NEURAL INFORMATION PROCESSING SYSTEMS 2025

Learning to Think: Information-Theoretic Reinforcement Fine-Tuning for LLMs

Jingyao Wang*, Wenwen Qiang*, Zeen Song*, Changwen Zheng, Hui Xiong

Abstract

Large language models (LLMs) excel at complex tasks thanks to advances in their reasoning abilities. However, existing methods overlook the trade-off between reasoning effectiveness and efficiency, often encouraging unnecessarily long reasoning chains and wasting tokens. To address this, we propose Learning to Think (L2T), an information-theoretic reinforcement fine-tuning framework for LLMs to make the models achieve optimal reasoning with fewer tokens. Specifically, L2T treats each query-response interaction as a hierarchical session of multiple episodes and proposes a universal dense process reward, i.e., quantifies the episode-wise information gain in parameters, requiring no extra annotations or task-specific evaluators. We propose a method to quickly estimate this reward based on PAC-Bayes bounds and the Fisher information matrix. Theoretical analyses show that it significantly reduces computational complexity with high estimation accuracy. By immediately rewarding each episode's contribution and penalizing excessive updates, L2T optimizes the model via reinforcement learning to maximize the use of each episode and achieve effective updates. Empirical results on various reasoning benchmarks and base models demonstrate the advantage of L2T across different tasks, boosting both reasoning effectiveness and efficiency.

Contributions

1. We explore the trade-off between reasoning effectiveness and efficiency and propose Learning to Think (L2T), an information-theoretic reinforcement fine-tuning framework for LLMs.
2. We propose a universal information-theoretic process reward based on internal model signals, eliminating the need for external annotations or specialized evaluators. Leveraging PAC-Bayes bounds and the Fisher matrix, we derive a scalable approximation of the intractable information gain with theoretical guarantees.
3. Across diverse complex reasoning benchmarks and base models, L2T consistently achieves great performance, delivering boosts in both effectiveness and efficiency.

