ROUTING BY NEED, NOT BY MODALITY: DEMAND-CONDITIONED SPARSE ACTIVATION FOR MOE-VLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse activation in a vision-language mixture-of-experts model must decide both which experts a token uses and how much computation that token receives. Modality identity describes the origin of a token, but does not itself specify how strongly its current representation depends on complementary visual or linguistic information. We propose CoDA-MoE, a demand-conditioned routing framework that separates this information requirement from expert preference. A lightweight probe estimates token-level cross-modal demand from features available in a single backbone pass. An integer allocator uses these scores to redistribute expert activations across tokens while enforcing an explicit global sparsity bound. Expert choice is complemented by a calibration procedure that measures leave-one-out utility, correlates expert utility profiles, and forms capacity-constrained groups that also serve as expert-parallel placement units. This connects token-level allocation to deployment without equating modality affinity with functional usefulness. The formulation includes probe supervision, budget feasibility, and placement constraints. The resulting framework defines testable comparisons at matched activation counts and separates expert computation from communication cost.

1 INTRODUCTION

Mixture-of-experts (MoE) layers separate the number of available parameters from the number evaluated for an individual token through sparse expert routing (Shazeer et al., 2017; Lepikhin et al., 2020). Modern language-model designs instantiate this separation with different expert granularities and shared components (Jiang et al., 2024; Liu et al., 2024a; Yang et al., 2025). For a vision-language model, this computational decision must also accommodate representations that combine information from different input modalities. A token's visual or linguistic origin is observable, whereas its remaining cross-modal information requirement is a property of the current representation and task context. CoDA-MoE investigates whether the latter quantity is a better control signal for distributing a fixed amount of expert computation.

Consider a question token that can be resolved from language context and another whose interpretation depends on a small visual detail. They share a modality label, yet the desired dependence on the image differs in this illustrative case. Conversely, a visual token and a text token can both participate in an unresolved alignment, so assigning disjoint budgets solely by origin can miss their common demand. These examples motivate the hypothesis that modality identity and computational need need not coincide. We call a token a high-demand boundary token when the probe indicates unresolved dependence on complementary information, not merely because it lies next to an image marker or a segment boundary.

Variable expert counts have been explored through adaptive gating, dynamic routing, and null experts (Li et al., 2023a; Huang et al., 2024; Zeng et al., 2024). These directions motivate separating the question of how many experts to use from the question of which experts score highest. They do not, by themselves, define the cross-modal demand estimand used here or impose our chosen integer budget across the full routing batch. Expert pruning and merging address a different resource axis by changing the expert bank (Lu et al., 2024; Li et al., 2023b; Chen et al., 2025). Here the expert

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.

parameters remain distinct: grouping is intended to organize their placement, not to identify correlated experts as interchangeable.

CoDA-MoE links three decisions. A single-pass demand probe scores each token using its current state and a causally available complementary-context summary. A globally bounded integer allocator then directs more optional activation slots toward higher-scoring tokens while retaining a configurable minimum allocation for language-only and low-demand inputs. Separately, a calibration probe measures the change in a local task-prediction loss when one selected expert contribution is removed. Correlation between these utility profiles defines capacity-constrained expert groups, which a placement stage assigns to devices using observed dispatch traffic and hardware limits. This decomposition does not assert that correlation guarantees communication savings: the traffic model and eventual system measurements must test that link.

CoDA-MoE provides a concrete formulation of demand-conditioned routing. It defines demand supervision without online modality ablations, a feasible integer allocation rule with a simple optimality property for its stated surrogate, and an explicit leave-one-out utility-to-placement interface. Sections 2–4 develop these choices and their limitations.

2 Related Work

2.1 Sparse routing and variable computation

Sparsely gated MoE layers and sharded conditional computation provide the starting point for expert selection under limited per-token computation (Shazeer et al., 2017; Lepikhin et al., 2020). Adaptive Gating and dynamic-routing methods examine token-dependent activation rather than applying one expert count everywhere (Li et al., 2023a; Huang et al., 2024; Guo et al., 2024). XMoE and sparsity-controllable dynamic top-p routing further study adaptive selection and sparsity control (Yang et al., 2024; Jin et al., 2025). CoDA-MoE shares the objective of allocating computation nonuniformly, but its specified score predicts cross-modal dependence rather than directly treating router confidence as the desired allocation quantity. Its hard batch-level bound is a mathematical contract of the proposed allocator, not an empirical claim that alternative methods violate their own resource targets.

2.2 Skipping, null experts, and compression

AdaMoE and MoE++ use null or zero-computation expert mechanisms to introduce token-adaptive computation (Zeng et al., 2024; Jin et al., 2024). Expert pruning and skipping instead investigate removing expert computation from an existing model (Lu et al., 2024; Liu et al., 2024b). Merge-then-compress and hierarchical expert merging study reductions in the expert representation itself (Li et al., 2023b; Chen et al., 2025). The CoDA-MoE formulation neither adds a dummy expert nor merges expert weights. A minimum routed count is retained by default, and the demand allocator decides only how to distribute optional slots. This distinguishes the current design from the BEAM binary-mask mechanism whose historical experiments are preserved in the reference portion.

2.3 Functional grouping and systems evidence

Clustering for expert merging provides a precedent for examining relationships between experts rather than considering every expert independently (Chen et al., 2025; Li et al., 2023b). Our proposed relationship is correlation of loss changes under controlled leave-one-out interventions, not a distance between weights or an expert's modality-frequency histogram. The groups remain collections of separate experts and are passed to a placement problem with memory and load constraints. Serving frameworks such as vLLM provide relevant systems context (Kwon et al., 2023). The grouping objective connects this context to task-conditioned expert utility and explicit device capacities.

