ICML 2026
Seoul, Korea

DTop-*p* MoE: Sparsity-Controlled Dynamic Top-*p* MoE for Foundation Model Pre-training

Can Jin[*1] Hongwu Peng[*2] Mingcan Xiang[23] Qixin Zhang[4] Xiangchi Yuan[5] Amit Hasan[2] Ohi Dibua[2] Yifan Gong[2] Yan Kang[2†] Dimitris N. Metaxas[1†]

[1]Rutgers University · [2]Adobe Research · [3]UMass Amherst · [4]Nanyang Technological University · [5]Georgia Institute of Technology | [*]Equal Contribution · [†]Equal Advising

PMLR 306, 2026
can.jin@rutgers.edu
Work done during internship at Adobe Research

MOTIVATION

Sparse **Mixture-of-Experts (MoE)** scales model capacity by activating only a few experts per token, decoupling parameters from compute cost. Two routing schemes dominate pre-training:

- **Top-*k*** — fixed number of experts per token; ignores token difficulty & layer needs.
- **Fixed Top-*p*** — selects experts until cumulative probability exceeds threshold *p*.

PROBLEM: FIXED TOP-*P* IS UNSTABLE

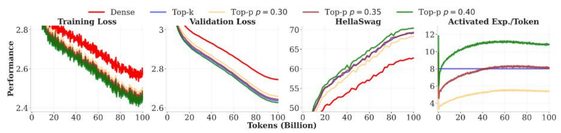

Figure 2. Performance comparison of the Dense model, Top-*k* MoE, and fixed-threshold Top-*p* MoE ($p \in \{0.30, 0.35, 0.40\}$). Top-*p* yields only marginal gains over Top-*k* MoE at comparable activation levels, while the number of activated experts fluctuates unpredictably.

- ❶ **Uncontrolled compute:** activated-expert count fluctuates unpredictably — incompatible with strict pre-training budgets.
- ❷ **Hypersensitive to *p*:** p=0.40 over-activates (>12 experts); p=0.35 only matches Top-*k* — gains are marginal.

KEY INSIGHT

Top-*p* is inherently *adaptive* — confident tokens use fewer experts, uncertain tokens recruit more — but the threshold receives **no gradient** (it only binarizes the mask), so it cannot be learned by SGD.

💡 **Idea:** treat target sparsity as a control *setpoint* and make the threshold effectively learnable, so average activated experts converge to a budget *T*.

CONTRIBUTIONS

- **Analysis** showing fixed-threshold Top-*p* gives only marginal gains over Top-*k* with uncontrolled cost.
- **DTop-*p* MoE**: a PI-controlled, sparsity-controllable dynamic Top-*p* with Dynamic Routing Normalization (+ a layer-wise variant).
- **Comprehensive study** on NLP & CV: better performance and scaling at matched FLOPs.

METHOD — DTOP-*P* MOE

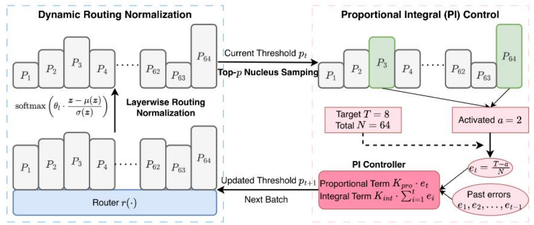

Figure 1. Overview of DTop-*p* MoE. We employ a Proportional-Integral (PI) controller to dynamically adjust the global probability threshold, aligning the number of activated experts with a target value. The Dynamic Routing Normalization modules layer-wise logit distributions to support varying sparsity needs, enabling distinct patterns across network depths under the global threshold.

① PI CONTROLLER — LEARNABLE SPARSITY

A Proportional-Integral controller adjusts the global threshold *p* between batches from the sparsity error $e_t=(T-a_t)/N$:

$$p_{t+1} = p_0 + \underbrace{K_{pro} \cdot e_t}_{\text{Proportional}} + \underbrace{K_{int} \cdot \sum_{i=1}^{t} e_i}_{\text{Integral}} \qquad (6)$$

The proportional term reacts to deviation; the integral term removes steady-state bias, driving activated experts $a_t \to$ target *T*.

② DYNAMIC ROUTING NORMALIZATION

$$\boldsymbol{P}(\boldsymbol{x}) = \text{softmax}\left(\theta_l \cdot \frac{\boldsymbol{z} - \mu(\boldsymbol{z})}{\sigma(\boldsymbol{z})}\right), \text{with } \boldsymbol{z} = \boldsymbol{W}\boldsymbol{x} \quad (7)$$

A learnable per-layer scale θ_l rescales normalized logits, letting each layer sharpen/flatten its routing — enabling distinct sparsity per depth under one global threshold.

