Mixture-of-Trees: Learning to Select and Weigh Reasoning Paths for Efficient LLM Inference

Yangbo Wei[1,2*], Zhen Huang [2,4*] Shaoqiang Lu[1,2], Junhong Qian[3], Dongge Qin[3], Ting Jung Lin[2], WEI W. XING[5], Chen Wu[2†], Lei He[2†]

[1]Shanghai Jiao Tong University, Shanghai, China
[2]Eastern Institute of Technology, Ningbo, China
[3]Southeast University, Nanjing, China
[4]University of Science and Technology of China, Hefei, China
[5]University of Sheffield, Sheffield, England
yangforever@sjtu.edu.cn

Abstract

We introduce Mixture-of-Trees (MoT), a novel framework that integrates sparse expert activation with structured tree-based reasoning for efficient LLM inference. MoT employs a learned gating mechanism to selectively activate only the most relevant expert reasoning trees for each problem, where experts use models of varying capacities based on task complexity. The framework features three key innovations: (1) sparse expert activation through unified gating networks, (2) specialized expert trees that leverage domain-specific expertise while optimizing the quality-efficiency trade-off, and (3) collaborative debate mechanisms for conflicting solutions. Additionally, MoT includes a shared baseline tree with early stopping—activated experts perform lightweight validation and terminate early when confidence is high. Experiments across five benchmarks (GSM8K, MATH, AIME 2024, MMLU, HotpotQA) show that MoT achieves 2-7 percentage point accuracy improvements while reducing LLM calls by 37-40% compared to existing multi-path methods.

Introduction

"In crowds, individual foolishness cancels out while wisdom reinforces itself."

— The Wisdom of Crowds

The remarkable performance of LLMs in complex reasoning tasks has driven significant advances in artificial intelligence (OpenAI 2023; Dubey et al. 2024; Touvron et al. 2023; Wei et al. 2025), yet their deployment remains constrained by fundamental trade-offs between computational efficiency and reasoning quality. As modern AI systems increasingly tackle complex problems involving mathematical reasoning, logical inference, and multi-step decision-making, the limitations of single-path reasoning approaches become increasingly apparent. While Chain-of-Thought (CoT) (Wei et al. 2022; Wang et al. 2022) prompting has demonstrated the power of explicit intermediate reasoning steps, its sequential nature often fails to capture the inherent uncertainty and multiple valid pathways present in human problem-solving processes (Plaat et al. 2024).

Method	Specialized Reasoning	Multi-Expert Collaboration	Efficiency Optimization	Complexity Adaptivity
MAD	✔	✔	✗	✗
ToT	✗	✗	✗	✗
FoT	✔	✗	✗	✗
GoT	✗	✗	✗	✔
BoT	✗	✗	✗	✗
XoT	✗	✗	✗	✔
MoT	✔	✔	✔	✔

Table 1: Comparison among different reasoning techniques.

Recent advances in multi-path reasoning approaches (Liu et al. 2025; Mo and Xin 2024), including Tree-of-Thought (ToT) (Yao et al. 2023) and Forest-of-Thought (FoT) (Wu et al. 2025), have shown promising improvements by exploring multiple reasoning trajectories and leveraging ensemble techniques. However, these methods suffer from significant computational overhead, as they typically allocate uniform resources across all reasoning paths without considering problem complexity or domain specificity. This one-size-fits-all paradigm becomes particularly problematic in large-scale deployments, where computational costs grow exponentially with the number of explored paths, making such approaches prohibitively expensive for practical applications (Chen et al. 2025).

The field has witnessed growing interest in mixture-of-experts architectures, particularly Mixture-of-Experts (MoE) (Shazeer et al. 2017; Zhou et al. 2022) models that achieve computational efficiency through sparse activation of specialized components. While traditional MoE approaches focus on parameter-level efficiency within neural network architectures, their principles of selective activation and specialized expertise offer compelling insights for reasoning-level optimization. The key insight lies in recognizing that different types of problems may benefit from different reasoning strategies, expert knowledge domains, and computational resources—much like how human experts naturally specialize in specific domains while collaborating on complex interdisciplinary challenges.

*These authors contributed equally.
†Corresponding author.

