HAMoE: Heterogeneity-Aware Mixed-Precision Quantization for Mixture-of-Experts Models

Anonymous authors
Paper under double-blind review

Abstract

Mixture-of-Experts (MoE) models face deployment challenges in resource-constrained devices due to their massive parameter scales. Quantization is recognized as one of the most effective approaches to address such challenges. In this work, we conduct a systematic quantization study of MoE models and identify two critical heterogeneity insights that have not been considered in prior work: 1) inter-expert heterogeneity, i.e., distinct activation frequencies and processed token types across experts, and 2) intra-expert heterogeneity, i.e., differential quantization sensitivity of weights within a single expert. To this end, we introduce HAMoE, a mixed-precision quantization framework that considers both heterogeneities. HAMoE assigns fine-grained mixed precision to inter-experts based on activation frequency and token type. In addition, it proposes an adaptive column replacement mechanism, which replaces the most quantization-sensitive intra-expert weight columns with INT8-quantized versions, thereby effectively reducing quantization loss. Evaluation results show that HAMoE achieves competitive results against strong baselines under equivalent bit budgets, reducing WikiText-2 perplexity by 4.0% relative to GPTQ at 3.35-bit, recovering about 45% of the quantization loss. HAMoE also consistently outperforms MxMoE in both average accuracy and perplexity across all four models.

1 Introduction

As artificial intelligence enters the era of large language models (LLMs), the applications of these models have experienced explosive growth. However, during both training and inference, dense LLMs exhibit a strict positive correlation between computational cost and the number of parameters, making further parameter scaling constrained by hardware limits. To overcome this bottleneck, the Mixture-of-Experts (MoE) architecture has emerged as a promising solution (Cai et al., 2025; Fedus et al., 2022). The feature of the MoE architecture lies in its sparse activation mechanism, which replaces dense multi-layer perceptron (MLP) blocks with several expert models, each adapted to a specific type of input data, enabling better fitting of diverse data distributions. During inference, MoE introduces gates to handle dynamic routing, activating a subset of experts for each input token.

Despite superior scalability and performance, MoEs face deployment challenges due to their massive total parameter count. Taking Mixtral 8×7B (Jiang et al., 2024) as an example, it activates only 13B parameters during inference, yet the total parameters amount to 47B, imposing high demands on memory capacity and bandwidth. Particularly in resource-constrained devices, e.g., edge device or single-card setups, this storage overhead renders MoE deployment unfeasible, making compression to reduce footprint urgently needed (Gan et al., 2026; Aghdam et al., 2024; Zhong et al., 2025). Among various compression techniques, quantization is widely recognized as one of the most direct and effective approaches. Quantization maps model parameters from high precision (e.g., FP16) to low-precision representations (e.g., INT4, INT8), enabling multiplicative compression of model size (Gong et al., 2025).

Current MoE quantization methods primarily adopt mixed-precision strategies, assigning higher precision to more important experts (Duanmu et al., 2025; Hu et al., 2025; Zheng et al., 2025). However, existing methods suffer from two key limitations under a fixed bit budget. First, they

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.

largely evaluate expert importance through usage or parametric vulnerability, while overlooking a third, orthogonal dimension, that of function. Specifically, we observe that experts exhibit distinct token-type preferences, and those responsible for semantically rich tokens (e.g., nouns, adjectives) induce significantly greater performance degradation when quantized to the same low precision, compared to those handling semantically narrow tokens (e.g., punctuation, determiners). In addition, they assign precision at the expert or layer level, overlooking intra-expert weight heterogeneity and treating all weights within an expert equally despite their widely varying quantization sensitivity.

To address these issues, we propose HAMoE, a heterogeneity-aware mixed-precision quantization framework for MoE models that accounts for inter-expert and intra-expert heterogeneity under a fixed bit budget. This method involves initially classifying experts based on two key dimensions, activation frequency and token type, and then assigning fine-grained mixed precision to each expert group. To address the quantization sensitivity heterogeneity within individual experts, we propose an adaptive column replacement mechanism, which identifies the most quantization-sensitive weight columns in each expert and replaces them with INT8-quantized versions. The main contributions are summarized as follows.

- We propose HAMoE, a fine-grained mixed-precision quantization method that addresses inter-expert heterogeneity by grouping experts based on activation frequency and token type to assign differentiated precision. Compared to the strongest single dimension baseline for each metric, HAMoE recovers 19.8% of the accuracy loss and 7.5% of the perplexity loss from the FP16 baseline on Qwen1.5-MoE.
- HAMoE further addresses intra-expert heterogeneity with an adaptive column replacement mechanism, which reduces quantization-induced accuracy degradation by protecting highly sensitive weight columns in expert layers. Protecting 1% of sensitive columns yields perplexity reductions ranging from 1.0% to 4.1% across four MoE models, achieving near-optimal performance.
- Extensive experiments show that HAMoE consistently outperforms state-of-the-art MoE quantization baselines on representative benchmarks, achieving a superior accuracy and compression tradeoff. For example, at 3.35-bit on WikiText-2, HAMoE reduces perplexity by 4.0% relative to GPTQ, and by 3.5% relative to MxMoE.

2 Background and Related Work

2.1 Mixture-of-Experts

The MoE architecture replaces the feed-forward network (FFN) layer with a gate and a set of expert networks (Fedus et al., 2022; Shen et al., 2024). The gate activates only the top-K experts per token, enabling sparse computation (Cai et al., 2025; Muennighoff et al., 2025). This design allows MoE models to scale total parameters while keeping inference FLOPs comparable to a smaller dense model. Some recent MoE architectures adopt a hybrid design with both routed and shared experts, e.g., DeepSeek-V2-Lite (DeepSeek-AI, 2024) and Qwen1.5-MoE-A2.7B (Team, 2024). Figure 1 illustrates the overall architecture.

2.2 Quantization

For a weight matrix W, the uniform quantization function is defined as

$$\hat{W} = \text{round}\left(\frac{W - z}{\Delta}\right) \cdot \Delta + z,$$

where Δ is scale, z is zero point, and round$(\cdot)$ denotes rounding to the nearest integer (Jacob et al., 2018; Gholami et al., 2021; Yao et al., 2021). Dequantization recovers $\hat{W}$ from the quantized representation.

2.3 Related Work

Post-training quantization (PTQ) compresses pre-trained models using a small calibration set without retraining, favoring efficiency and simplicity. For instance, GPTQ uses approximate second-order information to quantize LLMs to 3-4 bits with negligible accuracy loss (Frantar et al., 2022).

