CIPHER-MOE: BALANCING EFFICIENCY AND ROUTING FIDELITY IN TRILLION-SCALE MOE TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

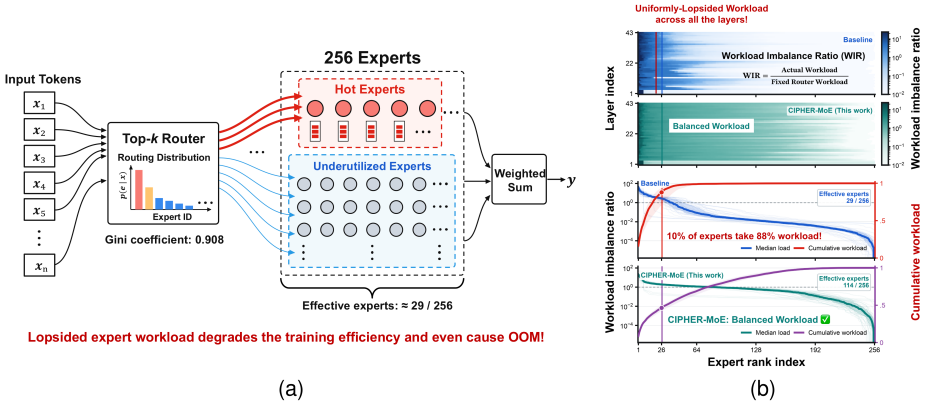
Mixture-of-Experts (MoE) has been widely adopted in recent large language model (LLM) architectures. However, scaling up MoE in LLM training introduces system-level challenges on training, where non-uniform token routing can lead to highly imbalanced workloads across experts and devices, further destabilizing the training process. With trillion-scale LLMs, imbalanced expert workloads further amplify the resource cost of MoE training, resulting in degraded training efficiency and hardware utilization for underloaded experts, while hot experts require additional resources to accommodate excessive workloads. Recent studies address imbalanced MoE training through intricate parallelism strategies or resource reallocation. However, these system-level approaches often introduce additional resource requirements and considerable orchestration complexity, which become increasingly difficult to afford when training trillion-parameter LLMs under constrained computational resources. This work introduces CIPHER-MoE, which mitigates MoE workload imbalance while keeping the router's token-side Top-K selection unchanged. CIPHER-MoE applies affinity-aware Expert-to-Token filtering with explicit capacity control to reduce hotspot expert workloads without additional hardware resources or complex runtime design. The proposed method has been evaluated on large-scale MoE models, including DeepSeek-V4-Pro, showing up to 64.9 percentage points Top-1 expert workload reduction and 1.10×–1.94× training acceleration, while preserving the training quality. The source code will be released soon.

1 INTRODUCTION

Large language models (LLMs) have become foundational components of modern AI systems (Bommasani et al., 2021). The continued effectiveness of scaling laws has demonstrated that increasing model capacity can substantially improve performance (Kaplan et al., 2020). However, naively scaling dense Transformers toward the trillion-parameter regime incurs prohibitively high computational and memory costs, motivating the adoption of Mixture-of-Experts (MoE) architectures (Fedus et al., 2022). In particular, MoE decouples model capacity from per-token computation by replacing the dense feed-forward network (FFN) with multiple sparsely activated experts and dynamically routing each token to only a small subset of them, leading to a balanced tradeoff between scalability and computational efficiency (Shazeer et al., 2017).

Recent studies have demonstrated the strong potential of domain-specific supervised fine-tuning (SFT) and reinforcement learning (RL) for advancing LLM post-training (Guo et al., 2025). Training MoE models introduces substantial system-level challenges due to workload imbalance, which degrades communication efficiency, hardware utilization, and training stability (Jin et al., 2026b). The inherently non-uniform data distribution across domains and tasks can induce highly skewed token routing, resulting in uneven workloads across different experts. For the expensive trillion-parameter LLMs, cross-rank expert parallelism is almost mandatory for efficient training. However, the lopsided computation impedes efficient distributed training with unbalanced communication and computation (Rajbhandari et al., 2022). Moreover, the overloaded experts may incur excessive memory consumption and instantly trigger out-of-memory failures on large-scale computing clusters, ultimately undermining training efficiency and stability (Nguyen et al., 2026).

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.



(a) (b)

Figure 1: Expert workload imbalance in DeepSeek-V4-Flash. (a) Illustration of the MoE architecture with data-dependent routing, which leads to the imbalanced workload. (b) Lopsided workload statistics across all layers of the DeepSeek-V4-Flash model. With the domain-specific dataset, 10% of the experts consume over 88% routing workload, leading to degraded training efficiency.

Recent studies on efficient MoE training are two-fold; system-level approaches redistribute workloads through dynamic expert migration or replication (Nie et al., 2023). Meanwhile, MoE-specific kernels optimize expert-parallel execution through aggressive communication-computation pipeline design (Zhang et al., 2025). At the algorithm level, recent studies investigate global-batch balancing, routing regularization, or transfer the pre-trained routing strategy to fresh training (Lewis et al., 2021). However, system-level approaches largely accommodate routing imbalance after it arises, requiring additional expert replicas, collective communication, or specialized runtime scheduling, whose overhead and implementation complexity become increasingly pronounced with trillion-parameter LLMs training. On the other hand, regularization-based routing penalty failed to handle the instantaneous memory peak (Fedus et al., 2022), while the heuristic batch balancing or experience-based routing distillation exhibits poor scalability to the trillion-parameter MoE models. Given the research gaps in existing algorithm and system-level approaches for MoE LLM training, this work aims to investigate the following problem:

How to achieve efficient and stable training for trillion-parameter MoE LLMs without intricate system design or heuristic-based routing?

In this work, we propose `CIPHER-MoE`, a *plug-and-play* algorithm for stable-and-efficient trillion-parameter MoE model training. Although the fixed routing strategy ensures the perfect load balance with stabilized training process, the reduced expert specialization leads to the suboptimal intelligence in the resulting trained model (Dikkala et al., 2023). On the other hand, freely-exploited expert routing amplifies the workload imbalance (Guo et al., 2026), especially with domain-specific post-training phase. In contrast, the proposed `CIPHER-MoE` tackles the problem by investigating a bounded learning capacity of different experts and proves the feasibility of dropping tokens from expert training. Different from heuristic token dropping, this work first reveals the relationship between optimal token budget and training efficiency for trillion-parameter MoE training. We validate the proposed method via both theoretical analysis and empirical experiments across multiple large-scale MoE models (from 284B to 1.6T). The major contributions of this paper are:

- **Simplicity**. CIPHER-MoE directly exploits the bounded token budget for MoE training, achieving high efficiency while preserving the LLM training quality.
- **Theoretical Foundation**. CIPHER-MoE derives a sufficient expert capacity for similarity-based admission and a necessary capacity for natural-order admission.
- **Scalability**. CIPHER-MoE scales effectively across different model sizes, architectures, and datasets. Across DeepSeek-V3, DeepSeek-V4-Flash, GLM-5, and DeepSeek-V4-Pro, achieving $\mathbf{1.10}\times$–$\mathbf{1.94}\times$ measured training-time speedups.

2 Related Work

System-level workload balancing and efficient MoE execution. Workload imbalance is a major bottleneck for efficient distributed MoE training (Hwang et al., 2023), where skewed token routing creates expert stragglers, uneven communication, and poor accelerator utilization. System-level work mitigates this imbalance by adapting the mapping between experts and hardware. FasterMoE and FlexMoE replicate, resize, or migrate experts according to runtime load (He et al., 2022; Nie et al., 2023), while some works combine balancing with parameter offloading, decoupled parameter/optimizer-state placement, or sharded expert re-layout (Kim et al., 2024; Skiadopoulos et al., 2025; Liu et al., 2026). In parallel, another line of work reduces the overhead of sparse and irregular MoE computation. Tutel and PipeMoE (Hwang et al., 2023; Shi et al., 2023) optimize expert-parallel execution through adaptive parallelism and communication–computation pipelining; MegaBlocks (Gale et al., 2023) uses block-sparse kernels to process variable expert workloads without padding or token dropping; MegaScale-MoE (Jin et al., 2026a) jointly optimizes parallelism, communication–computation overlap, and communication compression. While these techniques improve MoE execution efficiency, they operate after routing and rely on expert replication, parameter migration or re-layout, runtime profiling, and specialized communication scheduling, increasing system complexity at scale.

Algorithmic load balancing and expert specialization. Routing offers an alternative approach to load balancing. GShard (Lepikhin et al., 2020) and Switch Transformer (Fedus et al., 2022) use auxiliary load-balancing objectives to encourage uniform expert utilization. However, overly strong local balancing can interfere with the language-modeling objective and suppress expert specialization. Loss-Free Balancing (Wang et al., 2024a), adopted by DeepSeek-V3 (Liu et al., 2024), adjusts per-expert routing biases based on historical loads without introducing auxiliary-loss gradients. Global-Batch Load Balancing (Qiu et al., 2025) aggregates global statistics to preserve domain specialization. Related work further highlights the tension between strict balancing and expert differentiation (Dai et al., 2024), while ϕ-balancing (Chen et al., 2026) formulates expert utilization as a population-level objective rather than relying on noisy mini-batch statistics. These methods improve average utilization through auxiliary objectives, historical statistics, or batch-level assignments rather than bounding instantaneous loads, although peak memory and latency depend on the most heavily loaded expert.

Capacity-aware token–expert assignment. Several works are closely related to our capacity-aware perspective. LocMoE (Li et al., 2024) jointly considers workload balance and communication locality, deriving a minimum expert capacity from the angular relationship between token representations and gating weights. This result motivates capacity-aware routing by showing that expert capacity need not be overprovisioned to preserve model quality. Expert Choice Routing (Zhou et al., 2022) instead fixes per-expert capacity by reversing token-to-expert assignment. These approaches show that expert-side token selection can control workloads, but either impose fixed expert-centric assignments or couple capacity optimization with specific locality and routing assumptions. In contrast, `CIPHER-MoE` analyzes expert capacity for large-scale training and establishes a sufficient capacity bound that enables workload-aware *bidirectional* token–expert assignment. Tokens first identify preferred experts, which then select the highest-affinity tokens within the derived budget, while excess assignments are rerouted. This design preserves token–expert affinity and controls hotspot workloads without expert replication, parameter migration, or complex runtime orchestration. We evaluate CIPHER-MoE on MoE models ranging from 284B to 1.6T parameters.

