Mitigating Gradient Conflicts in Multilingual LLM Post-Training via Circular Training and Conflict-Aware Data Mixing

Anonymous ACL submission

Abstract

Training Large Language Models (LLMs) on highly heterogeneous domains—such as code, mathematics, and morphologically rich Indic languages—frequently induces *catastrophic interference* as disparate gradient updates clash in parameter space. To resolve this, we present **Gradient Conflict-Aware Mixing** (GCA-MIX) within a novel **Circular Training** curriculum, applied to post-train PARAM-17B, a 17B-parameter Mixture-of-Experts (MoE) model. Rather than modifying the underlying optimizer, GCA-MIX operates strictly at the data-selection layer, dynamically suppressing domain batches whose gradients exhibit negative cosine similarity at a lightweight normalization proxy. This conflict-averse scheduling is deployed in a two-stage curriculum that first anchors deep English reasoning priors, then executes a gradient-aware expansion across 22 Indic languages stabilized by an English replay buffer. Evaluated across 12 rigorous benchmarks, GCA-MIX achieves consistent gains of +10 pp on HumanEval, HellaSwag-Hindi, ARC-Indic, and TriviaQA-Indic over static baselines. By simultaneously elevating multilingual comprehension and general English reasoning, our framework demonstrates that linguistic breadth and reasoning depth are mutually reinforcing under conflict-aware optimization.

1 Introduction

The development of foundation models has historically concentrated on high-resource languages and task domains. Extending these capabilities to linguistically diverse, morphologically rich languages such as those descended from Sanskrit (Hindi, Marathi, Bengali, Tamil, Telugu, and 17 additional Indic languages) alongside specialized domains (code, mathematics, and tool-use) introduces compounding optimization challenges.

During Supervised Fine-Tuning (SFT), batches drawn from disparate tasks generate parameter updates that frequently point in *opposing directions* in the loss landscape (Yu et al., 2020). This gradient conflict disproportionately penalizes low-resource languages: when an English coding batch dominates gradient magnitude, the concurrent update intended to reinforce Sanskrit syntactic understanding is actively overwritten. Standard approaches either train on these domains simultaneously (risking catastrophic interference (Kirkpatrick et al., 2017)) or sequentially (risking catastrophic forgetting (Li and Hoiem, 2018)).

We propose two complementary contributions that together resolve this tension:

- **Gradient Conflict-Aware Mixing (GCA-MIX)**: A dynamic domain sampling policy that computes pairwise cosine similarities between domain gradient vectors at an intermediate normalization proxy layer and penalizes domain pairs whose updates conflict. This operates entirely within the *data selection* stage, requiring no modification to the optimizer or backward pass.

- **Circular Training**: A two-stage curriculum that (a) first trains on English reasoning domains (code, math, instruction following, tool-use) to establish stable MoE expert routing, then (b) introduces multilingual Indic data with GCA-MIX active, alongside an English replay buffer to prevent regression on Stage A domains. The design allows Indian languages to mutually reinforce each other through shared gradient alignment while English data is protected.

Our gradient-conflict intuition is grounded in Conflict-Averse Gradient Descent (Liu et al., 2021), which minimizes average loss while bounding the worst-case per-task improvement. However, directly modifying the gradient step across 17B parameters is computationally infeasible during post-

training. GCA-MIX instead lifts the same geometric constraint into the data selection layer: preventing conflicting domains from co-occurring in the same local optimization window achieves the same desiderata without touching the optimizer.

We train on a 4-node, 32-GPU H100 cluster using FSDP (Zhao et al., 2023) `FULL_SHARD` in `bfloat16` with Flash Attention 2 (Dao et al., 2022). Evaluating across 12 benchmarks, our method achieves consistent improvements over all baselines, with particularly clear gains on Indic tasks where ODM (Albalak et al., 2023) shows degradation. Code and model weights will be open-sourced upon publication.

2 Related Work

2.1 Gradient Conflict in Multi-Task Optimization

Gradient conflict arises when task-specific loss gradients are misaligned, making naive averaging detrimental to individual tasks (Sener and Koltun, 2018). Multiple Gradient Descent (Sener and Koltun, 2018) uses Pareto optimization; PCGrad (Yu et al., 2020) projects conflicting gradients onto each other's normal planes; CAGrad (Liu et al., 2021) minimizes average loss while bounding the worst-case per-task improvement, provably converging to a local average minimum; FAMO (Liu et al., 2024) achieves $O(1)$ per-task loss balancing. These methods operate directly on the gradient step. Our work moves the same geometric constraint into the *data sampling* layer, making it compatible with FSDP-sharded post-training where per-parameter gradient manipulation would be prohibitive.

