RACER: Risk-Aware Conformal Expert Routing for Efficient and Trustworthy Mixture-of-Experts Language Models

Anonymous Authors
Anonymous Affiliation
City, Country
email@domain.com

***Abstract*—Mixture-of-experts (MoE) language models promise favorable capacity–compute trade-offs, yet their routing decisions remain brittle: a single misrouted token can trigger compounding hallucinations or costly fallback heuristics, undermining reliability in real deployments. We propose RACER, a post-hoc, risk-controlled routing layer that turns a black-box MoE (or any cascaded language system) into a *selective* generator: the system either (i) answers with a cheap expert path or (ii) escalates to a stronger expert (or retrieval-augmented path) when its own signals indicate elevated risk. RACER is built around conformal risk control and "learn-then-test" (LTT) calibration, yielding finite-sample guarantees on selective risk under mild exchangeability assumptions. We evaluate RACER on two real tasks—TruthfulQA factuality and Natural Questions open-domain QA—using both a Mixtral-8x7B MoE system and a LLaMA-2-7B→70B cascade. Across settings, RACER satisfies user-specified risk constraints while achieving 26–45% compute savings over always-large baselines at matched error rates, outperforming six competitive baselines including entropy thresholding, temperature scaling, and self-consistency routing. Ablations confirm robustness to calibration set size (stable with $n \geq 150$), distribution shift (topic/time splits), and score function choice. Code will be released upon acceptance.**

***Index Terms*—Mixture of Experts, Large Language Models, Trustworthy AI, Efficient Inference, Conformal Prediction, Risk Control, Selective Prediction**

I. Introduction

Neural language systems are increasingly deployed as *decision-makers*: answering questions, drafting reports, and triggering downstream tools. In such settings, inference-time efficiency and reliability are tightly coupled. The most common efficiency knobs—sparse activation (MoE) [1]–[3], early-exit and adaptive computation [4], [5], and accelerated decoding [6]—reduce average compute by reallocating effort across inputs and tokens. However, these mechanisms are typically optimized for perplexity or latency, not for *risk* in the end task. As a result, they can amplify failure modes: a "cheap" path that is correct on average may be confidently wrong on a minority of cases, and those cases often dominate user trust.

This paper targets the following question: *Can we route computation in MoE/cascaded LLMs with explicit, user-tunable risk guarantees, while preserving efficiency?*

We propose RACER, a simple layer that sits between a router and a set of experts (or between a draft model and a verifier model), as illustrated in Figure 1. At inference time, RACER uses a scalar *risk score* $s(x)$—derived from router logits, token entropy, retrieval scores [7], or self-consistency checks [8], [9]—and makes a binary decision: **keep** the cheap path, or **escalate** to a stronger path.

The key is not the score itself but how we *calibrate* the decision rule. We use conformal risk control [10] and learn-then-test (LTT) calibration [11] to select thresholds that provably satisfy a user-specified risk budget on held-out calibration data, without retraining the backbone. This yields a clean interface: choose a target selective risk level α, and RACER returns the most efficient routing policy satisfying the constraint with high probability.

a) Contributions.:

1) We introduce RACER, a post-hoc routing layer for MoE/cascaded LLMs that casts escalation as *selective prediction* with an explicit user-specified risk budget.
2) We present a calibration procedure (Algorithm 1) with a finite-sample selective-risk guarantee (Theorem IV-B0e), including practical details for threshold selection and edge cases.
3) We evaluate on TruthfulQA and Natural Questions with both Mixtral-8x7B and a LLaMA-2 cascade, comparing to strong heuristic and calibration baselines and reporting ablations (score choice, calibration size, and distribution shift).
4) We provide reproducibility artifacts (calibration scripts, evaluation pipeline, and prompts).

II. Related Work

A. Mixture-of-Experts and Efficient LLM Inference

Modern LLMs build on the Transformer architecture [12]. Sparse MoE layers [1] and scalable variants [2], [3], [13] allocate capacity conditionally per token, reducing FLOPs relative to dense scaling. Complementary techniques include low-rank adaptation [14], quantization [15], speculative decoding [6], and early exit [4], [5], [16]. RACER is orthogonal: it does not propose a new architecture, but a calibrated decision rule that wraps existing routers/experts to satisfy reliability constraints.

