MACS: Modality-Aware Capacity Scaling for Efficient Multimodal MoE Inference

Bo Li [1]*, **Chuan Wu** [2,3]*, **Shaolin Zhu** [3]†
[1] School of Software, Tsinghua University, Beijing, China
[2] School of New Media and Communication, Tianjin University, China
[3] TJUNLP Lab, School of Computer Science and Technology, Tianjin University, China
{wuchuan, zhushaolin}@tju.edu.cn, li-b19@tsinghua.org.cn

Abstract

Mixture-of-Experts Multimodal Large Language Models (MoE MLLMs) suffer from a significant efficiency bottleneck during Expert Parallelism (EP) inference due to the straggler effect. This issue is worsened in the multimodal context, as existing token-count-based load balancing methods fail to address two unique challenges: (1) Information Heterogeneity, where numerous redundant visual tokens are treated equally to semantically critical ones, and (2) Modality Dynamics, where varying visual to text ratios across tasks lead to resource misallocation. To address these challenges, we propose **MACS** (**M**odality-**A**ware **C**apacity **S**caling), a training-free inference framework. Specifically, MACS introduces an Entropy-Weighted Load mechanism to quantify the semantic value of visual tokens, addressing information heterogeneity. Additionally, the Dynamic Modality-Adaptive Capacity mechanism allocates expert resources based on the real-time modal composition of the input. Extensive experiments demonstrate that MACS significantly outperforms existing methods on various multimodal benchmarks, providing a novel and robust solution for the efficient deployment of MoE MLLMs in EP inference.

1 Introduction

Multimodal Large Language Models (MLLMs) have demonstrated remarkable capabilities in perceiving and reasoning across diverse modalities (OpenAI, 2025; Liu et al., 2024a; Bai et al., 2025). To efficiently scale MLLMs, the Mixture-of-Experts (MoE) architecture has become a mainstream choice (Fedus et al., 2022; Qu et al., 2024; Wang et al., 2025; Team et al., 2025; Bai et al., 2025). By sparsely activating a subset of experts for each token, MoE theoretically decouples the size of the model parameters from the inference computation, striking a balance between efficiency and performance (Fedus et al., 2022; Zhu et al., 2025).

In practice, MoE MLLMs are often deployed using **Expert Parallelism** (EP) (Cai et al., 2024), where different experts are distributed across multiple computational devices to improve throughput. However, this paradigm introduces an unavoidable synchronization bottleneck: all devices must wait after processing their respective tokens until the most heavily loaded device has finished its computation before proceeding to the next layer. CAI-MoE (He et al., 2025) formally defines this phenomenon as the straggler effect, where the overall inference latency is determined by the most heavily loaded straggler expert. Although this work proposes effective mitigation strategies, such as token drop, its methods are primarily designed for unimodal text models, under the core assumption that each token represents roughly equal computational load.

Recent studies indicate that the straggler effect is significantly worsened in MoE MLLMs under EP inference (Li et al., 2025c; Wu et al., 2025). Specifically, multimodal inputs highlight two deeper sources of load imbalance: (i) Information Heterogeneity. Unlike text tokens, which have a relatively uniform semantic density (Li et al., 2023), a single visual input is typically encoded into hundreds of patch tokens, many of which correspond to low-information background regions (Liang et al., 2025; Wu et al., 2025). However, token-count-based capacity management, as used in CAI-MoE, treats redundant background tokens and semantically critical object or text tokens equally, inevitably causing severe misestimation of true computational load and resource misallocation. (ii) Modality Dynamics. The ratio of visual to textual tokens varies dramatically between tasks, ranging from image-intensive document understanding or OCR tasks to text-dominant reasoning tasks. With such highly

* Equal contribution.
† Corresponding Author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 22134–22148
July 2-7, 2026

dynamic modality compositions, traditional token-count-based load modeling fails to accurately capture the actual computational pressure on experts, further increasing load imbalance and synchronization delays (Xue et al., 2024; Zhang et al., 2025a).

To address these challenges, we propose **MACS** (**M**odality-**A**ware **C**apacity **S**caling), a training-free inference framework for MoE MLLMs. We revisit expert capacity allocation under EP inference from a modality-aware perspective. Specifically, we employ an Entropy-Weighted Load mechanism to quantify and differentiate the semantic value of visual tokens, thereby mitigating load imbalance caused by information heterogeneity. In addition, the Dynamic Modality-Adaptive Capacity mechanism adjusts the expert capacity based on the real-time modality composition of each input batch, effectively alleviating the amplified straggler effect in multimodal settings and significantly improving the inference efficiency. Finally, to handle inevitable capacity overflows, we design a two-phase overflow handling mechanism to minimize information loss.

The main contributions of our work are summarized as follows: **(I)** We systematically analyze the core mechanisms through which the straggler effect is acutely exacerbated in MoE MLLMs under EP inference, driven by visual token redundancy and modality dynamics. **(II)** We propose MACS, which enables more fine-grained and robust expert load scheduling at the inference stage through its Entropy-Weighted Load and Dynamic Modality-Adaptive Capacity mechanisms. **(III)** We demonstrate through extensive experiments that MACS outperforms existing methods on various multimodal benchmarks, offering a novel and effective solution for the efficient deployment of MoE MLLMs in EP inference.

2 Related Work

This work addresses the efficiency bottleneck of MoE MLLMs under EP inference.

MoE Models under EP. MoE models are often deployed using the EP distributed strategy to improve throughput (Cai et al., 2024). However, this approach introduces a synchronization bottleneck that leads to the straggler effect (He et al., 2025), where overall system latency is determined by the slowest expert. To mitigate this issue, existing research primarily falls into two categories: (I) Capacity Management and Token Dropping. Capacity-Aware Inference (CAI-MoE) (He et al., 2025) addresses the straggler effect by imposing a capacity limit on experts and discarding excess tokens. While effective, its core mechanism relies on token counting, assuming all tokens have equal computational value, a premise with significant limitations in multimodal contexts. (II) Expert Pruning and Dynamic Skipping. Stun (Lee et al., 2025) and MoE-Pruner (Xie et al., 2024) reduce the computational load by decreasing the number of activated experts, including permanently removing redundant experts through structured pruning. NAEE (Lu et al., 2024) and MC-MoE (Huang et al., 2024) dynamically skip non-essential experts during inference, primarily making decisions based on signals such as routing probabilities. However, these methods are mostly designed for unimodal text models and often suffer from performance degradation when directly applied to multimodal architectures, as they cannot handle the unique behaviors of different modalities.

Imbalance in MoE MLLMs. Recent studies on MLLM interpretability have revealed that the straggler effect is acutely exacerbated in multimodal contexts, stemming from two deeper challenges: (I) Information Heterogeneity. Wu et al. (2025); Zhang et al. (2026) identified significant internal functional specialization. Li et al. (2025c); Liang et al. (2025) have shown that multimodal inputs themselves exhibit high information heterogeneity, many visual tokens correspond only to regions with low-information background. For a load balancing system based solely on token counting, this intrinsic information difference is imperceptible. (II) Modality Dynamics. The ratio of visual to textual tokens varies dramatically between tasks, ranging from image-intensive document understanding or OCR tasks (Li et al., 2025a; Zhu et al., 2023) to text-dominant reasoning tasks (Li et al., 2025b; Zuo et al., 2025). With such highly dynamic modality compositions, traditional token-count-based load modeling fails to accurately capture the actual computational pressure on experts, further increasing load imbalance and synchronization delays (Xue et al., 2024; Zhang et al., 2025a,b).