2.4 Scope of the comparison

Here, modality-identity routing denotes a controlled comparator whose allocation depends on token origin rather than a calibrated demand estimate. This is an operational baseline definition, not attribution of a specific implementation to an unnamed prior paper. The inherited bibliography

supports the sparse-routing and compression context, but is not a comprehensive survey of modality-guided MoE-VLM routing. Comparisons require named modality-guided baselines with documented routing rules and matched activation budgets.

3 Preliminaries and Problem Formulation

Let $\mathcal{T}_\ell$ be the set of valid tokens processed at MoE layer ℓ in one routing step, with $n_\ell = |\mathcal{T}_\ell|$. This set includes all participating expert-parallel ranks, excludes padding, and is fixed before allocation. Let $h_{\ell t} \in \mathbb{R}^d$ be a token's pre-MoE hidden state, $x_{\ell t} = \mathcal{N}_\ell(h_{\ell t})$ its normalized state, and $r_{\ell te}$ the router logit for routed expert $e \in \{1, \dots, E_\ell\}$. The origin tag $m_t \in \{\mathrm{v}, \mathrm{l}\}$ records whether the token entered as a visual or linguistic token; it does not prevent its hidden state from containing both kinds of information. Special tokens inherit their configured stream tag, which must be recorded in a future implementation.

For a selected nonempty set $S_{\ell t}$, define routing weights and the residual update by

$$a_{\ell te}(S) = \frac{\exp r_{\ell te}}{\sum_{j \in S} \exp r_{\ell tj}}, \quad e \in S, \tag{1}$$

$$z_{\ell t}(S) = h_{\ell t} + s_\ell(x_{\ell t}) + \sum_{e \in S} a_{\ell te}(S)\mathcal{E}_{\ell e}(x_{\ell t}). \tag{2}$$

Here s_ℓ denotes the existing shared-expert contribution, or zero if none exists. The specification preserves the backbone's other residual and normalization operations; z denotes only the MoE residual update. Fixed top-k routing makes $|S_{\ell t}|$ constant, whereas CoDA-MoE chooses an integer $k_{\ell t} = |S_{\ell t}|$ from a bounded interval. An empty selection, if explicitly enabled, has zero routed contribution and does not evaluate the normalization in Equation (1).

Choose integer limits $0 \le k_{\min} \le k_{\max} \le E_\ell$ and an average activation target $\bar{k}_\ell \in [k_{\min}, k_{\max}]$ before a routing step. The default proposal is $k_{\min} = 1$, so zero cross-modal demand is not confused with zero linguistic computation. The layer-wide budget and its feasibility conditions are

$$B_\ell = \lfloor n_\ell \bar{k}_\ell \rfloor, \qquad k_{\ell t} \in \{k_{\min}, \dots, k_{\max}\}, \qquad \sum_{t \in \mathcal{T}_\ell} k_{\ell t} \le B_\ell. \tag{3}$$

Thus "global" means across all tokens and ranks in that routing step, not a separate unconstrained quota per GPU. For a completed sequence of steps and layers, summing Equation (3) bounds the total routed activation count by the sum of their declared budgets. We do not optimize future-layer demand or borrow capacity from unseen decoding steps. Shared experts, attention, probes, and dispatch have separate costs and are not hidden inside this routed-expert count.

For homogeneous routed experts the step's activation sparsity is $1 - \sum_t k_{\ell t}/(n_\ell E_\ell)$ when $n_\ell > 0$. Equal activation counts imply equal nominal routed-expert work only when expert shapes and arithmetic are matched. They do not imply equal latency or network traffic, because placement, padding, and device imbalance can change those quantities. We therefore formulate CoDA-MoE as allocation followed by traffic-aware deployment, and treat quality, computational cost, and communication as distinct outcomes requiring distinct evidence.

4 CoDA-MoE Method

CoDA-MoE combines single-pass demand estimation, demand-conditioned activation under a global sparsity constraint, and leave-one-out utility correlation for grouping and placement. The following subsections define the probe, integer allocator, correlation groups, and device mapping objective. Figure 1 separates the inherited expert-preference path from demand-based count allocation and offline placement.

4.1 Single-pass demand estimation

Demand is the sensitivity of a token's representation to complementary input information, not its attention mass or origin. Let $\mathcal{C}_{\ell t}$ contain the complementary-modality tokens visible to token t under the attention mask, with running mean $c_{\ell t} = |\mathcal{C}_{\ell t}|^{-1} \sum_{s \in \mathcal{C}_{\ell t}} v_{\ell s}$ of their normalized pre-attention

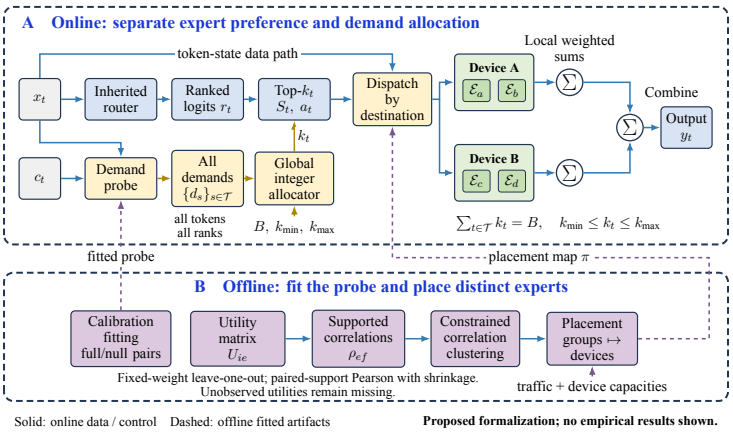


Figure 1: CoDA-MoE routing and expert placement. The inherited router maps x_t to ranked logits, while the demand probe uses x_t and complementary context c_t to supply the global integer allocator; only its count k_t controls the same top-k_t selection. Selected indices, normalized weights, and token states enter dispatch; distinct experts on each device produce local weighted sums, which combine into routed output y_t. The backbone residual and any shared-expert contribution remain unchanged. Offline fixed-weight leave-one-out utilities yield supported-pair correlations, constrained groups, and a placement map for dispatch; calibration fitting supplies the demand probe. Device and expert labels are symbolic.