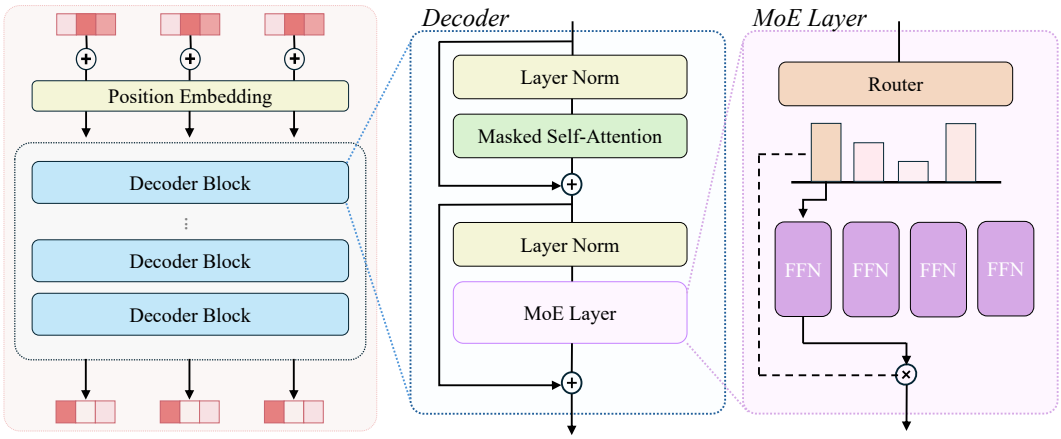

Figure 1: MoE layer architecture.

QuantEase frames layer-wise quantization as a discrete optimization problem solved via coordinate descent (Behdin et al., 2023). SmoothQuant migrates quantization difficulty from activations to weights via a mathematically equivalent transformation for W8A8 quantization (Xiao et al., 2023). AWQ protects salient weight channels identified by activation distribution via scaling to reduce quantization error (Lin et al., 2024). RPTQ reorders channels and quantizes them in clusters to address varying activation ranges across channels (Yuan et al., 2023). These methods effectively address the quantization challenges specific to dense LLMs.

Several recent studies have also explored quantization strategies for MoE models. QMoE (Frantar & Alistarh, 2023) is a system-level compression method that achieves sub-1-bit quantization of trillion-parameter MoE models. MoQa (Zheng et al., 2025) attributes the failure of traditional methods on MoE to complex data-model distribution characteristics and addresses them via expert-level base quantization and channel-level dynamic adjustment. MoQAE (Tao et al., 2025) treats different bit-width configurations as experts for dynamic selection at inference time. While these methods adopt different technical approaches, they share a common reliance on usage-based signals such as routing frequency, activation statistics, or data-model distribution to guide quantization decisions. Other methods incorporate parametric vulnerability as an additional signal. MxMoE (Duanmu et al., 2025) combines expert activation frequency with linear-block-level Hessian-based sensitivity for mixed-precision allocation. MoEQuant (Hu et al., 2025) uses expert-balanced sampling and affinity-guided quantization to implicitly capture parametric sensitivity alongside usage patterns. MoQE (Li et al., 2024) explores MoE quantization heuristics across granularities from MoE blocks to individual linear layers to identify bit-width allocation principles. SFMoE (Qi et al., 2027) applies cooperative game theory and fuzzy logic to model cross-layer interactions and accelerate sensitivity profiling while maintaining near-FP16 performance. A detailed comparison of these methods across the three dimensions is provided in Appendix A.

3 Motivation

3.1 Inter-Expert Heterogeneity

Mixture-of-Experts architectures rest on the premise that different experts specialize in different aspects of the input distribution. While prior work measures expert importance via routing statistics, a more fundamental question remains unexplored: *do different experts process different types of tokens, and if so, does this functional specialization affect their sensitivity to quantization?*

To investigate this question, we analyze token-level routing behavior across four MoE models: Qwen1.5-MoE-A2.7B, DeepSeek-V2-Lite, OLMoE-1B-7B (Muennighoff et al., 2025), and MoE-Girl-1B-7B (Allura-org, 2024). For each model, we run inference on 300 WikiText-2 (Merity et al., 2016) training samples, recording each token's routing decision and linguistic category. spaCy (Honnibal et al., 2020) assigns part-of-speech (POS) tags, which we group into four functional categories by semantic role.

- **Content Words**: `NOUN`, `VERB`, `ADJ`, `ADV`, `PROPN`.

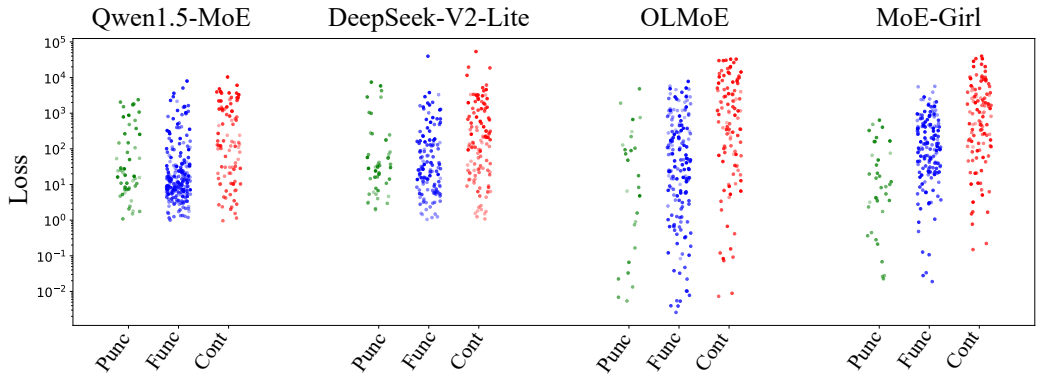

Figure 2: Quantization loss of experts with similar activation frequencies under INT2 quantization, grouped by their dominant functional category.

- **Function Words**: `ADP`, `AUX`, `CCONJ`, `SCONJ`, `DET`, `PART`, `PRON`.
- **Special Symbols**: `NUM`, `SYM`, `X`, `INTJ`.
- **Punctuation**: `PUNCT`.

Definition of Specialization. For an expert, we consider a category significantly preferred if its token proportion exceeds 50%, and extremely specialized if it exceeds 80%. For example, Expert 14 at Layer 2 of Qwen1.5-MoE processes 90.42% function words, while Expert 42 at Layer 25 of DeepSeek-V2-Lite processes 96.81% punctuation.

To quantify this phenomenon, we compute the maximum single-category proportion for every expert across all four models; the full distribution is provided in Appendix B. In Qwen1.5-MoE, 88.9% of experts have a dominant category proportion above 40% and 47.2% above 50%, both far exceeding the 25% expected under a random distribution. This pattern holds consistently across all four models, confirming that functional specialization is a universal characteristic of MoE architectures rather than a model-specific artifact.

Quantization Loss Across Functions. To eliminate the confounding effect of activation frequency, we select within each model a set of experts with nearly identical activation frequencies (differing by less than 1%), covering three functional categories (special symbols are excluded due to their low occurrence), each with a dominant category proportion exceeding 50%. We then quantize these experts to INT2 and measure the resulting loss. As shown in Figure 2, experts processing content words exhibit substantially higher quantization loss than those processing punctuation or function words, despite comparable activation frequencies. Notably, this gap is consistent across all four models, confirming that the phenomenon is model-agnostic.

Together, our analysis establishes that (1) MoE experts exhibit systematic functional specialization in token processing, and (2) this specialization directly determines their vulnerability to quantization.