Based on these observations, we propose **MACS**, which effectively mitigates the straggler effect under expert parallelism through its Entropy-Weighted Load and Dynamic Modality-Adaptive Capacity mechanisms.

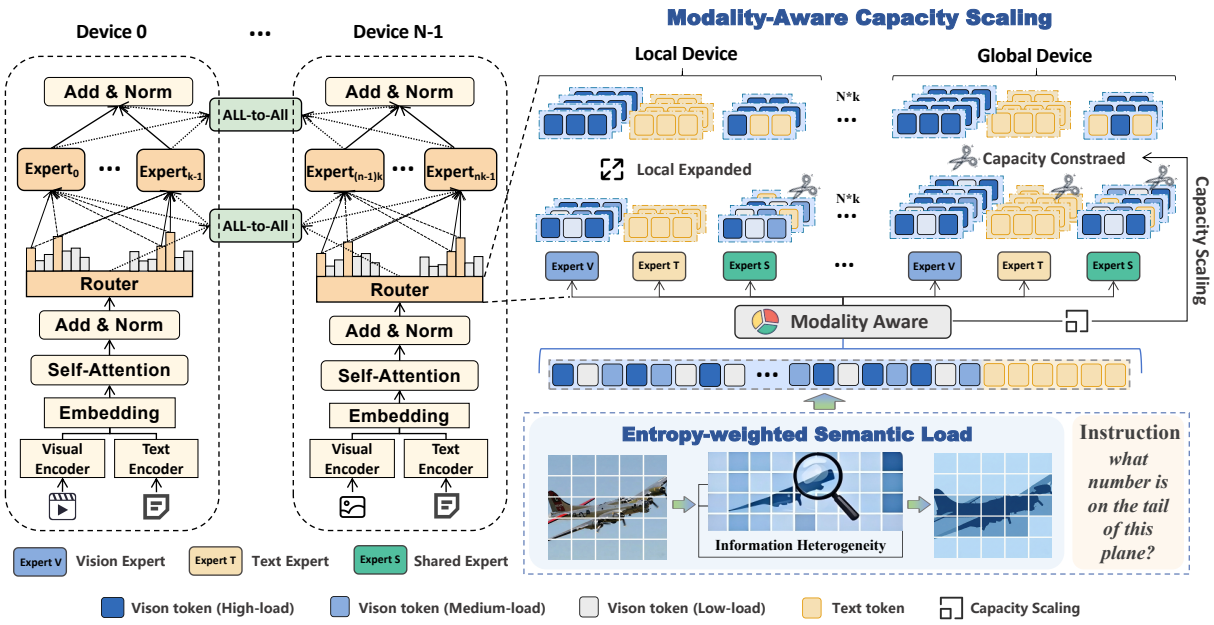

Figure 1: Overview of the MACS framework. It consists of three components: Entropy-Weighted Load, which models expert load based on token information; Dynamic Modality-Adaptive Capacity, which adjusts expert capacity according to batch-level modality composition, and Local Semantic Rerouting, which locally reroutes overflow tokens and applies a fail-safe drop when rerouting is infeasible.

3 Methodology

We present MACS, a training-free inference framework for MoE MLLMs. As illustrated in Figure 1, it consists of three components: (I) Entropy-Weighted Load (Sec. 3.2), which models expert load based on token information; (II) Dynamic Modality-Adaptive Capacity (Sec. 3.3), which adjusts expert capacity according to batch-level modality composition, and (III) Local Semantic Rerouting (Sec. 3.4), which locally reroutes overflow tokens and applies a fail-safe drop when rerouting is infeasible.

3.1 Problem Formulation

A standard MoE layer consists of N experts $\mathcal{E} = \{E_1, \ldots, E_N\}$ and a router network $G(\cdot)$ that produces gating scores for each input token x. The router selects the top-k experts and computes the output as

$$y(x) = \sum_{j \in \text{Top-}k(G(x))} G(x)_j \cdot E_j(x). \quad (1)$$

Let $\mathcal{T}$ denote the set of tokens in a batch and $\mathcal{I}_j \subset \mathcal{T}$ the set of tokens assigned to expert E_j. Under expert parallelism, the inference latency of an MoE layer, denoted as $\mathcal{L}_{\text{MoE}}$, is bounded by the slowest expert due to synchronization.

$$\mathcal{L}_{\text{MoE}} \propto \max_{j \in \{1,\ldots,N\}} |\mathcal{I}_j|. \quad (2)$$

The **straggler effect** arises when $\max_j |\mathcal{I}_j| \gg \text{mean}_j |\mathcal{I}_j|$, creating a severe bottleneck. Existing approaches typically mitigate this issue by imposing a static capacity limit

$$C = \gamma \cdot \frac{|\mathcal{T}| \cdot k}{N}, \quad (3)$$

where γ is a fixed capacity factor. However, in multimodal settings, raw token counts are a poor proxy for computational demand due to substantial information heterogeneity among tokens, particularly on the visual side.

3.2 Entropy-Weighted Expert Load

To reduce redundant visual tokens that consume expert capacity, we replace the count-based load metric with an information-based one, using entropy as a proxy for semantic importance.

Entropy Computation and Normalization. For a visual token x_v with a feature representation $z \in \mathbb{R}^D$, we compute its Shannon entropy $H(x_v)$ from the probability distribution obtained by $\text{Softmax}(z)$. To ensure robustness across different images and models, we apply image-wise z-score normalization over visual tokens:

$$\tilde{H}(x_v) = \frac{H(x_v) - \mu_{\mathcal{B}}}{\sigma_{\mathcal{B}} + \epsilon}, \quad (4)$$

where $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}$ denote the mean and standard deviation of the entropy values within the current

batch $\mathcal{B}$, and ϵ is a small constant for numerical stability.

Semantic Weighting and Effective Load. We define a semantic weight function

$$w(x) = \begin{cases} \sigma\left(-\delta \cdot \tilde{H}(x)\right), & x \in \mathcal{T}_{vis}, \\ 1.0, & x \in \mathcal{T}_{txt}, \end{cases} \quad (5)$$

where $\sigma(\cdot)$ is the Sigmoid function and δ controls the sensitivity of the entropy-to-weight mapping. Text tokens are assigned full weight due to their high semantic density. The effective load of the expert E_j is then defined as

$$\tilde{L}_j = \sum_{x \in \mathcal{I}_j} w(x), \quad (6)$$

allowing experts to process a larger number of low-information visual tokens without prematurely reaching capacity limits.

3.3 Modality-Aware Capacity Scaling

A static capacity factor is agnostic to the modality composition of the input batch. To prevent expert overload in vision-heavy scenarios and resource underutilization in text-heavy ones, we dynamically scale expert capacities based on the batch's effective modality ratio.

Effective Modality Ratio. Using semantic weights, we compute the effective visual ratio

$$R_v = \frac{\sum_{x \in \mathcal{T}_{vis}} w(x)}{\sum_{x \in \mathcal{T}} w(x)}, \quad (7)$$

which better reflects the true computational demand of the visual modality than raw token proportions.