TRAINING PROCEDURE

Algorithm 1 DTop-*p* MoE

Require: Dataset $\mathcal{D}$, target expert T, initial probability threshold p_0, PI gain coefficient K_{pro}, K_{int}, model parameters Θ (including dynamic scales $\{\theta_l\}_{l=1}^{L}$)

Initialize integral error accumulation $e_{sum} \leftarrow 0$
Initialize probability threshold $p_1 = p_0$
for step $t = 1, 2, \ldots$ with batch $\mathcal{B}_t \in \mathcal{D}$ **do**
  Accumulator for number of activated experts $a_{sum} \leftarrow 0$
  Forward Pass:
  for layer $l = 1$ to L **do**
    Compute raw logits for input representation $\boldsymbol{x}$: $\boldsymbol{z} \leftarrow \boldsymbol{W}\boldsymbol{x}$
    Dynamic Routing Normalization (Equation 7): $\boldsymbol{P} \leftarrow \text{softmax}\left(\theta_l \cdot \frac{\boldsymbol{z}-\mu(\boldsymbol{z})}{\sigma(\boldsymbol{z})}\right)$
    Nucleus Sampling with global threshold p_t:
      Select minimal set of experts S such that $\sum_{i \in S} P_i \geq p_t$
      $r_i(\boldsymbol{x}) \leftarrow \frac{P_i}{\sum_{j \in S} P_j}$ for $i \in S$, else 0 (Equation 4)
    Record number of activated experts: $a_{sum} \leftarrow a_{sum} + |S|$
    Compute layer output: $\boldsymbol{y} \leftarrow \sum_{i \in S} r_i(\boldsymbol{x}) E_i(\boldsymbol{x})$
  end for
  PI controller Update (Equation 6):
  Calculate average activation per token: $a_t \leftarrow a_{sum}/(L \cdot |\mathcal{B}_t|)$, $|\mathcal{B}_t|$ is total tokens in $\mathcal{B}_t$
  Calculate sparsity error: $e_t \leftarrow (T - a_t)/N$
  Update integral term: $e_{sum} \leftarrow e_{sum} + e_t$
  Update global threshold:
    $p_{t+1} \leftarrow p_0 + K_{pro} \cdot e_t + K_{int} \cdot e_{sum}$
  Clip p_{t+1} to range $(0, 1)$
  Optimization:
  Compute Total Loss $\mathcal{L}$
  Update parameters Θ via gradient descent
end for

NLP — TRAINING DYNAMICS (100B TOKENS)

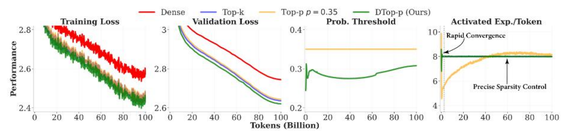

Figure 3. Training and validation performance of the Dense-1.3B and MoE-1.3B-6.9B-64E8A models using Top-*k*, Top-*p*, and DTop-*p* routing on NLP tasks. DTop-*p* achieves the best overall performance.

MoE-1.3B-6.9B-64E8A (1.3B active / 6.9B total). **DTop-*p*** reaches the best loss *and* locks sparsity to T=8, unlike fixed Top-*p* which overshoots.

NLP — INFERENCE BENCHMARKS (13 DATASETS)

Table 2. Inference performance comparison between Dense-1.3B and MoE-1.3B-6.9B-64E8A models with Top-*k*, Top-*p*, and DTop-*p* MoE trained on 100B tokens. **Bold** indicates the best performance across all settings. Numbers in parentheses indicate the number of few-shot examples used in evaluation. DTop-*p* MoE achieves the highest average performance.

Benchmark	Dense-1.3B	MoE-1.3B-6.9B-64E8A		
	Dense	Top-*k*	Top-*p*	DTop-*p* (Ours)
SVAMP(5)	5.3	10.3	8.3	**16.0**
MMLU(5)	25.2	26.6	26.8	**27.4**
ARC-Easy(0)	60.9	65.7	65.5	**67.1**
ARC-Challenge(0)	34.4	40.6	40.9	**41.7**
COPA(5)	69.0	82.0	82.0	**85.0**
PIQA(5)	75.7	**78.9**	77.7	78.1
HellaSwag(0)	62.7	69.1	69.2	**70.9**
WinoGrande(5)	62.7	64.0	66.3	**67.2**
LAMBADA(5)	56.7	61.5	**63.5**	62.5
BoolQ(5)	55.4	63.9	63.9	**65.4**
AGIEval-LSAT-RC(5)	26.2	22.7	23.8	**27.2**
AGIEval-LSAT-LR(5)	26.0	**26.4**	24.9	25.5
AGIEval-SAT-EN(5)	26.5	24.8	**27.7**	**27.7**
Average	45.1	49.0	49.3	**50.9**

CV — DIT PRE-TRAINING (2T PIXEL TOKENS)

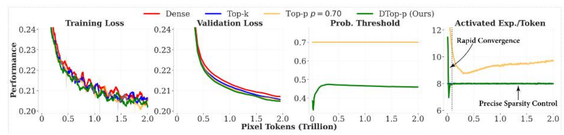

Figure 4. Training and validation performance of the 0.9B Dense model versus the 64E8A MoE model (0.9B activated / 3.4B total parameters) using Top-*k*, Top-*p*, and DTop-*p* MoE on CV tasks. DTop-*p* achieves the best performance.