3.2 Intra-Expert Heterogeneity

Within a single expert, weight columns tolerate quantization to markedly different degrees. To quantify this column-level disparity, we draw upon the Optimal Brain Surgeon (OBS) framework (Hassibi et al., 1993), where parameter importance is measured by the second-order derivative of the loss. For an expert weight matrix $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, we compute the Hessian matrix $H = XX^\top$ from calibration activations X, and define the sensitivity of column j as

$$\text{sensitivity}_j = |H_{jj}| \cdot \|W_{:,j}\|_2$$

where H_{jj} is the j-th diagonal element of the Hessian, reflecting the activity level of the corresponding input feature, and $\|W_{:,j}\|_2$ is the column's weight norm, reflecting its magnitude. Their product serves as an effective proxy for the expected impact of quantizing that column on the model output.

To validate this metric, we compute the sensitivity of every column in each expert, sort the columns in descending order, and accumulate their sensitivity. Figure 3 shows the average cumulative distribution across experts for all four MoE models, revealing a striking pattern that **1% of columns**

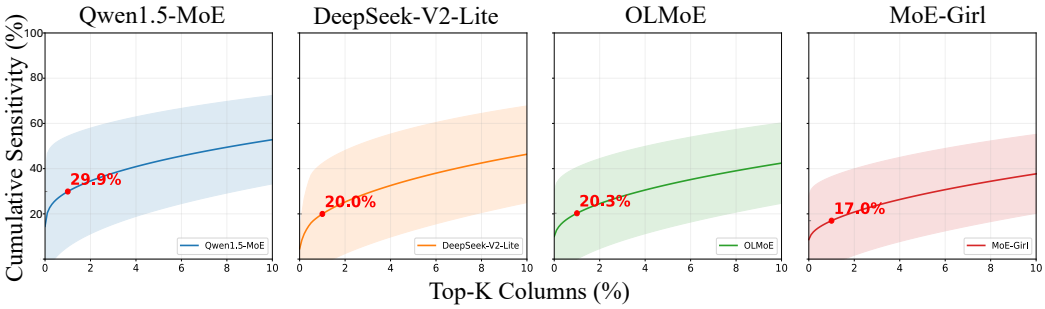

Figure 3: Cumulative sensitivity captured by the top-$k\%$ most sensitive columns across experts.

account for roughly 20% of the total sensitivity mass. This concentration holds consistently across all four models, confirming that it reflects a general structural property of MoE expert weight matrices rather than a model-specific artifact. This motivates us to assign higher precision, such as INT8, to high-sensitivity columns within each expert to substantially reduce quantization loss with negligible storage overhead.

4 METHOD

4.1 TOKEN AND ACTIVATION-AWARE MOE QUANTIZATION

Motivated by the two heterogeneity patterns presented in Section 3, We design a fine-grained mixed precision quantization method that scores expert importance from two dimensions, activation frequency and token type, and assigns precision accordingly. High-frequency experts amplify quantization errors and thus warrant higher precision, while experts specializing in semantically rich tokens are more vulnerable to low-bit quantization, as shown in Section 3.1. We quantify both dimensions and combine them into a single importance score, based on which precision is allocated under a fixed average bit budget.

Activation Frequency Dimension. As shown in Figure 4, expert activation frequencies follow a long-tail distribution, with high-frequency experts amplifying quantization errors and thus warranting higher precision. We therefore regard activation frequency as a key indicator of expert importance. Let $\mathcal{D}$ denote the calibration dataset, and let $N_{l,i}$ be the number of samples in $\mathcal{D}$ that activate expert i in layer l. We define the activation frequency $F_{\text{expert}_{l,i}}$ as

$$F_{\text{expert}_{l,i}} = \frac{N_{l,i}}{|\mathcal{D}|}.$$

Since activation frequencies may differ by several orders of magnitude, we apply a logarithmic transformation before normalization.

Token Type Dimension. To quantify the importance of tokens processed by each expert, we use spaCy to assign each token a part-of-speech (POS) tag and group the tags into four functional categories, i.e., content words, function words, special symbols, and punctuation, each assigned a semantic weight $w(t)$ (see Appendix C). Let $\mathcal{T}_{l,i}$ be the set of tokens routed to expert i in layer l. We define the token importance $T_{\text{expert}_{l,i}}$ as

$$T_{\text{expert}_{l,i}} = \frac{1}{|\mathcal{T}_{l,i}|} \sum_{t \in \mathcal{T}_{l,i}} w(t).$$

Comprehensive Importance Score. We normalize both dimensions to $[0, 1]$ using min-max normalization

$$F_{\text{norm}_{l,i}} = \frac{\log(F_{\text{expert}_{l,i}}) - \min(\log(F))}{\max(\log(F)) - \min(\log(F))},$$

and

$$T_{\text{norm}_{l,i}} = \frac{T_{\text{expert}_{l,i}} - \min(T)}{\max(T) - \min(T)},$$

where $\min(\log(F))$ and $\max(\log(F))$ are the minimum and maximum of $\log(F_{\text{expert}})$, and $\min(T)$ and $\max(T)$ are the minimum and maximum of T_{expert} over all experts, respectively. Based on the

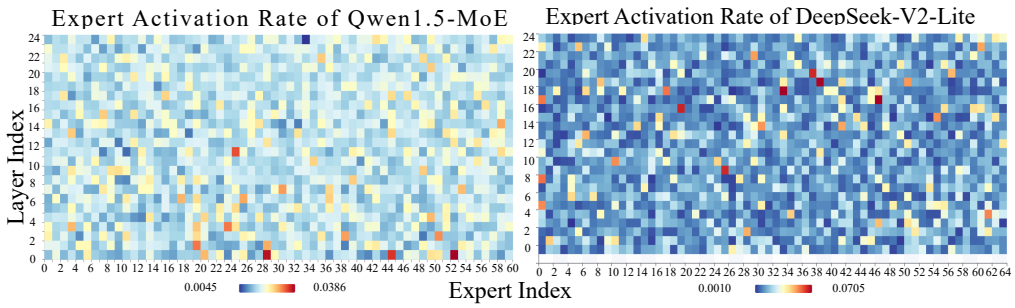

Figure 4: Activation frequency distribution across experts, highlighting a long-tail pattern.

normalized two dimensions, we define the comprehensive importance score for expert $e_{l,i}$ as

$$\text{Score}_{l,i} = \alpha F_{\text{norm}_{l,i}} + (1-\alpha) T_{\text{norm}_{l,i}},$$

where α balances the two dimensions. Using perplexity on the WikiText-2 validation set, we searched $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$ and found $\alpha = 0.7$ best on three of the four models. We use it as the shared default (Appendix D). A higher Score indicates higher quantization sensitivity and thus deserves higher precision.

Expert Classification and Precision Allocation. Given the comprehensive importance score of each expert, we allocate precision such that higher-scoring experts receive higher precision under a target average bit budget B_{target}. Let N be the total number of experts, $\mathcal{B} = \{2, 3, 4, 6, 8\}$ the allowable bit-widths, and $b_i \in \mathcal{B}$ the precision assigned to expert i. We formulate precision allocation as a constrained optimization problem.

$$\textbf{MINIMIZE} \quad \mathcal{L}(\mathbf{b}) = \sum_{i=1}^{N} \frac{s_i}{b_i}$$

$$\textbf{WHERE} \quad s_i = \text{ImportanceScore}(E_i)$$

$$\textbf{SUBJECT TO} \quad \left| \frac{1}{N} \sum_{i=1}^{N} b_i - B_{\text{target}} \right| \leq \epsilon$$

$$\textbf{SUBJECT TO} \quad \sum_{i=1}^{N} \mathbb{I}(b_i = 8) \geq \lceil 0.02N \rceil, \quad \sum_{i=1}^{N} \mathbb{I}(b_i = 6) \geq \lceil 0.03N \rceil$$

We solve this discrete problem with Gurobi (Gurobi Optimization, LLC, 2026), assigning one precision to each expert subject to the bit budget and group quotas.