Adaptive Capacity Scaling. Following prior analyses of expert specialization, we categorize experts into three groups based on their activation frequencies on a held-out calibration set: visual experts $\mathcal{E}_{vis}$, text experts $\mathcal{E}_{txt}$ and shared experts $\mathcal{E}_{shared}$. We define a modality bias indicator

$$m_j = \begin{cases} +1, & E_j \in \mathcal{E}_{vis}, \\ -1, & E_j \in \mathcal{E}_{txt}, \\ 0, & E_j \in \mathcal{E}_{shared}. \end{cases} \quad (8)$$

Let C_{base} denote the base capacity derived from the static formulation. We scale the capacity of each expert as

$$C_j = C_{base} \cdot \left(1 + \rho \cdot m_j \cdot (R_v - 0.5)\right), \quad (9)$$

where ρ controls the adaptation strength. In practice, we clamp C_j to a minimum value to avoid degenerate capacities. When $R_v > 0.5$, visual experts receive increased capacity while text experts are constrained, and vise versa.

3.4 Local Semantic Rerouting

Even with information aware load modeling and adaptive capacity scaling, transient expert overflows may still occur. When an expert E_j exceeds its capacity C_j, we first attempt to reroute overflow tokens locally to avoid unnecessary token dropping and cross-device communication.

Let $\mathcal{E}_{cand}$ denote the set of experts on the same computational device whose effective loads satisfy $\tilde{L}_k < C_k$. For an overflow token x with feature representation z_x, we score each candidate expert $E_k \in \mathcal{E}_{cand}$ by combining router preference and semantic affinity:

$$S(x, E_k) = (1-\eta)\, G(x)_k + \eta \cdot \mathrm{sim}(z_x, \mu_k), \quad (10)$$

where μ_k is the semantic centroid of the expert E_k and $\mathrm{sim}(\cdot,\cdot)$ denotes the cosine similarity. The overflow token is rerouted to the candidate expert with the highest score that satisfies the capacity constraint.

If no candidate experts are available on the local device ($\mathcal{E}_{cand} = \emptyset$), rerouting is infeasible. In this case, we activate a fail-safe drop mechanism. We define a retention score for an overflow token x as

$$r(x) = w(x) \cdot \max_j G(x)_j, \quad (11)$$

which jointly considers the token's semantic importance and routing confidence. When dropping is unavoidable, tokens with the lowest retention scores are discarded first, ensuring that only tokens of low-importance and low-confidence are removed.

4 Experiments

4.1 Setup

Models and Implementation. We conducted experiments on three SOTA MoE MLLMs: Qwen3-VL (30B-A3B) (Bai et al., 2025) , InternVL3.5 (30B-A3B) (Wang et al., 2025). , and Kimi-VL (16B-A3B) (Team et al., 2025) These models employ distinct MoE configurations (e.g., Qwen and InternVL use 48 layers with 128 experts, whereas Kimi-VL adopts a hybrid architecture with shared experts and 64 routing experts). All experiments

Method	Image Understanding							Video Understanding					Avg.
	TextVQA	ChartQA	MMStar	MMBench	MMVet	MME	RWQA	MVBench	EgoSch	VMME	LVB	VMMMU	(%)
QWEN3-VL-30B-A3B-INSTRUCT ($\gamma_0 = 1.0$)													
Vanilla MoE	83.54	85.36	72.10	86.82	85.67	2500	73.72	72.29	63.28	74.53	62.47	68.64	100.00
CAI-MoE (Token Drop)	77.42	79.13	66.58	81.34	79.21	2214	67.89	66.41	58.12	68.34	56.89	62.17	91.80
CAI-MoE (Expanded)	79.86	81.57	68.42	83.05	81.63	2305	69.46	68.74	60.15	70.92	58.73	64.28	94.69
MACS (w/o Expanded)	83.04	84.89	71.48	86.12	85.03	2478	73.15	71.63	62.84	73.91	61.88	67.92	99.20
MACS (Ours)	**83.41**	**85.22**	**71.93**	**86.67**	**85.48**	**2492**	**73.58**	**72.11**	**63.14**	**74.36**	**62.33**	**68.49**	**99.78**
INTERNVL3.5-30B-A3B ($\gamma_0 = 1.0$)													
Vanilla MoE	85.68	84.14	72.03	84.68	85.43	2324	64.87	72.06	60.37	68.65	63.76	65.24	100.00
CAI-MoE (Token Drop)	78.13	76.52	65.47	78.23	77.19	2056	58.34	64.21	54.12	61.88	57.24	58.63	90.22
CAI-MoE (Expanded)	80.94	79.28	67.91	80.76	80.14	2147	61.05	67.33	56.49	64.52	59.81	61.17	93.93
MACS (w/o Expanded)	84.92	83.47	71.36	84.02	84.66	2298	64.12	71.14	59.83	67.94	63.11	64.58	98.96
MACS (Ours)	**85.51**	**83.98**	**71.84**	**84.49**	**85.17**	**2315**	**64.69**	**71.89**	**60.21**	**68.42**	**63.55**	**65.03**	**99.72**
KIMI-VL-A3B-INSTRUCT ($\gamma_0 = 1.0$)													
Vanilla MoE	88.39	87.26	61.25	83.11	77.84	2218	68.07	62.73	78.32	66.84	64.37	57.58	100.00
CAI-MoE (Token Drop)	82.56	81.04	56.83	77.45	71.27	2013	63.42	57.18	72.56	61.29	59.14	52.87	92.24
CAI-MoE (Expanded)	84.72	83.91	58.17	79.62	73.84	2096	65.18	59.42	74.89	63.56	61.22	54.63	95.28
MACS (w/o Expanded)	87.94	86.68	60.89	82.76	77.12	2198	67.63	62.15	77.84	66.21	63.95	57.12	99.28
MACS (Ours)	**88.27**	**87.14**	**61.16**	**83.02**	**77.63**	**2212**	**67.96**	**62.58**	**78.19**	**66.67**	**64.24**	**57.49**	**99.81**

Table 1: Performance comparison of MACS against the SOTA distributed MoE inference acceleration method CAI-MoE on multimodal benchmarks. We evaluate on Qwen3-VL, InternVL3.5, and Kimi-VL, comparing against CAI-MoE's *Token Drop* and *Expanded Drop* variants. "Vanilla MoE" denotes the unconstrained baseline. All acceleration methods use a base capacity factor $\gamma_0 = 1.0$. "w/o Expanded" denotes the variant without local expansion, while "Ours" represents the full method.

were implemented using DeepSpeed. We performed distributed inference on 8 NVIDIA A100 GPUs, employing 8-way EP to simulate a high-performance production environment.

Datasets. We evaluated MACS on a comprehensive multimodal benchmarks. For Image Understanding, we utilize 8 Zero-Shot benchmarks, including $\text{TextVQA}_{\text{val}}$ (Singh et al., 2019) and ChartQA (Masry et al., 2022), $\text{MMBench}_{\text{en}}$ (Liu et al., 2024b), MMStar (Chen et al., 2024), MMVet (Yu et al., 2023), MME (Fu et al., 2023) and RealWorldQA (x.ai, 2024). For Video Understanding, we extend our evaluation to dynamic visual tasks using MVBench (Li et al., 2024b), EgoSchema (Mangalam et al., 2023), VideoMME (Fu et al., 2025), $\text{LongVideoBench}_{\text{val}}$ (Wu et al., 2024), and VideoMMMU (Hu et al., 2025). Performance is reported using standard accuracy metrics, while efficiency is measured via End-to-End Latency and Speedup.

Baselines. We compared our approach with the original model and the SOTA distributed MoE inference acceleration method, CAI-MoE (He et al., 2025). Vanilla MoE serves as both the performance upper bound and the latency lower bound. For CAI-MoE, we evaluate both its Token Drop and Expanded Drop variants.