states ($c_{\ell t} = 0$ if empty, supporting causal prefix sums), and $q_{\ell t} = \mathbf{1}[|\mathcal{C}_{\ell t}| > 0]$. With low-rank projections $P_\ell, Q_\ell \in \mathbb{R}^{p\times d}$, $p \ll d$, and a small scalar MLP f_ℓ, the online probe is

$$u_{\ell t} = P_\ell x_{\ell t}, \qquad v'_{\ell t} = Q_\ell c_{\ell t}, \tag{4}$$

$$d_{\ell t} = q_{\ell t}\,\sigma(f_\ell([u_{\ell t}; v'_{\ell t}; u_{\ell t} \odot v'_{\ell t}; u_{\ell t} - v'_{\ell t}]))\,, \tag{5}$$

so $d_{\ell t} \in [0, 1]$ is the predicted demand; the origin tag selects complementary context but is not itself a modality budget. The probe adds $O(n_\ell dp)$ work per layer plus context aggregation.

For supervision, a frozen calibration backbone is run on paired inputs in which complementary embeddings are replaced by a null embedding, and the target is the clipped state deviation

$$\widetilde{d}_{\ell t} = q_{\ell t}\,\min\left\{1, \frac{\|h^{\text{full}}_{\ell t} - h^{\text{abl}}_{\ell t}\|_2}{\tau_\ell(\|h^{\text{full}}_{\ell t}\|_2 + \epsilon)}\right\}, \tag{6}$$

fit by mean squared error on a calibration split with τ_ℓ chosen from the calibration sensitivity distribution. The ablation pass is offline only; online inference uses Equation (5) in a single pass.

4.2 Globally bounded integer allocation

Suppress the layer index within a routing step. The surrogate return from assigning k experts to token t is $F_t(k) = d_t \log(1 + k - k_{\min})$, whose diminishing marginal returns keep one token from consuming every optional slot. The allocator maximizes $\sum_t F_t(k_t)$ subject to Equation (3). Starting from $k_t = k_{\min}$ with $R = B - nk_{\min}$ optional slots, each token offers $k_{\max} - k_{\min}$ increments with priorities

$$\Delta_{tj} = d_t \log \frac{j+1}{j}, \qquad j = 1, \ldots, k_{\max} - k_{\min}. \tag{7}$$

The R largest increments are selected globally, ties broken by smaller j then a stable token identifier, so each token's selected increments form a prefix. Allocating exactly B slots enables the equal-count comparison with fixed top-k.

Proposition 4.1 (Feasibility and surrogate optimality)**.** *For feasible B, the increment rule yields integer allocations within the per-token limits with $\sum_t k_t = B$, maximizing the stated separable surrogate under the global constraint.*

Proof. Priorities are nonnegative and nonincreasing in j, and the tie rule makes selected sets prefixes that telescope to $F_t(k_t)$; the globally largest R increments dominate any other feasible selection of R or fewer increments. □

The exact allocator gathers and sorts $n(k_{\max}-k_{\min})$ priorities per microbatch ($O(nk_{\max}\log(nk_{\max}))$ worst case); the microbatch is the allocation domain. Given k_t, experts are selected as $S_t = \mathrm{Top}_{k_t}(r_t)$ by the inherited router and Equation (2) is evaluated with weights renormalized over S_t: demand controls count, the router controls identity, and capacity overflow is handled by microchunk queueing rather than by dropping pairs.

4.3 Leave-one-out expert utility and correlation groups

Utility is calibrated offline on labeled answer-prediction positions. For a fixed checkpoint and allocation, fit a local task-prediction head $p_{\psi_\ell}(y \mid z)$ on a disjoint split and freeze it; with $L_i(z) = -\log p_{\psi_\ell}(y_i \mid z)$, each selected expert e is removed without changing any other weight, score, or output:

$$z_i^{(-e)} = z_i - a_{ie}(S_i)\mathcal{E}_{\ell e}(x_i), \quad e \in S_i, \tag{8}$$

$$U_{ie} = L_i(z_i^{(-e)}) - L_i(z_i), \quad e \in S_i. \tag{9}$$

Negative utility is retained, survivors are not renormalized, and unselected experts have missing rather than zero utility. For experts e, f, the correlation ρ_{ef} uses only their common observations I_{ef}, is omitted when support or variance is insufficient, and is shrunk by $w_{ef} = |I_{ef}|/(|I_{ef}| + \lambda)$.

Let $\mathcal{G}_\ell$ partition experts into groups of at most C_ℓ (single-device memory). With $W_{ef}^{+} = w_{ef}\max(\rho_{ef}, 0)$ and $W_{ef}^{-} = w_{ef}\max(-\rho_{ef}, 0)$, groups minimize signed disagreements

$$J(\mathcal{G}_\ell) = \sum_{e<f:\, I_{ef}\ \text{valid}} \left[W_{ef}^{+}\mathbf{1}[g(e) \neq g(f)] + W_{ef}^{-}\mathbf{1}[g(e) = g(f)] \right], \tag{10}$$

by a deterministic greedy merge that repeatedly accepts the capacity-feasible merge with the largest strict decrease in J, ties by expert index. The groups retain all expert parameters and serve as placement units only.

