTQ-based PTQ under W4A16 and W3A16 settings, while keeping embedding layer, LM head, and routers in full precision. We compare: (i) GPTQ without router alignment, (ii) conventional uniform router alignment across all calibration tokens (RA), and (iii) the proposed OPERA, which prioritizes token contributions to router alignment based on observable output discrepancy. Calibration data is sampled from the Pile dataset (Gao et al., 2020).

5.2 Main Results

Table 1 summarizes the main results on Qwen3-30B-A3B. Compared to GPTQ without router alignment, both RA and OPERA generally improve performance across most benchmarks. Under W4A16, OPERA matches or improves upon conventional RA across all reported benchmarks, achieving the highest average performance of 77.01 compared with 75.66 of RA. The improvements become more noticeable under the more aggressive W3A16 setting, where routing perturbations become more severe after quantization. Although OPERA performs worse than RA on GPQA-D (54.05 vs. 55.50), it improves performance on the other seven benchmarks, with particularly noticeable gains on BBH, AIME24/25, and MMLU-Pro. Consequently, the average performance increases from 68.48 with RA to 71.15 with OPERA, indicating a larger overall benefit under aggressive low-bit quantization.

At the same time, the mixed behavior across methods and benchmarks suggests that routing per-

Table 2: Results on large-scale MoE models under W4A16. OPERA achieves competitive or improved performance across most settings.

Model	Method	AIME24	AIME25	GPQA-D	MMLU-P
Solar Open 100B	Baseline	77.92	62.50	58.59	65.92
	GPTQ	73.33	60.00	59.60	63.54
	RA	70.00	61.25	58.33	63.80
	OPERA	76.67	63.75	57.01	65.79
GLM 4.5 Air	Baseline	72.08	51.67	69.63	64.50
	GPTQ	64.58	51.25	68.18	61.69
	RA	63.75	52.08	67.87	62.08
	OPERA	67.92	51.67	69.26	62.96

turbations may differ substantially in how strongly they are reflected in observable output discrepancy after quantization. Uniformly enforcing routing consistency across all tokens may therefore allocate alignment emphasis inefficiently in certain settings. Overall, these observations suggest that selectively prioritizing routing perturbations associated with observable output discrepancy can become increasingly beneficial under more aggressive low-bit quantization settings.

5.3 Scaling to Larger MoE Models

Table 2 presents results on large-scale MoE models, including Solar-Open-100B (Park et al., 2025) and GLM-4.5-Air (GLM Team et al., 2025), under W4A16 quantization. Conventional RA shows mixed effectiveness across models and tasks, occasionally improving performance but also introducing degradation in certain settings. OPERA also does not uniformly outperform other methods on every individual benchmark: on Solar-Open-100B, GPQA-D decreases to 57.01 compared with 59.60 for GPTQ and 58.33 for RA, while on GLM-4.5-Air, AIME25 reaches 51.67 compared with 52.08 for RA. Nevertheless, OPERA achieves the highest average performance on both large-scale models. On Solar-Open-100B, the average score improves to 65.81 compared with 64.12 for GPTQ and 63.35 for RA; on GLM-4.5-Air, OPERA achieves 62.95, compared with 61.43 for GPTQ and 61.45 for RA. These results suggest that, despite benchmark-level variation, prioritizing routing perturbations associated with observable output discrepancy provides a consistent aggregate benefit across different MoE models and scales.

5.4 Ablation Study

Router Alignment Scope. Table 3 compares different expert scopes for the MSE router alignment

Table 3: Ablation study on router alignment scope evaluated on MMLU-Pro. Experiment is conducted on RA. Top-3k MSE achieves the strongest performance across both quantization settings.

Bit	Top- k	Top-2 k	Top-3 k	Top-4 k	Top-5 k	Full
4	70.88	70.88	71.09	70.94	70.96	70.72
3	63.93	63.92	64.73	64.65	64.30	64.23

Table 4: Ablation study on discrepancy threshold τ on Qwen3-30B-A3B under W4A16. Underlined results indicate improvements over conventional router alignment ($\tau = 0$). Small discrepancy thresholds maintain alignment effectiveness.

τ	Ratio (%)	AIME24	AIME25	GPQA-D	MMLU-P
0	100	79.58	67.50	60.73	71.09
0.0001	48.93	<u>79.58</u>	<u>70.00</u>	<u>61.87</u>	<u>71.10</u>
0.001	35.75	<u>80.00</u>	67.08	60.54	<u>71.70</u>
0.05	11.24	78.33	<u>67.92</u>	60.10	70.93
0.1	4.33	77.92	66.25	<u>62.37</u>	70.93

objective ranging from Top- k to Top-5 k , as well as Full MSE. Among them, Top-3 k achieves the highest MMLU-Pro performance under both quantization settings, while nearby alignment scopes show comparable results without further improvements. This suggests that narrower scopes may miss near-boundary experts that become newly selected after quantization, whereas expanding the scope beyond this region increasingly includes lower-ranked experts with limited additional benefit.