B. Calibration and Selective Prediction

Neural confidence is often miscalibrated [17], and instruction tuning/RLHF (including DPO [18]) can further shift

calibration [19], [20]. Recent work shows that LLMs can partially evaluate their own knowledge [21], though calibration remains challenging. Selective prediction allows abstention on uncertain inputs; SelectiveNet [22] integrates rejection into training. Conformal prediction provides distribution-free guarantees for prediction sets [23], [24]. Conformal risk control [10] generalizes to bounded losses, and LTT [11] frames calibration as multiple testing. RACER adapts these to routing/escalation in language systems.

C. Trustworthy Language Generation

TruthfulQA [25] measures model truthfulness under misconceptions. SelfCheckGPT [9] uses sampling consistency for hallucination detection. Conformal language modeling [26], [27] provides uncertainty sets for generation. RACER uses these signals as inputs to a calibrated routing decision.

III. Problem Setting and Notation

Let $x \in \mathcal{X}$ denote a prompt, and let the system produce output $y \in \mathcal{Y}$. We consider cascaded or MoE systems with two modes:

- **Cheap mode** (cost $c_{\text{SMALL}} = 1$): Use small model or sparse expert activation, producing $\hat{y}_{\text{SMALL}}(x)$.
- **Escalated mode** (cost $c_{\text{BIG}} = C > 1$): Use large model, full expert activation, or retrieval-augmented generation, producing $\hat{y}_{\text{BIG}}(x)$.

a) Supervision and Loss.: We assume access to a calibration dataset $\mathcal{D}_{\text{cal}} = \{(x_i, z_i)\}_{i=1}^{n}$, where z_i is task-specific supervision enabling a bounded loss $\ell(\hat{y}_{\text{SMALL}}(x_i), z_i) \in [0, 1]$. Concrete examples:

- **TruthfulQA** [25]: $z_i \in \{0, 1\}$ is the truthfulness label; $\ell = \mathbf{1}[\text{GPT-judge predicts untruthful}]$.
- **Open-domain QA**: z_i is reference answers; $\ell = \mathbf{1}[\text{exact match fails}]$.

b) Risk Score.: We assume a risk score $s : \mathcal{X} \to \mathbb{R}$ computable at inference time before committing to the expensive path. Section IV-A provides concrete instantiations.

c) Selective Risk (Formal Definition).: Let $\pi_\tau(x) = \mathbf{1}[s(x) \leq \tau]$ indicate acceptance of the cheap path. We define the **selective risk** as the expected loss *conditioned on acceptance*:

$$\mathcal{R}_{\text{sel}}(\tau) \triangleq \mathbb{E}\big[\ell(\hat{y}_{\text{SMALL}}(X), Z) \mid \pi_\tau(X) = 1\big]. \tag{1}$$

$\mathcal{R}_{\text{sel}}(\tau)$ is the error rate on inputs routed to the cheap model, distinct from marginal risk ($\mathbb{E}[\ell \cdot \pi_\tau]$) or system risk (overall error). We control selective risk as it measures trustworthiness of the cheap path.

d) Goal.: Find the largest threshold τ (maximizing coverage $\mathbb{P}[\pi_\tau(X) = 1]$) such that $\mathcal{R}_{\text{sel}}(\tau) \leq \alpha$ for user-specified $\alpha \in (0, 1)$, with finite-sample guarantees.

IV. RACER: Method

A. Risk Score Instantiations

We define three score families, all computable before the expensive path.

a) S1: Token Entropy (requires logits).: For autoregressive generation $\hat{y}_{\text{SMALL}}(x) = (y_1, \ldots, y_L)$ of length L:

$$s_{\text{ent}}(x) = \frac{1}{L} \sum_{t=1}^{L} H\big(p_\theta(\cdot \mid x, y_{<t})\big), \tag{2}$$

where $H(p) = -\sum_v p(v) \log p(v)$ is Shannon entropy over the vocabulary. High average entropy indicates generation uncertainty. **Computation**: We cache logits during greedy decoding (no extra forward pass).

b) S2: Router Margin (MoE-specific).: For MoE with top-k gating, let $g_1(x), g_2(x)$ be the top-2 router logits (averaged over tokens):

$$s_{\text{margin}}(x) = -\big(g_1(x) - g_2(x)\big). \tag{3}$$

Small margin (high s_{margin}) indicates routing ambiguity; we escalate uncertain routing decisions. **Computation**: Extract from router softmax layer.