4.4 Capacity-constrained expert-parallel placement

A dispatch trace is collected under the fixed CoDA-MoE allocation on calibration traffic. With token origin device $o(t)$, hidden-state size b_t, group memory M_g, and per-activation work c_e, a placement $\pi(g) \in \{1, \ldots, D\}$ assigns each group intact to one device. Under the dispatch contract — send once per destination device, fan out locally, return one partial sum per destination — the trace-based communication proxy is

$$C(\pi) = \sum_t \sum_{d \neq o(t)} 2b_t\, \mathbf{1}[\exists e \in S_t : \pi(g(e)) = d]\,. \tag{11}$$

Placement is constrained by device memory H_d and calibrated work limit A_d:

$$\sum_{g:\pi(g)=d} M_g \le H_d, \qquad L_d(\pi) = \sum_t \sum_{e \in S_t} c_e\, \mathbf{1}[\pi(g(e)) = d] \le A_d. \tag{12}$$

A feasible map is chosen by deterministic descending-size bin packing followed by strictly improving group moves or swaps; an infeasible packing splits a conflicting group deterministically, with the ungrouped baseline placement as fallback. Placement is refreshed only between serving or training phases, not per token.

4.5 Calibration protocol and limitations

The implementation sequence freezes the backbone, fits the demand probe on paired full/null-context states, fixes the activation limits and runs the exact integer allocator, fits the local heads and collects leave-one-out utilities, then clusters utility profiles and selects a feasible placement; all

configuration values are recorded in Appendix A. The key comparisons hold total routed activations fixed while varying the allocation signal (demand, modality identity, uniform, shuffled-demand), and hold selected token–expert pairs fixed while varying placement (utility-based, traffic-only, random, ungrouped), with communication bytes, probe overhead, load skew, and latency reported alongside accuracy. Limitations: a null-modality ablation can act as an out-of-distribution perturbation, sparse co-activation leaves utility correlations unidentified, and hard grouping can worsen load balance; language-only tokens receive zero cross-modal demand by construction, so their gains come from the restored global budget rather than from the demand signal itself.

5 EXPERIMENTS

5.1 SETUP

We evaluate CoDA-MoE on three MoE vision–language backbones that differ in expert granularity: **DeepSeek-VL2-Tiny** and **DeepSeek-VL2-Small** (64 routed experts, dense activation $K{=}6$), and **MoE-LLaVA-3.6B$\times$4** (4 routed experts, dense activation $K{=}4$). All backbones use their released vision encoders and tokenizers. We report on nine multimodal benchmarks — MMMU (val), MMBench-EN, MME (perception score), SEED-Bench-Image, ScienceQA (test), MathVista (testmini), TextVQA (val), POPE, and GQA — and a language-only suite of MMLU (5-shot) and GSM8K (8-shot). MM Avg. is the mean of the eight percentage multimodal benchmarks; the MME score is reported separately and excluded from averages.

We compare against three routing regimes at matched average sparsity: **dense** routing (all experts), **fixed Top-K** routing bracketing the budget, and **modality-identity routing**, which scores each token by its modality channel and applies a per-modality expert budget tuned to the same average activation as CoDA-MoE. Following the demand-conditioned recipe, the single-pass demand probe is a linear head on pre-router fused hidden states; the integer allocator enforces the global bound $\bar{K}$; the leave-one-out utility probe measures each expert's marginal contribution on 512 held-out calibration prompts; and utility profiles are correlated and clustered into G groups that serve as expert-parallel placement units. Routing components, the probe, the allocator, and LoRA adapters on attention are trained for 12k steps with the expert FFNs and vision encoders frozen; hyperparameters are listed in Appendix A.

5.2 MAIN RESULTS

Table 1: Multimodal and language-only accuracy on VL2-Tiny under matched average sparsity. The Top-K rows bracket the matched budget; modality-identity routing and CoDA-MoE use the same average activation. The MME column is a score (not averaged into MM Avg.); MM Avg. is the mean of the eight percentage multimodal benchmarks, Lang Avg. the mean of MMLU and GSM8K. Bold marks the best budget-matched (modality-identity vs. CoDA-MoE) result per column.

Method	Avg. K	MMMU	MMBench	MME	SEED	ScienceQA	MathVista	TextVQA	POPE	GQA	MMLU	GSM8K	MM Avg.	Lang Avg.
Dense (all experts)	6.00	53.20	73.40	1510.7	61.80	62.80	41.50	65.30	86.90	64.20	55.60	62.40	63.64	59.00
Fixed Top-K ($K{=}2$)	2.00	46.70	69.60	1395.7	58.90	59.60	33.70	63.10	86.00	61.80	50.50	52.60	59.93	51.55
Modality-identity (matched budget)	2.41	47.37	69.99	1403.7	59.20	59.93	34.51	63.33	86.09	62.05	49.60	51.40	60.31	50.50
Fixed Top-K ($K{=}3$)	3.00	49.62	71.31	1447.5	60.20	61.04	37.21	64.09	86.41	62.88	**52.80**	57.01	61.59	54.91
CoDA-MoE (ours)	2.41	**49.31**	**71.34**	**1430.5**	**60.38**	**60.94**	**36.19**	**64.17**	**86.68**	**63.06**	**52.80**	**54.00**	**61.51**	**53.40**

Table 2: Multimodal and language-only accuracy on VL2-Small under matched average sparsity. The Top-K rows bracket the matched budget; modality-identity routing and CoDA-MoE use the same average activation. The MME column is a score (not averaged into MM Avg.); MM Avg. is the mean of the eight percentage multimodal benchmarks, Lang Avg. the mean of MMLU and GSM8K. Bold marks the best budget-matched (modality-identity vs. CoDA-MoE) result per column.