PRECISE & ADAPTIVE SPARSITY CONTROL

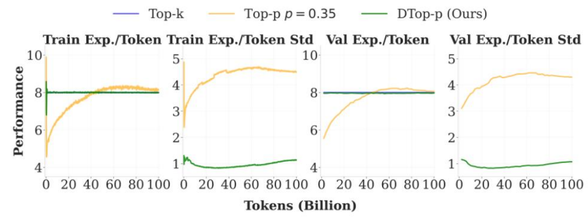

Figure 6. Mean and standard deviation of activated experts per token on training and validation sets for Top-*k*, Top-*p*, and DTop-*p* MoE. DTop-*p* effectively and rapidly converges to the target activation level on both training and validation datasets.

DTop-*p* converges to T=8 with low variance (σ≈1); fixed Top-*p* drifts with σ≈4. It activates fewer experts in shallow layers, more in deep layers.

ABLATION — PI CONTROLLER & DRN

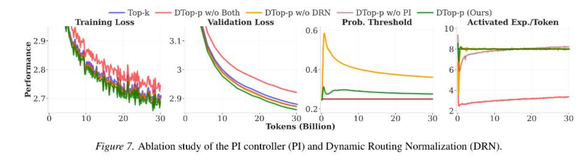

Figure 7. Ablation study of the PI controller (PI) and Dynamic Routing Normalization (DRN).

Both components are needed: PI enforces the budget; DRN adaptively rescales layer logits. Together they give the best loss and stable threshold.

SCALING — MODEL SIZE (0.4B → 2.4B)

Table 8. Inference performance of Dense models vs. varying 64E8A MoE models across different model sizes (0.4B, 1.3B, 2.4B) trained on 100B tokens. DTop-*p* consistently achieves the best performance and scales effectively with model size.

Area	Benchmark	Dense-0.4B	MoE-0.4B-3.7B		Dense-1.3B	MoE-1.3B-6.9B		Dense-2.4B	MoE-2.4B-13.6B	
		Dense	Top-*k*	DTop-*p*	Dense	Top-*k*	DTop-*p*	Dense	Top-*k*	DTop-*p*
Symbolic Problem Solving	SVAMP(5)	6.3	4.0	6.6	5.3	10.3	16.0	11.0	13.6	**22.6**
World Knowledge	MMLU(5)	25.0	23.9	24.8	25.2	26.6	27.4	24.6	28.2	**29.0**
	ARC-Easy(0)	52.9	61.3	62.1	60.9	65.7	67.1	64.7	70.2	**70.7**
	ARC-Challenge(0)	26.9	32.5	33.7	34.4	40.6	41.7	37.7	**45.5**	44.6
Commonsense Reasoning	COPA(5)	63.0	72.0	74.0	69.0	82.0	85.0	80.0	84.0	**87.0**
	PIQA(5)	71.2	74.5	75.0	75.7	78.9	78.1	77.2	79.1	**79.5**
Language Understanding	HellaSwag(0)	48.8	60.6	60.7	62.7	69.1	70.9	67.9	73.9	**74.4**
	WinoGrande(5)	55.8	59.7	59.9	62.7	64.0	67.2	67.4	70.6	**71.8**
	LAMBADA(5)	46.8	54.5	56.2	56.7	61.5	62.5	63.8	68.6	**69.4**
Reading Comprehension	BoolQ(5)	58.9	62.4	64.1	55.4	63.9	65.4	69.2	68.0	**71.2**
	AGIEval-LSAT-RC(5)	21.2	23.9	23.9	26.2	22.7	**27.2**	24.2	24.2	26.2
	AGIEval-LSAT-LR(5)	24.5	22.1	23.5	26.0	**26.4**	25.5	24.5	22.1	26.1
	AGIEval-SAT-EN(5)	23.7	22.3	23.3	26.5	24.8	27.7	27.6	27.7	**30.5**
Average		40.4	44.1	45.2	45.1	49.0	50.9	49.2	52.0	**54.1**

CONCLUSIONS

+1.9% avg. over Top-*k* (NLP, matched FLOPs)

σ≈1 stable activated-expert count

2 modalities: LLM & DiT

- **DTop-*p*** reconciles token-adaptive routing with strict compute control via PI control — no gradient needed for the threshold.
- Beats **Top-*k*** & **fixed Top-*p*** on LLMs and DiTs; robust scaling across granularity, model & dataset size.

Trained on DCLM-Baseline (NLP) & 2T pixel tokens (CV) · 64–128× NVIDIA A100 80GB

can.jin@rutgers.edu · Rutgers University & Adobe Research