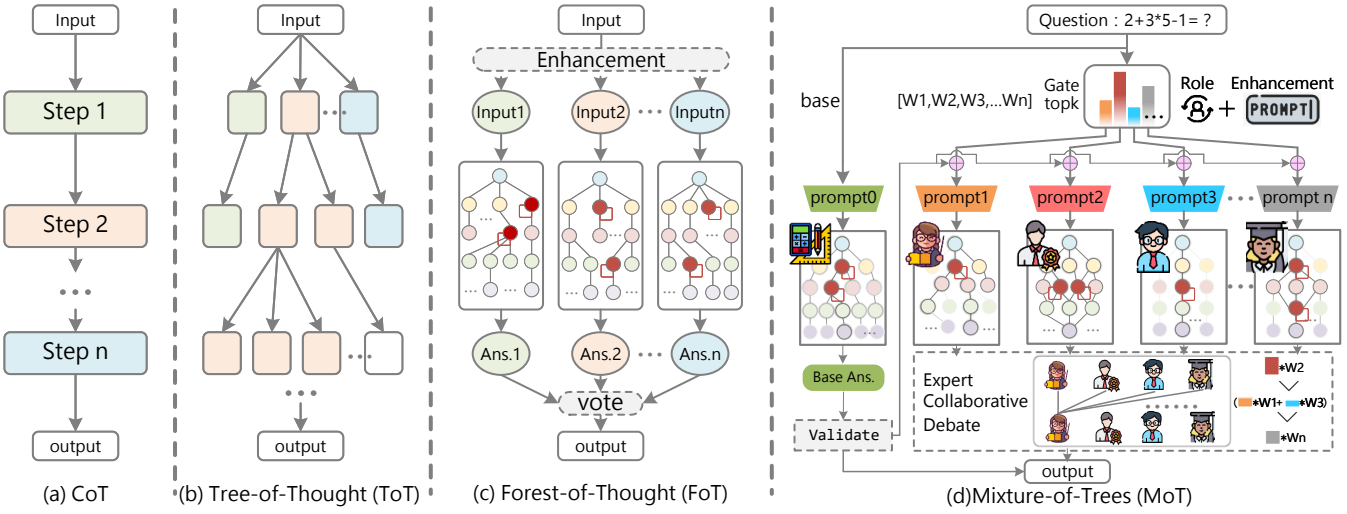

Figure 1: Overview of LLM reasoning paradigms. (a) CoT: single sequential reasoning path. (b) ToT: tree-structured multi-path search with higher cost. (c) FoT: parallel expansion of multiple reasoning trees but uniform resource allocation. (d) MoT (ours): introduces sparse expert activation via gating, heterogeneous expert trees, a shared baseline tree with early stopping, and expert debate, achieving improved accuracy and efficiency.

Despite these advances, existing reasoning frameworks face three critical limitations that hinder their practical deployment. **First, computational inefficiency**: Current multi-path methods uniformly activate all reasoning components, creating unnecessary computational overhead for problems that could be solved with simpler approaches. For instance, basic arithmetic problems receive the same computational treatment as advanced mathematical proofs, leading to suboptimal resource utilization. **Second, lack of adaptive specialization**: Existing systems fail to effectively leverage domain-specific expertise, treating all reasoning paths as equally relevant regardless of problem characteristics or required expertise. **Third, limited collaboration mechanisms**: While some methods explore multiple reasoning paths, they lack sophisticated mechanisms for experts to learn from each other's insights and converge toward higher-quality solutions through structured collaboration.

To tackle these fundamental challenges, we propose **Mixture-of-Trees (MoT)** (Figure 1), a novel framework that integrates the sparse activation principles of mixture-of-experts with the structured exploration of tree-based reasoning. As shown in Table 1, MoT stands out from existing techniques (e.g., ToT, FoT, GoT) by uniquely combining specialized reasoning, multi-expert collaboration, efficiency optimization, and complexity adaptivity.

Our framework introduces several key innovations. **Sparse Expert Activation**: Unlike traditional approaches that activate all reasoning components, MoT employs a learned gating mechanism to selectively activate only the most relevant expert reasoning trees for each specific problem. This selective activation is achieved through a unified gating network that simultaneously handles expert selection and weight assignment, inspired by MoE architectures but specifically optimized for reasoning tasks. **Expert Trees**: Recognizing that different reasoning tasks require different levels of computational complexity, MoT strategically assigns models of varying capacities to different expert roles. Complex mathematical reasoning experts utilize more powerful models, while basic arithmetic experts employ lightweight models, optimizing the quality-efficiency trade-off. **Collaborative Debate Mechanism**: When multiple experts produce conflicting solutions, MoT facilitates structured multi-round debates where experts can examine each other's reasoning and identify potential flaws.

Another key efficiency innovation in MoT is the **shared baseline tree with early stopping mechanism**. Rather than requiring all experts to perform complete reasoning from scratch, the system first generates a baseline solution using a shared reasoning tree. Activated experts then perform lightweight validation of this baseline solution. If weighted expert validation indicates high confidence in the baseline answer, the system terminates early, avoiding the computational overhead of full expert reasoning.

Our results demonstrate that MoT consistently outperforms state-of-the-art reasoning methods across all benchmarks, achieving 2-7 percentage point accuracy improvements across all benchmarks. Crucially, these accuracy gains are achieved while maintaining superior computational efficiency—MoT reduces LLM calls by 37-40% compared to FoT while delivering higher accuracy.

Related Work

CoT and Tree-based Reasoning Methods

Chain-of-Thought (CoT) enables LLMs to tackle complex problems through explicit generation of intermediate reasoning steps (Wei et al. 2022). Self-Consistency CoT (SC-CoT) further enhances robustness by introducing multi-path generation and voting mechanisms (Wang et al. 2022). Tree-of-Thought (ToT) models reasoning paths as search trees,

exploring multiple solution candidates through BFS/DFS traversal (Yao et al. 2023). Derivative methods such as BoT, XoT, GoT and FoT (Yang et al. 2024; Ding et al. 2024; Besta et al. 2024; Bi et al. 2024) propose template buffering, reinforcement learning-based tree search, and multi-tree parallel expansion, respectively. However, these approaches typically employ homogeneous models without differentiating resource allocation based on model complexity and task difficulty, leading to rapidly escalating costs when confronting challenging tasks. MoT introduces sparse expert activation and heterogeneous model configurations while inheriting tree-based search structures, significantly optimizing the efficiency-quality trade-off.

