ThinkLess: A Training-Free Inference-Efficient Method for Reducing Reasoning Redundancy

Anonymous ACL submission

Abstract

While Chain-of-Thought (CoT) prompting improves reasoning in large language models (LLMs), the excessive length of reasoning tokens increases latency and KV cache memory usage, and may even truncate final answers under context limits. We propose **ThinkLess**, an inference-efficient framework that terminates reasoning generation early and maintains output quality without modifying the model. Attention analysis reveals that answer tokens focus minimally on earlier reasoning steps and primarily attend to the reasoning terminator token, due to information migration under causal masking. Building on this insight, ThinkLess inserts the terminator token at earlier positions to skip redundant reasoning while preserving the underlying knowledge transfer. To prevent format discruption casued by early termination, ThinkLess employs a lightweight post-regulation mechanism, relying on the model's natural instruction-following ability to produce well-structured answers. Without fine-tuning or auxiliary data, ThinkLess achieves comparable accuracy to full-length CoT decoding while greatly reducing decoding time and memory consumption.

1 Introduction

Large language models (LLMs) (Vaswani et al., 2017; Zhang et al., 2025b) have achieved remarkable progress in natural language understanding and generation, but still struggle with tasks requiring multi-step reasoning. Chain-of-Thought (CoT) prompting (Wei et al., 2022) has emerged as a popular approach to address this issue, enabling models to decompose problems into intermediate reasoning steps before producing an answer.

While CoT improves accuracy on challenging benchmarks (Zhang et al., 2022; Jaech et al., 2024), it comes at a cost: reasoning tokens tend to be long and autoregressively generated, introducing substantial latency and memory overhead during inference. As shown in Figure 1, increasing the token budget does improve accuracy–but the gains diminishes rapidly, indicating clear marginal returns. Beyond a certain point, longer generations incur significantly higher computational cost without meaningful performance improvement. In deployment scenarios where user experience and response time are critical, such overhead becomes a practical bottleneck, making blind expansion of reasoning length both inefficient and unsustainable.

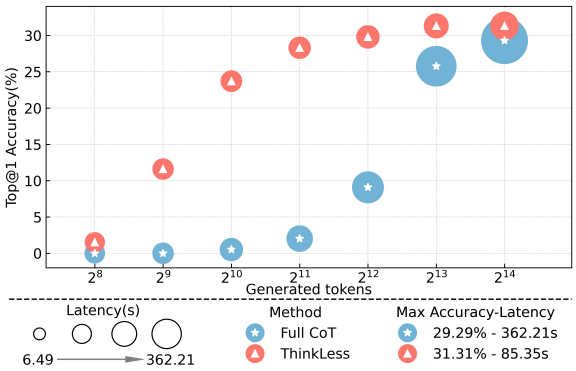


Figure 1: GPQA (Rein et al., 2024) accuracy of DeepSeek-R1-Distill-LLaMA-8B (Guo et al., 2025) under varying token budgets. Red: ThinkLess (compressed reasoning); Blue: full CoT reasoning.The left part of the legend illustrates the relationship between marker size and latency, the middle part denotes each methods, and the right part presents the maximum accuracy and corresponding latency of each method.

Several efforts aim to improve CoT efficiency through techniques such as feedback-based refinement (Yao et al., 2023b), search and planning (Bi et al., 2024; Ye et al., 2024), and iterative optimization (Zhang et al., 2024). While effective in controlled settings, these approaches typically rely on *additional training*, *curated datasets*, or *supervised fine-tuning (SFT)*–introducing significant engineering overhead. Moreover, their reliance on task-specific data or model customization limits generalizability, making them difficult to scale or deploy in real-world systems where flexibility, modularity,

and minimal intervention are critical.

We introduce **ThinkLess**, an inference-efficient framework that reduces DeepSeek-R1 (Guo et al., 2025) distilled CoT reasoning overhead *without any model modification or additional training*. Our key insight stems from an attention analysis: during answer generation, models rely minimally on earlier reasoning steps and focus on disproportionately on the reasoning terminator tokens (*e.g.*, </think>). This indicates reasoning information is progressively migrated and compressed toward the end of the reasoning sequence due to causal attention (Lin et al., 2025).

However, naively truncating reasoning by inserting the terminator token early often results in disrupted output formats. To solve, ThinkLess employs a lightweight output regulation that guides the model to produce well-structured responses. This is implemented simply by appending a small instruction prompt after early termination, leveraging the model's inherent instruction-following capabilities. This post-regulation step requires no model modification or fine-tuning, yet proves essential for maintaining output consistency and restoring accuracy degraded by premature reasoning truncation.