4.2 Adaptive Column Replacement Mechanism

In Section 3.2, we demonstrated that a small fraction of columns within each expert exhibit disproportionately high sensitivity to quantization. Motivated by this observation, we propose an adaptive column replacement mechanism to address the quantization sensitivity heterogeneity within individual experts. The core idea is to identify the most quantization-sensitive columns(1%) within each expert and replace them with INT8-quantized versions. The optimality of this 1% ratio will be discussed in the ablation studies.

To quantitatively measure column sensitivity, we draw inspiration from the Optimal Brain Surgeon (OBS) framework (LeCun et al., 1989; Hassibi et al., 1993). This framework quantifies parameter importance using the Hessian matrix for model pruning. We adapt this concept to the quantization task by defining column sensitivity based on the diagonal elements of the Hessian matrix and the magnitude of the weight columns. For expert weight matrix $W_e \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, we compute the Hessian $H = XX^\top$ from input activations X and the L2 norm $\|W_{e,:,j}\|_2$ for each column. The sensitivity of column j is defined as

$$\text{sensitivity}_j = |H_{jj}| \cdot \|W_{e,:,j}\|_2,$$

where H_{jj} is the j-th diagonal element of H. A larger product indicates higher sensitivity to quantization error. We select the top $k\%$ columns with the highest value, with $k = 1$ typically to protect only the most critical parameters.

After identifying sensitive columns, we first quantize all columns to target low precision, then replace the sensitive columns with their INT8-quantized versions. Let $\mathcal{C}_e$ denote the set of sensitive column indices for expert e. The quantization process is

$$\hat{W}_{e,:,j} = \begin{cases} \text{quantize}_{\text{INT8}}(W_{e,:,j}), & j \in \mathcal{C}_e, \\ \text{quantize}_{\text{target-precision}}(W_{e,:,j}; s, z), & j \notin \mathcal{C}_e, \end{cases}$$

where s and z are scale and zero point. Thus, quantization loss mainly comes from non-sensitive columns, while sensitive columns incur substantially smaller loss than the target low precision. Protecting a small fraction of columns mitigates accuracy degradation without significantly increasing storage overhead.

5 EXPERIMENT

5.1 EXPERIMENTAL SETUP

Model Configurations. We evaluate four mainstream MoE architectures: DeepSeek-V2-Lite, Qwen1.5-MoE-A2.7B, OLMoE-1B-7B, and MoE-Girl-1B-7B (see Table 8 for specifications). The first two adopt a hybrid design with shared and routed experts, while the latter two use routed experts only. We quantize only the MoE modules and keep attention modules in full precision.

Calibration and Hardware Configuration. We use 128 sequences of 4096 tokens randomly sampled from the WikiText-2 training set with seed 42 for offline calibration. Evaluation follows LM-Evaluation-Harness (v0.4.0) (Gao et al., 2024) with zero-shot prompts, batch size 32, and maximum sequence length 4096. WikiText-2 perplexity is computed on the test split, which is disjoint from the calibration data. All experiments run on a single NVIDIA RTX 4090 GPU, costing roughly 6 GPU-hours in total, with per-model execution time ranging from 1 to 2 hours depending on model size.

Baselines. We compare against FP16 and five quantization methods, GPTQ(Frantar et al., 2022), MoQE(Li et al., 2024), MxMoE(Duanmu et al., 2025), MoQa(Zheng et al., 2025), and SFMoE(Qi et al., 2027). All baselines are reproduced under identical evaluation settings, except MoQa, whose official code is unavailable and whose results are taken from the original paper. Other MoE-specific methods such as MoEQuant, MoQAE, and QMoE are excluded, as their code is unavailable and their reported metrics are not directly comparable.

5.2 ACCURACY RESULTS

Table 1 presents per-dataset accuracy (ARC-Challenge/Easy (Clark et al., 2018), HellaSwag (Zellers et al., 2019), LAMBADA-openai/standard (Paperno et al., 2016), PIQA (Bisk et al., 2020), WinoGrande (Sakaguchi et al., 2019)) and perplexity for four MoE models across multiple bit-width settings. Overall, quantization leads to inevitable performance degradation compared to the FP16 baseline. This degradation is reflected in increased perplexity (PPL, lower values indicate better performance) and decreased average accuracy (Avg, higher values indicate better performance).

HAMoE consistently outperforms GPTQ, MoQE, MxMoE, and SFMoE across all four MoE models and all tested bit-widths, in both average accuracy and perplexity metrics. For instance, at 3.35-bit on Qwen1.5-MoE, HAMoE achieves an average accuracy of 67.19%, surpassing GPTQ (66.20%) by 0.99 points, MoQE (66.19%) by 1.00 points, MxMoE (66.33%) by 0.86 points, and SFMoE (66.44%) by 0.75 points, while reducing PPL from 7.46, 7.93, 7.42, and 7.58 to 7.16. This performance trend is consistent across all four evaluated MoE models, suggesting generality across the tested architectures. Under the same bit setting of MoQa, HAMoE also achieves favorable performance. In the 4-bit setting, HAMoE achieves near-lossless performance, as evidenced on the MoE-Girl model, where the perplexity increases by only 0.02 compared to the FP16 baseline and the average accuracy is almost fully preserved. Although HAMoE slightly lags behind or ties with MxMoE on a small number of individual metrics, the differences are marginal and do not undermine its overall effectiveness. These experimental results indicate that HAMoE effectively preserves the critical weight distributions of MoE models, enabling a favorable trade-off between compression rate and accuracy preservation.

5.3 ABLATION STUDIES

Ablation on Token and Activation-Aware MoE Quantization. To validate the individual contributions of our token and activation-aware MoE quantization strategy, we compare the full HAMoE

Table 1: Quantization Results of GPTQ, MoQE, MxMoE, SFMoE, MoQa and HAMoE on Perplexity and Accuracy. Bold indicates the best result among all methods under the same bit setting.