Threshold Ablation. Table 4 ablates the discrepancy threshold τ used for token selection. Using a small threshold $\tau = 0.0001$ achieves comparable or improved performance over conventional RA while selecting substantially fewer tokens for alignment optimization. As the threshold increases further, the number of selected tokens decreases substantially, occasionally improving specific benchmarks but generally reducing overall performance consistency. These observations suggest that excessively aggressive filtering may exclude routing perturbations still relevant to output distribution, while a small threshold effectively removes near-zero discrepancy tokens without significantly reducing useful alignment signals.

RTN-GPTQ Mask Consistency. OPERA precomputes the token selection mask using RTN-based output discrepancy before GPTQ optimization. To examine whether this signal remains representative under GPTQ, we compare RTN- and GPTQ-derived discrepancy masks on Qwen3-30B-A3B under W4A16. At the default threshold of

Table 5: Ablation of token selection criteria on Qwen3-30B-A3B under W3A16. Ratio denotes the fraction of calibration tokens selected for router alignment. At nearly matched selection ratios, output-discrepancy-based selection achieves the highest MMLU-Pro performance. MMLU-Pro results in this table are averaged over two runs.

τ	R_{th}	Ratio	MMLU-P
0.0	0.0	100%	64.70
0.0	0.1	56%	64.92
0.1	0.1	22%	64.71
0.0001	0.1	56%	65.09
0.0001	0.0	57%	66.92

$\tau = 0.0001$, the two masks achieve a Jaccard overlap of 0.825, and 87.6% of RTN-selected tokens are also selected under GPTQ. Their token-wise discrepancy scores are also strongly correlated, with a Spearman correlation of 0.816. These results indicate that RTN provides a consistent token-prioritization signal for GPTQ-based alignment.

A GPTQ-derived mask computed once before alignment may likewise not remain fully representative throughout sequential optimization, as earlier alignment updates can affect the hidden states propagated to later layers. Recomputing discrepancy masks during optimization would require additional model evaluations. We therefore use the RTN-derived mask as a simple, precomputable proxy for token prioritization.

Token Selection Criteria. Table 5 compares different criteria for selecting tokens for router alignment, including routing shift, observable output discrepancy, and their combination. At nearly matched selection ratios of 56~57%, selecting tokens based solely on routing shift achieves an MMLU-Pro score of 64.92, while jointly applying output discrepancy and routing shift thresholds achieves 65.09. In contrast, the default OPERA configuration, which selects tokens solely based on observable output discrepancy, achieves 66.92 while selecting a comparable 57% tokens. This comparison indicates that the advantage of OPERA does not simply arise from the number of selected tokens; rather, observable output discrepancy provides a more effective signal for determining where router alignment should be emphasized. Further restricting the selected set to 22% using aggressive joint thresholding reduces performance to 64.71, suggesting that overly restrictive filtering may discard useful alignment signals.

Router Alignment Objective. We further examine whether the benefit of OPERA is specific to the

Table 6: Ablation study on router alignment objectives on Qwen3-30B-A3B under W3A16. We compare conventional uniform alignment (RA) and OPERA using the default MSE objective and an MSE+KLD objective. OPERA consistently improves over RA under both objectives. MMLU-Pro results in this table are averaged over two runs.

Objective	Method	AIME24	AIME25	MMLU-P
MSE	RA	69.58	52.50	64.70
	OPERA	72.08	55.42	66.92
MSE+KLD	RA	69.17	44.58	65.09
	OPERA	70.42	45.42	65.14

MSE objective used in our default configuration. Existing router alignment methods primarily focus on designing objectives that minimize routing discrepancies between the baseline and quantized models. In contrast, OPERA addresses a complementary aspect of router alignment by determining where alignment emphasis should be allocated across tokens based on observable output discrepancy. To examine whether this output-aware prioritization can be combined with different alignment objectives, we apply both conventional RA and OPERA using the default MSE objective and an MSE+KLD objective following EAQuant (Fu et al., 2025).