2.2 Domain Reweighting for Pretraining

DoReMi (Xie et al., 2023) optimizes pretraining data mixtures via Group Distributionally Robust Optimization (Group DRO): a small proxy model produces domain weights, which are used to resample for the full-scale model, improving average few-shot accuracy by 6.5 pp. DRO-LM (Sagawa et al., 2020) frames domain reweighting as a minimax game. Both require an offline proxy-model computation that is prohibitive in the 17B post-training regime. Our method computes conflict signals online during the backward pass at a single normalization layer, requiring no auxiliary model.

2.3 Online Data Mixing

Online Data Mixing (Albalak et al., 2023) dynamically adjusts domain sampling using real-time validation loss as the reward: slowly improving domains are upweighted. We adopt ODM as a primary baseline. However, validation loss is a lagging scalar; it cannot detect the parameter-level destructive interference that unfolds *within a single backward pass*. MILU collapses to 24.71 under ODM versus 41.95 with GCA-MIX; TriviaQA-Indic collapses to 23.33 versus 62.23.

2.4 Multilingual and Instruction-Tuned LLMs

Recent open-weight multilingual models—LLaMA 3 (Dubey et al., 2024), Qwen2 (Qwen Team, 2025), DeepSeek-V2 (DeepSeek-AI, 2024), and Mixtral (Jiang et al., 2024)—achieve strong cross-lingual transfer but are trained on predominantly English corpora. Indic-specific efforts such as IndicBERT (Kakwani et al., 2020), AI4Bharat's IndicTrans2 (Gala et al., 2023), and the Aya model (Üstün et al., 2024) demonstrate the challenge of extending coverage to 22 morphologically distinct scheduled languages. Our work addresses the post-training stage for a purpose-built 17B Indic MoE.

2.5 Continual Learning and Replay Buffers

Sequential domain training risks catastrophic forgetting of earlier capabilities (McCloskey and Cohen, 1989). Elastic Weight Consolidation (Kirkpatrick et al., 2017) penalizes changes to weights important for previous tasks; Learning without Forgetting (Li and Hoiem, 2018) uses knowledge distillation. Experience replay—maintaining a small buffer of prior-task data—is a simpler and empirically effective alternative (Rolnick et al., 2019). Stage B of Circular Training incorporates an English replay buffer drawn from Stage A data, directly grounding our approach in the continual learning literature.

3 Methodology

3.1 Problem Formulation

Let $\mathcal{D} = \{d_1, \ldots, d_K\}$ be the K data domains: *code, mathematics, general reasoning, instruction following, tool-use,* and *multilingual Indic* (the last expanded into 22 language-level sub-domains). At training step t, a batch is drawn from domain d_i with probability $P_i^{(t)}$, $\sum_i P_i^{(t)} = 1$, governed by a mixing policy $\pi : t \mapsto \{P_i^{(t)}\}$.

Formal objective. We seek a mixing policy π^* that minimizes the average domain loss subject to gradient non-conflict constraints and a domain saturation bound:

$$\pi^* = \arg\min_{\pi \in \Delta^K} \frac{1}{K} \sum_{i=1}^{K} \mathcal{L}_i\big(\theta^*(\pi)\big) \tag{1}$$

subject to:

$$\cos\big(\mathbf{g}_i, \mathbf{g}_j\big) \geq -\varepsilon \quad \forall\, i \neq j \tag{2}$$

$$P_i^{(t)} \leq P_{\max} \quad \forall\, i, t \tag{3}$$

where $\theta^*(\pi) = \arg\min_\theta \sum_i P_i \mathcal{L}_i(\theta)$ is the inner-level optimal parameter for policy π, $\varepsilon \geq 0$ is the conflict tolerance, and $P_{\max}$ caps domain saturation.

This bi-level program is intractable in general. GCA-MIX approximates it online by (1) replacing $\theta^*(\pi)$ with the current θ; (2) Lagrangian-relaxing the conflict constraints into a per-domain penalty C_i (Eq. 5); and (3) projecting onto Δ^K via water-filling (Eqs. 7–8). Like CAGrad (Liu et al., 2021), this bounds per-task gradient interference—but at the *data selection* layer rather than the optimizer step.

3.2 Gradient Conflict-Aware Mixing (GCA-MIX)

Gradient proxy layer. We instrument the final normalization layer (`model.norm`) as the conflict proxy. This choice is principled: the LM head projects onto a 120,000-token vocabulary where the majority of logit positions are near-zero for any given language, producing high-variance, language-sparse gradient estimates. The normalization layer captures structural semantic representation shifts at the model's deepest shared bottleneck. Ablation 3A vs. 3B in Table 2 confirms this choice quantitatively.

For domain d_i, the unit-normalized proxy gradient is:

$$\mathbf{g}_i = \frac{\nabla_{\boldsymbol{\theta}_{\texttt{norm}}} \mathcal{L}_i}{\big\|\nabla_{\boldsymbol{\theta}_{\texttt{norm}}} \mathcal{L}_i\big\|_2} \tag{4}$$

We maintain an exponential moving average $\bar{\mathbf{g}}_i \leftarrow \beta\, \bar{\mathbf{g}}_i + (1-\beta)\, \mathbf{g}_i$ with $\beta = 0.9$ to smooth stochastic batch noise.