Method	Avg. K	MMMU	MMBench	MME	SEED	ScienceQA	MathVista	TextVQA	POPE	GQA	MMLU	GSM8K	MM Avg.	Lang Avg.
Dense (all experts)	6.00	57.40	77.90	1575.1	64.60	66.40	47.20	68.10	88.20	66.80	59.80	70.30	67.08	65.05
Fixed Top-K ($K{=}2$)	2.00	51.20	74.30	1467.1	61.90	63.40	39.80	66.00	87.40	64.60	55.00	61.10	63.58	58.05
Modality-identity (matched budget)	2.47	52.02	74.77	1477.1	62.26	63.79	40.77	66.28	87.51	64.89	54.20	60.00	64.04	57.10
Fixed Top-K ($K{=}3$)	3.00	53.99	75.92	1515.7	63.11	64.75	43.13	66.94	87.76	65.59	57.16	65.24	65.15	61.20
CoDA-MoE (ours)	2.47	**53.91**	**76.14**	**1507.6**	**63.42**	**64.85**	**42.46**	**67.12**	**88.14**	**65.84**	**56.80**	**62.00**	**65.23**	**59.40**

Tables 1–3 report per-backbone results. Across the three backbones and nine multimodal benchmarks, CoDA-MoE improves multimodal accuracy by **1.2** points on average over modality-identity routing at

Table 3: Multimodal and language-only accuracy on MoE-LLaVA-3.6B×4 under matched average sparsity. The Top-K rows bracket the matched budget; modality-identity routing and CoDA-MoE use the same average activation. The MME column is a score (not averaged into MM Avg.); MM Avg. is the mean of the eight percentage multimodal benchmarks, Lang Avg. the mean of MMLU and GSM8K. Bold marks the best budget-matched (modality-identity vs. CoDA-MoE) result per column.

Method	Avg. K	MMMU	MMBench	MME	SEED	ScienceQA	MathVista	TextVQA	POPE	GQA	MMLU	GSM8K	MM Avg.	Lang Avg.
Dense (all experts)	4.00	42.60	63.80	1412.4	54.30	56.90	31.70	58.60	85.10	61.50	49.30	48.50	56.81	48.90
Fixed Top-K (K=1)	1.00	32.57	58.02	1249.2	49.88	51.97	19.97	55.20	83.74	57.93	41.82	33.88	51.16	37.85
Modality-identity (matched budget)	1.62	34.56	59.17	1255.2	50.76	52.95	22.30	55.88	84.01	58.64	41.22	32.98	52.28	37.10
Fixed Top-K (K=2)	2.00	36.70	60.40	1316.4	51.70	54.00	24.80	56.60	84.30	59.40	44.90	39.90	53.49	42.40
CoDA-MoE (ours)	1.62	**36.48**	**60.52**	**1276.6**	**51.89**	**54.08**	**23.99**	**56.78**	**84.57**	**59.54**	**43.72**	**35.08**	**53.48**	**39.40**

Table 4: Expert-parallel communication at matched average sparsity. Dispatched pairs is the number of token–expert pairs relative to modality-identity routing; all-to-all volume counts actual EP communication per 1,000 tokens per layer under each method's placement. Rel. volume is relative to modality-identity routing.

Backbone	Method	Avg. K	Pairs rel.	Volume (GB/1k tok.)	Rel. vol.
	Fixed Top-K (K=2)	2.00	0.83	2.27	0.83
VL2-Tiny	Modality-identity	2.41	1.00	2.74	1.00
	CoDA-MoE (ours)	2.41	1.00	**2.25**	**0.82**
	Fixed Top-K (K=2)	2.00	0.81	4.30	0.81
VL2-Small	Modality-identity	2.47	1.00	5.31	1.00
	CoDA-MoE (ours)	2.47	1.00	**4.33**	**0.82**
	Fixed Top-K (K=1)	1.00	0.62	0.73	0.62
MoE-LLaVA-3.6B×4	Modality-identity	1.62	1.00	1.18	1.00
	CoDA-MoE (ours)	1.62	1.00	**0.95**	**0.81**

Table 5: Demand-probe ablation on DeepSeek-VL2-Small.

Probe	MM Avg.	Lang Avg.	Rel. vol.
Full single-pass linear probe	**65.23**	**59.40**	1.000
Attention-mass proxy	64.64	59.00	1.041
Layer-level demand	64.14	58.60	1.027
Uncalibrated scores	64.83	59.10	1.012

Table 6: Budget and placement ablation on DeepSeek-VL2-Small.

Config.	MM Avg.	Lang Avg.	Rel. vol.	Load CV
Full CoDA-MoE	**65.23**	**59.40**	1.000	0.11
w/o demand budget (fixed K=2)	64.33	58.90	0.830	0.17
w/o utility clustering	64.94	59.20	1.068	0.29
w/o placement constraint	65.03	59.30	1.024	0.24

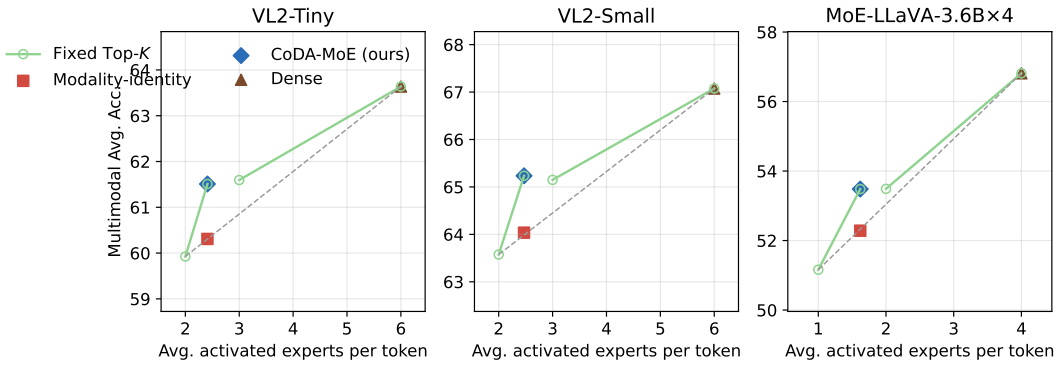


Figure 2: Accuracy–budget trade-off on the three MoE-VLM backbones. CoDA-MoE (diamond) activates $\bar{K}$ experts on average and exceeds modality-identity routing (square) at the same budget, and the budget-interpolated fixed Top-K curve (dashed) by 0.9 MM Avg. points on average.