ThinkLess achieves substantial efficiency gains. As illustrated in Figure 1, ThinkLess reaches strong performance at a much lower token budget compared to full CoT decoding, and further reduces inference latency, as reflected by smaller sizes. These results demonstrate that long-form reasoning is not always necessary; with proper output regulation, shortened reasoning can retain accuracy while dramatically improving inference efficiency.

Our contributions are as follows:

- We present an attention-based analysis revealing that answer tokens in CoT generation attend minimally to earlier reasoning steps, indicating substantial redundancy.
- We propose ThinkLess, a training-free early termination strategy that injects a reasoning terminator token to truncate redundant reasoning while preserving core information.
- To mitigate format disruption caused by early termination, we introduce a lightweight *output regulation* mechanism that restores structured answers using a minimal instruction prompt.
- ThinkLess achieves comparable performance than full CoT decoding with fewer tokens and lower inference cost, offering a plug-and-play solution deployable across models and tasks.

2 Related Work

2.1 LLMs Reasoning

Reasoning is a fundamental capability of LLMs, enabling them to tackle complex multi-step tasks across diverse domains (Qiao et al., 2022). To enhance this ability, recent work has explored various prompting and architectural strategies. Chain-of-Thought (CoT) prompting (Wei et al., 2022) has emerged as a foundational method, guiding models to generate intermediate reasoning steps before producing final answers. This decomposition of complex problems into sub-goals significantly improves performance on arithmetic, commonsense, and scientific reasoning benchmarks (Kojima et al., 2022; Feng et al., 2023; Rein et al., 2024; Lyu et al., 2023). Building on CoT, techniques such as Self-Consistency (Wang et al., 2022) aggregate multiple reasoning paths to improve robustness, while Tree-of-Thoughts (Yao et al., 2023a) introduces structured planning via tree-based exploration. More recently, advanced frameworks like OpenAI's o1, Alibaba's QwQ (Team, 2025), and DeepSeek's R1 (Guo et al., 2025) have extended CoT by incorporating reflective reasoning modes such as trial-and-error, backtracking, and self-correction (Shinn et al., 2023).

2.2 CoT Compression

While deeper reasoning improves performance, it often comes with diminishing returns and increasing computational cost (Chen et al., 2024; Wu et al., 2024). Excessively long reasoning sequences not only prolong inference but also strain memory and may even degrade output quality (Liu et al., 2025b,a). Recent work has thus focused on efficient CoT generation, which falls into two broad categories: training-based compression and inference-time optimization (Qu et al., 2025a; Sui et al., 2025). Training-based methods learn more compact reasoning traces through supervised fine-tuning. Some approaches compress CoT chains at the token level (Han et al., 2024; Xia et al., 2025), dynamically adjusting reasoning length based on task difficulty (Hao et al., 2024; Zhang et al., 2025a). Others replace explicit token-level reasoning with latent or abstract representations (Chen et al., 2024; Shen et al., 2025; Qu et al., 2025b), compressing the reasoning into a hidden state or

learned vector. Inference-time methods, by contrast, improve efficiency without modifying model weights. These include Sketch-of-Thought (Aytes et al., 2025; Xu et al., 2025), which generate concise draft reasoning before producing final outputs, balancing coherence and computational cost.

Our ThinkLess, aligns with this line of inference-time CoT optimization but differs by being entirely training-free and model-agnostic, particularly for DeepSeek-R1 distilled models. Rather than compressing reasoning through learning, ThinkLess truncates redundant reasoning tokens based on attention insights and restores output quality through a lightweight post-regulation mechanism.

3 Methodology

We present **ThinkLess**, a *training-free* framework designed to improve inference efficiency for CoT reasoning in LLMs. ThinkLess achieves this by (1) identifying redundancy in long reasoning traces via attention and hidden state analyses, and (2) introducing a lightweight termination and regulation mechanism that preserves output accuracy and format while significantly reducing decoding cost.