As shown in Table 6, OPERA consistently outperforms uniform RA under both objectives. Compared with the MSE objective, MSE+KLD does not improve absolute performance in this setting; nevertheless, the relative advantage of OPERA over uniform RA is preserved across both objectives. This result indicates that the effectiveness of output-aware token prioritization is not tied to a particular router-alignment loss formulation.

6 Conclusion

We analyze routing perturbations in quantized MoE models and observe that routing perturbations exhibit highly heterogeneous relationships with observable output discrepancy. Motivated by this observation, we propose OPERA, an output-aware router alignment strategy that allocates greater alignment emphasis toward tokens exhibiting observable output discrepancy. Experiments across multiple MoE LLMs and quantization settings demonstrate consistent aggregate improvements over conventional router alignment, with more noticeable gains under aggressive low-bit quantization.

Limitations

Our work has several limitations. First, OPERA uses a shared token-selection criterion across layers. Although we explored a simple layer-wise threshold variant, it did not improve over the uniform formulation, suggesting that more principled approaches are needed to account for layer-specific routing or quantization sensitivity. Second, OPERA precomputes output discrepancy using RTN before GPTQ optimization. While we observe strong consistency between RTN- and GPTQ-derived discrepancy signals, whether this consistency generalizes to other PTQ methods, models, and quantization settings remains to be studied. Finally, our main formulation adopts binary masking for simplicity. Preliminary results with continuous soft weighting show comparable or modestly improved performance, indicating that OPERA is not inherently restricted to binary selection; however, the systematic design of continuous or adaptive output-aware weighting strategies remains an interesting direction for future work.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2026-25507427, Development of Efficient Architectures and Training Techniques for High-Performance Lightweight AI Models). This work was also supported by Artificial intelligence industrial convergence cluster development project funded by the Ministry of Science and ICT (MSIT, Korea) & Gwangju Metropolitan City.

Ethical Considerations

This work focuses on improving PTQ techniques for MoE LLMs through output-aware router alignment. Our method is intended to improve computational efficiency and reduce memory requirements during inference, which may contribute to lowering the hardware cost and energy consumption associated with deploying large-scale MoE models.

At the same time, improving the efficiency and accessibility of LLMs may also facilitate broader deployment of these systems, including applications that could generate misleading, harmful, or biased content. OPERA does not introduce new capabilities beyond those already present in the underlying pretrained models, and it does not specifically

address issues related to fairness, bias, hallucination, privacy, or misuse.

In addition, our output-aware token selection relies on observable output discrepancy estimated from calibration data. The behavior of the proposed method may therefore depend on the representativeness of the calibration dataset and the underlying quantization setting. Future work may explore more robust and adaptive alignment strategies across diverse deployment scenarios.

References

- Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Yuanteng Chen, Yuantian Shao, Peisong Wang, and Jian Cheng. 2025a. Eac-moe: Expert-selection aware compressor for mixture-of-experts large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12942–12963.
- Zhixuan Chen, Xing Hu, Dawei Yang, Zukang Xu, Xu Chen, Zhihang Yuan, Sifan Zhou, and Jiangyong Yu. 2025b. Moequant: Enhancing quantization for mixture-of-experts large language models via expert-balanced sampling and affinity guidance. In *International Conference on Machine Learning*, pages 8245–8260. PMLR.
- Wenhua Cheng, Weiwei Zhang, Heng Guo, and Haihao Shen. 2025. Signroundv2: Closing the performance gap in extremely low-bit post-training quantization for llms. *arXiv preprint arXiv:2512.04746*.
- Mohammed Nowaz Rabbani Chowdhury, Kaoutar El Maghraoui, Hsinyu Tsai, Naigang Wang, Geoffrey W Burr, Liu Liu, and Meng Wang. 2026. Efficient quantization of mixture-of-experts with theoretical generalization guarantees. In *The Fourteenth International Conference on Learning Representations*.
- Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, and 1 others. 2024. Deepseek-moe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1280–1297.