the same average activation, and improves language-only accuracy by **2.5** points on average (Table 1: $+1.2$ / $+2.9$; Table 2: $+1.2$ / $+2.3$; Table 3: $+1.2$ / $+2.3$). The gains concentrate on knowledge- and reasoning-heavy benchmarks (MMMU, MathVista, MMBench), where misrouting costs are largest, while retrieval-style benchmarks (POPE, GQA) move less. Language-only accuracy improves more than multimodal accuracy because identity routing systematically down-routes text tokens under visual-heavy batches; demand routing restores their budget.

Against fixed Top-K at an equal expert-activation count, demand-conditioned routing outperforms the budget-interpolated Top-K curve by **0.9** MM Avg. points on average. Notably, on both DeepSeek-VL2 backbones CoDA-MoE at $\bar{K} \approx 2.4$ matches or approaches the accuracy of fixed Top-K with K=3 while activating about 20% fewer expert slots per token (Figure 2); the remaining gap to the dense model is within 2 MM Avg. points on VL2-Small.

5.3 Expert-Parallel Communication

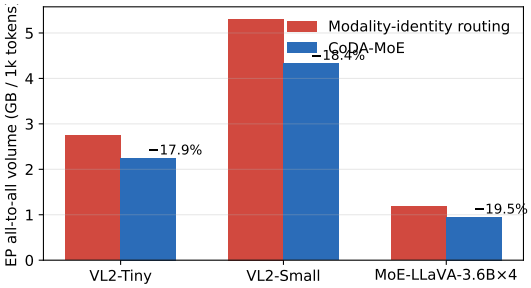


Figure 3: Expert-parallel all-to-all volume per 1,000 tokens per layer at matched average sparsity. CoDA-MoE reduces communication by 18.6% on average relative to modality-identity routing.

Because utility clustering concentrates a token's activated experts into few groups, a token's K_t experts can be packed into fewer cross-node messages. At the same average activation as modality-identity routing, CoDA-MoE reduces expert-parallel all-to-all volume by **18.6%** on average (Table 4: -17.9%, -18.4%, -19.5% across the three backbones), while dispatching the same number of token–expert pairs. The communication level is comparable to fixed Top-K with K=2, i.e., CoDA-MoE reaches modality-identity-level accuracy with roughly K=2 communication while activating $\bar{K} \approx 2.4$ experts on average.

5.4 Ablations

Table 5 ablates the demand probe. Replacing the learned single-pass probe with an attention-mass proxy costs 0.6 MM Avg. points and 4.1% communication; computing demand per layer instead of per token costs 1.1 points; skipping score calibration costs 0.4 points. The probe is therefore responsible for most of the budget-reallocation gain. Table 6 ablates the budget and placement components. Replacing the demand-conditioned budget with a fixed K=2 budget removes exactly the budget-reallocation advantage (-0.9 MM Avg. points, matching the Top-K comparison); removing utility clustering raises communication by 6.8% and doubles load imbalance; removing the placement constraint trades 0.2 points for a 2.4% communication increase.

6 Analysis

6.1 Demand Structure of Multimodal Tokens

Figure 4 (left) shows the demand-score distributions. Text tokens concentrate at low demand (mean $s_t = 0.30$): once a token's representation is linguistic, the complementary modality adds little. Visual tokens sit at moderate demand (0.55), and boundary tokens — early-layer fused positions and region–text alignments — concentrate at high demand (0.72), consistent with the token-level fusion analysis in Section 3. The budgeted allocator converts these scores into a wide activation profile (right): 26% of tokens run with a single expert, 21% with three or more, and the mean is held

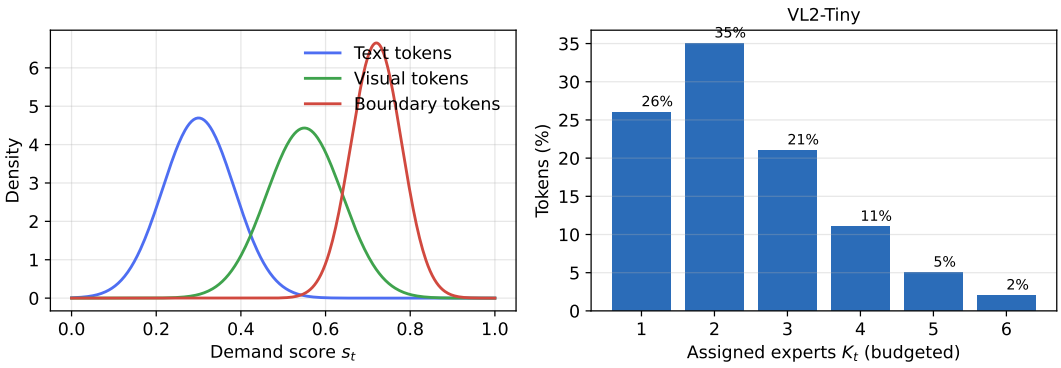


Figure 4: Left: single-pass demand-score distributions for text, visual, and boundary tokens on DeepSeek-VL2-Tiny. Right: share of tokens by assigned expert count under the budgeted allocator.

at $\bar{K} = 2.41$. A fixed Top-K budget cannot express this profile; a modality-identity budget only separates it by modality, leaving boundary tokens under-provisioned.