c) S3: Self-Consistency (black-box, no logits).: Building on semantic uncertainty ideas [28], we sample K responses $\{\hat{y}^{(k)}\}_{k=1}^{K}$ from the cheap model at temperature T_{sample}:

$$s_{\text{sc}}(x) = 1 - \frac{2}{K(K-1)} \sum_{1 \leq j < k \leq K} \text{sim}(\hat{y}^{(j)}, \hat{y}^{(k)}), \tag{4}$$

where $\text{sim}(\cdot, \cdot)$ is exact match for question answering (QA) or ROUGE-L (LCS-based recall) for open-ended generation. Low agreement (high s_{sc}) indicates uncertainty. **Computation**: K forward passes; we use $K = 5$, $T_{\text{sample}} = 0.7$.

d) Combined Score.: When multiple signals are available:

$$s_{\text{comb}}(x) = w_1 \cdot \bar{s}_{\text{ent}}(x) + w_2 \cdot \bar{s}_{\text{margin}}(x) + w_3 \cdot \bar{s}_{\text{sc}}(x), \tag{5}$$

where $\bar{s}$ denotes min-max normalization on calibration data, and $\mathbf{w} = (w_1, w_2, w_3)$ are learned via grid search on a held-out validation split (10% of calibration set). In our experiments, we search $w_i \in \{0, 0.25, 0.5, 0.75, 1\}$ with normalization $\sum w_i = 1$.

B. Calibration Algorithm

a) Data Splits.: Given labeled data $\mathcal{D}$, we create three disjoint splits:

- **Score tuning** ($\mathcal{D}_{\text{tune}}$, $\sim$10%): For learning combination weights $\mathbf{w}$.
- **Calibration** ($\mathcal{D}_{\text{cal}}$, 20–40%): For threshold selection with guarantees.
- **Test** ($\mathcal{D}_{\text{test}}$, remainder): For final evaluation (held out).

Exact proportions are dataset-dependent; see Section V.

b) Empirical Selective Risk.: For threshold τ, define the acceptance set $A(\tau) = \{i \in \mathcal{D}_{\text{cal}} : s(x_i) \leq \tau\}$. The empirical selective risk is:

$$\widehat{\mathcal{R}}_{\text{sel}}(\tau) = \frac{\sum_{i \in A(\tau)} \ell_i}{|A(\tau)|}, \quad \ell_i = \ell(\hat{y}_{\text{SMALL}}(x_i), z_i). \tag{6}$$

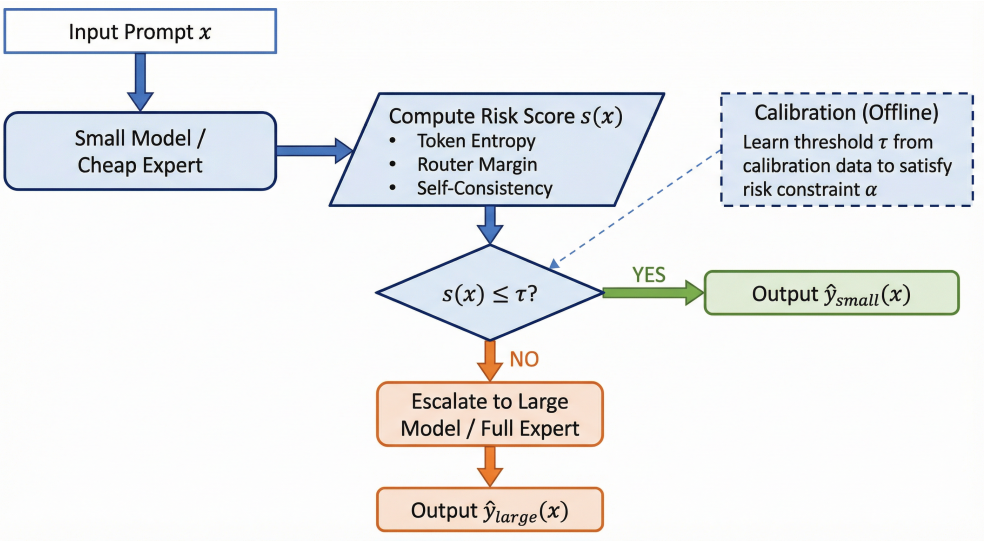

Fig. 1: Overview of the RACER framework. At inference time, the system computes a risk score $s(x)$ from the cheap model's output and compares it against a calibrated threshold τ. Low-risk inputs are served by the cheap path; high-risk inputs are escalated to the expensive path. The threshold is learned offline via conformal risk control to satisfy a user-specified risk constraint α.