Mixture of Experts and Sparse Activation

Classical sparsely-gated MoE (Shazeer et al. 2017) activates only top-k experts at each step, decoupling parameter scale from computational cost and enabling larger-capacity training and inference. Subsequent research such as GShard (Lepikhin et al. 2020) and GLaM (Du et al. 2022) focuses on expert routing diversity and interpretability. Traditional MoE approaches concentrate on the training phase, typically maintaining homogeneous activation strategies during inference and lacking dynamic expert selection mechanisms based on task type or difficulty, resulting in coarse-grained resource matching. MoT extends the MoE (Eigen, Ranzato, and Sutskever 2013) paradigm to the inference level, employing non-normalized gating mechanisms based on input task characteristics to dynamically activate the most suitable expert combinations. By integrating tree-based search with early-exit strategies, it achieves energy-efficient inference.

Multi-Agent Systems and Collaborative Reasoning

Recent years have witnessed significant advantages of multi-agent systems in complex reasoning tasks. Multi-Agent Debate frameworks such as MAD (Liang et al. 2023) effectively improve answer consistency through interaction and critical feedback mechanisms between model agents. The "more agents is all you need" (Li et al. 2024) paradigm further validates that increasing the number of agents through sampling and voting strategies can achieve performance scaling across reasoning tasks, obtaining 69-200% improvements on challenging problems. The Mixture of Agents (MoA) framework (Wang et al. 2024) employs multiple LLMs as proposers and aggregators in a collaborative mechanism, while iterative refinement methods enable agents to continuously improve performance through peer criticism and self-reflection. MoT's expert collaborative debate mechanism builds upon these foundations but introduces structured multi-round analysis, where experts specifically examine others' reasoning paths to refine their own approaches, then conduct weighted voting based on tree-level confidence scores rather than simple output aggregation. This differs from existing multi-agent systems that typically operate on final outputs rather than intermediate reasoning structures.

MoT: Mixture-of-Trees

Expert Tree Modeling

➤ **Core Definition:** We formalize each expert tree as a specialized reasoning function $T_i^{(r_i,m_i)}$, jointly parameterized by role-specific prompts r_i and model specifications m_i:

$$T_i^{(r_i,m_i)} = \Pi_i \left(\mathcal{G}_{m_i}^{r_i}, \mathcal{V}_{m_i}^{r_i} \right) \tag{1}$$

Architecture Components: The expert tree consists of three core components:

① A **generator** $\mathcal{G}_{m_i}^{r_i}$ that uses model m_i with role-specific prompts to generate k candidate thought steps in parallel at the current reasoning state;

② An **evaluator** $\mathcal{V}_{m_i}^{r_i}$ that employs the same model to score the quality of each candidate step;

③ A **search strategy** Π_i (such as BFS, DFS, A*, etc.) that performs node expansion, pruning, and backtracking based on evaluation scores until termination conditions are met or optimal solutions are found.

The complete reasoning process of each expert on enhanced input is represented as:

$$s_i = T_i^{(r_i,m_i)}(x_i') = \Pi_i \left(\mathcal{G}_{m_i}^{r_i}, \mathcal{V}_{m_i}^{r_i}, x_i' \right). \tag{2}$$

Expert Diversity Strategy: Expert diversity is achieved through systematic role assignment and **heterogeneous model configuration**:

$$\mathcal{R} = \{r_1, r_2, ..., r_n\} = \{\text{"Algebra Expert"}, \text{"Statistics Expert"}, ...\}, \tag{3}$$

$$\mathcal{M} = \{m_1, m_2, ..., m_n\} = \{\text{Qwen-32B}, \text{Qwen-7B}, ...\}. \tag{4}$$

The heterogeneous model allocation strategy optimizes configuration based on the complexity requirements of expert roles and computational budget constraints. High-complexity reasoning experts (such as university mathematics experts) are assigned stronger models to ensure reasoning quality, while basic functional experts use lightweight models to reduce costs.

Input Enhancement Mechanism: Expert-specific input enhancement is performed during the activation phase, with customized enhancement strategies tailored to each expert's role characteristics:

$$x_i' = \text{enhance}(x, r_i) = \text{similar\_examples}(x, r_i) \oplus \text{rephrase}(x, r_i), \tag{5}$$

where $\oplus$ denotes text concatenation, *similar_examples* retrieves similar questions and solutions related to the expert's role from the knowledge base, and *rephrase* generates problem formulations tailored to the expert's domain.

Sparse Activation and Weight Allocation

➤ **Optimization Objective:** To optimize computational efficiency and achieve intelligent expert weight allocation, we design a unified gating network to simultaneously handle both expert selection and weight computation tasks. This network is inspired by the gating mechanism in Mixture-of-Experts but is specifically optimized for our collaborative reasoning scenario.

Activation Probability Computation: Given problem input x and expert role set $\mathcal{R} = \{r_1, r_2, ..., r_n\}$, the gating network computes activation probabilities for each expert:

$$p_i = \text{Gate}_\theta(x, r_i) = \sigma(\text{MLP}_\theta([\text{embed}(x), \text{embed}(r_i)])), \quad (6)$$

where σ is the sigmoid function and $[\cdot, \cdot]$ denotes feature concatenation. Unlike standard MoE, our activation probabilities $p_i \in [0, 1]$ do not need to satisfy the constraint $\sum_i p_i = 1$, as each expert's activation decision is mutually independent.

Selection Strategy: The **sparse activation strategy** combines threshold filtering and top-k selection to determine the active expert set:

$$\mathcal{A} = \{i : p_i \geq \tau\} \cap \{\text{top-}k(\{p_j\}_{j=1}^n)\}, \quad (7)$$

where τ is the confidence threshold and k ensures maximum activation count.

Network Architecture: We employ a lightweight two-layer MLP structure to implement the gating network, ensuring a balance between model capacity and computational efficiency. The network input is the joint embedding representation of question text and expert roles, outputting activation probability p_i for each expert.