Method	SC	SL	DC	SR	TextVQA	RWQA	MMB	VMMMU
$\gamma_0 = 0.5, \quad \rho = 0.6$								
Baseline	✓	-	-	-	68.91	63.21	74.29	54.64
+ SL	✓	✓	-	-	74.96	66.19	77.94	60.85
+ DC	-	✓	✓	-	79.11	69.74	83.41	64.29
MACS (Ours)	-	✓	✓	✓	**81.04**	**71.28**	**84.21**	**67.17**
$\gamma_0 = 1.0, \quad \rho = 0.6$								
Baseline	✓	-	-	-	77.42	67.89	81.34	62.17
+ SL	✓	✓	-	-	79.28	69.96	82.41	64.37
+ DC	-	✓	✓	-	83.04	73.15	86.12	67.92
MACS (Ours)	-	✓	✓	✓	**83.41**	**73.58**	**86.67**	**68.49**

Table 2: Ablation study on Qwen3-VL. "SC" denotes the baseline with only Static Capacity and conventional token counting. "+SL" indicates the addition of Entropy-weighted Semantic Load. "+DC" further incorporates Modality-aware Dynamic Capacity. Experiments are conducted with two base capacity factors ($\gamma_0 = 0.5$ and $\gamma_0 = 1.0$) to evaluate performance under varying pressure levels.

4.2 Main Results

To comprehensively evaluate the effectiveness of MACS, we compare it with a representative capacity-aware MoE inference acceleration method, CAI-MoE (He et al., 2025), across three mainstream MoE MLLMs. All methods are evaluated under an EP inference setting, with a unified base capacity factor of $\gamma_0 = 1.0$ to ensure a fair comparison.

As shown in Table 1, the results demonstrate

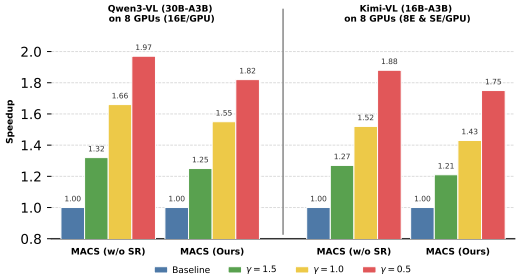

Figure 2: Compared to the capacity-unconstrained baseline, the speedup of a single MoE layer is achieved through two capacity-aware inference methods: Capacity Constrained and Semantic Rerouting.

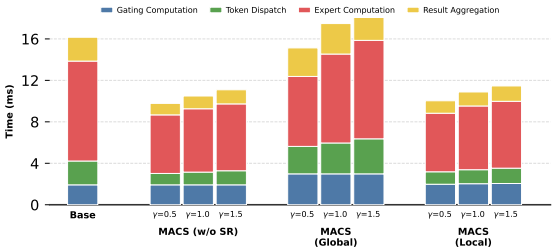

Figure 3: Inference latency speedup across different stages of Qwen3-VL. Global refers to rerouting overflowed tokens within the global expert scope, while Local denotes rerouting overflowed tokens exclusively to local experts.

that MACS consistently outperforms both variants of CAI-MoE on all three models (Qwen3-VL, InternVL3.5, and Kimi-VL). Specifically, the full MACS preserves more than 99.7% of the original Vanilla MoE performance, corresponding to an average degradation of less than 0.3%. In contrast, CAI-MoE incurs substantial performance losses: its Token Drop variant degrades performance by over 7%, while the Expanded Drop variant still results in an approximately 5% drop. These results suggest a fundamental limitation of token-count-based capacity management, which inevitably discards semantically valuable tokens when experts are overloaded. By contrast, MACS makes information-aware allocation decisions that better preserve multimodal reasoning capability.

Importantly, this comparison isolates the contribution of our core design. Even MACS (w/o Expanded), which excludes local semantic rerouting, significantly outperforms CAI-MoE. For instance, on the Kimi-VL model, MACS (w/o Expanded) improves performance from 92.24% (CAI-MoE Token Drop, which also lacks rerouting) to 99.28% through Entropy-Weighted Load and Dynamic Modality-Adaptive Capacity alone. This observation indicates that information-aware load balancing, rather than token manipulation, is the primary factor in mitigating performance degradation under EP inference. Building upon this foundation, the full MACSfurther improves performance from 99.28% to 99.81% by incorporating Local Semantic Rerouting, which effectively recovers overflowed tokens and nearly closes the performance gap with the original Vanilla MoE.

4.3 Ablation Study

Table 2 presents an ablation study on Qwen3-VL under two base capacity settings, $\gamma_0 = 0.5$ and $\gamma_0 = 1.0$, where we progressively introduce its core components. We start from a static-capacity baseline (SC) and incrementally add Semantic Load (SL), Dynamic Modality-Adaptive Capacity (DC), and Semantic Rerouting (SR) to analyze their individual contributions.

We can find that SL consistently improves performance in both capacity settings. Compared to the static-capacity baseline, SL enables the model to distinguish between high-information and low-information visual tokens, reducing unnecessary capacity consumption by redundant background tokens. This results demonstrate that more accurate load modeling alone can effectively alleviate performance degradation under expert-parallel inference. Building on SL, the addition of DC yields further consistent improvements. DC dynamically adjusts expert capacity based on the effective modality composition of each input batch, leading to more stable performance across tasks with varying visual-to-text ratios. This effect is particularly pronounced in the constrained capacity setting ($\gamma_0 = 0.5$), indicating that dynamic capacity allocation is especially beneficial when expert resources are limited, and load imbalance is more severe. Incorporating SR provides additional and stable gains across all benchmarks and capacity settings. Although SR is not the primary source of improvement, it consistently narrows the remaining performance gap to the full-capacity model, complementing SL and DC to improve robustness under extreme load conditions.

Overall, the ablation results indicate that SL and DC are the core performance drivers of MACS,

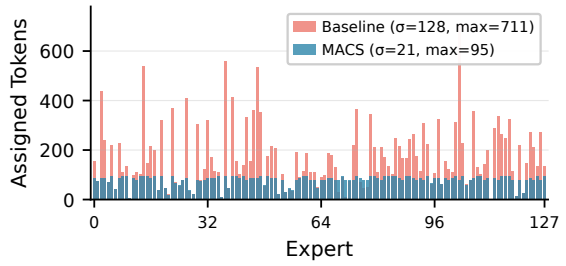

Figure 4: Mitigating the Straggler Effect in Qwen3-VL ($\gamma_0 = 0.5$). The x-axis shows the expert index, and the y-axis shows the expert load.

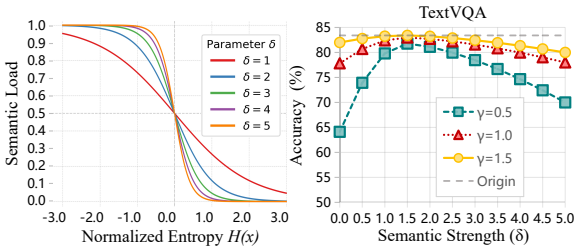

Figure 5: Sensitivity analysis of Semantic Strength (δ). (Left) Entropy-to-load mapping curves. (Right) TextVQA performance trends under varying base capacities (γ).