- Google DeepMind. 2026. [Gemma 4 model card](#). Accessed: 2026-05-04.
- DeepSeek-AI. 2026. Deepseek-v4: Towards highly efficient million-token context intelligence.
- Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 1286–1305.
- Wei Du, Shubham Toshniwal, Branislav Kisanin, Sadegh Mahdavi, Ivan Moshkov, George Armstrong, Stephen Ge, Edgar Minasyan, Feng Chen, and Igor Gitman. 2025. Nemotron-math: Efficient long-context distillation of mathematical reasoning from multi-mode supervision. *arXiv preprint arXiv:2512.15489*.
- Yi-Zeng Fang and Juinn-Dar Huang. 2025. [Router choice matters: Rank-aware post-training quantization for moe models](#).
- William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.
- Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2023. Gptq: Accurate post-training quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*. OpenReview.
- Zhongqian Fu, Tianyi Zhao, Ning Ding, Xianzhi Yu, Xiaosong Li, Yehui Tang, and Yunhe Wang. 2025. Eaquant: Enhancing post-training quantization for moe models via expert-aware optimization. *arXiv preprint arXiv:2506.13329*.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, and 1 others. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.
- GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Hao-ran Wang, and 168 others. 2026. [Glm-5: from vibe coding to agentic engineering](#). *Preprint*, arXiv:2602.15763.
- GLM Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, and 152 others. 2025. [Glm-4.5: Agentic, reasoning, and coding \(arc\) foundation models](#). *Preprint*, arXiv:2508.06471.
- Wei Huang, Yue Liao, Jianhui Liu, Ruifei He, Haoru Tan, Shiming Zhang, Hongsheng Li, Si Liu, and XIAOJUAN QI. 2025. Mixture compressor for mixture-of-experts llms gains more. In *The Thirteenth International Conference on Learning Representations*.
- Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. [Adaptive mixtures of local experts](#). *Neural Computation*, 3:79–87.
- M.I. Jordan and R.A. Jacobs. 1993. [Hierarchical mixtures of experts and the em algorithm](#). In *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, volume 2, pages 1339–1344 vol.2.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Janghwan Lee, Seongmin Park, Sukjin Hong, Minsoo Kim, Du-Seong Chang, and Jungwook Choi. 2024. Improving conversational abilities of quantized large language models via direct preference alignment. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11346–11364.
- LG AI Research. 2025. K-evaone technical report. *arXiv preprint arXiv:2601.01739*.
- Pingzhi Li, Xiaolong Jin, Zhen Tan, Yu Cheng, and Tianlong Chen. 2024. Quantmoe-bench: Examining post-training quantization for mixture-of-experts. *arXiv preprint arXiv:2406.08155*.
- Zhaoyi Li, Jiatong Li, Gangwei Jiang, Linqi Song, Defu Lian, and Ying Wei. 2026. Scaling reasoning hop exposes weaknesses: Demystifying and improving hop generalization in large language models. *arXiv preprint arXiv:2601.21214*.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let’s verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.
- Haokun Lin, Haobo Xu, Yichen Wu, Jingzhi Cui, Yingtao Zhang, Linzhan Mou, Linqi Song, Zhenan Sun, and Ying Wei. 2024a. Duquant: Distributing outliers via dual transformation makes stronger quantized llms. *Advances in Neural Information Processing Systems*, 37:87766–87800.
- J. Lin. 1991. [Divergence measures based on the shannon entropy](#). *IEEE Transactions on Information Theory*, 37(1):145–151.

- Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024b. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6:87–100.
- Xuan Luo, Weizhi Wang, and Xifeng Yan. 2025. Adaptive layer-skipping in pre-trained llms. *arXiv preprint arXiv:2503.23798*.
- Maxwell-Jia. 2025. Aime 2024 dataset. https://huggingface.co/datasets/Maxwell-Jia/AIME_2024.
- Thong Nguyen. 2024. [Logit Prisms: Decomposing Transformer Outputs for Mechanistic Interpretability](#).
- NVIDIA. 2025a. Nemotron-sft-competitive-programming-v2. <https://huggingface.co/datasets/nvidia/Nemotron-SFT-Competitive-Programming-v2>. Hugging Face dataset.
- NVIDIA. 2025b. [Nvidia nemotron 3: Efficient and open intelligence](#). White Paper.
- Hancheol Park, Geonho Lee, Tairen Piao, and Tae-Ho Kim. 2026. Value-and-structure alignment for routing-consistent quantization of mixture-of-experts models. *arXiv preprint arXiv:2606.05688*.
- Sungrae Park, Sanghoon Kim, Jungho Cho, Gyoungjin Gim, Dawoon Jung, Mikyoung Cha, Eunhae Choo, Taekgyu Hong, Minbyul Jeong, SeHwan Joo, Minsoo Khang, Eunwon Kim, Minjeong Kim, Sujeong Kim, Yunsu Kim, Hyeonju Lee, Seunghyun Lee, Sukyung Lee, Siyoung Park, and 18 others. 2025. [Solar open technical report](#). *arXiv preprint arXiv:2601.07022*.
- Qwen Team. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.
- Qwen Team. 2026. [Qwen3.6-35B-A3B: Agentic coding power, now open to all](#).
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.
- Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. 2024. Omniquant: Omnidirectionally calibrated quantization for large language models. In *The Twelfth International Conference on Learning Representations*.
- Zunhai Su, Qingyuan Li, Hao Zhang, Weihao Ye, Qibo Xue, YuLei Qian, Yuchen Xie, Ngai Wong, and Kehong Yuan. 2025. Unveiling super experts in mixture-of-experts large language models. *arXiv preprint arXiv:2507.23279*.
- Zhou, and 1 others. 2023. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051.
- Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, and 1 others. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290.
- William Webber, Alistair Moffat, and Justin Zobel. 2010. [A similarity measure for indefinite rankings](#). *ACM Trans. Inf. Syst.*, 28(4).
- Hossein Entezari Zarch, Lei Gao, Chaoyi Jiang, and Murali Annamaram. 2025. Del: Context-aware dynamic exit layer for efficient self-speculative decoding. *arXiv preprint arXiv:2504.05598*.
- Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