3 Proposed Method: CIPHER-MoE

3.1 Preliminaries of Mixture-of-Expert training

Sparse expert computation and training. An MoE layer replaces the dense FFN with E expert networks $\{f_e(\cdot;\theta_e)\}_{e=1}^{E}$ and a router that activates only $K \ll E$ experts for each token (Fedus et al., 2022). Consider N token representations $\{\boldsymbol{x}_i\}_{i=1}^{N}$, with $\boldsymbol{x}_i \in \mathbb{R}^d$, processed by one expert-parallel group in a training micro-batch (the layer index is omitted for clarity). Let $\boldsymbol{W}_r \in \mathbb{R}^{E\times d}$ denote the router parameters and $\boldsymbol{w}_e^\top$ its e-th row. A conventional softmax router computes the token–expert

affinity $p_{i,e}$ and selects the corresponding Top-K expert set $\mathcal{S}_i$:

$$s_{i,e} = \boldsymbol{w}_e^\top \boldsymbol{x}_i, \qquad p_{i,e} = \frac{\exp(s_{i,e})}{\sum_{j=1}^{E} \exp(s_{i,j})}, \qquad \mathcal{S}_i = \mathrm{TopK}_{e\in\{1,\dots,E\}}(p_{i,e}). \tag{1}$$

Here, $s_{i,e}$ is the routing logit, and $\mathcal{S}_i$ contains the indices of the K selected experts. Define the assignment indicator $a_{i,e} = \mathbf{1}[e \in \mathcal{S}_i]$ and the sparse gate $g_{i,e} = a_{i,e}p_{i,e}$. The routed MoE output is

$$\boldsymbol{y}_i = \sum_{e=1}^{E} g_{i,e} f_e(\boldsymbol{x}_i; \theta_e), \qquad \sum_{e=1}^{E} a_{i,e} = K. \tag{2}$$

Training optimizes the task loss $\mathcal{L}_{\text{task}}$ over the expert parameters and router parameters. Each token contributes task gradients through its selected expert paths. Consequently, increasing E expands the available model capacity while the expert computation per token remains governed by K.

Models	Method	Latency / Iteration (sec)	NL4Opt	OptiBench	Bench4Opt Feasi.	Bench4Opt OR
DeepSeek-V4-Flash	Fixed Router	**28.30** (**1.00**×)	88.93	66.17	70.93	48.73
DeepSeek-V4-Flash	Vanilla MoE Training	38.69 (1.37×)	**93.08**	**68.00**	**71.51**	**51.02**
DeepSeek-V4-Pro	Fixed Router	**24.68** (**1.00**×)	89.27	64.67	74.42	54.31
DeepSeek-V4-Pro	Vanilla MoE Training	OOM	-	-	-	-

Table 1: Post-training SFT of the DeepSeek-V4-Flash and DeepSeek-V4-Pro model on linear programming and operations research (OR) dataset. OR capabilities evaluated on NL4Opt (Ramamonjison et al., 2023), OptiBench (Yang et al., 2025), Bench4Opt-Feasi. (Wang et al., 2025), and Bench4Opt-OR (Wang et al., 2025). Fixed router exhibits suboptimal performance compared to the vanilla MoE training due to the homogenized expert training. Detailed training setup and resource configurations are summarized in the Appendix A.5.

Fixed Router and homogenized experts. With expert parallelism (EP), experts are distributed across R devices, with $\mathcal{E}_r$ denoting the set hosted on device r. Given the unpredictable routing for each incoming token, *fixed routing* is introduced to enforce the deterministic workload balancing between different experts. For a local routing tensor with M tokens, the j-th expert assigned to token position t is

$$e_{t,j}^{\text{fix}} = 1 + \big((tK + j) \bmod E\big), \qquad t = 0, \dots, M-1, \quad j = 0, \dots, K-1, \tag{3}$$

For $K \le E$, each token is dispatched to K **distinct** experts, and each expert receives either $\lfloor MK/E \rfloor$ or $\lceil MK/E \rceil$ assignments from this local tensor.

The predictable routing strategy ensures the stable and balanced training workload for MoE model at the cost of removing affinity-based expert selection. Semantically unrelated tokens occupying the same positions across micro-batches are repeatedly dispatched to the same expert combinations, potentially weakening expert differentiation, leading to expert homogenization. As a result, the trained model exhibits suboptimal domain specialization with homogenized experts (Qiu et al., 2025). We compare the routing strategy with large-scale MoE model training with the results in Table 1. The standard training paradigm with freely-explored expert workload reaches the ideal performance, with the cost of the degraded training efficiency. With the scaled computing resources, DeepSeek-V4-Pro (1.6T parameters) even failed before the very first iteration of training with the error of out-of-memory (OOM), where the freely-explored expert routing explodes the imbalanced workload, resulting in the instant OOM error.

3.2 CIPHER-MoE: Bidirectional Expert Workload Balancing

Expert-token similarity. Given the token hidden state $\boldsymbol{x}_i$ and expert gating vector $\boldsymbol{w}_e$, their cosine similarity is defined as

$$c_{i,e} = \frac{\boldsymbol{w}_e^\top \boldsymbol{x}_i}{\|\boldsymbol{w}_e\|_2 \|\boldsymbol{x}_i\|_2}. \tag{4}$$

For each expert e, CIPHER-MoE uses $c_{i,e}$ to rank the candidate tokens in $\mathcal{P}_e$. The similarity threshold $\tau \in [-1, 1]$ excludes tokens with low similarity, while the expert capacity C_e limits the number of tokens processed in each micro-batch.

Specifically, let $\mathcal{P}_e^\tau = \{i \in \mathcal{P}_e : c_{i,e} \geq \tau\}$ contain the tokens that satisfy the similarity threshold. If more than C_e tokens satisfy the threshold, the expert retains the C_e tokens with the highest cosine similarity; otherwise, it retains all tokens in $\mathcal{P}_e^\tau$:

$$\mathcal{T}_e^{\mathrm{S}} = \mathrm{Top}_{\min\{C_e, |\mathcal{P}_e^\tau|\}}\big(\{c_{i,e} : i \in \mathcal{P}_e^\tau\}\big), \qquad b_{i,e}^{\mathrm{S}} = \mathbf{1}[i \in \mathcal{T}_e^{\mathrm{S}}], \tag{5}$$

where Top_q returns the indices of the q tokens with the highest similarity. The budget C_e can vary with the layer, expert, and training stage, while remaining fixed for each micro-batch. Furthermore, the rejection by one expert does not remove a token from its other selected experts, which will be discussed in Section 3.2.2.

Although similarity-based token selection directly controls expert workload, two critical questions require further investigation: ① Under what conditions can similarity-based selection reduce the required expert capacity while retaining informative tokens? ② How should the routing weights of rejected assignments be handled?

3.2.1 Theoretical analysis of similarity-based token selection

To address the first question, we examine why selecting tokens by similarity can require less expert capacity than processing them in their original order. Fix an expert e and its N_e candidate tokens. A token is called *discriminative* if it carries the expert-specific signal described by our assumptions. Let A_e denote the event that the candidate pool contains at least one such token. We compare the two selection policies on the same pool, with the same target:

$$\Pr(\text{at least one discriminative token is retained} \mid A_e) \geq 1 - \varepsilon, \qquad 0 < \varepsilon < 1. \tag{6}$$

Natural-order selection processes the first C candidates, whereas CIPHER-MoE ranks the candidates by similarity before allocating its C expert-execution slots.

Assumptions. Our analysis uses the assumptions of A1–A3 in Appendix A.4.1. First, candidates are independent draws from an expert's exposure distribution, with signal availability $\Pr(A_e) \geq a_0 > 0$. Under the spherical reference model, discriminative tokens have normalized affinity at least $\delta \in (0, 1)$, and expert exposure has probability at least $1/(\beta E)$, where $\beta \geq 1$. Second, conditional on A_e, a chosen discriminative token survives candidate filtering and thresholding with probability at least $1 - \eta$, where $0 \leq \eta < \varepsilon$. Third, after conditioning on its survival and previously examined candidates, the probabilities that competing tokens rank ahead of this token have sum at most μ. Thus, η measures the risk of excluding the signal before ranking, while μ controls the capacity needed to retain it after ranking. The signal margin δ is distinct from the threshold τ.

Theorem 1 (Similarity-based Capacity Reduction) *Under A1–A3, with $d \geq 2$, define*

$$\ell = \log \frac{1}{\varepsilon - \eta}, \qquad G(d, \delta) = -\frac{d-1}{2} \log(1 - \delta^2),$$
$$B_e = \min\left\{ N_e, \left\lceil 1 + \mu + \frac{\ell}{3} + \sqrt{2\mu\ell + \frac{\ell^2}{9}} \right\rceil \right\}. \tag{7}$$

Let C^*_{natural} and C^*_{CIPHER} be the minimum capacities required by the respective policies to satisfy Eq. 6. Then

$$C^*_{\mathrm{CIPHER}} \leq B_e, \qquad C^*_{\mathrm{natural}} \geq \frac{(1-\varepsilon)a_0}{\beta E} e^{G(d,\delta)}. \tag{8}$$

Consequently, a factor-κ reduction, $C^*_{\mathrm{CIPHER}} \leq C^*_{\mathrm{natural}}/\kappa$ for $\kappa > 1$, is guaranteed whenever the natural-order lower bound is at least κB_e.

Key proof steps. ① **Signal scarcity.** Spherical concentration bounds the discriminative-token probability by $\pi_e \leq \beta E e^{-G(d,\delta)}$. Processing the first C tokens retains a discriminative token with conditional probability at most $C\pi_e/a_0$. Reaching the target therefore requires $C \geq (1-\varepsilon)a_0/\pi_e$, giving the natural-order lower bound. ② **Similarity ranking.** Let S count candidates ranked ahead of the surviving discriminative token. Its rank is $S + 1$. A conditional concentration bound on S,

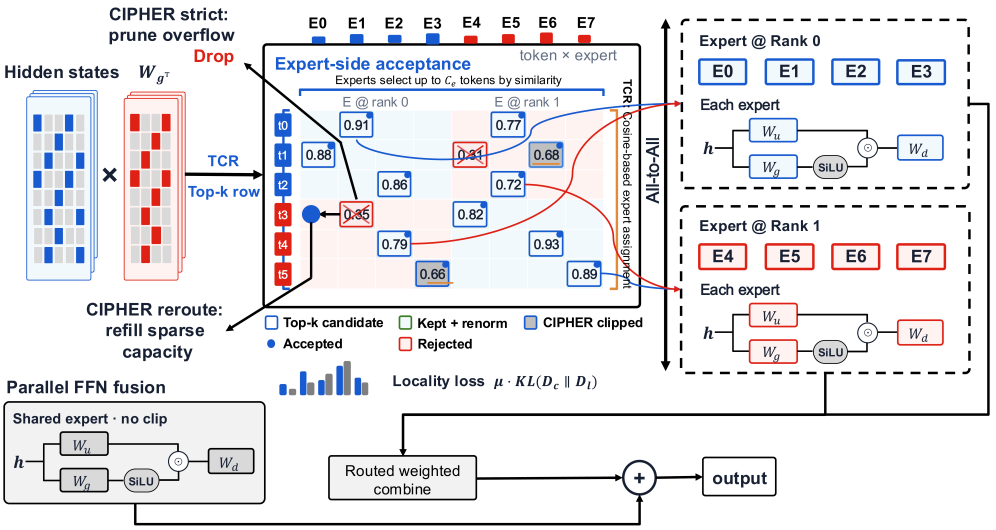


Figure 2: The overall design of the proposed CIPHER-MoE. The incoming tokens start with the normal Top-K expert selection, followed by the proposed expert-side token drop and reroute.

using A3, gives the sufficient capacity B_e with ranking-failure probability at most $\varepsilon - \eta$. Including filtering failures keeps the total failure probability at most ε. ③ **Capacity comparison.** Comparing CIPHER-MoE's sufficient capacity with natural-order selection's necessary capacity establishes the reduction. The full proof and explicit dimension condition are given in Appendix A.4.2.