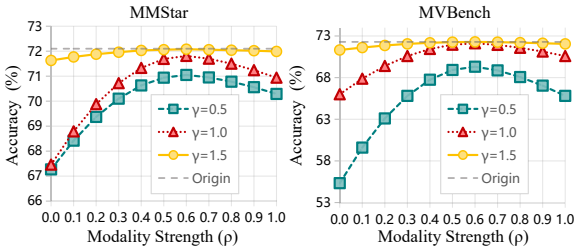

Figure 6: Sensitivity analysis of Modality Adaptation Strength (ρ). Performance on MMStar (image) and MVBench (video) under varying base capacities (γ_0). The gray dashed line denotes the unconstrained upper bound.

while Semantic Rerouting serves as a lightweight yet effective auxiliary mechanism to handle unavoidable overflows in EP inference.

4.4 Efficiency Analysis

As shown in Figure 4, our method significantly mitigates the straggler effect, reducing the maximum load from 711 to 95, thereby achieving more efficient expert parallel inference. To quantify the inference acceleration provided by MACS, we evaluated the end-to-end inference speed of the MoE layer and analyzed the sources of these gains through a latency breakdown. All experiments are conducted under an EP setting.

End-to-End Speedup. As shown in Figure 2, MACSsignificantly improves the inference speed of each MoE layer compared to the unconstrained baseline. This speedup is consistently observed on Qwen3-VL and Kimi-VL. As the capacity factor γ decreases, the inference speedup continues to increase, reaching up to 1.97× on Qwen3-VL. Furthermore, our Local Semantic Rerouting reassigns overflow tokens to local idle experts, and introduces negligible computational overhead while yielding substantial performance recovery (as shown in Table 1), achieving both efficiency and performance.

Latency Breakdown Analysis. To analyze the components of these speedups, we decompose the MoE layer's latency, as illustrated in Figure 3. The most significant latency reduction consistently occurs in the Expert Computation stage. For example, at $\gamma = 0.5$, expert computation latency is reduced by over 40% compared to the unconstrained baseline. This demonstrates that our capacity constraint mechanism effectively limits the number of tokens assigned to the busiest experts, reducing the system's waiting time. This analysis also highlights the superiority of our Local Semantic Rerouting. In MACS (Global), the overhead of the Token Dispatch and Result Aggregation stage increases noticeably. This is because global rerouting requires broadcasting and synchronizing more token information across all GPUs. In contrast, the communication overhead of our MACS (Local) is nearly identical to that of the MACS (w/o SR), demonstrating the significant advantage of the local strategy in controlling communication costs.

4.5 Parameters Sensitivity Analysis

As illustrated in Figure 5, δ governs the nonlinearity of the entropy mapping. Specifically, a low δ (< 1.0) results in a flattened weight distribution that fails to effectively suppress background noise. Experimental results on TextVQA demonstrate that $\delta \approx 1.5$ strikes the optimal balance, particularly in low-capacity scenarios ($\gamma = 0.5$), as it effectively distinguishes between foreground and background information without compromising subtle yet critical visual details (e.g., small OCR characters).

We investigated the impact of the modality adaptation strength ρ on multimodal tasks under varying base capacities (γ), as illustrated in Figure 6.

Method	γ	TextVQA	ChartQA	MMStar	MMBench	MMVet	MME	RealWorldQA	COCO	Avg. (%)
Baseline	$+\infty$	83.54	85.36	72.10	86.82	85.67	2500	73.72	80.28	100.00
Random	1.5	80.89	82.67	69.83	84.11	82.97	2422	71.41	78.92	97.04
Router-based		81.67	83.48	70.49	84.92	83.78	2446	72.09	78.97	97.87
Modality-Prior		82.59	84.37	71.27	85.83	84.66	2471	72.87	79.47	98.86
Entropy-weighted Load(Ours)		**82.93**	**84.72**	**71.53**	**86.17**	**84.97**	**2481**	**73.18**	**79.52**	**99.22**
Random	1.0	69.94	71.53	60.36	72.76	71.74	2092	61.69	68.71	83.97
Router-based		73.61	75.19	63.53	76.49	75.46	2204	64.93	71.69	88.25
Modality-Prior		77.42	79.13	66.58	81.34	79.21	2214	67.89	72.36	91.83
Entropy-weighted Load(Ours)		**79.28**	**81.01**	**68.39**	**82.41**	**81.32**	**2372**	**69.96**	**76.21**	**94.90**
Random	0.5	53.79	54.98	46.43	55.91	55.17	1611	47.47	50.47	64.21
Router-based		61.88	63.26	53.41	64.32	63.47	1852	54.61	59.53	74.09
Modality-Prior		68.91	68.79	59.81	74.29	68.79	2007	63.21	61.98	81.89
Entropy-weighted Load(Ours)		**74.96**	**76.62**	**64.71**	**77.94**	**76.88**	**2246**	**66.19**	**72.37**	**89.82**

Table 3: Comparison of different token selection strategies under varying capacity factors γ. We evaluate Random, Router-based, Modality-Prior, and MACS (Entropy-Weighted Load). The baseline operates without capacity constraints ($\gamma = +\infty$). γ controls the severity of capacity pressure during EP inference.

Specifically, for image tasks (MMStar), increasing ρ significantly overcomes the static bottleneck in resource-constrained scenarios ($\gamma = 0.5$), while under the standard setting ($\gamma = 1.0$), setting $\rho = 0.6$ restores the model to its original performance level. It is worth noting that when $\rho > 0.8$, a "Cannibalization Effect" is observed, where the excessive capacity expansion of visual experts encroaches upon the resources required for text reasoning, leading to a performance regression. In the context of video tasks (MVBench) with high-frame-rate inputs, a static low-capacity configuration ($\gamma = 0.5$) results in a performance collapse. However, MACS achieves a significant recovery of +12.5%, demonstrating its critical capability in handling highly redundant visual streams. In conclusion, $\rho = 0.6$ exhibits the best robustness across various modalities and resource constraints, achieving an effective balance between visual throughput and textual reasoning.

4.6 Analysis of Entropy-Weighted Load

To evaluate the effectiveness of Entropy-Weighted Load for quantifying token-level information value and expert load regulation, we conduct experiments on Qwen3-VL by comparing token dropping strategies based on three alternative weighting criteria: (1) Random, which discards tokens uniformly at random; (2) Router-based, which prioritizes tokens according to routing confidence; and (3) Modality-Prior, which preferentially drops visual tokens assumed to be more redundant. We vary the capacity factor $\gamma \in \{1.5, 1.0, 0.5\}$ to simulate different levels of capacity constraints under EP inference.

As shown in Table 3, our Entropy-weighted Load consistently achieves the best performance under all capacity settings. When $\gamma = 0.5$, the Entropy-weighted Load still preserves 89.82% of the baseline performance, while all alternative strategies show substantial degradation. The Random strategy performs the worst across all settings, indicating that capacity reduction alone, without information-aware token prioritization, is insufficient to maintain model performance. The Router-based strategy improves over random selection, but its routing scores primarily reflect an expert's preference for individual tokens rather than the tokens' intrinsic semantic importance. As a result, semantically critical tokens with low routing confidence may be erroneously discarded. The Modality-Prior strategy improves performance on language-sensitive tasks by enforcing the retention of text tokens, but it applies a coarse-grained approach to visual tokens. By failing to distinguish informative foreground regions from redundant background content, it leads to notable performance degradation on vision-centric tasks such as COCO.

These results demonstrate that the Entropy-Weighted Load provides an effective measure of token priority by enabling fine-grained modeling of the density of cross-modal information. This property allows for more reliable token retention and dropping decisions under capacity-constrained EP inference, and forms a solid foundation for the subsequent dynamic capacity scaling and semantic rerouting mechanisms in MACS.