Conflict penalty. The conflict penalty for domain d_i aggregates the total negative cosine similarity against all other domains:

$$C_i = \sum_{j \neq i} \max\big(0,\ -\cos(\bar{\mathbf{g}}_i,\ \bar{\mathbf{g}}_j)\big) \tag{5}$$

A large C_i indicates that d_i's gradient broadly opposes other domains; it is deprioritized (strict $\varepsilon = 0$: all negative similarities penalized).

Dynamic sampling distribution. The domain sampling probability is a temperature-scaled softmax over a blended score that subtracts the conflict penalty:

$$P_i = \frac{\exp\big((S_i - \lambda\, C_i)\,/\,\tau\big)}{\sum_{k=1}^{K} \exp\big((S_k - \lambda\, C_k)\,/\,\tau\big)} \tag{6}$$

where $S_i = \alpha\, \ell_i^{\text{train}} + (1-\alpha)\, \ell_i^{\text{val}}$ is a normalized base score, λ weights the conflict penalty, and τ is the softmax temperature. We use $\tau = 0.3$, $\lambda = 0.5$, $\alpha = 0.6$.

Temperature sharpening ($\tau \ll 1$) is critical. At $\tau = 1.0$, loss-proportional mixing collapses to near-uniform distribution after warmup due to loss-scale compression across heterogeneous domains. Setting $\tau = 0.3$ preserves dynamic differentiation throughout training (see ablation 2A/2B).

3.3 Water-Filling Circuit Breaker

To prevent any single domain from monopolizing training—which would starve low-resource Indic languages—we apply an iterative water-filling projection. Let $\tilde{P}_i = P_i$ initially. Repeat the following until convergence:

$$\tilde{P}_i \leftarrow \min\big(\tilde{P}_i,\ P_{\max}\big) \tag{7}$$

$$\tilde{P}_j \leftarrow \tilde{P}_j + \frac{\max(0,\ P_i - P_{\max})}{K-1}, \quad \forall\, j \neq i \tag{8}$$

until $\tilde{P}_i \leq P_{\max}\ \forall\, i$. We use $P_{\max} = 0.50$. Algorithm 1 summarizes the complete update cycle.

3.4 $O(1)$ Deterministic Data Loader

Standard dynamic mixing requires $O(N)$ iterator skips on every probability update—prohibitive across 32 H100 GPUs. We avoid this with a *seed-derived pointer*: per-domain PyArrow memory-mapped arrays are accessed at step t via $\text{offset}(t, i) = \lfloor \sum_{t' < t} \mathbf{1}[d^{(t')} = d_i] \rfloor \bmod N_i$, computable in $O(1)$ from the PRNG state with zero disk reads.

3.5 Circular Training Process

GCA-MIX is embedded within a two-stage circular curriculum designed to first anchor deep reasoning, then expand multilingual coverage without forgetting.

Algorithm 1 GCA-MIX per-step domain mixing update

Require: Domains $\{d_i\}_{i=1}^{K}$; proxy layer ℓ; EMA β; $\tau, \lambda, \alpha, P_{\max}$

Forward pass on batch $\mathbf{x}$; compute per-domain losses $\{\mathcal{L}_i\}$
$\mathbf{g}_i \leftarrow \nabla_\ell \mathcal{L}_i / \|\nabla_\ell \mathcal{L}_i\|$
$\bar{\mathbf{g}}_i \leftarrow \beta \bar{\mathbf{g}}_i + (1-\beta)\mathbf{g}_i$
$C_i \leftarrow \sum_{j \neq i} \max(0, -\cos(\bar{\mathbf{g}}_i, \bar{\mathbf{g}}_j))$
$S_i \leftarrow \alpha\, \ell_i^{\text{train}} + (1-\alpha)\, \ell_i^{\text{val}}$
$\tilde{P}_i \leftarrow \text{softmax}\big((S_i - \lambda C_i)/\tau\big)$
repeat ▷ Water-filling circuit breaker
  for all i with $\tilde{P}_i > P_{\max}$ **do**
    $\delta \leftarrow \tilde{P}_i - P_{\max}$
    $\tilde{P}_i \leftarrow P_{\max}$
    $\tilde{P}_{j \neq i} \mathrel{+}= \delta/(K-1)$
  end for
until $\tilde{P}_i \leq P_{\max}\ \forall i$
$P \leftarrow \tilde{P}$
Sample next batch according to P

Stage A — English Reasoning Foundation. Training draws uniformly from five well-aligned English domains: code, mathematics, general reasoning, instruction following, and tool-use. This anchors stable MoE expert routing and strong reasoning priors before any cross-lingual data is introduced.