The result explains the benefit of similarity-based selection: **An expert can retain an informative token without processing all candidates that precede it in the original order.** Under the stated assumptions, $C_e \geq B_e$ is sufficient for signal retention. This is a requirement on the available capacity, not a minimum number of tokens that must actually be processed; equation 5 still caps the workload at C_e. The theorem does not guarantee unchanged training loss or final model quality, and it does not recover signals excluded by an overly restrictive τ. Stability conditions and output-fidelity bounds are provided in Appendices A.4.3 and A.4.4.

3.2.2 Strict Token Drop vs. Token–Expert Reroute

Given the theoretical analysis of bounded expert capacity, handling rejected token–expert assignments becomes critical for both the efficiency and quality of LLM training. `Strict Token Drop` removes each rejected assignment and sets its routing weight to zero, while retaining the token's other accepted expert paths. `Token--Expert Reroute` instead redirects the rejected assignment to another candidate expert that satisfies the similarity threshold and has available capacity. Both strategies apply **only** to routed experts. Shared experts continue to process all incoming tokens, and rejecting an expert assignment does not remove the token from the training sequence or the task loss. Figure 2 illustrates these two mechanisms within CIPHER-MoE.

We evaluate the impact of **both** strategies on large-sized (MoE models >250B) with domain-specific training tasks. Surprisingly, rerouting tokens to other experts leads to sub-optimal performance compared to the strict token dropping, as shown in Table 2. Although rerouting preserves more expert computation and distributes overflow to experts with spare capacity and sufficient affinity, the additional computation does not necessarily improve training quality.

For domain-specific training, the token–expert rerouting shifts the token distributions on which those experts specialize. Even threshold-qualified tokens can induce updates that interfere with an expert's existing specialization, since cosine affinity alone does not constrain the direction of the task gradient. Strict token dropping avoids the additional assignments and gradient interference, which allows each expert to train on the highest-affinity candidates retained within its budget.

3.2.3 TRAINING OBJECTIVE OF CIPHER-MOE

On top of the proposed token selection during MoE training, we adopt the Locality loss proposed by LocMoE (Li et al., 2024), which minimizes the KL divergence between the current expert-assignment distribution and the ideal distribution target corresponding to all the locally-deployed experts (e.g., within the same EP rank), softly encouraging tokens to be routed to local experts while preserving load balance. Overall, the training objective of CIPHER-MoE is formulated as:

$$\mathcal{L}^{\rho}_{\text{CIPHER}}(\Theta) = -\frac{1}{|\mathcal{U}|}\sum_{t\in\mathcal{U}} \log p^{\rho}_{\Theta}\left(z_t \mid z_{<t}\right) + \lambda_{\text{loc}} \sum_{\ell\in\mathcal{M}} \mathcal{L}^{\rho,\ell}_{\text{loc}}(\Theta), \tag{9}$$

$$\text{where} \qquad \mathcal{L}^{\rho,\ell}_{\text{loc}} = -\int D^{\rho,\ell}_{c}(x) \log\left(\frac{D^{\rho,\ell}_{l}(x)}{D^{\rho,\ell}_{c}(x)}\right) \mathrm{d}x = \mathrm{KL}\left(D^{\rho,\ell}_{\mathrm{c}} \,\middle\|\, D^{\rho,\ell}_{l}\right). \tag{10}$$

where D_c is the current distribution of expert assignments for tokens in the batch, and D_l is the hypothetical fully localized assignment distribution. The loss pushes the actual routing distribution toward one where tokens are preferentially processed by experts that are local to their current communication group of expert parallelism (EP). $M^{\rho,\ell}$ can be determined by the aforementioned strict or reroute strategies, where $\rho \in \{\text{strict}, \text{reroute}\}$. Other auxiliary loss terms (e.g., for MTP head and sparse attention indexer) are omitted for better clarity.

4 EXPERIMENTAL RESULTS

We validate the proposed CIPHER-MoE algorithm on MoE models ranging from 284B to 1.6T parameters, including DeepSeek-V4-Pro (1.6T) (DeepSeek-AI, 2026), DeepSeek-V4-Flash (284B) (DeepSeek-AI, 2026), GLM-5 (744B) (GLM-5-Team et al., 2026), and DeepSeek-V3 (671B) (Liu et al., 2024). The selected models are trained on domain-specific datasets including OR-based mathematical reasoning, Medical knowledge MedMCQA (Pal et al., 2022), and Social interactions SIQA (Sap et al., 2019). After training the model on domain-specific datasets, we validate the robustness of the trained model on the general-knowledge datasets, as presented in Appendix A.3. We would like to highlight the fact that this work is not deliberately picking the skewed data distributions to enforce the lopsided expert workloads. In fact, the domain-specific fine-tuning on powerful foundation LLMs is widely employed in practical scenarios. The detailed training configurations are summarized in Appendix A.5.

While most existing works are validated on small-sized MoE models rather than the mega-sized model, this work uses the vanilla MoE training (provided by the official recipes from Megatron), Fixed-routing enabled MoE training, and LocMoE (Li et al., 2024) as the comparison counterpart.

Table 2: Evaluation results and training speedup of domain-specific SFT. The proposed CIPHER-MoE achieves the optimal speedup and quality.

Model	Method	Operations Research						MedMCQA		SIQA	
		NL4Opt	OptiBench	B4O-Feas.	B4O-OR	W. Avg.	S_{train}	Acc.	S_{train}	Acc.	S_{train}
DeepSeek-V4-Flash	Vanilla MoE Training (Baseline)	93.08	68.00	71.51	51.02	69.09	-	76.60	-	51.84	-
	CIPHER-strict	93.77	66.67	70.64	51.27	68.59	**1.42**×	**77.36**	**1.50**×	54.81	**1.58**×
	CIPHER-reroute	90.66	66.33	69.77	51.27	67.73	1.38×	76.79	1.46×	**57.83**	1.52×
	LocMoE ($\tau = 0.005$)	94.12	**68.17**	66.28	50.76	68.16	1.17×	–	–	–	–
	LocMoE ($\tau = 0.01$)	**94.81**	67.67	70.06	**53.05**	**69.46**	1.12×	77.22	1.12×	55.94	1.19×
	LocMoE ($\tau = 0.02$)	93.08	67.50	**72.67**	51.52	69.27	1.12×	–	–	–	–
	LocMoE ($\tau = 0.03$)	93.77	65.67	70.93	52.79	68.65	1.10×	–	–	–	–
DeepSeek-V3	Vanilla MoE Training (Baseline)	83.39	60.17	58.43	46.45	60.60	-	72.32	-	**53.63**	-
	CIPHER-strict	83.74	59.83	**61.34**	**47.46**	61.40	**1.94**×	72.84	**1.84**×	52.00	**1.78**×
	CIPHER-reroute	82.70	61.50	61.05	47.20	**61.71**	1.68×	**73.44**	1.62×	52.25	1.56×
	LocMoE ($\tau = 0.01$)	**84.08**	**62.00**	59.30	45.94	61.46	1.75×	72.70	1.62×	53.07	1.50×
GLM-5	Vanilla MoE Training (Baseline)	85.47	64.67	**61.05**	45.94	63.06	-	80.66	-	**55.83**	-
	CIPHER-strict	**90.66**	**65.67**	57.56	46.95	**63.86**	**1.14**×	**81.31**	**1.14**×	55.02	**1.13**×
	CIPHER-reroute	88.58	63.33	58.14	46.95	62.75	1.10×	80.56	1.11×	53.63	1.10×
	LocMoE ($\tau = 0.01$)	88.93	62.33	57.56	**47.72**	62.51	1.12×	77.77	1.11×	50.15	1.11×

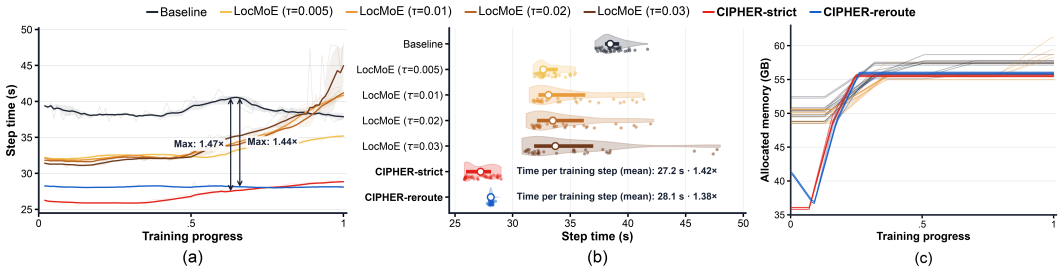


Figure 3: Training efficiency and memory footprint on DeepSeek-V4-Flash. (a) Per-step training time over the training progress, and CIPHER-MoE achieves up to 1.47× speedup. (b) Step-time distributions, where CIPHER-strict and CIPHER-reroute achieve an average speedup of 1.38–1.42× over the baseline. (c) Allocated memory throughout training.

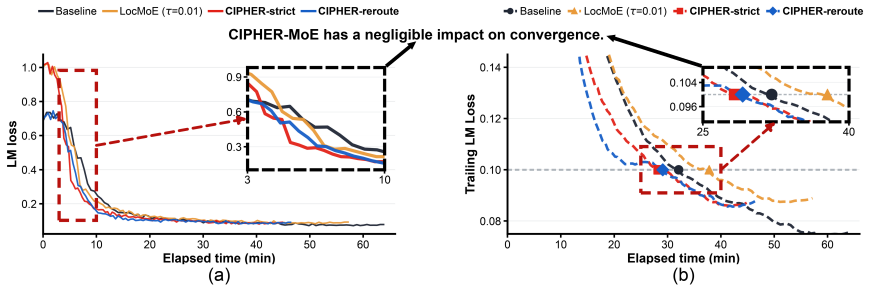


Figure 4: Training stability and convergence. (a) LM loss over elapsed training time. (b) Ten-step trailing LM loss and time to reach the 0.10 target.