Training Objective: The loss function design comprehensively considers task performance and computational cost:

$$\mathcal{L} = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[-\log P(y|\text{MoT}(x, \mathcal{A})) + \lambda \cdot \frac{|\mathcal{A}|}{N} + \beta \cdot \frac{C(\mathcal{A})}{C_{\max}} \right], \quad (8)$$

where the first term is the cross-entropy loss measuring prediction accuracy, the second term $\frac{|\mathcal{A}|}{N}$ penalizes excessive expert activation (N is the total number of experts), and the third term $\frac{C(\mathcal{A})}{C_{\max}}$ penalizes high computational overhead ($C(\mathcal{A})$ is the token consumption of activated experts, $C_{\max}$ is the normalization constant).

Weight Computation: The **weight allocation mechanism** directly utilizes activation probabilities as voting weights, avoiding the introduction of additional weight networks. For activated experts, their voting weights are computed through normalized activation probabilities:

$$w_i = \begin{cases} \frac{p_i \cdot c_i^\alpha}{\sum_{j\in\mathcal{A}} p_j \cdot c_j^\alpha} & \text{if } i \in \mathcal{A} \\ 0 & \text{else} \end{cases}, \quad (9)$$

where c_i is the confidence score of expert i, and α is a balance factor controlling the influence of confidence on final weights.

➔ **Expected Outcome:** Through this unified gating mechanism, we expect to achieve significant computational efficiency improvements—avoiding redundant computation from all experts while maintaining the quality and flexibility of expert collaboration.

Shared Tree and Early Stopping Mechanism

Baseline Generation: Our framework employs a shared baseline tree T_{shared} as the reasoning starting point for all experts, using the original problem input to quickly generate initial solutions:

$$s_{\text{base}} = T_{\text{shared}}(x). \quad (10)$$

This shared tree uses general reasoning strategies and original input, providing reasonable baseline answers for most problems. This design both saves redundant basic reasoning computation and provides a unified reference benchmark for subsequent expert validation.

Validation Process: The **lightweight validation phase** is implemented through prompt-guided rapid validation mechanisms. All activated experts $i \in \mathcal{A}$ perform quick evaluation of the baseline answer rather than complete re-reasoning. Each expert makes binary judgments on the baseline answer through specialized validation prompts:

$$v_i = \text{Validate}_i(x, s_{\text{base}}) \in \{\text{accept}, \text{reject}\}. \quad (11)$$

The validation prompt design focuses on checking answer reasonableness, logical consistency, and format correctness, avoiding the computational overhead of re-solving.

Early Stop: The **early stopping decision mechanism** is implemented based on weighted validation voting. Early stopping is triggered when supporting weights exceed opposing weights:

$$\text{Early\_Stop} = \begin{cases} \text{True}, & \text{if } \sum_{i\in\mathcal{A}, v_i=\text{acc}} w_i > \sum_{i\in\mathcal{A}, v_i=\text{rej}} w_i \\ \text{False}, & \text{otherwise} \end{cases}. \quad (12)$$

➔ **Computational Benefit:** If early stopping is triggered, the system directly returns s_{base} as the final answer, skipping the computationally intensive complete expert reasoning process. This mechanism is particularly suitable for simple problems where baseline answers are often sufficiently accurate without requiring multi-expert collaboration.

Expert Collaborative Debate

Collaboration Trigger: When the early stopping mechanism is not triggered, the system enters the expert collaboration phase. We first generate role-specific enhanced inputs for each activated expert and collect their initial solutions, forming a candidate answer set $\mathcal{D} = \{s_i : i \in \mathcal{A}\}$.

Conflict Detection: If experts produce conflicting solutions (i.e., $|\text{unique}(\mathcal{D})| > 1$), the system initiates a structured debate mechanism to promote expert collaboration and consensus formation.

Iterative Refinement: The **multi-round debate process** sets fixed debate rounds T, allowing experts to refine their reasoning through mutual learning and criticism. In round t of debate, each expert i updates their answer based on other experts' current solutions:

$$s_i^{(t+1)} = \text{Debate}_i(x_i', s_i^{(t)}, \{s_j^{(t)} : j \neq i, j \in \mathcal{A}\}), \quad (13)$$

where Debate_i is a specially designed debate prompt that guides expert i to analyze other experts' solutions, identify potential flaws in their own reasoning, and improve their current answer accordingly. This process simulates collaborative discussion in human expert teams, achieving the emergence of collective intelligence through multi-perspective exchange and criticism.

Consensus Formation: Final weighted voting integrates expert opinions after debate. After T rounds of debate, the system uses activation probability weights w_i to perform weighted voting on experts' final solutions:

Algorithm 1: Mixture-of-Trees Reasoning

Require: Problem x, expert role set R, gating network parameters θ, threshold τ, minimum activation count k, debate rounds T
Ensure: Final answer s_{final}