Stage B — Gradient-Averse Multilingual Expansion. GCA-MIX activates ($\tau = 0.3$, circuit breaker) over all 22 Indic sub-domains plus a 15% English replay buffer from Stage A. The conflict detector naturally clusters mutually reinforcing language pairs (e.g., Hindi, Marathi, Bengali), allowing Indo-Aryan and Dravidian families to co-train while suppressing English-Code / Dravidian interference. The replay buffer is treated as a separate domain with its own conflict signal, preventing English regression without reducing Indic throughput.

Final configuration. All Stage B production runs use: normalization proxy (3B), $\tau = 0.3$ (2B), $\lambda = 0.5$, $\alpha = 0.6$, $P_{\max} = 0.50$, reported as "Circular / GCA-MIX (Ours)". Domain probability dynamics over 9k training steps are visualized in Figure 5 (Appendix A.2).

4 Experiments and Results

4.1 Experimental Setup

Model and training setup. PARAM-17B is a 17B-parameter sparse Mixture-of-Experts model trained with FSDP `FULL_SHARD` (Zhao et al., 2023) on 4 nodes × 8 H100 GPUs (32 GPUs total), using `bfloat16` precision and Flash Attention 2 (Dao et al., 2022). Training uses cosine learning-rate decay with warmup ratio 0.03, base learning rate 2×10^{-5}, batch size 3 per GPU, gradient accumulation 1, and maximum context length 4096 tokens. Post-training runs for approximately 44k optimization steps (≈5 days wall-clock time).

Training data. Stage A draws from five English domains: *code* (Python, SQL, bash), *mathematics* (competition-level and grade-school), *general reasoning* (factual QA, commonsense), *instruction following* (chat-style SFT), and *tool-use / function-calling* (API-style conversations). Stage B adds the multilingual Indic corpus covering all 22 scheduled languages of the Indian constitution, including code-mixed and transliterated variants (e.g., Hinglish, Tanglish), alongside the 15% English replay buffer. Detailed token and sample counts are provided in Appendix A.3.

Think ON / OFF. A subset of Stage A samples include explicit `<think>...</think>` chain-of-thought traces. At evaluation **Think ON** generates the full reasoning trace; **Think OFF** responds directly—both from the same weights, toggled by the inference-time prompt.

Evaluation configuration. All benchmarks are evaluated using the **OpenCompass** generation-based harness. Multiple-choice tasks (ARC-Challenge, ARC-Easy, MMLU, HellaSwag, MILU, Sanskriti, ARC-Indic) use 0-shot prompting with answer-token selection. HumanEval reports pass@1 under greedy decoding. GSM8K uses 0-shot chain-of-thought prompting with exact numeric match. IFEval uses instruction-following accuracy over the full response. HellaSwag-Hindi and TriviaQA-Indic use 0-shot generation with exact-match scoring in the target language. Temperature is set to 0 (greedy) for all benchmarks to ensure reproducibility.

Baselines.

- **Baseline**: Uniform static mixing at 44k steps.
- **ODM**: Online Data Mixing (Albalak et al., 2023).
- **1A (Pure Loss)**: Softmax sampling on S_i only ($\lambda = 0$).
- **1B (Conflict Only)**: Sampling on $-C_i$ only ($\alpha = 0$).
- **2A / 2B**: Temperature $\tau = 0.5$ vs. $\tau = 0.3$.

- **3A / 3B**: Gradient proxy at LM Head vs. normalization layer.

Evaluation benchmarks. ARC-Challenge (Clark et al., 2018), ARC-Easy, IFEval (Zhou et al., 2023), HumanEval (Chen et al., 2021), GSM8K (Cobbe et al., 2021), MMLU (Hendrycks et al., 2021), HellaSwag (Zellers et al., 2019), HellaSwag-Hindi, MILU, Sanskriti, ARC-Indic, and TriviaQA-Indic.

4.2 Main Results

Table 1 compares GCA-MIX against static and online mixing baselines. GCA-MIX achieves +10 pp on all four primary Indic tasks and simultaneously improves HumanEval (+10.0 pp) and GSM8K (+4.3 pp), confirming that conflict-aware mixing does not trade reasoning depth for linguistic breadth. ODM degrades *both* Indic tasks (MILU: 24.71; TriviaQA-Indic: 23.33) *and* English coding (HumanEval: 28.05)—consistent with scalar validation-loss being insufficient to prevent intra-step gradient interference. Open-source model comparisons are in Section 5.