5 Conclusion

In this work, we identify a significant efficiency bottleneck in MoE MLLMs under EP inference, where the straggler effect is worsened by two challenges unique to the multimodal domain: the information heterogeneity of visual tokens and modality dynamics across tasks. To address these challenges,

we propose MACS, which mitigates information heterogeneity through an Entropy-Weighted Load mechanism and adapts to modality dynamics with its Dynamic Modality-Adaptive Capacity. Extensive experiments demonstrate that MACS significantly outperforms existing methods across various MoE MLLMs, providing a practical solution for the efficient deployment of multimodal MoE models.

Limitations

Despite the significant efficiency improvements demonstrated by MACS in the expert parallel inference of multimodal MoE models, this work has the following limitations:

Experimental Scale. Our evaluation was primarily conducted within a standard 8-GPU environment. While this setup reflects mainstream high-performance scenarios, the scalability and communication overhead of MACS in ultra-large-scale clusters (e.g., spanning hundreds of GPUs) remain to be fully verified. In such expanded settings, inter-node communication latency may become a more dominant factor affecting overall efficiency.

Generalization across Modalities. We effectively validated MACS on Vision-Text multimodal models. However, its applicability to other modalities, such as audio or 3D point clouds, has not yet been explored. Given that information density distributions may vary significantly across different data types, direct application of the current entropy calculation mechanism may require further adaptation.

Ethics Statement

This study adheres to the ethical guidelines set forth by our institution and follows the principles outlined in the ACM Code of Ethics and Professional Conduct. All datasets used in our experiments are publicly available.

Acknowledgements

The present research was supported by the National Key Research and Development Program (Grant No.2023YFE0116400) and the National Natural Science Foundation of China Youth Fund (Grant No.62306210). We would like to thank the anonymous reviewers for their insightful comments.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025. Qwen3-vl technical report. *Preprint*, arXiv:2511.21631.

Weilin Cai, Juyong Jiang, Le Qin, Junwei Cui, Sunghun Kim, and Jiayi Huang. 2024. Shortcut-connected expert parallelism for accelerating mixture-of-experts. *arXiv preprint arXiv:2404.05019*.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. 2024. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*.

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.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, and 1 others. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, and 1 others. 2025. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24108–24118.

Shwai He, Weilin Cai, Jiayi Huang, and Ang Li. 2025. Capacity-aware inference: Mitigating the straggler effect in mixture of experts. *arXiv preprint arXiv:2503.05066*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. 2025. Video-mmmu: Evaluating knowledge acquisition from multi-discipline professional videos. *arXiv preprint arXiv:2501.13826*.

Wei Huang, Yue Liao, Jianhui Liu, Ruifei He, Haoru Tan, Shiming Zhang, Hongsheng Li, Si Liu, and Xiaojuan Qi. 2024. Mixture compressor for mixture-of-experts llms gains more. *arXiv preprint arXiv:2410.06270*.

Jaeseong Lee, Seung-won Hwang, Aurick Qiao, Daniel F Campos, Zhewei Yao, and Yuxiong He. 2025. Stun: Structured-then-unstructured pruning for scalable moe pruning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13660–13676.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Bo Li, Shaolin Zhu, and Lijie Wen. 2025a. MIT-10M: A large scale parallel corpus of multilingual image translation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5154–5167, Abu Dhabi, UAE. Association for Computational Linguistics.

Junchen Li, Qing Yang, Bojian Jiang, Shaolin Zhu, and Qingxuan Sun. 2025b. Lrm-llava: Overcoming the modality gap of multilingual large language-vision model for low-resource languages. In *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pages 24449–24457. AAAI Press.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. 2024b. Mvbench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*.

Shangjie Li, Xiangpeng Wei, Shaolin Zhu, Jun Xie, Baosong Yang, and Deyi Xiong. 2023. MMNMT: Modularizing multilingual neural machine translation with flexibly assembled MoE and dense blocks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4978–4990, Singapore. Association for Computational Linguistics.

Yi Li, Hualiang Wang, Xinpeng Ding, Haonan Wang, and Xiaomeng Li. 2025c. Token activation map to visually explain multimodal llms. *arXiv preprint arXiv:2506.23270*.

Jiawei Liang, Ruoyu Chen, Xianghao Jiao, Siyuan Liang, Shiming Liu, Qunli Zhang, Zheng Hu, and Xiaochun Cao. 2025. Explaining multimodal llms via intra-modal token interactions. *arXiv preprint arXiv:2509.22415*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. *Preprint*, arXiv:2310.03744.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024b. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Xudong Lu, Qi Liu, Yuhui Xu, Aojun Zhou, Siyuan Huang, Bo Zhang, Junchi Yan, and Hongsheng Li. 2024. Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models. *arXiv preprint arXiv:2402.14800*.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. 2023. Egoschema: A diagnostic benchmark for very long-form video language understanding. *arXiv preprint arXiv:2308.09126*.

Ahmed Masry, Do Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.

OpenAI. 2025. Chatgpt. Large language model.

Xiaoye Qu, Daize Dong, Xuyang Hu, Tong Zhu, Weigao Sun, and Yu Cheng. 2024. Llama-moe v2: Exploring sparsity of llama from perspective of mixture-of-experts with post-training. *arXiv preprint arXiv:2411.15708*.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326.

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

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, and 1 others. 2025. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Chuan Wu, Meng Su, Youxuan Fang, and Shaolin Zhu. 2025. Unveiling multimodal processing: Exploring activation patterns in multimodal llms for interpretability and efficiency. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 9005–9016.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. 2024. Longvideobench: A benchmark for long-context interleaved video-language understanding. *arXiv preprint arXiv:2407.15754*.

x.ai. 2024. Grok-1.5 vision preview. https://x.ai/blog/grok-1.5v.

Yanyue Xie, Zhi Zhang, Ding Zhou, Cong Xie, Ziang Song, Xin Liu, Yanzhi Wang, Xue Lin, and An Xu. 2024. Moe-pruner: Pruning mixture-of-experts large language model using the hints from its router. *arXiv preprint arXiv:2410.12013*.

Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739*.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*.

Danyang Zhang, Junhao Song, Ziqian Bi, Yingfang Yuan, Tianyang Wang, Joe Yeong, and Junfeng Hao. 2025a. Mixture of experts in large language models. *arXiv preprint arXiv:2507.11181*.

Dingkun Zhang, Shuhan Qi, Xinyu Xiao, Kehai Chen, and Xuan Wang. 2025b. Merge then realign: Simple and effective modality-incremental continual learning for multimodal LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13148–13164, Suzhou, China. Association for Computational Linguistics.

Yu Zhang, Jinlong Ma, Yongshuai Hou, Xuefeng Bai, Kehai Chen, Yang Xiang, Jun Yu, and Min Zhang. 2026. Evaluating and steering modality preferences in multimodal large language model. *Preprint*, arXiv:2505.20977.

Shaolin Zhu, Shangjie Li, Yikun Lei, and Deyi Xiong. 2023. PEIT: Bridging the modality gap with pretrained models for end-to-end image translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13433–13447, Toronto, Canada. Association for Computational Linguistics.

Shaolin Zhu, Leiyu Pan, Dong Jian, and Deyi Xiong. 2025. Overcoming language barriers via machine translation with sparse mixture-of-experts fusion of large language models. *Information Processing & Management*, 62(3):104078.

Fei Zuo, Kehai Chen, Yu Zhang, Zhengshan Xue, and Min Zhang. 2025. InImageTrans: Multimodal LLM-based text image machine translation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20256–20277, Vienna, Austria. Association for Computational Linguistics.