4.1 Training Efficiency and Memory Usage

Training efficiency and accelerator memory usage are reported in Figure 3. Since CIPHER-MoE targets very large-scale MoE models and few existing methods provide directly comparable results at this scale, we compare it with LocMoE (Li et al., 2024), which has already been applied to large-cluster training. All reported training times are measured from end-to-end training runs. CIPHER-MoE achieves 1.42× speedups compared to other baselines. In the meantime, CIPHER-MoE limits the allocated memory to approximately 87.5% of the device memory capacity. These gains arise from capacity-aware expert admission, which reduces the number of executed token–expert assignments and the associated computation and communication. The computational overheads introduced by the strict and reroute policies account for only 1.57% and 3.85% of the total training time, respectively, allowing both policies to preserve most of the efficiency gains.

4.2 LLM Training Quality with CIPHER-MoE

Figure 4 demonstrates that CIPHER-MoE accelerates convergence without compromising training stability. Both CIPHER-strict and CIPHER-reroute exhibit smooth loss trajectories and converge reliably to the same low-loss regime as the reference methods. The detailed information of gradient norm is also provided in Appendix A.1. With the smooth training process of CIPHER-MoE, the post-trained LLM achieves the optimal performance compared to the baselines, as presented in Table 2. The proposed CIPHER-MoE achieves the best quality-efficiency tradeoff regardless of the model sizes and architectures.

4.3 Expert Update Distributions and Degree of Homogenization

As displayed in Figure 5, CIPHER-MoE consistently promotes stronger expert differentiation and avoids expert homogenization. CIPHER-strict exhibits the clearest separation in router adaptation. The agreement between the update dynamics and routed-representation geometry indicates that the resulting specialization reflects a persistent division of labor rather than a transient routing fluctuation. This behavior arises because affinity-aware capacity control suppresses weak or repetitive

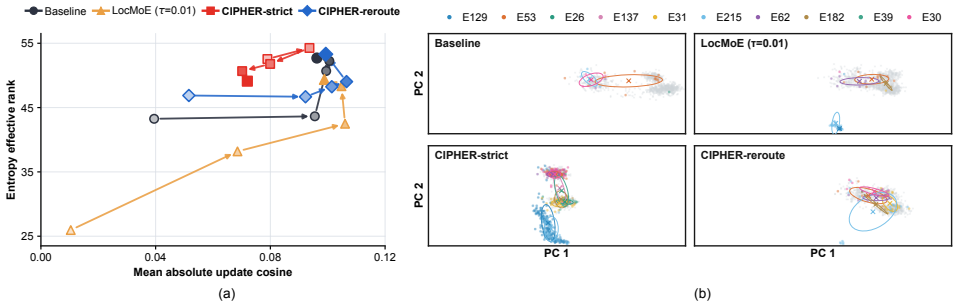


Figure 5: Router-update diversity and expert specialization. (a) Five consecutive training stages connected by arrows, with darker colors denoting later stages. Lower update-vector cosine similarity indicates clearer expert differentiation, while higher entropy effective rank indicates more dispersed, less homogeneous updates; the upper-left is better. (b) PCA of router-input hidden states. Compact same-color clusters indicate similar tokens routed to the same expert, while separation across colors indicates distinct token assignments and stronger specialization.

assignments, while selective rerouting directs overflow tokens toward compatible experts with available capacity, thereby preserving distinct expert learning signals.

4.4 Training Efficiency and Evaluation of DeepSeek-V4-Pro

(a) Training efficiency comparison.

Method	GBS	Elapsed Time (s)	Loss	Time Range (s)
DeepSeek-V4-Pro				
Baseline	256	— (OOM)	—	—
DeepSeek-V4-Pro	256	45.16	0.29–0.43	5.50
CIPHER-MoE	8192	538.65	1.41–1.51	91.11

(b) Evaluation results of OR domain benchmarks (%) for OR SFT.

Method	NL4Opt	OptiBench	B4O-Feas.	B4O-OR	Weighted Avg.
DeepSeek-V4-Pro					
Fixed Router	89.27	64.67	74.42	54.31	68.59
DeepSeek-V4-Pro					
CIPHER-MoE	92.04	65.33	67.44	59.14	69.02

Table 3: DeepSeek-V4-Pro training efficiency and OR-domain evaluation.

As shown in Table 3, the DeepSeek-V4-Pro baseline fails to train under limited memory resources due to OOM errors, whereas CIPHER-MoE's lower memory footprint enables successful training without fixed routing. Training remains stable even at a global batch size of 8192. CIPHER-MoE also outperforms the fixed-router baseline on multiple OR benchmarks and achieves a higher weighted average score.

5 Conclusion

This work addresses the expert workload imbalance that limits the efficiency and stability of MoE training, with empirical evaluations spanning hundred-billion- to trillion-parameter models. We propose `CIPHER-MoE`, a plug-and-play approach that balances workloads by constraining overloaded experts through capacity-aware token selection and rerouting. More importantly, this work reveals the bounded relationship between expert capacity and workload distribution. Experiments are conducted on the mainstream mega-sized MoE models ranging from 284B to 1.6T parameters, including DeepSeek-V3, DeepSeek-V4-Flash, GLM-5, and DeepSeek-V4-Pro. Across different architectures and datasets, CIPHER-MoE achieves measured training speedups of $1.10\times$–$1.94\times$, accelerating convergence while consistently preserving both domain-specific and general capabilities. These results demonstrate that CIPHER-MoE provides a simple, efficient, stable, and scalable foundation for large-scale MoE training.

AI USE STATEMENT

In this work, generative AI was used to generate synthetic examples for the OR SFT corpus and to assist with language polishing and the visual refinement of scientific figures. It also assisted in phrasing author-selected experimental findings and checking simple descriptive calculations reported in the manuscript; it did not generate, modify, or statistically analyze the underlying experimental records. Generative AI was not used to develop CIPHER-MoE, formulate its mathematical claims, produce or verify its proofs, design the experiments, implement the method, clean or reformat the experimental data, or determine the scientific conclusions. All AI-assisted text and visual modifications were reviewed by the authors against the original results and intended technical meaning. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work develops an efficiency-oriented routing method for large-scale MoE training and does not involve new human-subject recruitment or the collection of private personal data. The experiments use existing pretrained models, a validated synthetic OR corpus, and documented benchmark datasets; nevertheless, MedMCQA and Social IQA may inherit factual inaccuracies, demographic biases, or representational limitations from their source data. Results on MedMCQA should not be interpreted as evidence of clinical reliability, and CIPHER-MoE is not intended as a fairness, privacy, or safety intervention. Although improved training efficiency may reduce computational and communication costs, it may also facilitate the development of more capable models; downstream risks therefore remain dependent on the underlying model, training data, and deployment context. We report the relevant data provenance, experimental limitations, and scope of the supported claims to facilitate responsible interpretation.

REPRODUCIBILITY STATEMENT

Section 3.2 and Appendix A.4 specify the routing policies and implementation contract, while Appendix A.4.2 presents the assumptions and complete proofs underlying the theoretical results. Appendix A.5 documents the available training configurations and routing hyperparameters. The supplementary material provides figure-generation and verification scripts, source manifests, and checksums that link the reported results to their underlying records. We explicitly disclose the single-run evaluation design and any unavailable configurations or measurements. The CIPHER-MoE implementation and associated configuration files will be released publicly after code cleanup and documentation are complete.

REFERENCES

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Lizhang Chen, Jonathan Li, Qi Wang, Runlong Liao, Shuozhe Li, Chen Liang, Ni Lao, and Qiang Liu. phi-balancing for mixture-of-experts training. *arXiv preprint arXiv:2605.15403*, 2026.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1280–1297, 2024.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Nishanth Dikkala, Nikhil Ghosh, Raghu Meka, Rina Panigrahy, Nikhil Vyas, and Xin Wang. On the benefits of learning to route in mixture-of-experts models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9376–9396, 2023.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2368–2378, 2019.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Trevor Gale, Deepak Narayanan, Cliff Young, and Matei Zaharia. Megablocks: Efficient sparse training with mixture-of-experts. *Proceedings of Machine Learning and Systems*, 5:288–304, 2023.

GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Hongcan Guo, Haolang Lu, Guoshun Nan, Bolun Chu, Jialin Zhuang, Yuan Yang, Wenhao Che, Xinye Cao, Sicong Leng, Qimei Cui, et al. Advancing expert specialization for better moe. *Advances in Neural Information Processing Systems*, 38:48767–48809, 2026.

Jiaao He, Jidong Zhai, Tiago Antunes, Haojie Wang, Fuwen Luo, Shangfeng Shi, and Qin Li. Fastermoe: modeling and optimizing training of large-scale dynamic pre-trained models. In *Proceedings of the 27th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, pp. 120–134, 2022.

Changho Hwang, Wei Cui, Yifan Xiong, Ziyue Yang, Ze Liu, Han Hu, Zilong Wang, Rafael Salas, Jithin Jose, Prabhat Ram, et al. Tutel: Adaptive mixture-of-experts at scale. *Proceedings of Machine Learning and Systems*, 5:269–287, 2023.

Chao Jin, Ziheng Jiang, Zhihao Bai, Zheng Zhong, Juncai Liu, Xiang Li, Ningxin Zheng, Xi Wang, Cong Xie, Qi Huang, et al. Megascale-moe: Large-scale communication-efficient training of mixture-of-experts models in production. In *Proceedings of the 21st European Conference on Computer Systems*, pp. 366–382, 2026a.

Chao Jin, Xinming Wei, Yinmin Zhong, Chengxu Yang, Bingyang Wu, Ruidong Zhu, Zili Zhang, Yuliang Liu, and Xin Jin. Relibra: Routing-replay-guided load balancing for moe training in reinforcement learning. *arXiv preprint arXiv:2605.08639*, 2026b.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Yechan Kim, Hwijoon Lim, and Dongsu Han. Scaling beyond the gpu memory limit for large mixture-of-experts model training. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

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.

Mike Lewis, Shruti Bhosale, Tim Dettmers, Naman Goyal, and Luke Zettlemoyer. Base layers: Simplifying training of large, sparse models. In *International Conference on Machine Learning*, pp. 6265–6274. PMLR, 2021.

Jing Li, Zhijie Sun, Xuan He, Li Zeng, Yi Lin, Entong Li, Binfan Zheng, Rongqian Zhao, and Xin Chen. Locmoe: a low-overhead moe for large language model training. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/705. URL https://doi.org/10.24963/ijcai.2024/705.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Xinyi Liu, Yujie Wang, Fangcheng Fu, Xuefeng Xiao, Huixia Li, Jiashi Li, and Bin Cui. Laer-moe: Load-adaptive expert re-layout for efficient mixture-of-experts training. In *Proceedings of the 31st ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, pp. 1055–1072, 2026.

Mathematical Association of America. 2024 AMC 12a and AMC 12b competitions. https://maa.org/student-programs/amc/, 2024. American Mathematics Competitions.