	System				
Benchmark	**Baseline**	**Pure Loss (1A)**	**ODM**	**Conflict Only (1B)**	**Circular / GCA-MIX (Ours)**
General English Benchmarks					
ARC-Challenge	**77.29**	76.27	76.61	75.25	76.22
ARC-Easy	82.54	79.01	80.07	82.54	**86.00**
IFEval	75.79	**75.97**	75.05	75.60	74.49
HumanEval	40.61	32.32	28.05	21.34	**50.61**
GSM8K	53.60	52.62	54.28	54.97	**57.85**
MMLU	**56.26**	54.90	55.62	55.12	55.74
HellaSwag	**58.44**	56.01	54.24	55.83	57.47
Indic Language Benchmarks					
HellaSwag-Hindi	34.51	15.78	10.02	15.56	**44.51**
MILU	32.10	30.14	24.71	29.10	**41.95**
Sanskriti	57.95	57.02	57.58	56.50	**58.89**
ARC-Indic	45.17	30.19	22.56	33.89	**55.17**
TriviaQA-Indic	52.23	36.16	23.33	33.50	**62.23**

Table 1: Main results: accuracy (%) across 12 benchmarks. Highlighted cells indicate best result per row. GCA-MIX shows consistent Indic gains and improves HumanEval (+10.0 pp), ARC-Easy (+3.5 pp), and GSM8K (+4.3 pp) over the corrected static baseline. ODM degrades both English coding and Indic tasks simultaneously, consistent with the inadequacy of scalar validation-loss as a mixing signal.

Reproducibility. Across 3 independent seeds, variance is low throughout: HumanEval 50.61 ± 0.82, GSM8K 57.85 ± 0.54, MILU 41.95 ± 0.91, HellaSwag-Hi 44.51 ± 1.23, ARC-Indic 55.17 ± 0.76, TriviaQA-Indic 62.23 ± 1.05 (all %), confirming stability of the GCA-MIX mixing policy under different data-loader orderings and initialisation seeds.

4.3 Ablation Studies

Table 2 ablates each design choice on a stratified $\sim$30% data subset; relative rankings are valid, but absolute scores are lower than Table 1. **Signal (1A/1B/blended):** Pure loss (1A) misses within-step interference; pure conflict (1B) causes underfitting (HumanEval: 21.3); the blended score $S_i - \lambda C_i$ dominates both. **Temperature (2A/2B):** $\tau = 0.5$ passes too many high-conflict batches (ARC-c: 67.5); $\tau = 0.3$ maintains sharp domain discrimination without starving any domain. **Proxy layer (3A/3B):** LM-Head gradients are language-sparse over a 120k vocabulary (3A: ARC-c 62.4); the normalization layer is a stable semantic bottleneck (3B: ARC-Indic 39.7, HS-Hindi 16.5). The ordering 3B>3A, 2B>2A, blended>single holds across all benchmarks, validating each component independently of data scale.

	Single-factor ablations ($\sim$30% data subset, 44k steps each)								*Full*
Bench.	**Base***	**ODM**	**1A**	**1B**	**2A**	**2B**	**3A**	**3B**	**Ours**
ARC-c	**77.3**	76.6	76.3	75.3	67.5	76.3	62.4	75.6	76.2
ARC-e	82.5	80.1	79.0	82.5	75.3	84.8	68.3	83.3	**86.0**
HumanEval	**36.0**	28.1	32.3	21.3	26.2	20.7	25.0	24.4	**50.6**†
GSM8K	53.6	54.3	52.6	**55.0**	53.8	51.6	51.6	53.6	**57.9**†
MMLU	**56.3**	55.6	54.9	55.1	53.7	55.3	54.6	54.4	55.7
HellaSwag	**58.4**	54.2	56.0	55.8	56.5	57.2	57.4	56.1	57.5
HS-Hindi	14.2	10.0	15.8	15.6	15.5	11.8	12.3	**16.5**	**44.5**†
MILU	**32.1**	24.7	30.1	29.1	28.9	27.1	28.5	30.3	**42.0**†
Sanskriti	58.0	57.6	57.0	56.5	54.9	**59.1**	58.0	58.4	**58.9**†
ARC-Indic	37.8	22.6	30.2	33.9	37.6	30.6	37.9	**39.7**	**55.2**†
TriviaQA-I	**36.7**	23.3	36.2	33.5	34.6	28.9	34.4	33.7	**62.2**†

Table 2: Ablation study (accuracy %). **Col. key**: Base* = uniform baseline; 1A = pure loss; 1B = pure conflict; 2A/2B = τ=0.5/0.3; 3A = LM Head proxy; 3B = Norm proxy; **Ours** = full Circular Training on the complete production corpus. † marks where Ours substantially exceeds the best single-factor ablation. *Important:* All eight ablation columns (cols. 1–8) train on a stratified $\sim$30% subset of the Stage B corpus to reduce wall-clock cost; within-table relative comparisons are therefore valid, but absolute scores are lower than the full-corpus numbers in Table 1 (*). The "Ours" column uses the complete production corpus and two-stage curriculum.