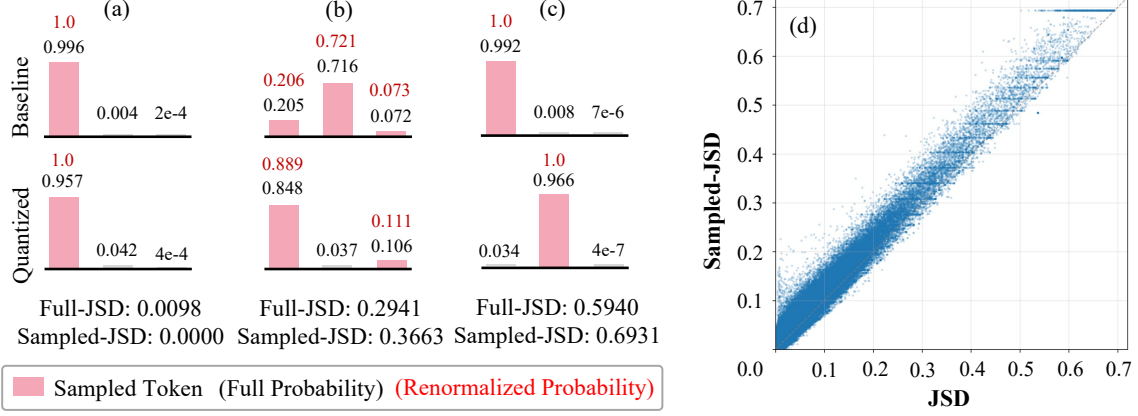


Figure 9: Comparison between standard JSD and Sampled-JSD under quantization. (a-c) Representative examples illustrating how Sampled-JSD suppresses low-probability vocabulary noise while emphasizing differences within sampled candidate tokens. Red values indicate probabilities after renormalization over the sampled candidate set. (d) Token-wise comparison between full-vocabulary JSD and Sampled-JSD. While both metrics remain strongly correlated overall, Sampled-JSD produces sharper separation between negligible and meaningful output discrepancy.

A Sampled-JSD

To quantify output discrepancy between the baseline and quantized models, we use Jensen-Shannon divergence (JSD) (Lin, 1991) over token-level probability distributions. Let p^* and $\hat{p}$ denote the probability distributions obtained from the baseline and quantized models, respectively. Standard JSD, a symmetric and bounded metric, is defined as:

$$\text{JSD}(p^*, \hat{p}) = \frac{1}{2} D_{\text{KL}}(p^* \parallel m) + \frac{1}{2} D_{\text{KL}}(\hat{p} \parallel m), \quad (10)$$

where $m = \frac{1}{2}(p^* + \hat{p})$.

However, full-vocabulary JSD may include differences originating from low-probability regions that are unlikely to affect practical decoding behavior, where generation is typically determined by a relatively small candidate token set. To better reflect observable output discrepancy, we compute JSD only over the union of candidate token sets considered during sampling. Specifically, we apply the same sampling procedure to both models, including temperature scaling and top- k /top- p filtering, to obtain candidate token sets P^* and $\hat{P}$. We then construct the union support:

$$U = P^* \cup \hat{P}. \quad (11)$$

We denote by p_U^* and $\hat{p}_U$ the distributions restricted to the union candidate set U , where tokens not present in the original candidate set are assigned zero probability. The proposed Sampled-JSD is computed as:

$$\text{JSD}_{\text{Sampled}} = \text{JSD}(p_U^*, \hat{p}_U). \quad (12)$$

By restricting comparison to tokens considered during generation, Sampled-JSD suppresses low-probability noise while preserving differences more directly related to observable decoding behavior.