Model	Method	Bits	AC↑	AE↑	HS↑	LO↑	LS↑	PQ↑	WG↑	Avg↑	PPL↓
Qwen1.5-MoE	FP16	16	44.28	69.15	77.27	71.92	64.82	80.47	68.90	68.12	6.79
	HAMoE	4	43.43	68.64	76.75	70.85	64.74	80.20	68.51	67.59	6.92
	GPTQ	3.35	42.74	66.50	75.81	67.69	62.22	79.54	68.92	66.20	7.46
	MoQE	3.35	43.46	66.99	75.28	67.48	62.09	79.48	68.54	66.19	7.93
	MxMoE	3.35	43.41	66.50	**75.90**	69.17	62.11	79.05	68.17	66.33	7.42
	SFMoE	3.35	43.37	67.29	**75.90**	68.85	61.93	79.44	68.33	66.44	7.58
	HAMoE	3.35	**44.11**	**68.39**	75.79	**69.32**	**63.35**	**79.38**	**70.01**	**67.19**	**7.16**
	MoQa*	3.04	27.13	43.69	/	/	/	68.72	/	/	8.51
	HAMoE	3.04	**39.59**	**62.16**	74.08	64.39	61.25	**78.29**	64.64	63.49	**7.84**
DeepSeekV2-Lite	FP16	16	45.39	70.12	73.45	68.58	61.81	79.05	68.77	66.74	7.01
	HAMoE	4	44.71	68.94	73.27	67.26	60.94	78.43	68.11	65.95	7.24
	GPTQ	3.35	41.89	68.40	70.24	62.79	56.06	77.85	66.85	63.44	8.06
	MoQE	3.35	41.93	68.39	70.64	63.94	55.50	77.30	66.73	63.49	8.12
	MxMoE	3.35	42.92	68.43	70.48	**65.24**	**57.22**	77.80	66.93	64.15	7.81
	SFMoE	3.35	42.87	68.27	70.43	64.79	56.12	77.63	66.57	63.81	7.94
	HAMoE	3.35	**42.94**	**68.56**	**70.83**	64.86	56.57	**78.23**	**67.48**	**64.21**	**7.68**
	MoQa*	2.99	34.71	64.19	/	/	/	75.34	/	/	7.94
	HAMoE	2.99	**40.78**	**67.55**	71.25	64.78	57.38	**78.07**	67.32	63.88	**7.86**
OLMoE	FP16	16	48.55	76.26	77.02	73.22	67.51	80.58	69.06	70.31	7.04
	HAMoE	4	47.53	75.21	75.53	72.21	66.00	80.03	66.69	69.03	7.26
	GPTQ	3.35	46.04	74.75	74.53	70.23	64.26	79.14	65.04	67.71	7.68
	MoQE	3.35	46.13	74.34	74.33	70.69	64.69	78.76	65.17	67.73	7.63
	MxMoE	3.35	**46.87**	**74.92**	74.44	71.12	64.87	**79.98**	65.98	68.31	7.50
	SFMoE	3.35	46.18	74.73	73.91	70.41	64.03	79.13	65.78	67.74	7.63
	HAMoE	3.35	46.08	74.87	**74.88**	**71.22**	**64.95**	79.71	**66.93**	**68.38**	**7.49**
	MoQa*	2.97	26.02	44.87	/	/	/	61.30	/	/	11.24
	HAMoE	2.97	**44.20**	**70.20**	72.43	67.28	62.00	**77.37**	65.90	65.63	**8.84**
MoE-Girl	FP16	16	49.57	76.18	77.51	71.03	67.90	80.96	68.75	70.27	8.20
	HAMoE	4	49.23	76.12	77.44	71.30	67.14	80.79	68.14	70.02	8.22
	GPTQ	3.35	48.12	74.54	75.86	68.60	65.36	79.82	66.54	68.41	8.83
	MoQE	3.35	48.98	74.92	76.14	68.66	64.57	78.79	68.50	68.65	8.73
	MxMoE	3.35	**49.49**	**75.20**	76.88	69.21	65.21	78.94	66.54	68.78	8.62
	SFMoE	3.35	48.78	74.99	76.26	69.12	64.72	78.82	67.06	68.54	8.91
	HAMoE	3.35	49.40	75.04	**77.01**	**69.78**	**65.79**	**80.14**	**67.25**	**69.20**	**8.51**
	MoQa*	2.88	25.85	43.52	/	/	/	58.87	/	/	10.59
	HAMoE	2.88	**46.59**	**73.23**	74.69	66.41	62.90	**78.13**	65.19	66.82	**9.59**

MoQa*(Zheng et al., 2025) results are from the original paper; other baselines are reproduced.

against two variants, including **Activation Frequency-only** (precision based solely on activation frequency) and **Token-only** (grouping solely by token semantic importance). Due to space constraints, Figure 5 presents only the two most important metrics, Avg (average accuracy) and PPL (perplexity); the complete per-task results are provided in Appendix F. Figure 5 shows the performance gains on all four models. HAMoE consistently outperforms both single-dimension variants. On Qwen1.5-MoE, it improves accuracy by 0.23 and 0.90 percentage points over Frequency-only and Token-only, respectively, while reducing perplexity by 0.09 and 0.03. On DeepSeek-V2-Lite, accuracy gains are 0.71 and 0.77 points, with perplexity reductions of 0.16 and 0.07. The same trend is observed on OLMoE and MoE-Girl. These results confirm that activation frequency and token-type dimensions are complementary. A single dimension degrades accuracy, while combining both better characterizes MoE expert heterogeneity and achieves superior quantization accuracy retention.

Ablation on Sensitive Column Replacement Ratio. We evaluate the impact of different protection ratios (0.5%, 1%, 2%) on quantization performance (PPL, ↓), with no protection (0%) as the baseline. Table 2 shows that protecting 0.5% of sensitive columns brings improvements with PPL reductions from 0.3% to 2.9%. Increasing the ratio to 1% yields further gains with cumulative re-

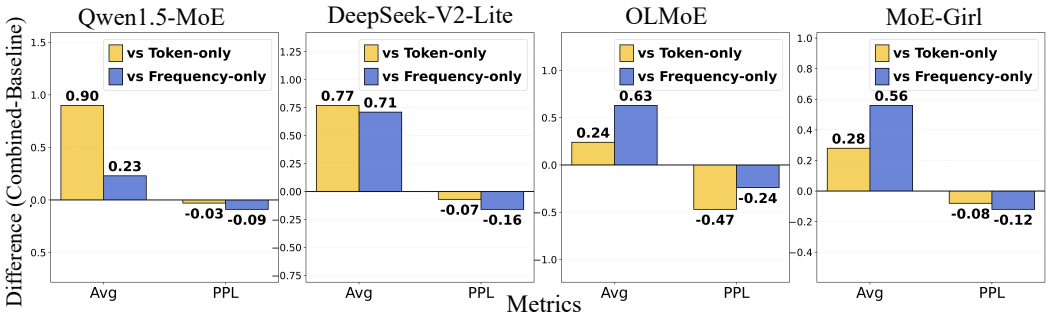

Figure 5: Ablation on dual-dimension grouping.

Table 2: Perplexity under different critical column protection ratios.

Protection Ratio	Qwen1.5-MoE	DeepSeek-V2-Lite	OLMoE	MoE-Girl
0%	7.29	7.76	7.81	8.75
0.5%	7.22	7.74	7.58	8.58
1%	7.16	7.68	7.49	8.51
2%	7.15	7.66	7.46	8.55

ductions from 1.0% to 4.1%, while increasing it to 2% leads to marginal saturation. Thus, protecting 1% of sensitive columns achieves near-optimal performance.

5.4 Storage Compression

Metadata Overhead. We use groups of 128 and store one FP32 scale and one UINT8 zero point per output row and group. In Qwen1.5-MoE, gate and up have shape $[1408, 2048]$ and down has shape $[2048, 1408]$. Across the base and protected paths, this requires 675,840 bytes per expert and about 973.2 MB for all 1,440 experts. Precision labels add little overhead. All metadata is included in the reported 10.15 GB full model size.