✦ **Expert activation and weight computation**
$P \leftarrow \{\text{Gate}_\theta(x, r_i) : r_i \in R\}$
$\mathcal{A} \leftarrow \{i : p_i \geq \tau\} \cap \{\text{top-}k(\{p_j\})\}$
$W \leftarrow \{w_i : i \in \mathcal{A}\}$
✦ **Shared baseline generation**
$s_{\text{base}} \leftarrow T_{\text{shared}}(x)$
✦ **Early stopping validation**
votes $\leftarrow \{\text{Validate}_i(x, s_{\text{base}}) : i \in \mathcal{A}\}$
if $\sum_{i \in \mathcal{A}, v_i = \text{approve}} w_i > \sum_{i \in \mathcal{A}, v_i = \text{reject}} w_i$ **then**
  return s_{base}
end if
✦ **Complete expert reasoning**
for $i \in \mathcal{A}$ **do**
  $x_i' \leftarrow \text{enhance}(x, r_i)$
  $s_i^{(0)} \leftarrow T_i^{(r_i)}(x_i')$
end for
$S^{(0)} \leftarrow \{s_i^{(0)} : i \in \mathcal{A}\}$
✦ **Collaborative debate**
if $|\text{unique}(S^{(0)})| > 1$ **then**
  for $t = 0$ to $T - 1$ **do**
    for $i \in \mathcal{A}$ **do**
      $s_i^{(t+1)} \leftarrow$ Eq. 13
    end for
  end for
  $S_{\text{final}} \leftarrow S^{(T)}$
else
  $S_{\text{final}} \leftarrow S^{(0)}$
end if
✦ **Final weighted voting**
$s_{\text{final}} \leftarrow \arg\max_{s \in S_{\text{final}}} \sum_{i: s_i = s, i \in \mathcal{A}} w_i$
return s_{final}

$$s_{\text{final}} = \arg\max_{s \in \mathcal{D}^{(T)}} \sum_{i: s_i^{(T)} = s, i \in \mathcal{A}} w_i, \tag{14}$$

where $\mathcal{D}^{(T)}$ is the solution set after round T of debate. This design ensures that experts with high activation probabilities (i.e., experts more suitable for the current problem) have greater influence in final decision-making, while improving overall solution quality and consistency through the debate mechanism.

Overall Algorithm

➤ **System Integration:** Our MoT framework combines all the above components to form an end-to-end reasoning system. The complete algorithmic flow is as shown Alg. 1

MoT first computes activation probabilities for each expert role through the gating network and selects top-k experts exceeding the threshold to participate in reasoning; then generates a shared baseline answer and performs quick validation—if weighted validation passes, directly returns the baseline answer (early stopping mechanism); otherwise, activated experts each generate initial solutions based on enhanced problems; if conflicting answers exist, experts engage in T rounds of collaborative debate, with each round considering other experts' viewpoints to optimize their own solutions; finally, the final answer is determined through weighted voting.

Experiments

Experimental Setup

Datasets. We select five benchmark datasets: **GSM8K** (Cobbe et al. 2021) , **MATH** (Hendrycks et al. 2021), **AIME 2024** (Zhang 2025), **MMLU** (Hendrycks et al. 2020), and **HotpotQA** (Yang et al. 2018), covering basic arithmetic reasoning, high-difficulty mathematical reasoning, cross-domain and multi-hop comprehensive reasoning.

Baselines. We compare MoT with several state-of-the-art reasoning methods, including Chain-of-Thought (CoT) for single-path step-by-step reasoning, Self-Consistency CoT (SC-CoT) with multi-path generation and voting, Tree-of-Thought (ToT) with beam width 5, Buffer-of-Thought (BoT) using meta-buffers to store thought templates, XoT integrating reinforcement learning and Monte Carlo tree search with 3 rounds, and Forest-of-Thought (FoT) simultaneously expanding 4 reasoning trees. Our MoT framework uses a default configuration of 12 expert trees with $k = 4$ sparse activation strategy and $T = 2$ rounds of debate mechanism.

Implementation Details. All experiments are conducted under unified settings to ensure fair comparison: using GPT-4o-mini and Qwen2.5-7B (Qwen et al. 2025) as base models, employing zero-shot CoT prompts, setting maximum token length to 2048 and temperature to 0.0 to ensure deterministic outputs. Datasets are split into training-test sets with a 2:8 ratio, and data augmentation is performed through semantic paraphrasing.

Accuracy Experiments

Table 2 shows the accuracy comparison between MoT and various baseline methods across five datasets. MoT achieves the best performance across all tasks and model configurations, with an average improvement of 2-7 percentage points compared to the strongest baseline FoT. Notably, MoT demonstrates exceptional performance on **competition-level reasoning tasks** (AIME 2024), achieving 40.0% accuracy on GPT-4o-mini compared to FoT's 33.3% (6.7 percentage point improvement), and improving from 23.3% to 26.7% on Qwen2.5-7B. This indicates that complex problems benefit more from multi-expert collaboration and structured debate.

Comparing different models' performance, we find that **Qwen2.5-7B achieves larger performance improvement margins compared to GPT-4o-mini**, a phenomenon consistent across all datasets. For example, on the MATH dataset, Qwen2.5-7B improves from baseline CoT's 51.8% to MoT's 58.1% (6.3 percentage point increase), while GPT-4o-mini only improves from 70.2% to 80.3% (10.1 percentage point increase but smaller relative improvement). This result suggests that weaker base models can more fully utilize multi-expert collaboration mechanisms, compensating for individual model limitations through knowledge complementarity and reasoning path diversification among experts,