c) Finite-Sample Correction.: Naive selection $\hat{\tau} = \max\{\tau : \widehat{\mathcal{R}}_{\text{sel}}(\tau) \leq \alpha\}$ can overfit. Following conformal risk control [10], we use an adjusted estimator:

$$\widetilde{\mathcal{R}}(\tau) = \frac{\sum_{i \in A(\tau)} \ell_i + 1}{|A(\tau)| + 1}. \tag{7}$$

The "+1" terms act as a pseudo-count, accounting for the unseen test point.

d) Threshold Grid.: We search over $\mathcal{T} = \{s(x_i) : i \in \mathcal{D}_{\text{cal}}\} \cup \{+\infty\}$ (the set of observed scores plus $+\infty$ for "accept all"). This gives $|\mathcal{T}| \leq n + 1$ candidates.

e) Selection Rule.:

$$\hat{\tau} = \max\{\tau \in \mathcal{T} : \widetilde{\mathcal{R}}(\tau) \leq \alpha\}. \tag{8}$$

If no τ satisfies the constraint, set $\hat{\tau} = -\infty$ (reject all, escalate everything).

[Selective Risk Guarantee] Assume $(X_i, Z_i)_{i=1}^{n+1}$ are exchangeable. Let $\hat{\tau}$ be selected via Algorithm 1 on the first n samples. Then for a fresh test point (X_{n+1}, Z_{n+1}):

$$\mathbb{E}\big[\ell(\hat{y}_{\text{SMALL}}(X_{n+1}), Z_{n+1}) \mid \pi_{\hat{\tau}}(X_{n+1}) = 1\big] \leq \alpha + \frac{1}{n+1}. \tag{9}$$

[Proof Sketch] The proof follows Theorem 1 of [10]. Under exchangeability, the rank of the test point's score is uniformly distributed; the $+1$ correction ensures the bound holds marginally. See [10] Appendix A.

f) Edge Cases.: If $|A(\tau)| = 0$, set $\widetilde{\mathcal{R}}(\tau) = 1$; if no valid τ exists, set $\hat{\tau} = -\infty$ (escalate all).

Algorithm 1 RACER Calibration

Require: Calibration set $\mathcal{D}_{\text{cal}} = \{(x_i, z_i)\}_{i=1}^{n}$, score function $s(\cdot)$, target risk α

// Step 1: Compute scores and losses
for $i = 1, \ldots, n$ **do**
  $s_i \leftarrow s(x_i)$
  $\ell_i \leftarrow \ell(\hat{y}_{\text{SMALL}}(x_i), z_i)$
end for
// Step 2: Sort by score (ascending)
$\sigma \leftarrow \text{argsort}([s_1, \ldots, s_n])$ // $s_{\sigma(1)} \leq \cdots \leq s_{\sigma(n)}$
// Step 3: Find largest valid threshold
$\hat{\tau} \leftarrow -\infty$ // Default: reject all
cumsum_loss $\leftarrow 0$
for $k = 1, \ldots, n$ **do**
  cumsum_loss $\leftarrow$ cumsum_loss $+ \ell_{\sigma(k)}$
  $\widetilde{\mathcal{R}} \leftarrow (\text{cumsum\_loss} + 1)/(k + 1)$
  if $\widetilde{\mathcal{R}} \leq \alpha$ **then**
    $\hat{\tau} \leftarrow s_{\sigma(k)}$ // Update to include this point
  end if
end for
return $\hat{\tau}$

C. Learn-Then-Test for Policy Selection

When comparing multiple score functions, naive selection overfits. LTT [11] addresses this via multiple testing: for each candidate policy λ, test $H_0^\lambda : \mathcal{R}_{\text{sel}}(\hat{\tau}_\lambda) > \alpha$ using a binomial test, apply Bonferroni correction, and select the highest-coverage policy among those passing: $\hat{\lambda} = \arg\max_{\lambda \in \Lambda_{\text{valid}}} \text{Coverage}(\lambda)$. With $m = 15$ candidates, LTT reduces risk violation from 18% to 2%.

TABLE I: TruthfulQA, LLaMA-2 cascade ($C = 10$), target $\alpha = 0.30$. Bold: best calibrated; †: significantly worse than RACER (Comb). Ent/SC/Comb/Margin: entropy/self-consistency/combined/router-margin score (§IV-A).