Compression Ratio. Quantizing only the expert weights of Qwen1.5-MoE reduces the full model size from 28.6 GB to 9.72 GB, a $2.95\times$ compression. Detailed compression ratios and group assignments for all models are reported in Table 3 and Appendix G.

Table 3: Full model compression results (all values in GB).

Model	Original	Quantized	Comp. Ratio	Peak Mem. (FP16)	Peak Mem. (HAMoE)
Qwen1.5-MoE	28.67	9.72	2.95×	31.969	10.40
DeepSeek-V2-Lite	31.40	10.41	2.91×	35.383	10.78
OLMoE	13.89	4.60	3.02×	14.08	4.63
MoE-Girl	13.89	4.60	3.02×	14.08	4.63

6 Conclusion

In this work, we propose HAMoE, a heterogeneity-aware mixed-precision quantization framework for MoE models. We identify two overlooked heterogeneity patterns, namely inter-expert heterogeneity in activation frequency and token-type preference, and intra-expert heterogeneity in weight column sensitivity, and address them with a token- and activation-aware precision allocation scheme plus an adaptive column replacement mechanism that protects the most sensitive 1% of columns in INT8. Experiments on four MoE models show that HAMoE consistently outperforms GPTQ, MoQE, MxMoE, SFMoE, and MoQa under equal bit budgets, reducing WikiText-2 perplexity by 4.0% over GPTQ at 3.35-bit and achieving near-lossless performance at 4-bit, with full model compression ratios between $2.91\times$ and $3.02\times$ after expert weight quantization.

Several limitations remain. First, we quantize only weights and leave activations in full precision, so the benefits of low-precision activation computation are not yet realized. Second, token category weights are assigned heuristically; although Appendix C confirms their robustness, more principled derivations are left for future work. Third, the allocation objective relies on the importance score as a proxy for downstream quantization loss. Fourth, inference optimizations required for industrial deployment, such as mixed-precision kernels and system-level co-design, have not yet been explored.

AI USE STATEMENT

In this work, generative AI tools were used to improve the clarity, grammar, and presentation of the manuscript; to assist with writing and debugging experimental code; and to provide limited suggestions and feedback on the experimental design. The central research hypothesis was independently proposed by the authors and was not generated or refined using AI tools. All experimental choices were critically evaluated and finalized by the authors. The authors manually reviewed and revised all AI-assisted text, inspected and tested all AI-assisted code, and independently verified the experimental results. Generative AI tools were not used to fabricate data, results, citations, or scientific claims. The authors take full responsibility for the final content of the manuscript and all associated artifacts.

REPRODUCIBILITY STATEMENT

We provide the information necessary to reproduce our experiments throughout the paper and supplementary materials. Section 5.1 describes the evaluated models, datasets, calibration procedure, random seed, evaluation framework, hardware configuration, and baseline settings. The appendices provide additional details on token-type weight assignment, hyperparameter selection, model architectures, ablation studies, and precision-group allocations. An anonymized implementation of HAMoE, together with the scripts and configurations used for quantization and evaluation, is submitted as supplementary material. The authors have verified that the supplementary code does not contain identifying information.

REFERENCES

Maryam Akhavan Aghdam, Hongpeng Jin, and Yanzhao Wu. Da-moe: Towards dynamic expert allocation for mixture-of-experts models. *CoRR*, abs/2409.06669, 2024. doi: 10.48550/ARXIV.2409.06669. URL https://doi.org/10.48550/arXiv.2409.06669.

Allura-org. Moe-girl. Online, 2024. URL https://huggingface.co/allura-org/MoE-Girl-1BA-7BT. Hugging Face.

Kayhan Behdin, Ayan Acharya, Aman Gupta, Sathiya Keerthi Selvaraj, and Rahul Mazumder. Quantease: Optimization-based quantization for language models - an efficient and intuitive algorithm. *CoRR*, abs/2309.01885, 2023. doi: 10.48550/ARXIV.2309.01885. URL https://doi.org/10.48550/arXiv.2309.01885.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. A survey on mixture of experts in large language models. *IEEE Trans. Knowl. Data Eng.*, 37(7):3896–3915, 2025. doi: 10.1109/TKDE.2025.3554028. URL https://doi.org/10.1109/TKDE.2025.3554028.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

DeepSeek-AI. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024.

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. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/duanmu25a.html.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:120:1–120:39, 2022. URL https://jmlr.org/papers/v23/21-0998.html.

Elias Frantar and Dan Alistarh. Qmoe: Practical sub-1-bit compression of trillion-parameter models. *CoRR*, abs/2310.16795, 2023. doi: 10.48550/ARXIV.2310.16795. URL https://doi.org/10.48550/arXiv.2310.16795.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: accurate post-training quantization for generative pre-trained transformers. *CoRR*, abs/2210.17323, 2022. doi: 10.48550/ARXIV.2210.17323. URL https://doi.org/10.48550/arXiv.2210.17323.

Wensheng Gan, Zhenyao Ning, Zhenlian Qi, and Philip S. Yu. Mixture of experts (moe): A big data perspective. *Inf. Fusion*, 127:103664, 2026. doi: 10.1016/J.INFFUS.2025.103664. URL https://doi.org/10.1016/j.inffus.2025.103664.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. *ArXiv*, abs/2103.13630, 2021. URL https://api.semanticscholar.org/CorpusID:232352683.

Ruihao Gong, Yifu Ding, Zining Wang, Chengtao Lv, Xingyu Zheng, Jinyang Du, Yang Yong, Shiqiao Gu, Haotong Qin, Jinyang Guo, Dahua Lin, Michele Magno, and Xianglong Liu. A survey of low-bit large language models: Basics, systems, and algorithms. *Neural Networks*, 192: 107856, 2025. doi: 10.1016/J.NEUNET.2025.107856. URL https://doi.org/10.1016/j.neunet.2025.107856.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2026. URL https://www.gurobi.com.

Babak Hassibi, David G. Stork, and Gregory J. Wolff. Optimal brain surgeon and general network pruning. In *Proceedings of International Conference on Neural Networks (ICNN'88), San Francisco, CA, USA, March 28 - April 1, 1993*, pp. 293–299. IEEE, 1993. doi: 10.1109/ICNN.1993.298572. URL https://doi.org/10.1109/ICNN.1993.298572.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spacy: Industrial-strength natural language processing in python. 2020. doi: 10.5281/zenodo.1212303. Version 3.0+.

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. *CoRR*, abs/2505.03804, 2025. doi: 10.48550/ARXIV.2505.03804. URL https://doi.org/10.48550/arXiv.2505.03804.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew G. Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 2704–2713. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00286. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Jacob_Quantization_and_Training_CVPR_2018_paper.html.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *CoRR*, abs/2401.04088, 2024. doi: 10.48550/ARXIV.2401.04088. URL https://doi.org/10.48550/arXiv.2401.04088.