Task Category	Dataset	Model	CoT (%)	SC-CoT (%)	ToT (b=5) (%)	BoT (%)	XoT (w/3 r) (%)	FoT (n=4) (%)	MOT (k=4, T=2) (%)
Basic Math	GSM8K	GPT-4o mini	89.3	90.1	91.8	92.4	93.1	95.5	**96.2**
		Qwen2.5-7B	85.4	87.2	88.5	89.1	90.3	92.8	**93.5**
Advanced Math	MATH	GPT-4o mini	70.2	71.6	73.1	74.1	74.8	78.5	**80.3**
		Qwen2.5-7B	51.8	53.5	54.2	54.9	55.4	57.2	**58.1**
Competition-level	AIME 2024	GPT-4o mini	8.7	10.3	13.3	16.7	16.7	33.3	**40.0**
		Qwen2.5-7B	6.9	8.2	10.7	13.3	16.7	23.3	**26.7**
General Reasoning	MMLU	GPT-4o mini	82.0	83.1	84.5	86.2	88.1	91.6	**93.9**
		Qwen2.5-7B	74.6	75.4	76.8	78.3	80.7	85.2	**88.8**
Multi-hop QA	HotpotQA	GPT-4o mini	68.1	71.6	73.2	74.8	76.5	78.9	**81.1**
		Qwen2.5-7B	67.9	69.8	71.1	72.5	74.2	76.1	**78.2**

Table 2: Accuracy comparison of MoT and baseline methods across five datasets.

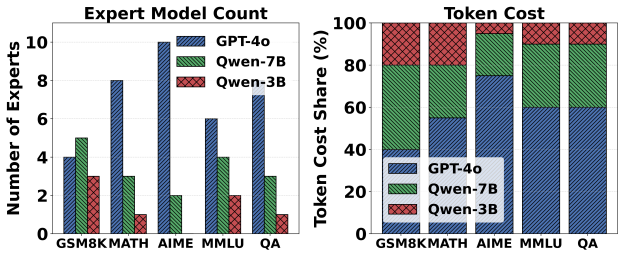

Figure 2: Expert model allocation and token cost across datasets. Left: simpler tasks (e.g., GSM8K) activate more lightweight models, while harder tasks (e.g., AIME) use high-capacity models. Right: token cost shifts toward larger models for challenging tasks.

demonstrating MoT framework's good adaptability and universality across different model scales.

Cost Experiments

Table 3 shows the cost-performance trade-off comparison of different reasoning methods on GSM8K and MMLU datasets. MoT achieves optimal cost-effectiveness through intelligent expert model allocation: maintaining the same cost level as ToT on GSM8K ($0.011) while achieving higher accuracy (96.2% vs 91.8%), and reducing cost by 25% on MMLU ($0.009 vs $0.012) while significantly improving accuracy (93.9% vs 84.5%). Compared to the computationally expensive FoT method, MoT reduces LLM calls by 40.6% and 37.0% on GSM8K and MMLU respectively, while maintaining the highest accuracy, fully demonstrating the effectiveness of sparse activation and early stopping mechanisms.

From the efficiency score perspective, MoT achieves significantly leading performance on both datasets. On GSM8K, the efficiency score reaches 87.4, improving 4.8% over ToT and 138% over FoT; on MMLU, the efficiency score is 104.3, improving 48% over ToT and 162% over FoT. This result validates the advantages of MoT's heterogeneous model strategy: by assigning appropriate models (GPT-4o-mini, Qwen2.5-7B/3B) to experts of different complexity levels, it significantly reduces computational costs while ensuring reasoning quality. Meanwhile, the shared baseline tree combined with lightweight verification mechanisms enables many simple problems to quickly obtain correct answers, avoiding unnecessary expert collaboration overhead and achieving optimal balance between cost and performance.

Dataset	Method	LLM Calls	Acc(%)	Cost($)	ES†
GSM8K	ToT (b=5)	**13.74**	91.8%	0.011	83.4
	FoT (n=4)	32.32	95.5%	0.026	36.7
	MoT (Ours)	19.21	**96.2%**	**0.011**	**87.4**
MMLU	ToT (b=5)	**15.21**	84.5%	0.012	70.4
	FoT (n=4)	28.67	91.6%	0.023	39.8
	MoT (Ours)	18.05	**93.9%**	**0.009**	**104.3**

Table 3: Cost-performance trade-off comparison. †Efficiency Score (ES) = Accuracy / Cost.

Method Opt.	Expert Debate	Sparse Activation	Early Stopping	Expert Trees	Acc (%)	LLM Invoked
MoT	✓				82.8	98.45
MoT	✓	✓			81.4	33.67
MoT	✓	✓	✓		79.2	23.91
MoT	✓	✓	✓	✓	**81.1**	**20.23**

Table 4: Ablation study of MoT components on HotpotQA. Sparse activation and early stopping greatly cut computation (LLM calls from 98.45 to 20.23), while expert trees help recover accuracy, highlighting their complementary roles in balancing efficiency and performance.

Furthermore, the **Expert Model Count & Token Cost** Fig. 2 reveals the **correlation between task difficulty and model allocation strategy**. In simple tasks (e.g., GSM8K), the system tends to activate more lightweight models (Qwen-7B/3B) to reduce costs, while in high-difficulty tasks (e.g., AIME), the number of high-performance model (GPT-4o) experts and token consumption ratio significantly increase to ensure reasoning quality. This result validates MoT's **adaptive expert allocation and cost control mechanism**: it can dynamically adjust the participation ratio of different models based on problem complexity, achieving efficient performance-cost trade-offs.