Mathematical Association of America. 2026 american invitational mathematics examination i, 2026a. URL https://maa.org/news/2025-26-aime-thresholds-are-now-available/. Administered February 5, 2026.

Mathematical Association of America. 2026 american invitational mathematics examination ii, 2026b. URL https://maa.org/news/2025-26-aime-thresholds-are-now-available/. Administered February 11, 2026.

Xuan-Phi Nguyen, Shrey Pandit, Austin Xu, Caiming Xiong, and Shafiq Joty. Least-loaded expert parallelism: Load balancing an imbalanced mixture-of-experts. *arXiv preprint arXiv:2601.17111*, 2026.

Xiaonan Nie, Xupeng Miao, Zilong Wang, Zichao Yang, Jilong Xue, Lingxiao Ma, Gang Cao, and Bin Cui. Flexmoe: Scaling large-scale sparse pre-trained model training via dynamic device placement. *Proceedings of the ACM on Management of Data*, 1(1):1–19, 2023.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann (eds.), *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 248–260. PMLR, 07–08 Apr 2022. URL https://proceedings.mlr.press/v174/pal22a.html.

Zihan Qiu, Zeyu Huang, Bo Zheng, Kaiyue Wen, Zekun Wang, Rui Men, Ivan Titov, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Demons in the detail: On implementing load balancing loss for training specialized mixture-of-expert models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5005–5018, 2025.

Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale. In *International conference on machine learning*, pp. 18332–18346. PMLR, 2022.

Rindranirina Ramamonjison, Timothy Yu, Raymond Li, Haley Li, Giuseppe Carenini, Bissan Ghaddar, Shiqi He, Mahdi Mostajabdaveh, Amin Banitalebi-Dehkordi, Zirui Zhou, et al. Nl4opt competition: Formulating optimization problems based on their natural language descriptions. In *NeurIPS 2022 competition track*, pp. 189–203. PMLR, 2023.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 4463–4473, 2019.

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.

Shaohuai Shi, Xinglin Pan, Xiaowen Chu, and Bo Li. Pipemoe: Accelerating mixture-of-experts through adaptive pipelining. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2023.

Athinagoras Skiadopoulos, Mark Zhao, Swapnil Gandhi, Thomas Norrie, Shrijeet Mukherjee, and Christos Kozyrakis. Accelerating mixture-of-experts training with adaptive expert replication. *arXiv e-prints*, pp. arXiv–2504, 2025.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

Lean Wang, Huazuo Gao, Chenggang Zhao, Xu Sun, and Damai Dai. Auxiliary-loss-free load balancing strategy for mixture-of-experts. *arXiv preprint arXiv:2408.15664*, 2024a.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Zhuohan Wang, Ziwei Zhu, Ziniu Li, Congliang Chen, Yizhou Han, Yufeng Lin, Zhihang Lin, Angyang Gu, Xinglin Hu, Ruoyu Sun, et al. Orgeval: Graph-theoretic evaluation of llms in optimization modeling. *arXiv preprint arXiv:2510.27610*, 2025.

Zhicheng Yang, Yiwei Wang, Yinya Huang, Zhijiang Guo, Shi Shi, Xiongwei Han, Liang Feng, Linqi Song, Xiaodan Liang, and Jing Tang. Optibench meets resocratic: Measure and improve llms for optimization modeling. In *International Conference on Learning Representations*, volume 2025, pp. 24726–24759, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Shulai Zhang, Ningxin Zheng, Haibin Lin, Ziheng Jiang, Wenlei Bao, Chengquan Jiang, Qi Hou, Weihao Cui, Size Zheng, Li-Wen Chang, et al. Comet: Fine-grained computation-communication overlapping for mixture-of-experts. *Proceedings of Machine Learning and Systems*, 7, 2025.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114, 2022.

A Appendix

A.1 Gradient Dynamics and Training Stability of MoE Models

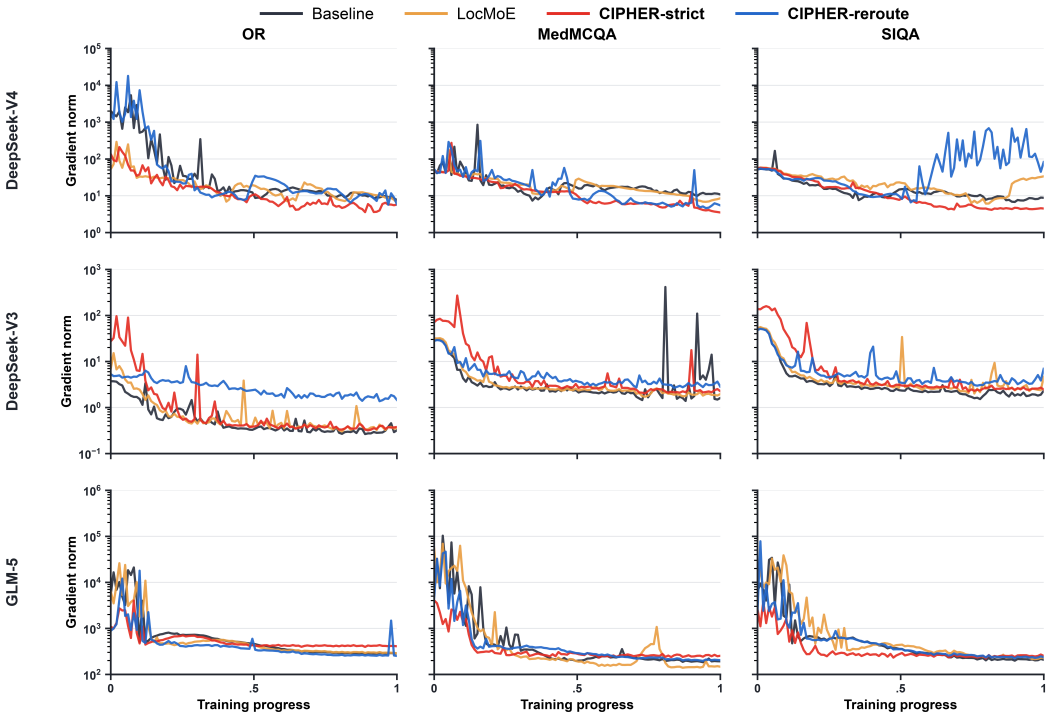


Figure 6: Gradient stability of CIPHER-MoE over training progress for DeepSeek-V4-Flash, DeepSeek-V3, and GLM-5 on MedMCQA, SIQA, and OR, shown on logarithmic scales.

In Figure 6, across all evaluated model-dataset configurations, CIPHER-MoE maintains finite, non-degenerate gradient norms that generally decay from their initial transients into stable ranges, with no evidence of gradient explosion or vanishing. CIPHER-strict often produces smoother and lower gradient norms than the baseline, while CIPHER-reroute remains bounded despite occasional task-dependent fluctuations. Together with the consistently decreasing loss curves, these results indicate that CIPHER-MoE preserves well-behaved optimization dynamics and can provide greater training stability than the baseline. Overall, capacity-aware routing enables favorable convergence without destabilizing gradient-based training.

In Figure 7, across early, middle, and late layers, both CIPHER policies suppress the severe routing hotspots observed under the baseline and LocMoE, reducing peak expert traffic from as high as 83.6% to 18.7%; CIPHER-strict minimizes routing volume, whereas CIPHER-reroute achieves the most balanced expert utilization while retaining broader expert coverage.

A.2 Training Efficiency and Stability of Qwen Models

Qwen3.6-35B-A3B adopts hidden-state dimensions of 2048 for Qwen3.6-35B-A3B, whose relatively limited widths are consistent with failure to satisfy the favorable dimension-margin regime required by CIPHER-MoE in Table 7, as shown in Figure 8. Consequently, the observed training-efficiency improvement remains below 5%, falling short of the gains achieved on larger MoE models. This contrast suggests that CIPHER-MoE is better suited to large-scale MoE architectures with higher-dimensional representations and may deliver increasingly substantial benefits as future models approach the 10-trillion-parameter scale.

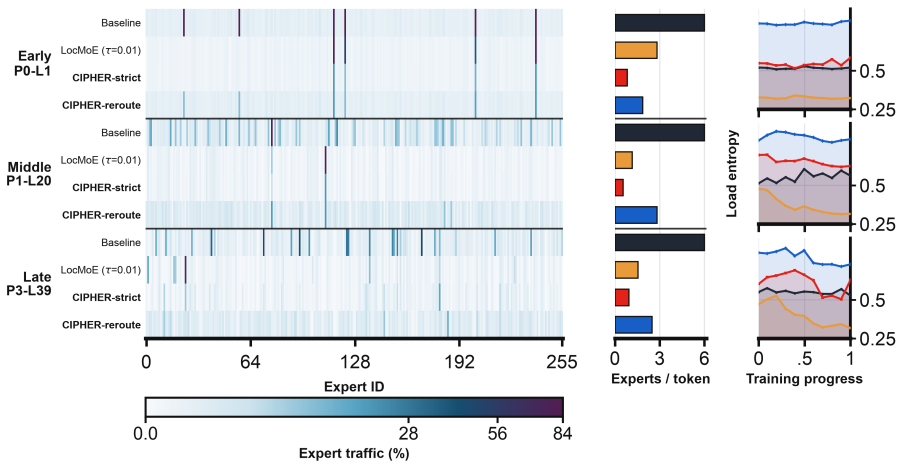


Figure 7: Expert routing behavior. Expert traffic across representative MoE layers (left), activated experts per token (middle), and load entropy during training (right).

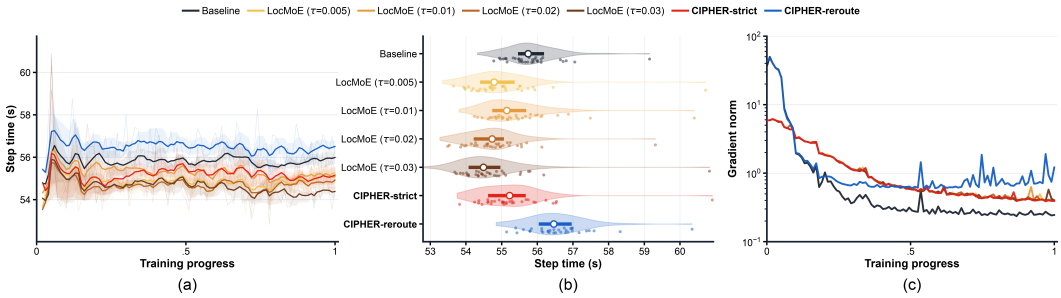


Figure 8: Training efficiency and optimization stability on Qwen3.6-35B-A3B. (a) Step time over training progress, shown with a 10-step rolling mean and interquartile range. (b) Distribution of post-warm-up step times. (c) Raw gradient norms on a logarithmic scale. CIPHER maintains stable step times and convergent gradient norms, with CIPHER-strict exhibiting smoother late-stage optimization.