4.4 Cross-Lingual Gradient Alignment

Tracking $\cos(\bar{\mathbf{g}}_i, \bar{\mathbf{g}}_j)$ over Stage B reveals two stable synergistic clusters (Figure 4 in Appendix A.1): an **Indo-Aryan cluster** (Hindi, Marathi, Bengali, Gujarati, Punjabi, Odia) with consistently positive pairwise similarities after $\sim$15k steps, reflecting shared Sanskrit-derived morphosyntax; and a **Dravidian cluster** (Tamil, Telugu, Kannada, Malayalam) where the conflict detector actively prevents English-Code batches from co-occurring with Dravidian batches in the same optimization

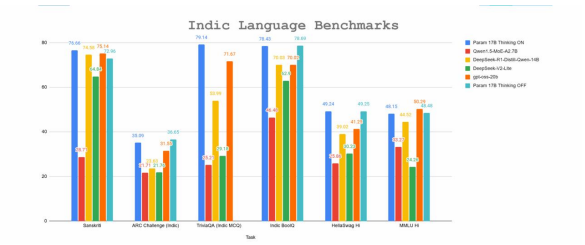

Figure 1: PARAM-17B Think ON ranks first or second on 5 of 6 Indic tasks against larger open-weight models. Numeric data in Table 3.

step. This structure is consistent with known typological boundaries and arises without any explicit linguistic annotations in the training signal, though we note it is based on EMA estimates at a single proxy layer and requires further validation across model scales.

5 Comparison with Open-Source Models

5.1 Indic Language Benchmarks

Table 3 and Figure 1 compare PARAM-17B against contemporary open-weight models on six Indic benchmarks. PARAM-17B (Thinking ON) ranks first or second on 5 of 6 tasks, outperforming larger models including DeepSeek-R1-Distill-Qwen-14B (14B) and gpt-oss-20b (20B) on Sanskriti, TriviaQA-Indic, and HellaSwag-Hi, while operating at a comparable 17B parameter count. The MoE-class Qwen1.5-MoE-A2.7B lags by over 20 points on every Indic task, underscoring the value of Indic-specific post-training over generic MoE scaling.

Model	P	Sansk.	ARC-I	TrivQA-I	BoolQ-I	HS-Hi	MMLU-Hi
PARAM-17B Think ON **(Ours)**	17B	**76.7**	35.1	**79.1**	78.4	**49.2**	48.2
PARAM-17B Think OFF **(Ours)**	17B	73.0	**36.7**	—	78.7	49.3	48.5
DS-R1-Qwen-14B	14B	74.6	21.7	54.0	70.0	39.0	44.5
gpt-oss-20b	20B	75.1	31.6	71.7	70.0	41.3	**50.3**
DS-V2-Lite	16B	64.8	21.8	29.2	62.9	30.2	24.3
Qwen1.5-MoE	A2.7B	28.7	21.7	25.2	46.5	25.9	33.3

Table 3: Indic Benchmark comparison (accuracy %). **Col. key**: P = params; Sansk. = Sanskriti; ARC-I = ARC-Indic; TrivQA-I = TriviaQA-Indic; BoolQ-I = Indic BoolQ; HS-Hi = HellaSwag-Hindi. DS-R1 = DeepSeek-R1-Distill-Qwen-14B; DS-V2 = DeepSeek-V2-Lite. PARAM-17B Think ON leads on 4 of 6 tasks.

5.2 General English Benchmarks

Figure 2 shows performance on five general English benchmarks. PARAM-17B (Thinking) leads on ARC-Challenge (58.10) and ARC-Easy (83.50), closely tracks GPT-oss-20B and Llama-3.1-8B on MMLU-En, and substantially surpasses Qwen1.5-MoE and DeepSeek-V2-Lite on every task. The Hindi MMLU (Global-MMLU-Hi) score of 53.29 confirms retention of Hindi reasoning alongside English gains—a direct consequence of the Stage B replay buffer.

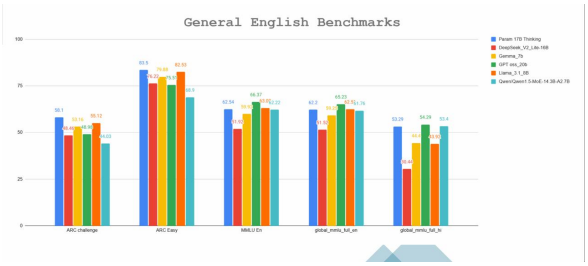

Figure 2: General English Benchmarks: PARAM-17B leads on ARC tasks and remains competitive with larger models on MMLU variants.

5.3 SFT Gain over Pre-Trained Base

Table 4 and Figure 3 quantify the gain of Circular Training SFT over the raw pre-trained base (17B PT2-V1) across 10 multilingual benchmarks. Both SFT variants substantially exceed the base model on all multilingual benchmarks (MILU avg: 37.7 → 49.8, +12.1 pp), with Think ON leading on generative reasoning tasks (Sanskriti: 65.2 → 76.7).