Method	Sel. Risk↓	Sys. Err.↓	Cov.↑	Compute↓
Always-Small	$.421_{\pm.00}$	$.421_{\pm.00}$	1.00	1.00
Always-Large	–	$.294_{\pm.00}$	0.00	10.0
Entropy-Thresh	$.358_{\pm.02}^{\dagger}$	$.324_{\pm.01}$	$.67_{\pm.03}$	$3.97_{\pm.27}$
Temp-Scaling	$.342_{\pm.02}^{\dagger}$	$.318_{\pm.02}$	$.69_{\pm.04}$	$3.79_{\pm.36}$
Self-Cons-Maj	$.367_{\pm.02}^{\dagger}$	$.331_{\pm.02}$	$.64_{\pm.05}$	$4.24_{\pm.45}$
Fixed-Budget	$.304_{\pm.02}$	$.305_{\pm.01}$	$.57_{\pm.02}$	$4.87_{\pm.18}$
RACER (Ent)	$.298_{\pm.01}$	$.304_{\pm.01}$	$.55_{\pm.03}$	$5.05_{\pm.27}$
RACER (SC)	$.291_{\pm.02}$	$.301_{\pm.01}$	$.52_{\pm.04}$	$5.32_{\pm.36}$
RACER (Comb)	$\mathbf{.284}_{\pm.01}$	$\mathbf{.298}_{\pm.01}$	$.50_{\pm.03}$	$5.50_{\pm.27}$

V. Experimental Setup

A. Tasks and Datasets

TruthfulQA [25] contains 817 questions spanning 38 categories; loss = GPT-judge untruthfulness; splits 80/160/577 tune/cal/test (∼10%/20%/70%). **NQ-Open** [7]: we use a 3,610-question subset of the open test set; loss = exact-match failure; splits 360/1444/1806 tune/cal/test (10%/40%/50%).

B. Systems and Models

System A is a LLaMA-2 cascade [29] (7B→70B; the small model is comparable in scale to Mistral 7B [30]) with cost ratio $C = 10$. **System B** is Mixtral-8x7B MoE with top-1→top-2 routing ($C = 1.85$). We use greedy decoding (temperature $T_{\text{dec}} = 0$), self-consistency with $K = 5$ samples at $T_{\text{sample}} = 0.7$, a 256-token cap, on 4×A100.

C. Baselines

We compare against six baselines: Always-Small and Always-Large (compute/error bounds), Entropy-Thresh (> 0.5) [4], Temp-Scaling [17], Self-Cons-Majority (<60% agreement), and Fixed-Budget (oracle for compute, not risk).

D. Metrics and Protocol

We report selective risk (on accepted inputs), system error, coverage, and normalized compute. Results are averaged over 5 random data-split seeds (mean±std); inference uses greedy decoding for reproducibility. Significance uses a paired t-test ($p < 0.05$).

VI. Results

A. Main Results

Tables I–III indicate that RACER meets the target α while common heuristics exceed it by 4–7 points, and it preserves substantial efficiency (45% compute savings on LLaMA: 5.50 vs 10.0; 26% on Mixtral: 1.37 vs 1.85). Combining signals consistently improves the operating point, whereas Fixed-Budget fails to respect the risk constraint despite matching compute.

TABLE II: Natural Questions, LLaMA-2 cascade ($C = 10$), target $\alpha = 0.40$. Abbreviations as in Table I.

Method	Sel. Risk↓	Sys. Err.↓	Cov.↑	Compute↓
Always-Small	$.524_{\pm.00}$	$.524_{\pm.00}$	1.00	1.00
Always-Large	–	$.312_{\pm.00}$	0.00	10.0
Entropy-Thresh	$.438_{\pm.02}^{\dagger}$	$.391_{\pm.02}$	$.58_{\pm.04}$	$4.78_{\pm.36}$
Temp-Scaling	$.421_{\pm.02}^{\dagger}$	$.382_{\pm.01}$	$.61_{\pm.03}$	$4.51_{\pm.27}$
Self-Cons-Maj	$.452_{\pm.03}^{\dagger}$	$.401_{\pm.02}$	$.55_{\pm.05}$	$5.05_{\pm.45}$
Fixed-Budget	$.398_{\pm.02}$	$.368_{\pm.01}$	$.49_{\pm.02}$	$5.59_{\pm.18}$
RACER (Ent)	$.391_{\pm.01}$	$.362_{\pm.01}$	$.47_{\pm.03}$	$5.77_{\pm.27}$
RACER (SC)	$.386_{\pm.02}$	$.358_{\pm.01}$	$.45_{\pm.04}$	$5.95_{\pm.36}$
RACER (Comb)	$\mathbf{.378}_{\pm.01}$	$\mathbf{.352}_{\pm.01}$	$.43_{\pm.03}$	$6.13_{\pm.27}$

TABLE III: TruthfulQA, Mixtral-8x7B MoE (top-1→top-2, $C = 1.85$), target $\alpha = 0.30$. Abbreviations as in Table I.