Table 4: Evaluation results of general benchmarks (%) for OR SFT.

Model	Method	AIME-26	AMC	ARC	BBH	DROP	GPQA-D	GSM8K	HellaSwag	IFEval	MATH-500	MMLU-Pro
DeepSeek-V4-Flash	Baseline	33.33	71.60	98.10	**84.00**	86.00	64.14	95.90	**85.40**	**87.80**	91.20	80.02
	CIPHER-strict	30.00	70.90	**98.20**	71.20	84.50	67.17	95.70	80.80	83.70	91.80	80.98
	CIPHER-reroute	40.00	70.60	97.80	58.00	72.30	55.56	95.70	81.80	80.40	90.40	75.47
	LocMoE (τ=0.005)	33.33	70.90	98.10	71.00	86.30	66.67	**96.80**	82.00	86.50	92.20	80.59
	LocMoE (τ=0.01)	33.33	70.90	98.10	71.30	86.30	65.15	96.30	81.80	85.20	93.20	80.86
	LocMoE (τ=0.02)	**46.67**	71.60	98.00	71.20	86.40	65.15	96.40	81.70	85.40	91.80	81.44
	LocMoE (τ=0.03)	36.67	**72.40**	98.10	71.90	**86.90**	**70.20**	96.30	81.30	83.90	**94.20**	**81.70**
DeepSeek-V3	Baseline	**46.67**	**71.64**	**97.94**	87.87	87.19	52.53	95.68	75.97	73.74	**88.80**	75.51
	CIPHER-strict	**46.67**	64.18	97.91	85.99	**89.24**	**57.58**	96.13	87.45	84.66	87.60	**76.77**
	CIPHER-reroute	43.33	62.69	97.91	86.44	88.51	52.53	**96.29**	**87.48**	**93.05**	88.00	76.14
	LocMoE (τ=0.01)	40.00	66.42	97.07	**88.44**	87.27	56.57	95.68	80.65	81.65	88.60	75.57
GLM-5	Baseline	63.33	**86.57**	98.20	93.20	82.19	61.11	96.74	87.55	90.94	**95.80**	**81.67**
	CIPHER-strict	53.33	81.34	**98.59**	93.44	81.56	**64.65**	97.19	**88.19**	91.31	92.80	80.15
	CIPHER-reroute	**66.67**	82.84	98.37	91.34	**82.96**	58.59	**97.80**	87.65	**92.05**	95.20	80.33
	LocMoE (τ=0.01)	60.00	82.84	97.27	**93.87**	80.53	63.64	96.21	85.22	91.50	95.20	77.85

A.3 Evaluation results of general benchmarks

General capabilities are evaluated across five categories: mathematics (AIME-26 (Mathematical Association of America, 2026a;b), AMC (Mathematical Association of America, 2024), GSM8K (Cobbe et al., 2021), and MATH-500 (Lightman et al., 2024)), general knowledge

Table 5: Evaluation results of general benchmarks (%) for MedMCQA SFT.

Model	Method	AIME-26	AMC	ARC	BBH	DROP	GPQA-D	GSM8K	HellaSwag	IFEval	MATH-500	MMLU-Pro
DeepSeek-V4-Flash	Baseline	50.00	**80.60**	97.50	85.00	**85.20**	62.63	95.70	80.20	70.20	92.60	76.38
	CIPHER-strict	**56.67**	75.56	**98.03**	85.16	81.62	**66.67**	**96.44**	76.35	**78.74**	94.00	80.31
	CIPHER-reroute	50.00	75.56	97.74	**86.13**	81.40	60.10	96.13	**80.33**	74.12	92.80	78.77
	LocMoE (τ=0.01)	**56.67**	80.00	97.86	84.53	81.51	**66.67**	96.29	76.86	77.45	**94.20**	**80.38**
DeepSeek-V3	Baseline	46.67	**77.61**	96.65	87.70	82.65	**60.61**	**96.66**	74.38	80.10	92.20	74.28
	CIPHER-strict	46.67	74.63	**97.97**	86.28	**86.77**	55.56	96.29	**87.48**	82.07	88.00	**76.32**
	CIPHER-reroute	**53.33**	76.12	97.86	85.33	85.66	59.09	95.83	80.87	80.41	90.20	74.75
	LocMoE (τ=0.01)	43.33	76.86	96.99	**88.33**	85.89	57.07	**96.66**	75.08	**83.93**	**92.80**	75.01
GLM-5	Baseline	**63.33**	86.57	98.34	91.49	**84.49**	61.11	96.82	**87.46**	90.76	94.80	80.38
	CIPHER-strict	60.00	85.07	**98.68**	92.58	82.01	62.63	**97.88**	86.25	90.20	95.40	81.38
	CIPHER-reroute	60.00	**88.06**	98.39	92.40	79.38	**65.15**	96.89	85.74	**94.64**	95.80	**81.82**
	LocMoE (τ=0.01)	60.00	82.09	98.00	**93.12**	82.50	**65.15**	96.97	86.31	91.31	**96.20**	75.07

Table 6: Evaluation results of general benchmarks (%) for SIQA SFT.

Model	Method	AIME-26	AMC	ARC	BBH	DROP	GPQA-D	GSM8K	HellaSwag	IFEval	MATH-500	MMLU-Pro
DeepSeek-V4-Flash	Baseline	53.33	75.40	96.30	**87.90**	**84.70**	60.10	96.10	87.10	79.30	90.80	77.68
	CIPHER-strict	50.00	**77.78**	98.03	85.64	81.53	**62.63**	96.44	85.12	80.59	**92.00**	78.26
	CIPHER-reroute	**60.00**	73.33	**98.17**	85.73	79.46	60.10	95.75	**88.06**	**81.48**	91.60	74.67
	LocMoE (τ=0.01)	53.33	73.33	97.91	85.73	81.91	56.57	**96.51**	84.87	81.33	91.20	**78.27**
DeepSeek-V3	Baseline	53.33	72.39	96.62	85.30	84.31	54.55	93.10	78.87	76.14	87.60	71.67
	CIPHER-strict	46.67	65.67	**97.94**	**86.35**	**88.39**	**57.58**	**95.98**	**87.61**	**92.81**	87.40	**73.99**
	CIPHER-reroute	53.33	68.66	97.86	84.23	87.45	56.57	95.00	85.23	89.09	88.60	71.01
	LocMoE (τ=0.01)	**56.67**	**73.88**	97.10	81.97	87.21	54.55	95.68	80.01	87.29	**90.80**	72.22
GLM-5	Baseline	53.33	82.84	97.49	91.49	83.66	61.11	95.91	86.48	91.50	93.80	80.56
	CIPHER-strict	**56.67**	84.33	**98.03**	**93.56**	83.47	**62.63**	**97.04**	**89.12**	**91.68**	92.40	81.22
	CIPHER-reroute	**56.67**	**85.07**	97.89	92.87	**84.55**	61.11	96.89	88.74	90.57	93.80	**82.68**
	LocMoE (τ=0.01)	50.00	82.09	97.66	92.23	81.75	**62.63**	95.83	86.88	90.20	**95.20**	75.86

(ARC (Clark et al., 2018), GPQA-D (Rein et al., 2023), MMLU-Pro (Wang et al., 2024b), and HellaSwag (Zellers et al., 2019)), logical reasoning (BBH (Suzgun et al., 2023)), reading comprehension (DROP (Dua et al., 2019)), and instruction following (IFEval (Zhou et al., 2023)). Tables 4, 5, and 6 demonstrate that CIPHER-MoE can strengthen broad transfer performance rather than merely preserve it. For DeepSeek-V3 after SIQA SFT, CIPHER-strict improves the macro-average across ARC, BBH, GPQA-D, HellaSwag, and MMLU-Pro by 4.3% relative to the baseline. This pattern suggests that selective routing suppresses low-value signals and concentrates expert updates on more informative token–expert associations. The efficiency benefits of CIPHER-MoE therefore need not come at the expense of general capability and may instead be accompanied by better transferable reasoning.

A.4 Capacity analysis of Theorem 1

A.4.1 Comparison task and assumptions

Consider expert e and its candidate pool in one micro-batch. Let $N_e \geq 1$ be the number of offered tokens, and let $C \in \{1, \ldots, N_e\}$ be the number retained for expert execution. A token is *discriminative* if it carries the expert-specific signal specified by the model below. Write D_i for its indicator and $A_e = \{\sum_{i=1}^{N_e} D_i \geq 1\}$ for the event that the offered pool contains such a token. The common retention target is given in equation 6. Natural-order admission executes the first C offered tokens, whereas CIPHER admission ranks the available candidates before allocating its C slots. Computing affinities and selecting candidates are not counted as expert-execution capacity. The comparison therefore concerns signal retention per execution slot; its implications for end-to-end training time are evaluated separately.

A1: Exposure, geometry, and signal availability. The offered tokens are independent draws from a common exposure distribution, with discriminative probability $\pi_e \in (0, 1)$ and

$$\Pr(A_e) = 1 - (1 - \pi_e)^{N_e} \geq a_0 > 0. \tag{11}$$

For the geometric reference model, let $X \sim \mathrm{Unif}(S^{d-1})$, let $\boldsymbol{u}_e$ be the unit gating direction, and let $\mathcal{R}_e$ denote exposure to expert e. Suppose the discriminative event $\mathcal{D}_e$ and exposure probability

satisfy

$$\mathcal{D}_e \subseteq \{\langle \boldsymbol{u}_e, X\rangle \geq \delta\}, \qquad \Pr(\mathcal{R}_e) \geq \frac{1}{\beta E}, \qquad 0 < \delta < 1, \quad \beta \geq 1. \tag{12}$$

The offered distribution is the exposure-conditioned distribution, rather than an unconditioned uniform spherical distribution. Consequently,

$$\pi_e = \Pr(\mathcal{D}_e \mid \mathcal{R}_e) \leq \beta E \Pr(\mathcal{D}_e) \leq \beta E p_\delta(d), \qquad p_\delta(d) = \Pr(|\langle \boldsymbol{u}_e, X\rangle| \geq \delta). \tag{13}$$

No independence between discrimination and exposure is required for this inequality. The two-sided spherical probability gives a conservative bound for the one-sided discriminative event. Equal-norm orthogonal gates under the symmetric reference model yield $\beta = 1$; learned gates, expert biases, and Top-K selection require their own exposure justification. The margin δ is a certified property of the signal model and is distinct from the algorithmic affinity threshold τ.

A2: Candidate and threshold recall. Conditional on A_e, choose one discriminative token as a *witness* according to a fixed rule. Let $\mathcal{V}_e$ denote the event that this witness reaches the expert's ranking pool and survives thresholding. Assume

$$\Pr(\mathcal{V}_e \mid A_e) \geq 1 - \eta, \qquad 0 \leq \eta < \varepsilon. \tag{14}$$

This explicitly accounts for signal loss before capacity truncation. Increasing execution capacity cannot recover a witness that has already been excluded from the ranking pool.