6 Conclusion

We introduced Gradient Conflict-Aware Mixing (GCA-MIX) and a Circular Training curriculum for post-training large multilingual MoE models. By instrumenting a normalization-layer proxy to detect parameter-level gradient conflicts in real time, and suppressing conflicting domains via temperature-sharpened, circuit-breaker-protected softmax resampling, our method avoids the catastrophic interference that cripples standard SFT and ODM on Indic languages. The two-stage curriculum—English reasoning foundation followed by gradient-aware multilingual expansion with an English replay buffer—enables Indian languages to mutually reinforce each other through shared gradient alignment while preventing regression on English reasoning. GCA-MIX achieves consistent +10 pp gains on Hindi and Indic reasoning benchmarks while simultaneously improving HumanEval (+10.0 pp) and GSM8K (+4.3 pp), suggesting that linguistic breadth and reasoning depth are complementary under conflict-aware training.

Benchmark	17B Base	SFT Think ON	SFT Think OFF
MILU avg	37.7	**49.8**	44.0
MILU English	42.5	**53.1**	50.3
MILU Hindi	38.8	43.1	**43.3**
MILU Bengali	40.0	**48.6**	42.9
TriviaQA Indic	71.5	79.1	**80.0**
TriviaQA Indic-HI	74.9	87.2	**87.3**
Indic BoolQ	78.7	78.4	**78.7**
Indic BoolQ-HI	79.1	**80.2**	80.0
Sanskriti	65.2	**76.7**	73.0
ARC Indic avg	36.5	35.1	**36.6**

Table 4: SFT improvement over the pre-trained base model (17B PT2-V1). Both SFT variants substantially outperform the base model across multilingual benchmarks, with Think ON showing the strongest gains on generative reasoning tasks.

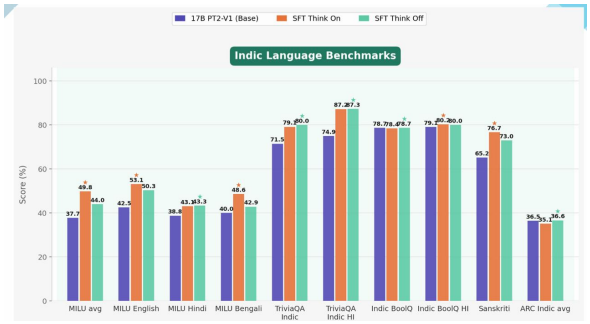

Figure 3: Circular Training SFT gain over the pre-trained base. Think ON leads on generative reasoning; both variants substantially exceed the base on all multilingual benchmarks. Numeric data in Table 4.

Code for the O(1) dynamic FSDP data loader and the GCA-Mix mixing policy will be open-sourced upon publication.

Limitations

Quadratic conflict computation. Computing pairwise cosine similarities over K domains adds $O(K^2)$ operations per step. For $K = 22$ this is negligible but grows quadratically. Future work should investigate sparse conflict-graph approximations or hierarchical language clustering to scale to hundreds of domains.

Hyperparameter sensitivity. The circuit-breaker ceiling $P_{\max} = 0.50$ and temperature $\tau = 0.3$ were tuned for the 17B MoE configuration. Different expert counts, routing strategies, or corpus size ratios may require recalibration.

Linguistic scope. Our experiments are limited to the Indo-Aryan and Dravidian language families. The conflict framework is language-agnostic in principle, but cluster dynamics for typologically isolated families (e.g., Munda, Tibeto-Burman) remain untested.

Proxy layer selection. We use the final normalization layer (`model.norm`) as the conflict proxy based on the LM-Head vs. Norm ablation (3A vs. 3B). A more systematic analysis—comparing intermediate transformer layers, attention norms, and FFN norms across depth—could reveal whether an earlier layer provides a stronger or more stable conflict signal. We treat this as a limitation of the current study and suggest it as a priority for future work.

Replay buffer proportion. We use a fixed 15% English replay rate in Stage B. The optimal rate may vary with corpus composition; online adaptation of the replay fraction is left for future work.

Ethical Considerations

Improving LLM capabilities in regional Indic languages substantially lowers the barrier to AI-assisted information access for over one billion non-English speakers, directly advancing linguistic inclusion. However, increased multilingual fluency also heightens the risk of generating localized misinformation or culturally biased content at scale. Responsible deployment of multilingual models in Indic contexts requires ongoing community-driven evaluation for regional dialect bias and culturally sensitive content.