Sampled-JSD is bounded between 0 and $\ln 2$ (approximately 0.69). A value of 0 indicates identical distributions, while values approaching $\ln 2$ indicate that the two models place probability mass on different candidate tokens within the sampled support.

Figure 9 illustrates several representative examples comparing full-vocabulary JSD and Sampled-JSD. In cases where both models place probability mass on the same sampled token despite minor low-probability differences elsewhere in the vocabulary, Sampled-JSD becomes nearly zero while full-vocabulary JSD remains nonzero (a). Conversely, when the sampled candidate token changes after quantization, Sampled-JSD approaches its maximum value even if full-vocabulary JSD remains substantially smaller than $\ln 2$ due to probability mass distributed across low-probability vocabulary regions (c).

Figure 9 (d) compares token-wise full-vocabulary JSD and Sampled-JSD. While both metrics remain strongly correlated overall, Sampled-JSD produces sharper separation between negligible and meaningful output discrepancy by reducing the influence of low-probability vocabulary noise. These observations suggest that Sampled-JSD provides a more stable estimate of discrepancy under practical decoding settings.

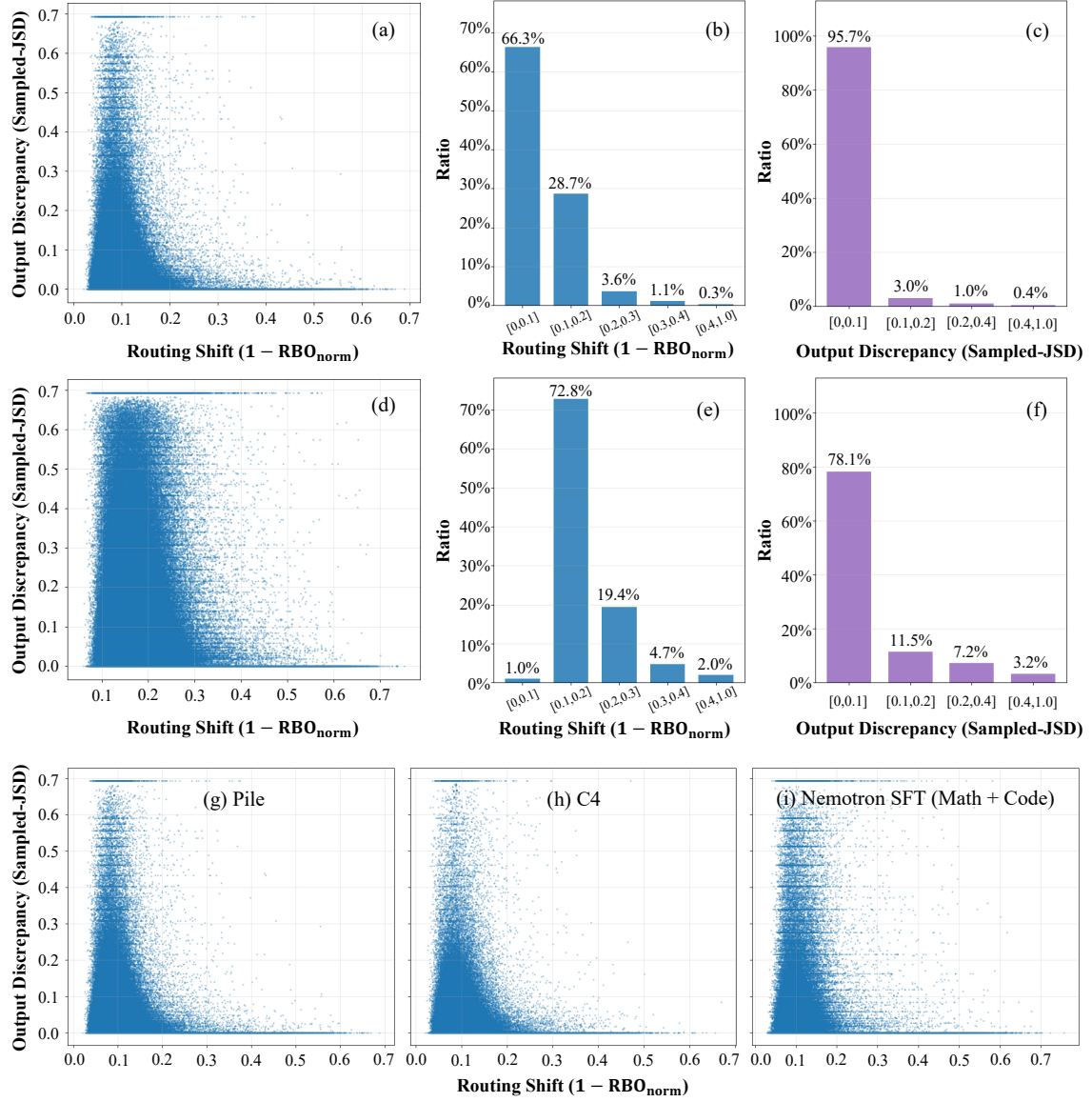


Figure 10: Additional routing perturbations and observable output discrepancy distributions across quantization settings and datasets. (a-c) Distribution of routing shift and output discrepancy under W4A16 quantization. (d-f) Distribution of routing shift and output discrepancy under W3A16 quantization. More aggressive quantization produces substantially larger routing perturbations, while observable output discrepancy remains concentrated in a relatively smaller subset of tokens. (g-i) Routing shift and observable output discrepancy across different datasets, including Pile, C4, and Nemotron-SFT. Similar heterogeneous relationships are consistently observed across different data distributions and domains.

B Routing Shift and Output Discrepancy Analysis

Distribution of Routing Shift and Output discrepancy. To complement the scatter plots in Section 3.3, we further analyze the distributions of routing shift and output discrepancy using histogram-based binning. Specifically, we report the proportion of tokens falling into different value ranges for both routing shift and Sampled-JSD under 4-bit (W4A16) and 3-bit (W3A16) quantization. The results are shown in Figure 10 (a-f).

Under the 4-bit setting, both routing shift and output discrepancy are heavily concentrated in low-value regions. In particular, more than 95% of tokens exhibit Sampled-JSD values below 0.1, indicating that the most tokens preserve highly similar output distributions after quantization. Routing shift also remains relatively small for most tokens under W4A16.