Project Page

https://wangjingyao07.github.io/L2T.github.io/

Motivation and Analysis

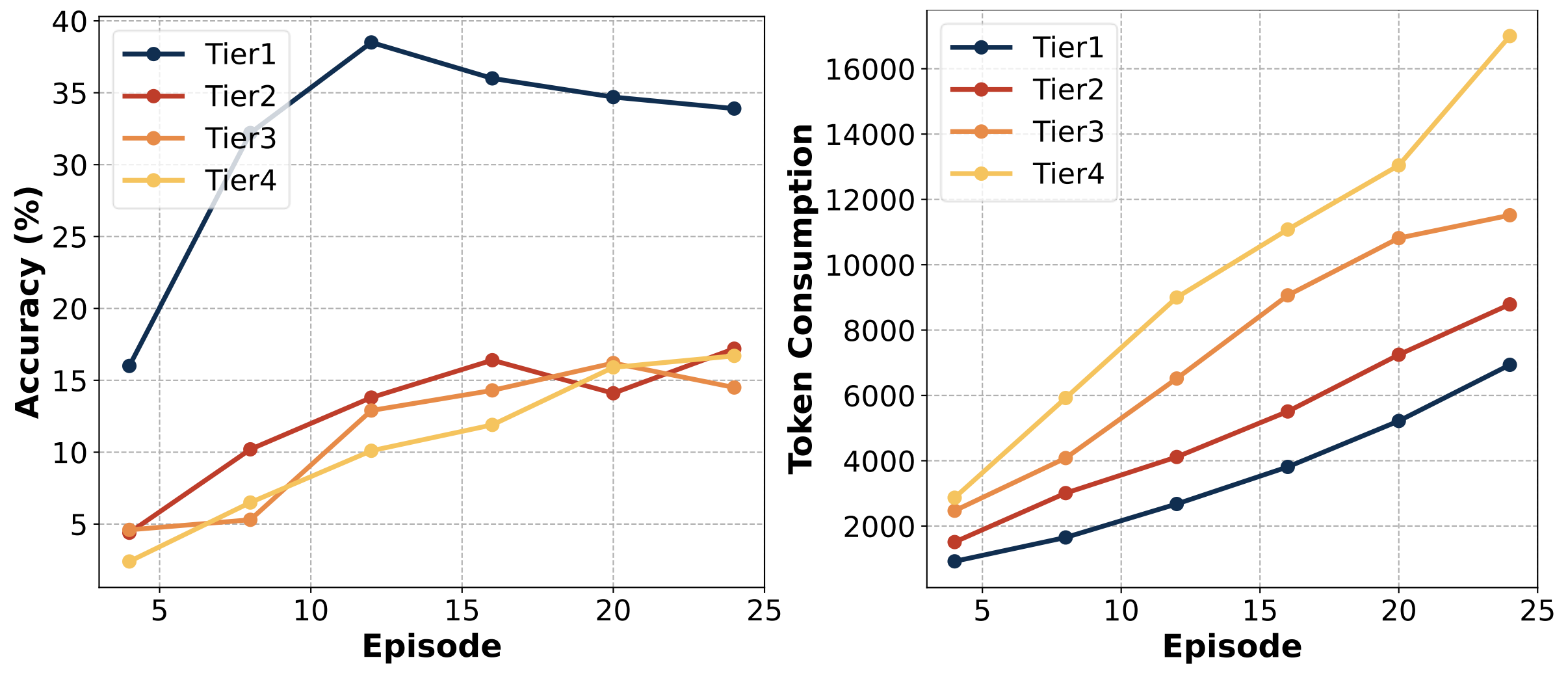


Figure 1: Results of DeepScaleR-1.5B-Preview on Omni-MATH. Left: Reasoning accuracy vs. episode number. Right: Token consumption vs. episode number.

Recent studies show that scaling test-time compute can markedly improve LLM reasoning, with longer inference producing logarithmic–linear accuracy gains. Building on this, a new class of models combines test-time compute scaling with RL, achieving comparable results on challenging benchmarks. These models rely on CoT tokens to structure multi-step reasoning, and by extending CoT paths beyond typical solution lengths, they explore the solution space more thoroughly and improve accuracy.

However, they still struggle to balance reasoning effectiveness and efficiency. Current approaches optimize only for final outcome rewards, providing no feedback on intermediate steps. With such delayed supervision, extra reasoning incurs no cost, so even tiny gains from large numbers of additional steps are treated as positive. As a result, models tend to adopt a "one more thought" strategy, excessively lengthening CoTs, which leads to redundant computation and reduced efficiency. Our experiments confirm this: outcome-reward-based RL often makes LLMs consume more than twice the tokens actually required, and in some cases, this redundancy even harms accuracy. Because task difficulty varies, no fixed chain length works universally.

Therefore, it is crucial to design dense process rewards that evaluate the contribution of each reasoning step. Such rewards guide the model to generate only the tokens that improve the answer, achieving effective reasoning with minimal budgets.

Methodology

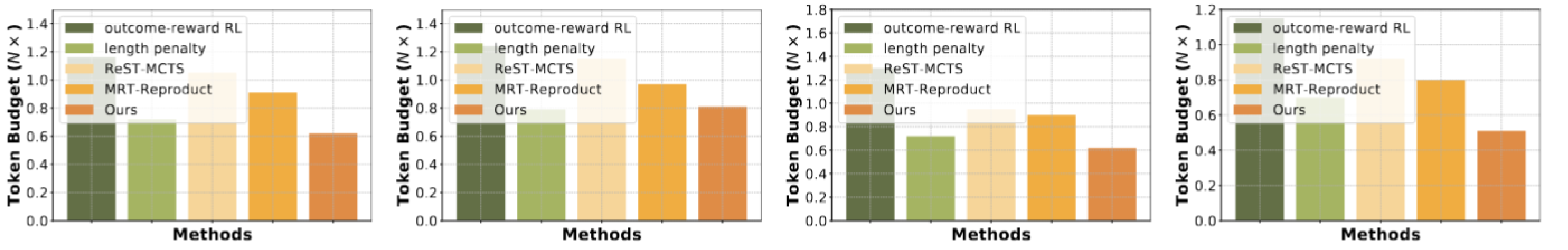
We propose Learning to Think (L2T), an information-theoretic reinforcement fine-tuning framework designed to improve both the effectiveness and efficiency of reasoning. At its core, L2T introduces a dense process reward that quantifies the incremental information gain in model parameters during reasoning. This reward combines two complementary components: (i) a fitting term that encourages the model to capture correctness-critical information in each update; and (ii) a compression penalty that discourages redundant optimization, preserving efficiency. By decomposing each question–answer process into multiple episodes and rewarding each step immediately, L2T guides the model to focus on progressive improvements, thereby curbing unnecessary reasoning steps and reducing computational overhead. To further improve efficiency, we leverage PAC-Bayes bounds together with the Fisher Information Matrix to approximate the information gain, making the process both theoretically grounded and computationally tractable. The final training objective integrates both outcome-level rewards (reflecting final correctness) and process-level rewards (capturing per-step informational progress), optimized under a token budget constraint. In practice, L2T is instantiated on GRPO, where episodic rewards are distributed to tokens via log-probability surprise, and optimized using clipped policy gradients with KL regularization. This enables reinforcement-guided reasoning that is both efficient and accurate across diverse tasks.