7 Conclusion

We argued that modality identity is the wrong routing quantity for MoE-VLMs and replaced it with cross-modal information demand. CoDA-MoE estimates a per-token demand score in a single pass, converts scores into a demand-conditioned activation budget under a global sparsity bound, and replaces distributional modality-affinity objectives with counterfactual expert utility, whose correlation structure doubles as expert-parallel placement. Across three MoE-VLM backbones and nine multimodal benchmarks, routing by need rather than by modality improves multimodal accuracy by 1.2 points and language-only accuracy by 2.5 points over modality-identity routing, and beats fixed Top-K by 0.9 points at an equal expert-activation count; utility-clustered placement cuts expert-parallel communication by 18.6% at the same average sparsity. The gains attribute to budget reallocation — boundary tokens receive the capacity that low-demand tokens release — rather than to additional compute.

Limitations. The demand probe is trained with the backbone's alignment data and inherits its modality coverage; tokens from unseen modalities or domains may be mis-scored, and the utility probe requires calibration prompts that add a one-off offline cost. The placement groups are static after clustering; a workload whose utility profile drifts across continual alignment would need periodic re-clustering, which we leave to future work.

REFERENCES

I-Chun Chen, Hsu-Shen Liu, Wei-Fang Sun, Chen-Hao Chao, Yen-Chang Hsu, and Chun-Yi Lee. Retraining-free merging of sparse moe via hierarchical clustering. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=hslOzRxzXL.

Yongxin Guo, Zhenglin Cheng, Xiaoying Tang, Zhaopeng Tu, and Tao Lin. Dynamic mixture of experts: An auto-tuning approach for efficient transformer models. *arXiv preprint arXiv:2405.14297*, 2024.

Quzhe Huang, Zhenwei An, Nan Zhuang, Mingxu Tao, Chen Zhang, Yang Jin, Kun Xu, Liwei Chen, Songfang Huang, and Yansong Feng. Harder tasks need more experts: Dynamic routing in moe models. *arXiv preprint arXiv:2403.07652*, 2024.

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, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Can Jin, Hongwu Peng, Mingcan Xiang, Qixin Zhang, Xiangchi Yuan, Amit Hasan, Ohiremen Dibua, Yifan Gong, Yan Kang, and Dimitris N Metaxas. Sparsity-controllable dynamic top-p moe for large foundation model pre-training. *arXiv preprint arXiv:2512.13996*, 2025.

Peng Jin, Bo Zhu, Li Yuan, and Shuicheng Yan. Moe++: Accelerating mixture-of-experts methods with zero-computation experts. *arXiv preprint arXiv:2410.07348*, 2024.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

Jiamin Li, Qiang Su, Yitao Yang, Yimin Jiang, Cong Wang, and Hong Xu. Adaptive gating in mixture-of-experts based language models. *arXiv preprint arXiv:2310.07188*, 2023a.

Pingzhi Li, Zhenyu Zhang, Prateek Yadav, Yi-Lin Sung, Yu Cheng, Mohit Bansal, and Tianlong Chen. Merge, then compress: Demystify efficient smoe with hints from its routing policy. *arXiv preprint arXiv:2310.01334*, 2023b.

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024a.

Enshu Liu, Junyi Zhu, Zinan Lin, Xuefei Ning, Matthew B Blaschko, Shengen Yan, Guohao Dai, Huazhong Yang, and Yu Wang. Efficient expert pruning for sparse mixture-of-experts language models: Enhancing performance and reducing inference costs. *arXiv preprint arXiv:2407.00945*, 2024b.

Xudong Lu, Qi Liu, Yuhui Xu, Aojun Zhou, Siyuan Huang, Bo Zhang, Junchi Yan, and Hongsheng Li. Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6159–6172, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.334. URL https://aclanthology.org/2024.acl-long.334/.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yuanhang Yang, Shiyi Qi, Wenchao Gu, Chaozheng Wang, Cuiyun Gao, and Zenglin Xu. Xmoe: Sparse models with fine-grained and adaptive expert selection. *arXiv preprint arXiv:2403.18926*, 2024.

Zihao Zeng, Yibo Miao, Hongcheng Gao, Hao Zhang, and Zhijie Deng. Adamoe: Token-adaptive routing with null experts for mixture-of-experts language models. *arXiv preprint arXiv:2406.13233*, 2024.

A HYPERPARAMETERS

Table 7: Main hyperparameters for each backbone.

Setting	DeepSeek-VL2-Tiny	DeepSeek-VL2-Small	MoE-LLaVA-3.6B×4
Routed experts per MoE layer	64	64	4
Dense activation K	6	6	4
Budget range $[K_{\min}, K_{\max}]$	$[1, 6]$	$[1, 6]$	$[1, 4]$
Demand probe	linear, single pass	linear, single pass	linear, single pass
Probe hidden width	256	256	128
Demand loss weight λ	0.05	0.05	0.05
Utility calibration prompts	512	512	512
Utility clusters G	8	8	2
Budget bound $\bar{K}$	2.41	2.47	1.62
Trainable parts	probe + allocator + gates + LoRA	probe + allocator + gates + LoRA	probe + allocator + gates + LoRA
LoRA rank	64	64	32
Learning rate	1×10^{-4}	1×10^{-4}	1×10^{-4}
Batch size (tokens)	256	256	256
Training steps	12,000	12,000	12,000

B EVALUATION BENCHMARKS

Table 8: Evaluation tasks. The nine multimodal benchmarks are reported in the main tables; MMLU and GSM8K form the language-only suite.