Ablation Studies

We conduct systematic ablation experiments on the HotpotQA dataset to analyze the contribution of each core component in the MoT framework. The experimental results show that the sparse activation mechanism plays a crucial role: without this mechanism, all 12 experts are simultaneously activated to participate in multi-round debates, leading to computational overhead surge to 98.45 LLM calls. After

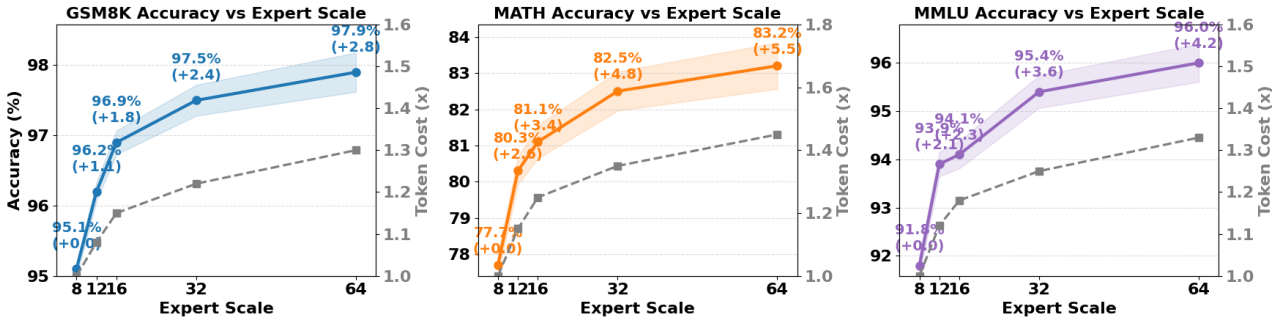

Figure 3: Accuracy Gains and Controlled Token Costs with Increasing Expert Scale in MoT across GSM8K, MATH, and MMLU Accuracy improves steadily as expert scale increases, with diminishing returns beyond 32 experts, while token costs rise only slightly due to MoT's sparse activation, demonstrating efficient scalability.

introducing sparse activation, the system intelligently selects the most relevant expert subset, dramatically reducing calls to 33.67, saving approximately 66% of computational cost with only a slight accuracy drop of 1.4% (from 82.8% to 81.4%). The early stopping mechanism further reduces calls from 33.67 to 23.91, saving an additional 26% of computational resources, particularly effective for problems solvable through simple reasoning.

The complete expert tree configuration demonstrates significant performance recovery capability, restoring accuracy from 79.2% to 81.1% while maintaining the lowest call count (20.23). This indicates that the specialized design of expert trees can maximize reasoning quality under resource constraints. Overall, the complete MoT framework achieves 79.5% computational cost savings compared to the baseline configuration through synergistic effects of four components, while maintaining competitive accuracy performance, fully demonstrating the rationality of each component design and the superiority of the overall architecture.

Expert Activation and Scalability

We conduct two sets of experiments to deeply analyze the dynamic behavior of the MoT framework in expert activation and scalability. First, expert activation experiments under different task difficulties show that as task complexity increases from GSM8K to AIME in Fig. 4, the number of activated experts significantly increases (from 3.5 to 8.7, $\tau = 0.5$), demonstrating MoT's adaptive expert allocation capability. Meanwhile, accuracy improves as the activation threshold decreases, but exhibits diminishing marginal returns: accuracy improvement from $\tau = 0.9$ to $\tau = 0.7$ is significant, while further reduction to $\tau = 0.5$ yields diminishing gains. This indicates that MoT can dynamically adjust reasoning resources based on problem complexity and achieve optimal balance between accuracy and computational cost through reasonable threshold control.

Expert scaling experiments as shown in Fig. 3 further validate MoT's scalability and cost controllability. As expert scale increases from 8 to 64, accuracy on three tasks (GSM8K, MATH, MMLU) steadily rises but with gradually diminishing improvements, showing typical diminishing marginal returns. Notably, token costs increase only slightly and limitedly with expert scale, especially more gradual in simple tasks (GSM8K). This result demonstrates that MoT's sparse activation mechanism effectively controls computational overhead from scaling expansion, enabling the system to maintain efficient resource utilization while expanding the expert pool, achieving the dual goals of accuracy improvement and efficiency balance.

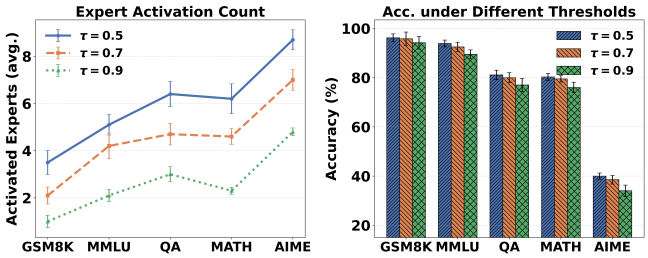

Figure 4: Impact of activation threshold (τ) on expert count and accuracy. Lower τ activates more experts, especially for harder tasks, showing MoT's adaptive scaling. Accuracy improves with lower τ via expert collaboration but plateaus beyond 0.7, indicating an accuracy-efficiency trade-off.

Conclusion

This paper presents Mixture-of-Trees (MoT), a unified framework that combines sparse expert activation with structured tree-based reasoning to balance reasoning quality and computational efficiency. Through a unified gating network, MoT dynamically selects and weighs expert reasoning trees based on task characteristics, while a shared baseline tree with early stopping enables adaptive and cost-aware inference. Experiments across five benchmarks demonstrate that MoT consistently improves accuracy by 2–7 percentage points while reducing LLM calls by around 40% compared to prior multi-path methods.

Beyond efficiency gains, MoT introduces a shift in perspective—from parameter-level MoEs to reasoning-level specialization, enabling models to decide how to think rather than merely what to predict. Future work will explore hierarchical gating, multimodal expert collaboration, and self-adaptive expert learning.