In contrast, under the 3-bit setting, both routing perturbations and output discrepancy become substantially larger. The proportion of tokens with Sampled-JSD below 0.1 decreases to approximately 78% reflecting increased observable output discrepancy under more aggressive quantization. Routing shift exhibits an even larger distributional shift, with only a small fraction of tokens remaining below 0.1 and most tokens distributed in higher routing shift ranges.

These observations suggest that routing perturbations become increasingly widespread under aggressive quantization, while observable output discrepancy remains comparatively concentrated in a smaller subset of tokens.

Generalization Across Datasets. We further examine whether the observed relationship between routing shift and output discrepancy generalizes across different datasets. In addition to the Pile used in the main experiments, we evaluate datasets with different characteristics, including C4 (Dodge et al., 2021) as another pretraining corpus and Nemotron-SFT datasets such as Nemotron-Math (Du et al., 2025) and Nemotron-Code (NVIDIA, 2025a) datasets containing supervised fine-tuning data for math and code domains.

Figure 10 (g-i) presents token-wise scatter plots of routing shift and output discrepancy across different datasets. Overall, we observe highly consistent patterns across all settings: many tokens retain near-zero output discrepancy despite noticeable routing perturbations, while observable output

Word/Subword
- cursor key zone (indicated by darker grey coloring)\n editing function zone (indicated by lighter grey coloring) - The current pal ly internal state, fields will be mutated by\n - a horse with stats MUST be the top rated horse for the profit able condition to be valid
Digit
- one for photos taken in timezone -02:00 (Brazil daylight the AltGr key.\n\nSpecial issues\n\nISO/IEC 9995-2 \n\nISO/ - Low-end RAID: 200-300 MB/s\n\nHigh-end RAID: 1000-2
Punctuation
- it's going to run.\n\nfunction add (x, y) { return x + y } - Let \$A_{1}\$ and \$A_{2}\$ denote the treatments. (Jockey (T)rainer (S)peed (F)inish (E)arnings (W)orkouts
Whitespace
- For example, do not use a 3.2 or\n earlier version of - Wait, I thought it would be positive 6 \non the bottom too? - The x1 range is split into three subranges 201, 202, and 203,

Figure 11: Representative examples of HRLO tokens (highlighted). Highlighted tokens exhibit high routing shift while maintaining near-zero observable output discrepancy after quantization. Many correspond to highly predictable token types such as common subwords, digits, punctuation, and whitespace.

discrepancy remains concentrated in a relatively smaller subset of tokens.

These results suggest that the heterogeneous relationship between routing perturbations and observable output discrepancy is not specific to a particular dataset, but rather reflects a more general behavior of quantized MoE models across different data distributions and domains.

C HRLO Examples

Figure 11 presents representative examples of HRLO tokens identified in our analysis. Highlighted tokens correspond to tokens exhibiting substantial routing perturbations while maintaining near-zero observable output discrepancy after quantization.