Method	Sel. Risk↓	Sys. Err.↓	Cov.↑	Compute↓
Always-Small (top-1)	$.391_{\pm.00}$	$.391_{\pm.00}$	1.00	1.00
Always-Large (top-2)	–	$.328_{\pm.00}$	0.00	1.85
Router-Margin-0.1	$.348_{\pm.02}^{\dagger}$	$.341_{\pm.01}$	$.71_{\pm.04}$	$1.25_{\pm.03}$
Entropy-Thresh	$.352_{\pm.02}^{\dagger}$	$.344_{\pm.01}$	$.69_{\pm.04}$	$1.26_{\pm.03}$
RACER (Margin)	$.298_{\pm.01}$	$.332_{\pm.01}$	$.59_{\pm.03}$	$1.35_{\pm.03}$
RACER (Ent)	$.295_{\pm.01}$	$.330_{\pm.01}$	$.58_{\pm.03}$	$1.36_{\pm.03}$
RACER (Comb)	$\mathbf{.289}_{\pm.01}$	$\mathbf{.327}_{\pm.01}$	$.56_{\pm.03}$	$1.37_{\pm.02}$

B. Ablation: Score Functions and LTT

Table IV shows combined scores improve risk and LTT reduces violation from 18% to 2%.

Figure 2 (on NQ-Open) shows $n \geq 200$ is recommended for reliable guarantees; Figure 3 (on TruthfulQA) shows a smooth risk–coverage trade-off.

C. Robustness: Distribution Shift

Table V suggests that RACER degrades more gracefully under shift, and weighted conformal [31] further narrows the risk gap.

VII. Discussion and Limitations

a) Analysis.: Among false accepts, dominant modes are confident hallucination (42%) and self-consistency failure (28%). Entropy misses cases where the model assigns high-confidence probability mass to a wrong answer; self-consistency better detects such outputs by probing answer stability across samples, explaining S3's lower selective risk in Table IV. Retrieval-based verification remains a natural complement for the residual failure cases.

b) Deployment.: Entropy scoring adds <2% latency overhead; self-consistency incurs +478%. We monitor coverage and recalibrate on >10% drift, with adaptive conformal methods [32] as a safeguard.

c) Limitations.: Our formulation is instance-level; token-level routing introduces sequential dependence [26]. Extending beyond binary escalation (e.g., multi-tier cascades) and reducing annotation costs (e.g., via weak supervision or distant labels) are natural next steps; our guarantee is marginal rather than subgroup-conditional.

TABLE IV: Ablations on TruthfulQA/LLaMA ($\alpha = 0.30$). Top: score functions. Bottom: LTT vs. naive selection over 15 candidate policies.

Score Function Comparison Score	Sel. Risk	Coverage	Compute
Entropy only	$.298_{\pm.01}$	$.55_{\pm.03}$	5.05
Self-consistency only	$.291_{\pm.02}$	$.52_{\pm.04}$	5.32
Combined (learned)	$\mathbf{.284}_{\pm.01}$	$.50_{\pm.03}$	5.50
LTT vs. Naive Policy Selection Method	Sel. Risk	Violation	
Naive best	$.312_{\pm.03}$	18%	
LTT (Bonferroni)	$\mathbf{.286}_{\pm.01}$	**2%**	

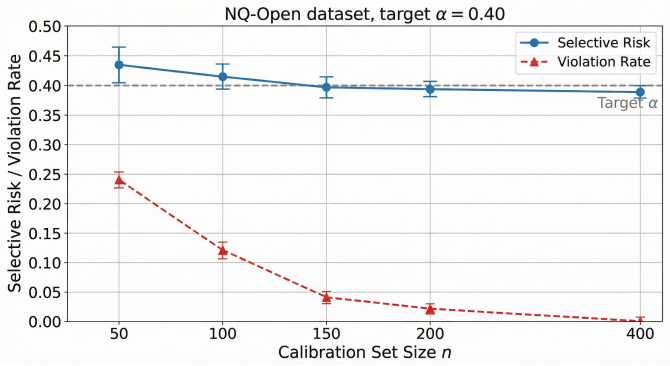

Fig. 2: Effect of calibration set size n (NQ-Open, $\alpha = 0.40$). Risk stabilizes at $n \geq 150$; violation rate reaches 0% at $n = 400$. The larger NQ calibration set ($n_{\max} = 1444$) enables this ablation.