A3: Stable conditional ranking. Condition on A_e and $\mathcal{V}_e$. An expert ranks candidates using affinity and a fixed deterministic rule for ties. Expose the competing candidates sequentially, and let Y_j indicate whether competitor j precedes the witness. For the information $\mathcal{F}_{j-1}$ revealed by previous competitors, assume deterministic bounds

$$\Pr(Y_j = 1 \mid \mathcal{F}_{j-1}, A_e, \mathcal{V}_e) \leq q_j, \qquad 0 \leq q_j \leq 1, \qquad \sum_{j=1}^{N_e - 1} q_j \leq \mu. \tag{15}$$

Here, μ bounds the total conditional number of competitors that can rank ahead of the witness. If fewer than N_e candidates reach the ranking pool, the remaining indicators are set to zero. With $S = \sum_j Y_j$, the witness occupies position $S + 1$. This assumption allows dependence between candidates, provided the bound holds after witness selection and after observing earlier competitors. A marginal ranking-error rate alone does not imply this conditional guarantee.

A.4.2 Capacity separation for affinity-based admission

We prove the capacity bounds in Theorem 1, using B_e and $G(d, \delta)$ as defined in equation 7. For any prescribed reduction factor $\kappa > 1$, we also derive the following sufficient condition for $C^*_{\text{CIPHER}} \leq C^*_{\text{natural}}/\kappa$:

$$d \geq 1 + \frac{2\log(\kappa\beta E B_e/[(1-\varepsilon)a_0])}{-\log(1-\delta^2)}. \tag{16}$$

Theorem 1 identifies two different controls on execution capacity. Natural-order admission must allocate enough slots to encounter a scarce discriminative token. Affinity admission instead needs enough slots to accommodate candidates ranked ahead of an available witness. When this conditional ranking budget is small, the latter policy can operate with substantially fewer slots. The quantity B_e is a sufficient budget, rather than an exact minimum or an optimal training-time configuration.

Spherical concentration. We first establish the finite-dimensional geometric bound used in the theorem. Rotational symmetry and the normalized Gaussian representation of a spherical vector give

$$p_\delta(d) = 1 - I_{\delta^2}\left(\frac{1}{2}, \frac{d-1}{2}\right) \leq (1-\delta^2)^{(d-1)/2} = e^{-G(d,\delta)}, \tag{17}$$

where $I_z(a, b)$ is the regularized incomplete beta function. To prove the inequality, let $Z_1, \ldots, Z_d$ be independent standard Gaussian variables, set $W = \sum_{j=2}^{d} Z_j^2$, and write $X = Z/\|Z\|$. Then

$X_1^2 = Z_1^2/(Z_1^2+W)$ has the beta distribution appearing above. With $b = \delta^2/(1-\delta^2)$, the cap event is $Z_1^2 \geq bW$. The Gaussian tail satisfies $2[1-\Phi(v)] \leq e^{-v^2/2}$ for $v \geq 0$, where Φ is the standard Gaussian cumulative distribution function. Indeed, the difference between the right and left sides vanishes at zero and at infinity, and its derivative is $e^{-v^2/2}(\sqrt{2/\pi} - v)$, so it remains nonnegative. Conditioning on W therefore yields

$$\begin{aligned} p_\delta(d) &= \mathbb{E}\Big[2\{1-\Phi(\sqrt{bW})\}\Big] \leq \mathbb{E}e^{-bW/2} \\ &= \prod_{j=2}^{d} \mathbb{E}e^{-bZ_j^2/2} = (1+b)^{-(d-1)/2} = (1-\delta^2)^{(d-1)/2}. \end{aligned} \tag{18}$$

This bound holds at finite d and does not require an asymptotic approximation. In particular, $G(d,\delta) \geq (d-1)\delta^2/2$.

Necessary capacity for natural-order admission. Success within the first C slots implies A_e. Independence in A1 thus gives

$$\begin{aligned} P_{\text{natural}}(C \mid A_e) &= \frac{1-(1-\pi_e)^C}{1-(1-\pi_e)^{N_e}}, \\ C^*_{\text{natural}} &= \left\lceil \frac{\log\big(1-(1-\varepsilon)[1-(1-\pi_e)^{N_e}]\big)}{\log(1-\pi_e)} \right\rceil. \end{aligned} \tag{19}$$

The exact expression follows by solving $P_{\text{natural}}(C \mid A_e) \geq 1-\varepsilon$; division by $\log(1-\pi_e) < 0$ reverses the inequality. For a simpler necessary bound, $1-(1-\pi_e)^C \leq C\pi_e$ and equation 11 imply

$$C^*_{\text{natural}} \geq \frac{(1-\varepsilon)a_0}{\pi_e} \geq \frac{(1-\varepsilon)a_0}{\beta E} e^{G(d,\delta)}, \tag{20}$$

where the second inequality uses equations 13 and 17. The union-bound argument requires only the marginal scarcity bound and positive availability; independence is needed for the exact expression in equation 19.

Conditional concentration of competing candidates. Write $\mathbb{Q}$ for the probability law conditioned on A_e and $\mathcal{V}_e$. For $\theta > 0$, A3 gives

$$\begin{aligned} \mathbb{E}_{\mathbb{Q}}[e^{\theta Y_j} \mid \mathcal{F}_{j-1}] &= 1 + (e^\theta - 1)\mathbb{E}_{\mathbb{Q}}[Y_j \mid \mathcal{F}_{j-1}] \\ &\leq 1 + q_j(e^\theta - 1) \leq \exp\{q_j(e^\theta - 1)\}. \end{aligned} \tag{21}$$

Iterating conditional expectations and using $\sum_j q_j \leq \mu$ yields $\mathbb{E}_{\mathbb{Q}} e^{\theta S} \leq \exp\{\mu(e^\theta - 1)\}$. Exponential Markov and minimization over $\theta > 0$ then give, for $\mu > 0$ and $x > 0$,

$$\begin{aligned} \Pr_{\mathbb{Q}}(S \geq \mu + x) &\leq \exp\{-\mu h(x/\mu)\} \leq \exp\left\{-\frac{x^2}{2(\mu + x/3)}\right\}, \\ h(v) &= (1+v)\log(1+v) - v. \end{aligned} \tag{22}$$

The first exponent follows by setting $\theta = \log(1 + x/\mu)$. For the second, the series inequality $e^\theta - 1 - \theta \leq \theta^2/[2(1-\theta/3)]$, valid for $0 < \theta < 3$, follows from $k! \geq 2 \cdot 3^{k-2}$ for $k \geq 2$. Substituting $\theta = x/(\mu + x/3)$ into the Markov exponent gives the stated bound. If $\mu = 0$, every q_j is zero, so $S = 0$ almost surely under $\mathbb{Q}$ and one slot suffices for an admitted witness.

Sufficient CIPHER capacity and proof of separation. Choose

$$x = \frac{\ell}{3} + \sqrt{2\mu\ell + \frac{\ell^2}{9}}, \qquad x^2 = 2\ell(\mu + x/3). \tag{23}$$

Equation 22 bounds the ranking failure by $e^{-\ell} = \varepsilon - \eta$. Since the witness has rank $S+1$, capacity $\lceil 1 + \mu + x \rceil$ is sufficient. If this exceeds N_e, admitting all available candidates eliminates capacity-induced failure, giving B_e in equation 7. The same expression remains sufficient when $\mu = 0$ or $N_e = 1$, although it can be conservative. Accounting for candidate-recall failure gives

$$\Pr(\text{failure} \mid A_e) \leq \eta + (1-\eta)(\varepsilon - \eta) \leq \varepsilon. \tag{24}$$

Thus $C^*_{\mathrm{CIPHER}} \le B_e$. Together with equation 20, this proves equation 8. Requiring the natural-order lower bound to be at least κB_e is equivalent to

$$G(d,\delta) \ge \log \frac{\kappa\beta E B_e}{(1-\varepsilon)a_0}, \tag{25}$$

which gives equation 16 after substituting the definition of G. This completes the proof of Theorem 1.

Finite-pool feasibility. Availability imposes the additional requirement

$$N_e \ge \frac{\log(1-a_0)}{\log(1-\pi_e)}. \tag{26}$$

Ranking cannot create a discriminative token absent from the offered pool. If the claimed natural-order lower bound exceeds N_e, the assumed scarcity and availability cannot both hold for that pool. The theorem does not prescribe execution of more tokens than are available. The degenerate case $\pi_e = 0$ violates positive availability, whereas $\pi_e = 1$ requires only one natural-order slot and admits no nontrivial separation of this kind.

The assumptions are jointly feasible. For example, consider a single expert, identity exposure, independent spherical tokens, and $\mathcal{D}_e = \{\langle \boldsymbol{u}_e, X\rangle \ge \delta\}$. Choose a sufficiently large pool to maintain a_0, and choose its highest-projection discriminative token as the witness. Ranking by exact projection places no competitor ahead of this witness, up to probability-zero ties. With complete candidate recall, $\eta = \mu = 0$ and one affinity-selected slot suffices, whereas natural-order admission still follows equation 19. This construction establishes compatibility of the assumptions without asserting that a trained checkpoint follows the reference distribution.

Retention guarantee for reroute. For the same input micro-batch, let $\mathcal{T}_e^{\mathrm{S}}$ be the tokens retained by expert e under strict selection and $\mathcal{T}_e^{\mathrm{R}}$ its final token set after rerouting. If rerouting preserves all strictly accepted assignments and adds only tokens that satisfy the destination expert's similarity threshold within its remaining capacity, then

$$\mathcal{T}_e^{\mathrm{S}} \subseteq \mathcal{T}_e^{\mathrm{R}}, \qquad |\mathcal{T}_e^{\mathrm{R}}| \le C_e, \tag{27}$$

Every discriminative witness retained by strict is therefore also retained by reroute. The strict-retention event is a subset of the reroute-retention event, so reroute inherits the same lower bound on the conditional retention probability under the assumptions of Theorem 1. This guarantee requires preserving accepted assignments and respecting each expert's capacity; it does not apply if rerouting evicts a strictly accepted token. Additional assignments can change the expert mixture and the resulting training updates, so the retention guarantee does not imply unchanged model outputs or improved downstream performance.