Consistent with the entropy and category analyses in Section 3.3.2, many HRLO tokens correspond to highly predictable token types, including whitespace, punctuation, digits, and common subwords. These examples qualitatively illustrate that output-preserving routing perturbations frequently occur on tokens whose next-token prediction remains relatively stable under local representation perturbations.

Table 7: Benchmark configuration.

Benchmark	Max Generation Tokens
AIME24,25	16384
MATH-500	16384
GPQA-Diamond	16384
MMLU-Pro	4096
IFEval	2048
HumanEval	2048
BBH	2048

D Benchmark Settings

Table 7 summarizes the maximum generation lengths used for each benchmark. AIME24/25 are evaluated using pass@4, while all remaining benchmarks use pass@1.

Table 8: Comparison of the default hard-selection OPERA and a Sampled-JSD-based soft-weighting variant on Qwen3-30B-A3B under W4A16. The soft-weighting variant provides modest improvements, suggesting that output-aware router alignment is not limited to binary token selection. MMLU-Pro results in this table are averaged over two runs.

Method	MMLU-Pro	GPQA-D
OPERA (hard selection)	71.16	61.87
Soft weighting	71.28	61.93

E Additional Ablations

Soft-weighting Variant. The default OPERA formulation uses a binary selection mask to provide a simple mechanism for prioritizing tokens with observable output discrepancy. To examine whether the proposed output-aware alignment perspective is specific to binary masking, we additionally evaluate a soft-weighting variant that continuously adjusts the contribution of each token to the router alignment objective based on its Sampled-JSD score.

As shown in Table 8, the soft-weighting variant yields small but consistent improvements over the default hard-selection scheme on both MMLU-Pro and GPQA-D. This result suggests that the benefit of OPERA is not restricted to a particular binary masking formulation, but can extend to continuous allocation of alignment emphasis. We view this experiment as a preliminary exploration rather than a fully developed weighting strategy. A more systematic design of continuous or adaptive output-aware weighting functions is left for future work.

Table 9: Comparison of the default uniform OPERA and a simple layer-wise variant on Qwen3-30B-A3B under W4A16. The layer-wise variant does not improve over the default formulation, suggesting that straightforward layer-dependent selection alone may be insufficient. MMLU-Pro results in this table are averaged over two runs.

Method	MMLU-Pro	GPQA-D
OPERA (uniform)	71.16	61.87
Layer-wise variant	70.47	61.24

Layer-wise Variant. The default OPERA formulation applies a shared token-selection threshold across layers. Motivated by the layer-wise variation in routing perturbations observed in Figure 6, we additionally evaluate a simple layer-wise variant that uses smaller thresholds in later layers and larger thresholds in earlier layers. As shown in Table 9, this variant does not improve over the default uniform formulation on either MMLU-Pro or GPQA-D. This result suggests that straightforward layer-dependent thresholding alone may be insufficient, and that more principled approaches accounting for how routing perturbations and output sensitivity vary across layers remain an interesting direction for future work.

F Latency

Table 10: Inference efficiency comparison between BF16 and W4A16. Inference is measured on Qwen3-30B-A3B using vLLM with a single H100 80GB GPU, input/output length of 15k tokens, and maximum concurrency of 4. Quantization improves throughput, TTFT, TPOT, and available KV cache capacity compared to BF16.

Bit	BF16	W4A16
Output TPS (tok/s)↑	389.04	469.26
Total TPS (tok/s)↑	778.09	938.53
TTFT (ms)↓	1353.27	1333.63
TPOT (ms)↓	10.19	8.44
Available KV Cache (tok)↑	126,368	583,216

Although inference acceleration is primarily a benefit of quantization itself rather than the proposed alignment method, we provide additional measurements to illustrate the practical efficiency gains enabled by low-bit MoE deployment.

We measure inference performance on Qwen3-30B-A3B using vLLM (Kwon et al., 2023) with

a single H100 80GB GPU. The evaluation is conducted with an input length of 15k tokens, an output length of 15k tokens, and a maximum concurrency of 4 requests. We compare BF16 and W4A16 settings in terms of throughput, time-to-first-token (TTFT), time-per-output-token (TPOT), and available KV cache capacity.

The results are summarized in Table 10. Quantization improves all measured metrics compared to BF16 inference. In particular, TPOT is substantially reduced under W4A16, leading to noticeable improvement in overall throughput. In addition, the reduced model memory footprint enables significantly larger available KV cache memory, allowing more tokens to be cached during inference. This is particularly beneficial for long-context generation scenarios, where KV cache capacity often becomes a major memory bottleneck.