Yann LeCun, John S. Denker, and Sara A. Solla. Optimal brain damage. In David S. Touretzky (ed.), *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*, pp. 598–605. Morgan Kaufmann, 1989. URL http://papers.nips.cc/paper/250-optimal-brain-damage.

Pingzhi Li, Xiaolong Jin, Yu Cheng, and Tianlong Chen. Examining post-training quantization for mixture-of-experts: A benchmark. *CoRR*, abs/2406.08155, 2024. doi: 10.48550/ARXIV.2406.08155. URL https://doi.org/10.48550/arXiv.2406.08155.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Guangxuan Xiao, and Song Han. AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. *GetMobile Mob. Comput. Commun.*, 28(4):12–17, 2024. doi: 10.1145/3714983.3714987. URL https://doi.org/10.1145/3714983.3714987.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Evan Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, and et al. Olmoe: Open mixture-of-experts language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=xXTkbTBmqq.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset, Aug 2016.

Hua-Chen Qi, Rui-Yu Zhuo, Bo-Wen Shi, Fei Chao, Xiang Chang, Changjing Shang, and Qiang Shen. Synergistic mixed-precision quantization for moe llms: Integrating cooperative shapley attribution with profiling-few fuzzy rule interpolation. *Information Sciences*, 759:124025, 2027. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2026.124025. URL https://www.sciencedirect.com/science/article/pii/S0020025526009564.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*, 2019.

Yikang Shen, Zhen Guo, Tianle Cai, and Zengyi Qin. Jetmoe: Reaching llama2 performance with 0.1m dollars. *CoRR*, abs/2404.07413, 2024. doi: 10.48550/ARXIV.2404.07413. URL https://doi.org/10.48550/arXiv.2404.07413.

Wei Tao, Haocheng Lu, Xiaoyang Qu, Bin Zhang, Kai Lu, Jiguang Wan, and Jianzong Wang. Moqae: Mixed-precision quantization for long-context LLM inference via mixture of quantization-aware experts. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 10810–10820. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.531/.

Qwen Team. Qwen1.5-moe: Matching 7b model performance with 1/3 activated parameters, February 2024. URL https://qwenlm.github.io/blog/qwen-moe/.

Guangxuan Xiao, Ji Lin, Mickaël Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, Proceedings of Machine Learning Research, pp. 38087–38099. PMLR, 2023. URL https://proceedings.mlr.press/v202/xiao23c.html.

Zhewei Yao, Zhen Dong, Zhangcheng Zheng, Amir Gholami, Jiali Yu, Eric Tan, Leyuan Wang, Qijing Huang, Yida Wang, Michael Mahoney, et al. Hawq-v3: Dyadic neural network quantization. In *International Conference on Machine Learning*, pp. 11875–11886. PMLR, 2021.

Zhihang Yuan, Lin Niu, Jiawei Liu, Wenyu Liu, Xinggang Wang, Yuzhang Shang, Guangyu Sun, Qiang Wu, Jiaxiang Wu, and Bingzhe Wu. RPTQ: reorder-based post-training quantization for large language models. *CoRR*, abs/2304.01089, 2023. doi: 10.48550/ARXIV.2304.01089. URL https://doi.org/10.48550/arXiv.2304.01089.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Zihao Zheng, Xiuping Cui, Size Zheng, Maoliang Li, Jiayu Chen, Yun (Eric) Liang, and Xiang Chen. Moqa: Rethinking moe quantization with multi-stage data-model distribution awareness. *CoRR*, abs/2503.21135, 2025. doi: 10.48550/ARXIV.2503.21135. URL https://doi.org/10.48550/arXiv.2503.21135.

Shuzhang Zhong, Yanfan Sun, Ling Liang, Runsheng Wang, Ru Huang, and Meng Li. Hybrimoe: Hybrid CPU-GPU scheduling and cache management for efficient moe inference. In *62nd ACM/IEEE Design Automation Conference, DAC 2025, San Francisco, CA, USA, June 22-25, 2025*, pp. 1–7. IEEE, 2025. doi: 10.1109/DAC63849.2025.11133274. URL https://doi.org/10.1109/DAC63849.2025.11133274.

A METHOD COMPARISON

Table 4 summarizes the positioning of the compared MoE quantization methods across the three dimensions discussed in 2.3.

Table 4: Comparison of existing MoE quantization methods across three dimensions of expert importance assessment.

Method	Usage-Aware	Parametric Vulnerability-Aware	Function-Aware
QMoE	✓	×	×
MoQa	✓	×	×
MoQAE	✓	×	×
MoQE	✓	×	×
MxMoE	✓	✓	×
MoEQuant	✓	✓	×
SFMoE	✓	✓	×
Ours	✓	✓	✓

B FULL SPECIALIZATION DISTRIBUTION

Table 5 presents the complete distribution of maximum single-category token proportions for all experts across the four evaluated MoE models.

Table 5: Distribution of maximum single-category token proportion across experts in four MoE models.

Maximum Proportion	Qwen1.5-MoE	DeepSeek-V2-Lite	OLMoE	MoE-Girl
0%–9%	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
10%–19%	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
20%–29%	0 (0.0%)	7 (0.4%)	0 (0.0%)	1 (0.1%)
30%–39%	158 (11.0%)	277 (16.6%)	148 (14.5%)	163 (15.9%)
40%–49%	601 (41.7%)	643 (38.6%)	432 (42.2%)	412 (40.2%)
50%–59%	334 (23.2%)	351 (21.1%)	306 (29.9%)	298 (29.1%)
60%–69%	123 (8.5%)	231 (13.9%)	72 (7.0%)	87 (8.5%)
70%–79%	103 (7.2%)	85 (5.1%)	40 (3.9%)	40 (3.9%)
80%–89%	73 (5.1%)	57 (3.4%)	16 (1.6%)	15 (1.5%)
90%–99%	48 (3.3%)	13 (0.8%)	10 (1.0%)	8 (0.8%)

C TOKEN TYPE WEIGHT ASSIGNMENT

We assign semantic importance weights $w(t)$ to tokens based on POS tags identified by spaCy (Honnibal et al., 2020), following the principle that content words receive higher weights than function words. **Specifically, punctuation is assigned 0.1, function words 0.2, and content words 0.5.** These values are set heuristically to establish a relative ordering rather than precise quantification, ensuring that experts processing content words are recognized as more quantization-sensitive. More principled weight derivation methods are deferred to future work.

C.1 ROBUSTNESS ANALYSIS OF TOKEN WEIGHT ASSIGNMENT

To validate the robustness of our token weight assignment, we design six additional weight variants covering diverse reasonable allocations.

- **V1 (large gap)**: content words=0.65, punctuation=0.05, function words=0.15

- **V2 (small gap)**: content words=0.35, punctuation=0.15, function words=0.25,
- **V3 (distinguish prep. vs. art.)**: content words=0.5, punctuation=0.1, function words=0.15
- **V4 (elevate function words)**: content words=0.3, punctuation=0.1, function words=0.3
- **V5 (ignore punctuation)**: content words=0.6, punctuation=0.0, function words=0.2
- **V6 (content dominant)**: content words=0.7, punctuation=0.1, function words=0.1

These variants cover a broad spectrum of weight assignments, including enlarging or reducing inter-category gaps, distinguishing subcategories, and extreme configurations. For each variant, we re-compute the expert precision groups under the same average bit budget (3.35-bit), and calculate the overall agreement rate between each variant and the original setting. Table 6 presents the results on all four MoE models.