A.4.3 Sharper capacity bounds and stability conditions

Sharper conditional ranking bounds. When A3 admits a common bound $q_j \le q$, retaining the exact Bernoulli moment-generating function gives

$$\begin{aligned} \mathbb{E}_{\mathbb{Q}} e^{\theta S} &\le (1-q+qe^{\theta})^{N_e-1}, \\ \Pr_{\mathbb{Q}}(S \ge k) &\le \exp\left[-(N_e-1)\,\mathrm{kl}\left(\frac{k}{N_e-1}\middle\|q\right)\right], \end{aligned} \tag{28}$$

for $N_e \ge 2$, $0 < q < 1$, and $(N_e-1)q < k \le N_e - 1$, where $\mathrm{kl}(u\|q) = u\log(u/q) + (1-u)\log[(1-u)/(1-q)]$. Optimizing the Markov exponent yields $e^{\theta} = u(1-q)/[q(1-u)]$ for $u = k/(N_e-1) < 1$; the endpoint $u = 1$ follows by taking $\theta \to \infty$. Numerically inverting this bound can reduce the sufficient integer budget. If the blockers are independent Bernoulli variables under $\mathbb{Q}$, the exact witness-retention budget is the corresponding binomial quantile,

$$C^*_{\mathrm{rank}}(\alpha) = 1 + \min\{k \in \{0,\dots,N_e-1\} : \Pr(\mathrm{Bin}(N_e-1,q) \le k) \ge 1-\alpha\}. \tag{29}$$

Here α is the conditional ranking-failure allowance, and $\alpha = \varepsilon - \eta$ accounts for candidate recall as above. A simpler sufficient budget associated with the comparison is

$$\widetilde{C} = \min\left\{N_e, \left\lceil \max\left\{2N_e q, \frac{16}{3}\log\frac{1}{\alpha}\right\}\right\rceil\right\}. \tag{30}$$

Indeed, with $m_0 = (N_e-1)q \le C/2$, the Bernstein exponent satisfies $3(C-m_0)^2/[2(C+2m_0)] \ge 3C/16$. Admitting the entire pool handles budgets exceeding N_e; one slot suffices when $q = 0$. These bounds concern the chosen witness, whereas retaining any one of several discriminative tokens may require fewer slots.

Data drift and affinity calibration. To connect A3 with changing training data, suppose a relevant signal has ideal normalized affinity at least δ_0, and data drift reduces this margin by at most Δ. Let $c^\star_{j,e}$ and $c_{j,e}$ be ideal and learned affinities, respectively. Assume both scores in a witness–competitor comparison have error at most ζ, except with conditional probability ϱ. Also assume the conditional competitor tail satisfies

$$\Pr(c^\star_{j,e} \ge z \mid \mathcal{F}_{j-1}, A_e, \mathcal{V}_e) \le \chi \Pr(\langle \boldsymbol{u}_e, X\rangle \ge z) + \nu, \qquad \chi \ge 1, \quad \nu \ge 0. \tag{31}$$

The parameters χ and ν account for tail inflation and contamination after conditioning on exposure, witness admission, and previous competitors. Unconditional isotropy or calibration accuracy does not establish this bound.

On a calibrated comparison, a competitor preceding the witness must have ideal affinity at least

$$c^\star_{j,e} \ge c_{j,e} - \zeta \ge c_{*,e} - \zeta \ge c^\star_{*,e} - 2\zeta \ge \delta_0 - \Delta - 2\zeta \equiv \delta_{\text{eff}} > 0. \tag{32}$$

For $\delta_{\text{eff}} < 1$, a union bound and the one-sided spherical tail yield

$$q_{\max} \le \min\left\{1, \varrho + \nu + \frac{\chi}{2} e^{-G(d,\delta_{\text{eff}})}\right\}, \qquad \mu = (N_e - 1) q_{\max}. \tag{33}$$

The factor $1/2$ arises from the one-sided competitor event; equation 13 used a conservative two-sided bound. A bounded total ranking budget is obtained along configurations satisfying $N_e(\varrho + \nu) = O(1)$ and $N_e \chi e^{-G(d,\delta_{\text{eff}})} = O(1)$. With a confidence gap $\varepsilon - \eta$ bounded away from zero, this gives $B_e = O(1)$. Thus, a larger candidate pool requires correspondingly stronger control of ranking errors.

For a certified budget B_e and desired factor κ, write $L_\kappa = \log[\kappa\beta E B_e/((1-\varepsilon)a_0)]$. Applying equation 25 with the conservative margin δ_{eff} gives the sufficient drift tolerance

$$\Delta + 2\zeta \le \delta_0 - \sqrt{1 - \exp\{-2L_\kappa/(d-1)\}}. \tag{34}$$

The right side must be nonnegative, and this condition must hold jointly with candidate recall, availability, and the bound on μ. If the data domain or router changes enough to invalidate these conditions, the admission budget or threshold must be reconsidered.

Dimension and finite-horizon scaling. Theorem 1 gives

$$\frac{C^*_{\text{CIPHER}}}{C^*_{\text{natural}}} \le \frac{\beta E B_e}{(1-\varepsilon)a_0} e^{-G(d,\delta)}. \tag{35}$$

A growing separation follows if $G(d,\delta) - \log(\beta E B_e/a_0) \to \infty$ with fixed confidence. For bounded μ and fixed exposure and availability parameters, $d\delta(d)^2 \to \infty$ is sufficient. In contrast, when $\delta = c/\sqrt{d}$ for fixed $c > 0$, the normalized Gaussian representation gives $\sqrt{d}X_1 \Rightarrow \mathcal{N}(0,1)$ and hence $p_{c/\sqrt{d}}(d) \to 2[1 - \Phi(c)]$. The geometric exponent remains of constant order, so increasing dimension alone does not yield an increasing separation in this regime. These conclusions require availability to remain valid, potentially through a growing offered pool. Nominal hidden width or total parameter count does not by itself certify the effective spherical dimension of an anisotropic learned representation.

For H expert–micro-batch pairs, set $\varepsilon = \varepsilon_{\text{tot}}/H$, require $\eta < \varepsilon$, and recompute B_e. If the assumptions hold for each pair, a union bound gives

$$\Pr\left(\bigcup_{h=1}^{H} (F_h \cap A_h)\right) \le \sum_{h=1}^{H} \Pr(F_h \mid A_h) \Pr(A_h) \le \varepsilon_{\text{tot}}, \tag{36}$$

where F_h denotes retention failure and A_h signal availability. Independence between micro-batches is unnecessary. The guarantee covers failures to retain an offered signal; it neither guarantees signal availability nor establishes an unlimited-horizon training guarantee.

Table 7: Sufficient representation dimensions for the illustrative capacity-separation regimes under A1–A3.

Certified margin δ	Twofold reduction ($\kappa = 2$)	Tenfold reduction ($\kappa = 10$)
0.05	$d \geq 6{,}586$	$d \geq 7{,}872$
0.06	$d \geq 4{,}571$	$d \geq 5{,}464$
0.07	$d \geq 3{,}357$	$d \geq 4{,}012$

Illustrative capacity regimes. For $E = 256$, $\beta = 1$, $\varepsilon = 0.1$, $\eta = 0.01$, $a_0 = 0.9$, and $\mu = 1$, equation 7 gives $B_e = 6$ when $N_e \geq 6$. Substituting these values into equation 16 gives Table 7. These are conditional illustrations rather than fitted properties of the evaluated models. In particular, $N_e \geq 6$ establishes only the displayed budget; the offered pool must also satisfy equation 26, and the margin cannot be replaced by τ without calibration.

A.4.4 Signal retention and output fidelity

Signal retention does not imply that clipping leaves the MoE output unchanged. To quantify a separate local fidelity property, fix a token i with original routed mass $m_i > 0$, normalize its native gates as $\omega_e = g_{i,e}/m_i$, and write $\boldsymbol{v}_e = m_i f_e(\boldsymbol{x}_i; \theta_e)$. Then $\sum_e \omega_e = 1$ and the original routed output is $\sum_e \omega_e \boldsymbol{v}_e$. Assume $\|\boldsymbol{v}_e\| \leq V$, absorbing the original routed mass and any model-specific output scale into this bound. For a retained expert set $\mathcal{U}$ with mass $r = \sum_{e \in \mathcal{U}} \omega_e > 0$, the strict output with renormalized weights satisfies

$$\begin{aligned} \left\| \sum_e \omega_e \boldsymbol{v}_e - \sum_{e \in \mathcal{U}} \frac{\omega_e}{r} \boldsymbol{v}_e \right\| &\leq V \sum_{e \notin \mathcal{U}} \omega_e + V \left(\frac{1}{r} - 1 \right) \sum_{e \in \mathcal{U}} \omega_e \\ &= 2V(1 - r). \end{aligned} \tag{37}$$

If $r = 0$ and the routed branch returns zero, its deviation is at most V. For general modified weights ω'_e, including reroute, define both weight vectors over the union of their supports. The triangle inequality gives

$$\left\| \sum_e \omega_e \boldsymbol{v}_e - \sum_e \omega'_e \boldsymbol{v}_e \right\| \leq V \|\omega - \omega'\|_1. \tag{38}$$

Unchanged shared-expert and residual contributions cancel in this comparison. If the downstream loss is L_{loss}-Lipschitz in the block output, its one-step change is bounded by $2L_{\text{loss}}V(1-r)$ for strict and by $L_{\text{loss}}V\|\omega - \omega'\|_1$ for a general weight change. These statements hold at fixed expert outputs and fixed downstream computation. They do not bound the accumulation of parameter changes during optimization. Consequently, the capacity analysis motivates affinity-aware workload reduction, while training stability, specialization, and final model capability remain separate empirical evaluations.

A.5 Training Configurations of CIPHER-MoE

Table 8: Training configurations for MoE models.

Model	Nodes	Parallel configuration	Seq. length	GBS	MoE configuration
DeepSeek-V4-Pro	32	TP2/PP8/EP64/CP1/ETP1/VPP4 + SP	4K	256/8192	384 experts; learned Top-6
DeepSeek-V4-Flash	8	TP2/PP4/EP16/CP1/ETP1/DP16 + SP	8K	128	256 experts; learned Top-6; sqrt-softplus gate
DeepSeek-V3	16	TP4/PP16/EP4/CP1/DP4 + SP	8K	128	256 experts; Top-8 routing; sigmoid gate
GLM-5	16	TP2/PP8/EP32/CP1/ETP1/VPP2/DP16/EDP1	8K	128	256 experts; Top-8 routing; sigmoid gate

Table 8 summarizes the training and routing configurations used in our experiments. The affinity threshold τ specifies the minimum normalized token–expert affinity required to retain a candidate assignment; assignments with affinity below this threshold are filtered out. On DeepSeek-V4-Flash, we evaluated LocMoE with $\tau \in \{0.005, 0.01, 0.02, 0.03\}$ and found that $\tau = 0.01$ provided the most favorable overall performance. We therefore adopt $\tau = 0.01$ for LocMoE in all subsequent experiments to ensure a consistent configuration across models and datasets. Both variants of CIPHER-MoE use a capacity factor of 8; CIPHER-reroute additionally uses a fill-target factor of 4. The

capacity factor denotes the capacity limit as a multiple of the uniform expert load. For locality loss, λ_{loc} is set to 0.01 for both LocMoE and CIPHER-MoE. These settings remain robust across different models and datasets.