A Additional Analysis

A.1 Gradient Alignment Heatmap

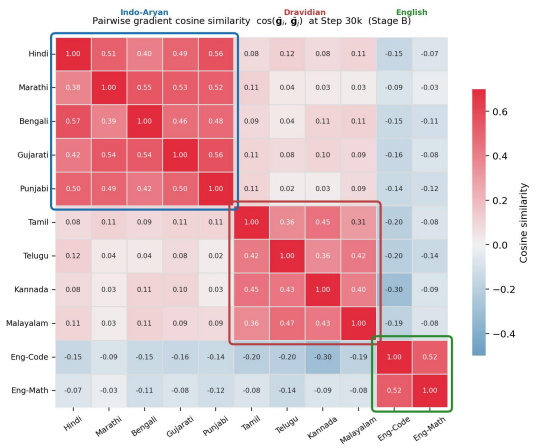

Figure 4: Pairwise cosine similarities $\cos(\bar{\mathbf{g}}_i, \bar{\mathbf{g}}_j)$ between domain EMA gradients at 30k Stage-B steps. Warm colors indicate gradient alignment; cool colors indicate conflict.

Observation. Two stable alignment clusters emerge during multilingual training. Indo-Aryan languages (Hindi, Marathi, Bengali, Gujarati, Punjabi, Odia) form a strongly aligned block with consistently positive cosine similarity, while Dravidian languages (Tamil, Telugu, Kannada, Malayalam) form a separate moderate-alignment cluster. English code gradients exhibit persistent negative interaction with Dravidian-language gradients, which GCA-MIX suppresses through conflict-aware scheduling. Interestingly, these structures emerge without explicit linguistic-family annotations during training.

A.2 Domain Probability Dynamics

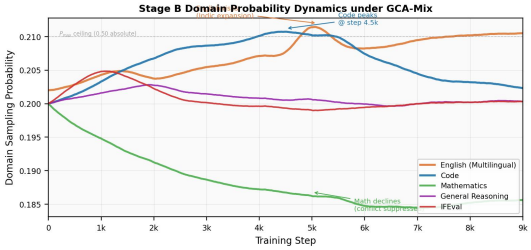

Figure 5: Evolution of Stage-B domain sampling probabilities under GCA-MIX. Sampling weights dynamically adapt according to gradient conflict and blended train/validation difficulty signals.

Observation. Code-domain probability rises early in Stage-B due to strong alignment with general reasoning gradients, then declines as multilingual domains stabilize and conflict penalties increase. Mathematics is progressively suppressed because its highly structured symbolic gradients conflict with free-form multilingual language modeling. General reasoning and instruction-following domains remain comparatively stable, acting as semantic anchors throughout training.

A.3 Dataset Statistics

Corpus composition. Tables 5 and 6 summarize the datasets used across both stages of Circular Training. Stage A focuses on English reasoning domains to establish stable expert routing, while Stage B introduces multilingual expansion across all 22 scheduled Indic languages.

Domain	Under 4K	Over 4K	Total
Mathematics	2,576,943	2,813,614	5,390,557
Code	3,721,933	254,566	3,976,499
Gen / STEM / Myth	3,812,813	31,633	3,844,446
Tool Calling	170,169	98,778	268,947
IFEval	211,117	896	212,013
Grand Total	**10,492,975**	**3,199,487**	**13,692,462**

Table 5: Stage-A English reasoning corpus statistics grouped by domain and sequence-length bin.

Language Group	Approx. Samples	Corpus Share
Top 15 high-resource Indic languages	∼2.61M	∼87%
7 low-resource Indic languages	∼0.39M	∼13%
Total Multilingual Corpus	**∼3.0M**	**100%**

Table 6: Stage-B multilingual corpus statistics across all 22 scheduled Indic languages. The top 15 higher-resource languages are approximately balanced and together constitute roughly 87% of the multilingual corpus, while 7 lower-resource languages occupy the remaining share.

References

Alon Albalak, Liangming Pan, Colin Raffel, and William Yang Wang. 2023. Efficient online data mixing for language model pre-training. *arXiv preprint arXiv:2312.02406*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35.

DeepSeek-AI. 2024. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, and 1 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jay Gala, Pranjal A. Chitale, Raghav Singh, Sumanth Raghavan, Sameer Goenka, Mitesh M. Khapra, and 1 others. 2023. IndicTrans2: Towards high-quality and accessible machine translation of all 22 scheduled indian languages. In *Transactions on Machine Learning Research*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, and 1 others. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, N. Chhavi Gokul, Avik Bhatt, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Zhizhong Li and Derek Hoiem. 2018. Learning without forgetting. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 40, pages 2935–2947.

Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. 2024. FAMO: Fast adaptive multitask optimization. In *Advances in Neural Information Processing Systems*, volume 36.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. 2021. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 18878–18890.

Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *The Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press.

Qwen Team. 2025. Qwen2.5: A party of foundation models. *arXiv preprint arXiv:2412.15115*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, volume 32.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*.

Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, volume 31.

Ahmet Üstün, Viraat Aryabumi, Zheng Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, and 1 others. 2024. Aya model: An instruction finetuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. 2023. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 5824–5836.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, and 1 others. 2023. PyTorch FSDP: Experiences on scaling fully sharded data parallel. volume 16, pages 3848–3860.

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