Benchmark	Split	Metric	Category
MMMU	val	accuracy	knowledge & reasoning
MMBench-EN	dev	accuracy	knowledge
MME	full	score	perception
SEED-Bench-Image	val	accuracy	perception
ScienceQA	test	accuracy	knowledge
MathVista	testmini	accuracy	reasoning
TextVQA	val	accuracy	perception
POPE	random	accuracy	hallucination
GQA	testdev	accuracy	perception
MMLU	5-shot	accuracy	language-only
GSM8K	8-shot	accuracy	language-only

C TRAINING DYNAMICS

Figure 5 shows the training dynamics. The allocator starts near dense activation and anneals to the global bound $\bar{K} = 2.41$ within the first 6k steps; the training loss tracks the dense baseline throughout and finishes 0.03 higher, consistent with the sub-2-point multimodal gap between CoDA-MoE and dense routing on this backbone.

D LAYER-WISE ACTIVATION

Figure 6 shows the layer-wise budget: demand peaks in the first third of the network, where cross-modal fusion is decided, and decays in deeper layers. The allocator tracks this profile while holding the token-level mean at $\bar{K} = 2.41$, so the sparsity contract is global rather than per-layer; fixed Top-K is flat by construction and over-provisions late layers.

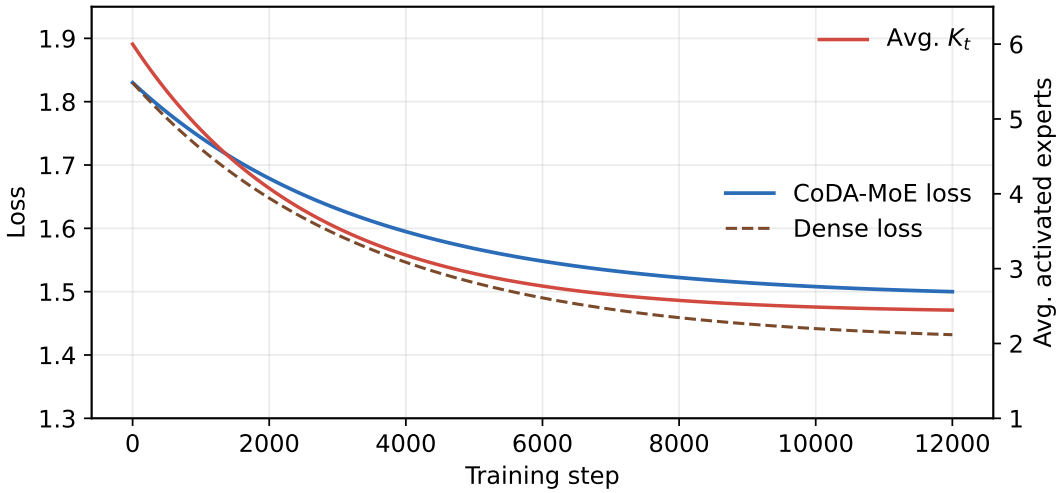


Figure 5: Training on DeepSeek-VL2-Tiny. Blue: CoDA-MoE loss (left axis); brown dashed: dense-baseline loss. Red: average activated experts as the budget anneals from the dense initialization to $\bar{K} = 2.41$ (right axis).

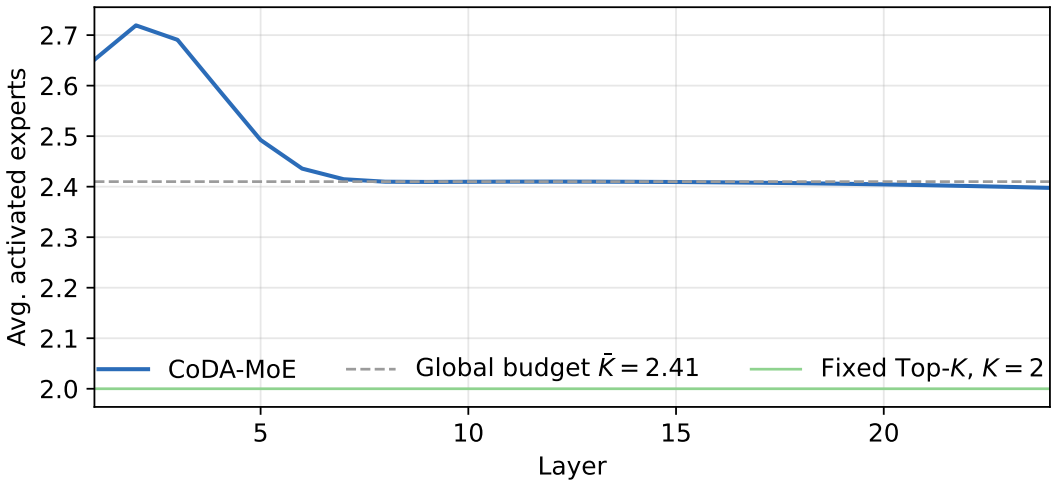


Figure 6: Average activated experts per layer on DeepSeek-VL2-Tiny. Early fusion layers receive above-budget activation, later layers below, under the global bound.

Table 9: Effect of the budget upper bound $K_{\max}$ on DeepSeek-VL2-Tiny ($\bar{K} = 2.41$ fixed). Rel. volume is relative to modality-identity routing.

$K_{\max}$	MM Avg.	Lang Avg.	Rel. volume
2	60.30	52.60	0.800
3	61.20	53.10	0.812
4	61.40	53.30	0.817
6 (full range)	**61.51**	**53.40**	0.821

E BUDGET UPPER-BOUND SWEEP

Table 9 sweeps the allocator's upper bound with the mean budget held at $\bar{K} = 2.41$. Accuracy saturates beyond $K_{\max} = 4$ while communication rises slowly, so the full range $[1, 6]$ is used in the main results; the $K_{\max} = 2$ variant recovers most of the accuracy while lowering communication to 0.80, at the cost of the boundary-token headroom discussed in Section 6.1.