Acknowledgements

This work is partially supported by the IDT Young Doctoral Innovation Program (S203.2.01.23.001).

References

Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; and Hoefler, T. 2024. Graph of Thoughts: Solving Elaborate Problems with Large Language Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17682–17690. Association for the Advancement of Artificial Intelligence (AAAI).

Bi, Z.; Han, K.; Liu, C.; Tang, Y.; and Wang, Y. 2024. Forest-of-thought: Scaling test-time compute for enhancing llm reasoning. *arXiv preprint arXiv:2412.09078*.

Chen, Q.; Qin, L.; Liu, J.; Peng, D.; Guan, J.; Wang, P.; Hu, M.; Zhou, Y.; Gao, T.; and Che, W. 2025. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ding, R.; Zhang, C.; Wang, L.; Xu, Y.; Ma, M.; Zhang, W.; Qin, S.; Rajmohan, S.; Lin, Q.; and Zhang, D. 2024. Everything of Thoughts: Defying the Law of Penrose Triangle for Thought Generation. In Ku, L.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 1638–1662. Bangkok, Thailand: Association for Computational Linguistics.

Du, N.; Huang, Y.; Dai, A. M.; Tong, S.; Lepikhin, D.; Xu, Y.; Krikun, M.; Zhou, Y.; Yu, A. W.; Firat, O.; et al. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International conference on machine learning*, 5547–5569. PMLR.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The LLaMA 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.

Eigen, D.; Ranzato, M.; and Sutskever, I. 2013. Learning factored representations in a deep mixture of experts. *arXiv preprint arXiv:1312.4314*.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Lepikhin, D.; Lee, H.; Xu, Y.; Chen, D.; Firat, O.; Huang, Y.; Krikun, M.; Shazeer, N.; and Chen, Z. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*.

Li, J.; Zhang, Q.; Yu, Y.; Fu, Q.; and Ye, D. 2024. More agents is all you need. *Transactions on Machine Learning Research*.

Liang, T.; He, Z.; Jiao, W.; Wang, X.; Wang, Y.; Wang, R.; Yang, Y.; Shi, S.; and Tu, Z. 2023. Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate. *arXiv preprint arXiv:2305.19118*.

Liu, Z.; Gou, Y.; Chen, K.; Hong, L.; Gao, J.; Mi, F.; Zhang, Y.; Li, Z.; Jiang, X.; Liu, Q.; and Kwok, J. T. 2025. Mixture of insighTful Experts (MoTE): The Synergy of Thought Chains and Expert Mixtures in Self-Alignment. arXiv:2405.00557.

Mo, S.; and Xin, M. 2024. Tree of Uncertain Thoughts Reasoning for Large Language Models. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 12742–12746.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*.

Plaat, A.; Wong, A.; Verberne, S.; Broekens, J.; van Stein, N.; and Back, T. 2024. Reasoning with large language models, a survey. *arXiv preprint arXiv:2407.11511*.

Qwen; :; Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; Lin, H.; Yang, J.; Tu, J.; Zhang, J.; Yang, J.; Yang, J.; Zhou, J.; Lin, J.; Dang, K.; Lu, K.; Bao, K.; Yang, K.; Yu, L.; Li, M.; Xue, M.; Zhang, P.; Zhu, Q.; Men, R.; Lin, R.; Li, T.; Tang, T.; Xia, T.; Ren, X.; Ren, X.; Fan, Y.; Su, Y.; Zhang, Y.; Wan, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; and Qiu, Z. 2025. Qwen2.5 Technical Report. arXiv:2412.15115.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q. V.; Hinton, G. E.; and Dean, J. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *ICLR*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Wang, J.; Wang, J.; Athiwaratkun, B.; Zhang, C.; and Zou, J. 2024. Mixture-of-Agents Enhances Large Language Model Capabilities. *arXiv preprint arXiv:2406.04692*.

Wang, X.; Zhou, D.; Wei, J.; et al. 2022. Self-Consistency Improves Chain of Thought Reasoning in Language Models. *arXiv preprint arXiv:2203.11171*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems(NIPS)*, 24824–24837. Red Hook, NY, USA.

Wei, Y.; Huang, Z.; Zhao, F.; Feng, Q.; and Xing, W. W. 2025. MECoT: Markov Emotional Chain-of-Thought for Personality-Consistent Role-Playing. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Findings of the Association for Computational Linguistics: ACL 2025*, 8297–8314. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-256-5.

Wu, X.; Tao, X.; Wu, W.; Li, Y.; and Li, L. 2025. Random Forest-of-Thoughts: Uncertainty-aware Reasoning for Computational Social Science. *ArXiv*, abs/2502.18729.

Yang, L.; Yu, Z.; Zhang, T.; Cao, S.; Xu, M.; Zhang, W.; Gonzalez, J. E.; and Cui, B. 2024. Buffer of thoughts: Thought-augmented reasoning with large language models. *Advances in Neural Information Processing Systems*, 37: 113519–113544.

Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W. W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T. L.; Cao, Y.; and Narasimhan, K. 2023. Tree of thoughts: deliberate problem solving with large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems(NIPS)*, 11809–11822. Red Hook, NY, USA.

Zhang, D. 2025. AIME_1983_2024 Dataset (including AIME 2024).

Zhou, Y.; Lei, T.; Liu, H.; Du, N.; Huang, Y.; Zhao, V.; Dai, A. M.; Le, Q. V.; Laudon, J.; et al. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35: 7103–7114.