A Appendix

A.1 Implementation Details

Hyperparameter Settings. For the main experiments, we employ a consistent hyperparameter configuration for the MACS framework, based on the sensitivity analysis presented in Section 4.5.

Specifically, the semantic strength parameter δ in the Entropy-Weighted Load mechanism is set to 1.5. This value was selected to reduce the weights of low-information background tokens while retaining necessary visual details (e.g., small OCR characters).

The modality adaptation strength ρ in the Dynamic Modality-Adaptive Capacity mechanism is set to 0.6. This setting balances visual throughput with the capacity required for textual reasoning, which helps mitigate resource competition between modalities (referred to as the "Cannibalization Effect"). Unless otherwise specified (e.g., in ablation studies), these parameters ($\delta = 1.5, \rho = 0.6$) are applied consistently across all evaluated models (Qwen3-VL, InternVL3.5, and Kimi-VL) and benchmarks.

A.2 Expert Calibration and Classification Details

Calibration Set Configuration To mitigate distribution bias and ensure statistical significance during expert classification, we construct a balanced calibration dataset $\mathcal{D}_{calib} = \mathcal{D}_{txt} \cup \mathcal{D}_{vis}$ comprising a total of 16,384 samples. The dataset is strictly stratified into two modalities:

- **Text Modality** ($\mathcal{D}_{txt}$, $N = 8192$): Randomly sampled from the MMLU benchmark(Hendrycks et al., 2021). This covers a broad spectrum of domains, including STEM and humanities, ensuring the generality of the text expert activation distribution.
- **Visual Modality** ($\mathcal{D}_{vis}$, $N = 8192$): Randomly sampled from the LLaVA-OneVision dataset(Li et al., 2024a), encompassing general imagery alongside complex visual scenarios such as OCR and charts.

Regarding convergence, our empirical observations demonstrate that the Kullback-Leibler (KL) divergence of the expert activation distribution reaches convergence ($\Delta KL < 10^{-3}$) at $N \approx 6000$. Consequently, allocating 8,192 samples per modality provides a sufficient margin to accurately capture the intrinsic routing preferences of the experts.

Classification Basis To quantify the modality bias of each expert, we first compute the activation frequency $f_j^{(m)}$ of expert E_j for a given modality $m \in \{txt, vis\}$:

$$f_j^{(m)} = \frac{1}{|\mathcal{D}_m|} \sum_{x \in \mathcal{D}_m} \mathbb{I}\big(E_j \in \text{TopK}(G(x))\big) \quad (12)$$

where $\mathbb{I}(\cdot)$ represents the indicator function and $G(x)$ denotes the routing network. We subsequently define the modality specialization score Δ_j for each expert as the difference in activation frequencies:

$$\Delta_j = f_j^{(vis)} - f_j^{(txt)} \quad (13)$$

Based on this score, we partition the expert set $\mathcal{E}$ into three mutually exclusive subsets using a threshold $\delta = 0.1$. Experts exhibiting a significant modality preference are categorized as *modality-specific experts*: visual experts are defined as $\mathcal{E}_{vis} = \{E_j \in \mathcal{E} \mid \Delta_j \geq \delta\}$, and text experts as $\mathcal{E}_{txt} = \{E_j \in \mathcal{E} \mid \Delta_j \leq -\delta\}$. Conversely, experts demonstrating minimal variance in activation frequency across modalities are designated as *multimodal shared experts*, defined as $\mathcal{E}_{shared} = \{E_j \in \mathcal{E} \mid |\Delta_j| < \delta\}$. These shared experts are primarily responsible for cross-modal alignment and general reasoning tasks.

A.3 Centroid Computation and Memory Overhead

Offline Centroid Computation The computation of expert centroids is performed offline prior to model deployment, introducing zero training overhead. To obtain stable representations, we reuse the aforementioned calibration dataset. For each expert E_k within a given layer, we aggregate the hidden states z of all tokens routed to it. The centroid μ_k is derived via mean pooling of these token embeddings:

$$\mu_k = \frac{1}{|Z_k|} \sum_{z \in Z_k} z \quad (14)$$

where Z_k represents the set of tokens assigned to expert E_k. This procedure is completed in advance, and the computed centroids μ_k remain static and frozen throughout the entire inference phase.

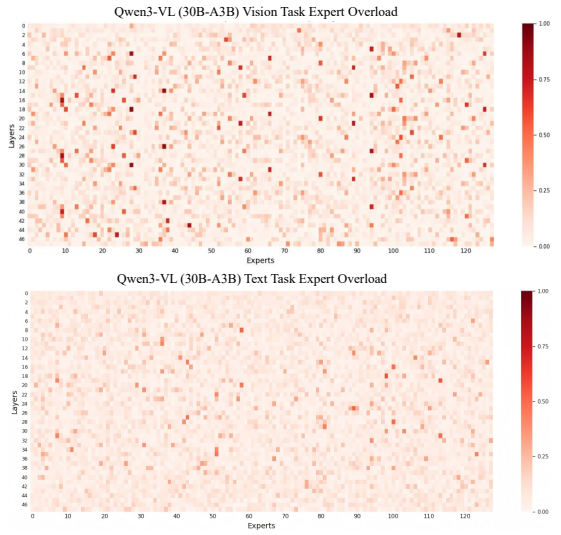

Figure 7: Normalized expert load on vision and text tasks in Qwen3-VL-30B-A3B.

Memory Overhead Analysis The memory footprint required to store these static centroids is negligible relative to the overall VRAM usage of the large language model. The storage overhead M is calculated as $M = L \times N \times D \times P$, where L denotes the total number of layers, N is the number of experts per layer, D represents the hidden dimension size, and P indicates the byte size of the numerical precision (e.g., $P = 2$ for FP16 precision).

Taking the Qwen3-VL-30B architecture as an illustrative example ($L = 48$, $N = 128$, $D = 2048$), the total memory overhead is approximately $48 \times 128 \times 2048 \times 2$ bytes, which equals roughly 25 MB. Given that the parameter size of the 30B model requires approximately 60 GB of memory, this additional centroid storage constitutes less than 0.05% of the total memory footprint. Consequently, the proposed mechanism introduces minimal impact on practical deployment resource constraints.

A.4 Straggler Effect in Multimodal Scenarios

We further analyze the normalized expert load heatmaps presented in Figure 7 to investigate the root cause of inference latency in multimodal MoE models.

Disparity in Load Distribution. As illustrated in Figure 7, there is a sharp contrast between the activation patterns of vision (top) and text (bottom) tasks. The text modality exhibits a relatively uniform and sparse distribution of expert utilization. In contrast, the vision task demonstrates significant load skewness, characterized by distinct "hotspots" (dark red clusters) where the normalized load approaches 1.0. This indicates that visual tokens tend to disproportionately congregate on specific experts, rapidly saturating their capacity.

Exacerbated Straggler Effect. The observed load imbalance in the vision modality directly contributes to a severe Straggler Effect. In MoE inference, the latency of a layer is bounded by the expert with the highest computational load (the straggler). The dense high-load clusters in the vision heatmap suggest that visual tasks create higher synchronization barriers compared to text tasks. Consequently, in a unified architecture, the vision modality acts as the primary bottleneck, intensifying the latency tail and justifying the necessity for modality-aware capacity management strategies.