Table 6: Group-wise agreement (%) between each token weight variant and the original setting across four MoE models.

Variant	Model			
	Qwen1.5-MoE	DeepSeek-V2-Lite	OLMoE	MoE-Girl
V1 (large gap)	98.19	98.80	98.24	98.05
V2 (small gap)	95.49	94.59	94.92	94.14
V3 (distinguish prep./art.)	98.61	98.68	98.83	98.44
V4 (elevate function words)	82.29	84.31	80.66	79.00
V5 (ignore punctuation)	98.33	98.44	99.02	98.44
V6 (content dominant)	95.28	96.63	96.29	95.90

C.2 Analysis.

Across all four models, five variants (V1, V2, V3, V5, V6) achieve over 94% agreement with the original setting, with V1, V3, and V5 reaching above 98%. V4 (elevating function words) yields relatively lower agreement (79%–84%), which corresponds to a less reasonable weight configuration that treats function words comparably to content words. This counterexample actually validates the rationale of our original design: distinguishing content words from function words is essential for accurately assessing expert importance.

Overall, as long as the basic semantic hierarchy of "content words $>$ function words $>$ punctuation" is preserved, our token importance scoring exhibits strong robustness to the specific numerical values. The five reasonable variants achieve over 94% agreement with the original setting, with an average agreement of 97.3% across all models and variants, fully demonstrating the stability of our token weight assignment.

D Selection of the Balancing Parameter α

The comprehensive importance score in 4.1 combines activation frequency and token semantic importance via a balancing parameter α:

$$\text{Score} = \alpha \cdot F_{\text{norm}} + (1 - \alpha) \cdot T_{\text{norm}}. \tag{1}$$

We conduct a grid search over $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$ on all four MoE models under the 3.35-bit setting. Table 7 shows that $\alpha = 0.7$ achieves the best perplexity on three out of four models (Qwen1.5-MoE, OLMoE, and MoE-Girl). Therefore, we set $\alpha = 0.7$ as the default balancing parameter across all experiments. The coarse step size of 0.1 is chosen for computational feasibility; finer-grained α tuning is left to future work.

E Model Architecture Details

Table 8 summarizes the architectural specifications of the four MoE models evaluated in our experiments, including total parameters, active parameters, BF16 memory footprint, number of experts, and Top-K routing settings.

Table 7: Ablation study on the balancing parameter α across four MoE models.

Model	α								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Qwen1.5-MoE	7.194	7.181	7.213	7.195	7.215	7.192	**7.159**	7.210	7.179
DeepSeek-V2-Lite	7.845	7.755	7.704	7.773	**7.624**	7.625	7.704	7.754	7.716
OLMoE	7.638	7.687	7.692	7.637	7.645	7.618	**7.492**	7.614	7.786
MoE-Girl	8.907	8.773	8.746	8.684	8.649	8.586	**8.517**	8.562	8.569

Table 8: Architectural specifications of evaluated MoE models.

Model	**Total Params**	**Active Params**	**BF16 Size**	**Experts**	**TopK**
DeepSeek-V2-Lite	15.7B	2.4B	31.4 GB	64+2	6
Qwen1.5-MoE-A2.7B	14.3B	2.7B	28.6 GB	60+4	4
OLMoE-1B-7B	6.9B	1.3B	13.8 GB	64+0	8
MoE-Girl-1B-7B	6.9B	1.3B	13.8 GB	64+0	8

F Ablation on Fine-grained Mixed Precision Assignment

To validate the effectiveness of our proposed HAMoE method, we conduct ablation studies comparing it against two variants: Token-only (assigning precision based solely on token significance) and Frequency-only (assigning precision based solely on expert activation frequency). As shown in Table 9, HAMoE consistently outperforms both variants across all four MoE models under the same 3.35-bit budget. On average, HAMoE achieves the highest accuracy and lowest perplexity for every model, demonstrating that combining token and frequency signals is crucial for fine-grained mixed-precision assignment. While Token-only captures per-instance importance and Frequency-only reflects expert usage patterns, neither alone matches the complementary benefits of our joint strategy. These results collectively confirm that token-level dynamics and expert-level statistics are complementary. Notably, HAMoE achieves the largest improvements on Qwen1.5-MoE and DeepSeek-V2-Lite. Moreover, the consistent perplexity reduction across all four models further demonstrates that our joint strategy better preserves generation quality under ultra-low bit quantization.

Table 9: Quantization results of HAMoE against two variants (Token-only and Frequency-only) under 3.35-bit setting. Bold indicates the best result among all variants for each model.

Model	**Variant**	**AC↑**	**AE↑**	**HS↑**	**LO↑**	**LS↑**	**PQ↑**	**WG↑**	**Avg↑**	**PPL↓**
Qwen1.5-MoE	HAMoE	44.11	68.39	75.79	69.32	63.35	79.38	70.01	**67.19**	**7.16**
	Token-Only	44.11	66.12	75.58	69.19	63.07	79.16	68.51	66.29	7.19
	Frequency-Only	43.54	67.97	75.62	69.15	63.32	79.33	69.77	66.96	7.25
DeepSeek-V2-Lite	HAMoE	42.94	68.56	70.83	64.86	56.57	78.23	67.48	**64.21**	**7.68**
	Token-Only	42.48	67.63	70.63	64.69	55.66	77.37	65.59	63.44	7.75
	Frequency-Only	42.92	67.76	70.33	63.94	55.65	77.80	66.11	63.50	7.84
OLMoE	HAMoE	46.08	74.87	74.88	71.22	64.95	79.71	66.93	**68.38**	**7.49**
	Token-Only	45.90	74.19	75.25	71.00	64.27	80.09	66.26	68.14	7.96
	Frequency-Only	45.26	74.76	74.19	70.52	63.67	79.43	66.42	67.75	7.73
MoE-Girl	HAMoE	49.40	75.04	77.01	69.78	65.79	80.14	67.25	**69.20**	**8.51**
	Token-Only	48.85	75.04	76.24	69.62	65.80	79.64	66.82	68.86	8.59
	Frequency-Only	48.30	73.50	76.52	69.53	65.41	79.96	66.14	68.48	8.63

G Compressions

This appendix provides the detailed expert group assignments for all four MoE models evaluated in our experiments. Table 10 reports, for each model, the number of experts assigned to each precision

group (INT8, INT6, INT4, INT3, INT2), the original full model size, the full model size after quantizing the expert weights, and the resulting compression ratio.

Table 10: Full model compression results after expert weight quantization and expert group assignments for all evaluated models.

Model	Expert Counts	Original	Quantized	Comp. Ratio
Qwen1.5-MoE	28 / 43 / 421 / 776 / 172	28.67	9.72	2.95×
DeepSeek-V2-Lite	33 / 49 / 486 / 897 / 199	31.40	10.41	2.91×
OLMoE	20 / 30 / 300 / 552 / 122	13.89	4.60	3.02×
MoE-Girl	20 / 30 / 300 / 552 / 122	13.89	4.60	3.02×