Experimental Results

Empirically, L2T delivers consistent gains across various reasoning benchmarks: Compared to outcome-reward methods, it improves accuracy by 3.7% and doubles token efficiency; against process-reward baselines, it yields about 2% higher accuracy and 1.3× greater efficiency. In multi-task evaluations, L2T achieves an average 3% accuracy gain across tasks of varying difficulty while maintaining stable performance under different token budgets. These highlight L2T's ability to balance reasoning effectiveness and efficiency across diverse scenarios.

Table 1: Pass@1 performance on various math reasoning benchmarks. We compare base models trained with different fine-tuning approaches. The best results are highlighted in **bold**.

Base model + Method	AIME 2024	AIME 2025	AMC 2023	MATH500	MinervaMATH	**Avg.**
DeepScaleR-1.5B-Preview	42.8	36.7	83.0	85.2	24.6	54.5
+outcome-reward RL (GRPO)	44.5 (+1.7)	39.3 (+2.6)	81.5 (-1.5)	84.9 (-0.3)	24.7 (+0.1)	55.0 (+0.5)
+length penalty	40.3 (-2.5)	30.3 (-6.4)	77.3 (-5.7)	83.2 (-2.0)	23.0 (-1.6)	50.8 (-3.7)
+ReST-MCTS	45.5 (+2.7)	39.5 (+2.8)	83.4 (+0.4)	84.8 (-0.4)	23.9 (-0.7)	55.4 (+0.9)
+MRT	47.2 (+4.4)	39.7 (+3.0)	83.1 (+0.1)	85.1 (-0.1)	24.2 (-0.4)	55.9 (+1.4)
+Ours	**48.5 (+5.7)**	**40.2 (+3.5)**	**85.4 (+2.4)**	**88.1 (+2.9)**	**26.5 (+1.9)**	**57.8 (+3.3)**
DeepSeek-R1-Distill-Qwen-1.5B	28.7	26.0	69.9	80.1	19.8	44.9
+outcome-reward RL (GRPO)	29.8 (+1.1)	27.3 (+1.3)	70.5 (+0.6)	80.3 (+0.2)	22.1 (+2.3)	46.0 (+1.1)
+length penalty	27.5 (-1.2)	22.6 (-3.4)	64.4 (-5.5)	77.1 (-3.0)	18.8 (-1.0)	42.0 (-2.9)
+ReST-MCTS	30.5 (+1.8)	28.6 (+2.6)	72.1 (+1.2)	80.4 (+0.3)	20.3 (+0.5)	46.4 (+1.5)
+MRT	30.3 (+1.6)	29.3 (+3.3)	72.9 (+3.0)	80.4 (+0.3)	22.5 (+2.7)	47.1 (+2.2)
+Ours	**32.9 (+4.2)**	**30.1 (+4.1)**	**73.5 (+3.6)**	**84.7 (+4.6)**	**24.5 (+4.7)**	**49.2 (+4.3)**



(a) AIME (b) AMC (c) MATH500 (d) MinervaMATH

Figure 2: Efficiency comparison across different benchmarks. We compute the token budget required for each benchmark and treat the budget of the base model w/o fine-tuning as reference (1×).