A.5 Capacity-Tradeoff Analysis

As illustrated in Figure 8, evaluating the base capacity factor γ_0 from 0.05 to 5.0 reveals a non-linear relationship between capacity allocation and routing behavior. We categorize this into three operational regimes:

Congestion Regime ($\gamma_0 < 1.5$). In this lower capacity range, both token drop and rerouting rates are high. The limited expert capacity requires the router to either discard tokens or redirect them to alternative experts. As γ_0 increases toward 1.5, the drop rate exhibits a continuous decrease, indicating that adding capacity directly mitigates token loss.

Optimal Efficiency Regime ($\gamma_0 \approx 1.8$). At $\gamma_0 = 1.8$, the system reaches an optimal operational point. The rerouting rate achieves its global minimum at 18.4%, and the drop rate is simultaneously reduced to a low level. This configuration provides an effective balance between preserving token information and maintaining stable routing assignments.

Saturation and Expansion Regime ($\gamma_0 > 2.0$). When $\gamma_0 > 2.0$, the drop rate stabilizes near zero, while the rerouting rate exhibits a increase. This rise in rerouting is driven by our local expert expansion strategy. As capacity becomes abundant, the routing network actively utilizes the available slots in idle experts to process tokens not originally assigned to them. This saturated expansion mechanism ensures load balancing and increases the utilization of experts with low loads. Although

Method	Image Understanding							Video Understanding					Avg.
	TextVQA	ChartQA	MMStar	MMBench	MMVet	MME	RWQA	MVBench	EgoSch	VMME	LVB	VMMMU	(%)
QWEN3-VL-30B-A3B-INSTRUCT ($\gamma_0 = 0.5$)													
Vanilla MoE	83.54	85.36	72.10	86.82	85.67	2500	73.72	72.29	63.28	74.53	62.47	68.64	100.00
CAI-MoE (Token Drop)	68.91	68.79	59.81	74.29	68.79	2007	63.21	60.22	50.37	61.58	49.73	54.64	81.89
CAI-MoE (Expanded)	71.51	68.47	58.18	72.58	70.66	2144	60.99	58.89	56.11	62.93	53.21	55.91	83.52
MACS (w/o Expanded)	79.11	**81.66**	67.61	83.41	81.04	2376	69.74	**71.06**	**60.42**	69.99	**60.43**	64.29	95.21
MACS (Ours)	**81.04**	80.96	**68.84**	**84.21**	**83.56**	**2424**	**71.28**	69.76	60.32	**71.61**	60.18	**67.17**	**96.47**
INTERNVL3.5-30B-A3B ($\gamma_0 = 0.5$)													
Vanilla MoE	85.68	84.14	72.03	84.68	85.43	2324	64.87	72.06	60.37	68.65	63.76	65.24	100.00
CAI-MoE (Token Drop)	71.14	64.54	59.76	68.46	68.32	1948	55.21	61.11	47.98	55.66	52.92	55.41	82.14
CAI-MoE (Expanded)	72.66	71.97	59.96	70.29	75.21	2077	55.67	63.78	52.46	57.93	55.41	58.61	86.36
MACS (w/o Expanded)	82.21	**82.76**	66.32	82.34	82.12	2220	62.36	67.03	57.76	66.27	**62.62**	62.12	95.84
MACS (Ours)	**82.67**	81.46	**70.74**	**82.67**	**82.67**	**2273**	**63.18**	**70.81**	**58.32**	**66.57**	61.79	**62.52**	**97.14**
KIMI-VL-A3B-INSTRUCT ($\gamma_0 = 0.5$)													
Vanilla MoE	88.39	87.26	61.25	83.11	77.84	2218	68.07	62.73	78.32	66.84	64.37	57.58	100.00
CAI-MoE (Token Drop)	71.76	70.72	49.01	65.71	57.87	1737	53.52	48.17	62.16	54.56	56.36	46.47	79.89
CAI-MoE (Expanded)	75.37	73.54	48.23	70.16	65.92	2038	57.13	53.57	66.11	54.14	56.61	49.96	84.89
MACS (w/o Expanded)	**85.73**	82.41	**59.97**	77.87	**75.26**	**2197**	64.21	59.56	75.16	63.53	60.17	55.22	95.70
MACS (Ours)	84.89	**85.14**	58.91	**81.99**	74.97	2146	**66.66**	**60.14**	**76.23**	**65.78**	**61.47**	**56.02**	**96.99**

Table 4: Performance comparison of MACS against the SOTA distributed MoE inference acceleration method CAI-MoE on multimodal benchmarks. We evaluate on Qwen3-VL, InternVL3.5, and Kimi-VL, comparing against CAI-MoE's *Token Drop* and *Expanded Drop* variants. "Vanilla MoE" denotes the unconstrained baseline. All acceleration methods use a base capacity factor $\gamma_0 = 0.5$. "w/o Expanded" denotes the variant without local expansion, while "Ours" represents the full method.

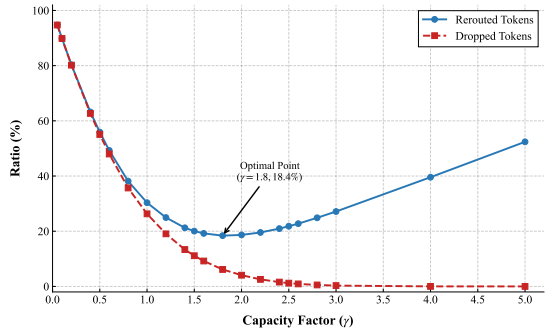

Figure 8: Impact of Capacity Factor (γ_0) on Token Rerouting and Dropping Rates.

this process may cause certain tokens to be processed by more than k experts, maintaining a strict k-expert constraint per token is unnecessary. Permitting the selection of additional experts improves the model's representational capacity, whereas enforcing a strict limit would introduce redundant computation.

A.6 Performance under Low Capacity Constraints

We present a performance comparison of MACS against the inference acceleration method, CAI-MoE, across three multimodal MoE architectures: Qwen3-VL, InternVL3.5, and Kimi-VL. As detailed in Table 4, all acceleration methods are evaluated under a constrained capacity setting of $\gamma_0 = 0.5$.

Performance against Baseline Methods. Under this constrained setting, existing methods such as CAI-MoE (Token Drop) experience performance decreases, retaining 81.89% of the Vanilla MoE performance on average for Qwen3-VL. This suggests that non-selective token dropping strategies may discard necessary visual information. MACS, by contrast, maintains higher performance levels, achieving 96.47% on Qwen3-VL and 97.14% on InternVL3.5. This indicates that modality-aware capacity allocation helps preserve model performance under tight capacity constraints.

Evaluation Across Modalities. The performance retention of MACS is observed across both image understanding (e.g., TextVQA, MMBench) and video understanding benchmarks (e.g., MVBench). For instance, on the ChartQA benchmark, the performance of Qwen3-VL drops from 85.36 to 68.79 when using CAI-MoE (Token Drop). MACS mitigates this decrease, achieving a score of 80.96. This suggests the proposed framework is more effective at retaining tokens necessary for these reasoning tasks.

Contribution of Local Expansion. The results in Table 4 also illustrate the effect of the local

expansion mechanism. Comparing *MACS (w/o Expanded)* with the full *MACS (Ours)*, there is a consistent performance improvement across the evaluated models. For example, on Qwen3-VL, the average performance increases from 95.21% to 96.47%, representing a $+1.26\%$ relative improvement. While the modality-aware capacity scaling provides the primary performance retention, the local expansion mechanism further utilizes available expert capacity to process additional tokens, contributing to overall performance without introducing significant communication overhead.