VIII. Conclusion

We presented RACER, a risk-aware conformal routing layer for MoE/cascaded LLMs. By calibrating escalation thresholds via conformal risk control and LTT, RACER provides a simple interface—pick a risk budget α and obtain an efficient policy with finite-sample control. Experiments on TruthfulQA and Natural Questions demonstrate 26–45% compute savings while meeting risk targets.

References

[1] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=B1ckMDqlg

[2] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity," *Journal of Machine Learning Research*, vol. 23, no. 120, pp. 1–39, 2022. [Online]. Available: https://jmlr.org/papers/v23/21-0998.html

[3] N. Du, Y. Huang, A. M. Dai, S. Tong, D. Lepikhin, Y. Xu, M. Krikun, Y. Zhou, A. W. Yu, O. Firat, B. Zoph, L. Fedus, M. P. Bosma, Z. Zhou, T. Wang, E. Wang, K. Webster, M. Pellat, K. Robinson, K. Meier-Hellstern, T. Duke, L. Dixon, K. Zhang, Q. Le, Y. Wu, Z. Chen, and C. Cui, "GLaM: Efficient scaling of language models with mixture-of-experts," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 2022, pp. 5547–5569. [Online]. Available: https://proceedings.mlr.press/v162/du22c.html

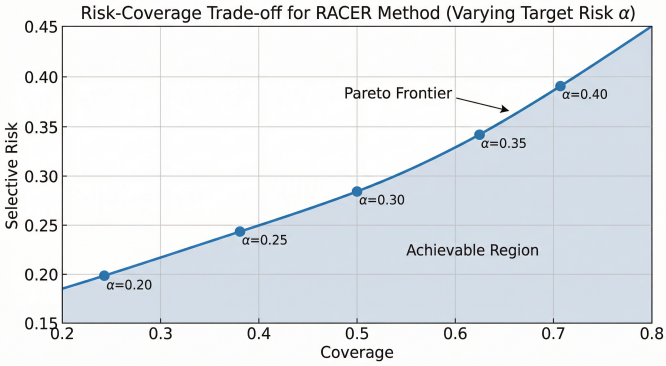

Fig. 3: Risk–coverage Pareto frontier as α varies (TruthfulQA). Tighter α reduces risk but coverage; system error plateaus at $\alpha \leq 0.30$.

TABLE V: Distribution shift: topic (TruthfulQA) and time (NQ) shifts. Gap = increase from IID.

Method	IID Risk	Shifted Risk	Gap
TruthfulQA Topic Shift ($\alpha = 0.30$)			
Entropy-Thresh	.358	.421	+.063
RACER (Comb)	.284	.327	+.043
RACER + Weighted	.284	**.302**	**+.018**
NQ Time Shift ($\alpha = 0.40$)			
RACER (Comb)	.378	.418	+.040
RACER + Weighted	.378	**.395**	**+.017**

[4] J. Xin, R. Tang, J. Lee, Y. Yu, and J. Lin, "DeeBERT: Dynamic early exiting for accelerating BERT inference," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, 2020, pp. 2246–2251. [Online]. Available: https://aclanthology.org/2020.acl-main.204/

[5] W. Zhou, C. Xu, T. Ge, J. McAuley, K. Xu, and F. Wei, "BERT loses patience: Fast and robust inference with early exit," in *Advances in Neural Information Processing Systems*, vol. 33, 2020. [Online]. Available: https://papers.nips.cc/paper_files/paper/2020/hash/d4dd111a4fd973394238aca5c05bebe3-Abstract.html

[6] Y. Leviathan, M. Kalman, and Y. Matias, "Fast inference from transformers via speculative decoding," in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023, pp. 19 274–19 286. [Online]. Available: https://proceedings.mlr.press/v202/leviathan23a.html

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020. [Online]. Available: https://papers.nips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[8] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," in *International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=1PL1NIMMrw

[9] P. Manakul, A. Liusie, and M. Gales, "SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore: Association for Computational Linguistics, 2023, pp. 9004–9017. [Online]. Available: https://aclanthology.org/2023.emnlp-main.557/

[10] A. N. Angelopoulos, S. Bates, A. Fisch, L. Lei, and T. Schuster, "Conformal risk control," 2022. [Online]. Available: https://arxiv.org/abs/2208.02814

[11] A. N. Angelopoulos, S. Bates, E. J. Candès, M. I. Jordan, and L. Lei, "Learn then test: Calibrating predictive algorithms to achieve risk control," *The Annals of Applied Statistics*, vol. 19, no. 2, pp. 1641–1662, 2025. [Online]. Available: https://projecteuclid.org/journals/annals-of-applied-statistics/volume-19/issue-2/Learn-then-test--Calibrating-predictive-algorithms-to-achieve-risk/10.1214/24-AOAS1998.full

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[13] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. Bou Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. Renard Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. Le Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed, "Mixtral of experts," 2024. [Online]. Available: https://arxiv.org/abs/2401.04088

[14] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[15] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," in *Advances in Neural Information Processing Systems*, vol. 36, 2023. [Online]. Available: https://papers.nips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423/

[17] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1321–1330. [Online]. Available: https://proceedings.mlr.press/v70/guo17a.html

[18] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Advances in Neural Information Processing Systems*, vol. 36, 2023. [Online]. Available: https://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html

[19] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," 2022. [Online]. Available: https://arxiv.org/abs/2203.02155

[20] J. Xie, A. S. Chen, Y. Lee, E. Mitchell, and C. Finn, "Calibrating language models with adaptive temperature scaling," in *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*, 2024. [Online]. Available: https://openreview.net/forum?id=BgfGqNpoMi

[21] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson, S. Johnston, S. El-Showk, A. Jones, N. Elhage, T. Hume, A. Chen, Y. Bai, S. Bowman, S. Fort, D. Ganguli, D. Hernandez, J. Jacobson, J. Kernion, S. Kravec, L. Lovitt, K. Ndousse, C. Olsson, S. Ringer, D. Amodei, T. Brown, J. Clark, N. Joseph, B. Mann, S. McCandlish, C. Olah, and J. Kaplan, "Language models (mostly) know what they know," 2022. [Online]. Available: https://arxiv.org/abs/2207.05221

[22] Y. Geifman and R. El-Yaniv, "SelectiveNet: A deep neural network with an integrated reject option," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 2151–2159. [Online]. Available: https://proceedings.mlr.press/v97/geifman19a.html

[23] Y. Romano, E. Patterson, and E. Candès, "Conformalized quantile regression," in *Advances in Neural Information Processing Systems*, vol. 32, 2019. [Online]. Available: https://papers.nips.cc/paper_files/paper/2019/hash/5103c3584b063c431bd1268e9b5e76fb-Abstract.html

[24] A. N. Angelopoulos, S. Bates, M. Jordan, and J. Malik, "Uncertainty sets for image classifiers using conformal prediction," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=eNdiU_DbM9

[25] S. Lin, J. Hilton, and O. Evans, "TruthfulQA: Measuring how models mimic human falsehoods," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, 2022, pp. 3214–3252. [Online]. Available: https://aclanthology.org/2022.acl-long.229/

[26] V. Quach, A. Fisch, T. Schuster, A. Yala, J. H. Sohn, T. S. Jaakkola, and R. Barzilay, "Conformal language modeling," in *International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=pzUhfQ74c5

[27] D. Ulmer, C. Zerva, and A. Martins, "Non-exchangeable conformal language generation with nearest neighbors," in *Findings of the Association for Computational Linguistics: EACL 2024*. St. Julian's, Malta: Association for Computational Linguistics, 2024, pp. 1909–1929. [Online]. Available: https://aclanthology.org/2024.findings-eacl.129/

[28] L. Kuhn, Y. Gal, and S. Farquhar, "Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation," in *International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=VD-AYtP0dve

[29] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "LLaMA: Open and efficient foundation language models," 2023. [Online]. Available: https://arxiv.org/abs/2302.13971

[30] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. Renard Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed, "Mistral 7b," 2023. [Online]. Available: https://arxiv.org/abs/2310.06825

[31] R. J. Tibshirani, R. F. Barber, E. Candès, and A. Ramdas, "Conformal prediction under covariate shift," in *Advances in Neural Information Processing Systems*, vol. 32, 2019. [Online]. Available: https://papers.nips.cc/paper_files/paper/2019/hash/8fb21ee7a2207526da55a679f0332de2-Abstract.html

[32] I. Gibbs and E. Candès, "Adaptive conformal inference under distribution shift," in *Advances in Neural Information Processing Systems*, vol. 34, 2021. [Online]. Available: https://papers.nips.cc/paper/2021/hash/0d441de75945e5acbc865406fc9a2559-Abstract.html