3.1 CoT Bottlenecks at Inference

Problem Formulation. Given a question q, LLM generates a sequence of tokens $x_{1:N}$ autoregressively, where each token x_i is sampled based on the conditional probability $p(x_i \mid q, x_{<i})$. In CoT prompting, this sequence can be divided into reasoning tokens $x^{\text{reason}}_{1:M}$ and answer tokens $x^{\text{answer}}_{1:N}$:

$$p(x^{\text{reason}}_{1:M} \mid q) = \prod_{i=1}^{M} p(x^{\text{reason}}_i \mid q, x^{\text{reason}}_{<i}) \quad (1)$$

$$p(x^{\text{answer}}_{1:N} \mid q, x^{\text{reason}}_{1:M}) = \prod_{i=1}^{N} p(x^{\text{answer}}_i \mid q, x^{\text{reason}}_{1:M}, x^{\text{answer}}_{<i}). \quad (2)$$

Inference-Time Bottlenecks. While reasoning tokens can enhance the model's ability to arrive at a more accurate answer during training, they introduce significant overhead during inference. Specifically, long reasoning sequences lead to increased computational costs, higher memory usage (due to the expanded KV cache (Qin et al., 2025)), and longer response times. This is particularly problematic in applications where quick answer responses are crucial, such as interactive AI systems.

Also, long reasoning paths may consume the context budget before generating answers, rendering the reasoning benefits inaccessible. This mismatch between computation and usable output severely undermines the efficacy of CoT at inference time. We empirically observe this issue in Figure 1, where the model's performance noticeably degrades when the total token length falls below 2^{13}. One key reason is that the answer segment is often truncated due to limited context, preventing the model from fully leveraging the reasoning process it has computed.

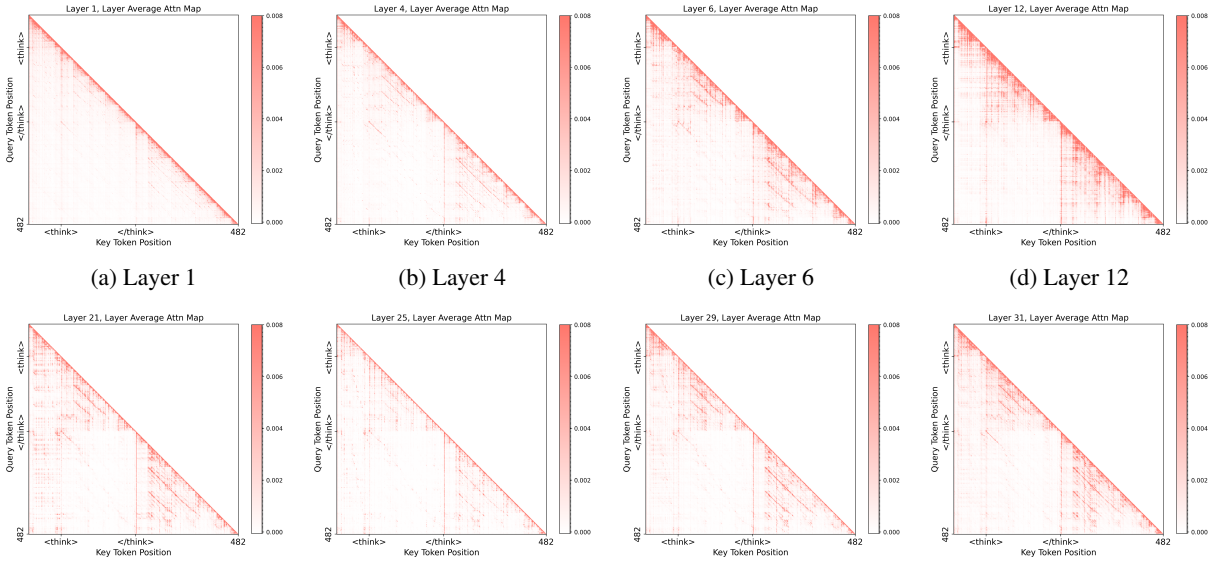
Motivation. These challenges expose a core inefficiency in current CoT generation: even if reasoning is computed, the final answer may not be delivered due to truncation, or its benefits may be outweighed by the added inference burden. These raises an important question: *how much of the reasoning is actually needed to support answer generation?* In Section 3.2, we examine the model's internal attention behavior during decoding to investigate this question more closely. Section 3.3 then presents a termination mechanism with minimal formatting disruption, enabling efficient and accurate CoT inference.

3.2 Attention Reveals Redundancy in CoT Reasoning

To understand why long-form CoT reasoning incurs high cost but limited benefit, we analzye the model's attention behavior during answer generation. Our goal is to examine whether all reasoning tokens are equally useful—or if, as we hypothesize, later reasoning tokens alone may carry the necessary information for generating accurate answers.

We visualize attention patterns across transformer layers using DeepSeek-R1-Distill-Llama-8B on GSM8K samples, as shown in Figure 2.

Each heatmap represents the attention weights from query tokens (rows) to key tokens (columns) during autoregressive decoding. The <think> and </think> tokens mark the boundaries of the reasoning span. In early layers, the model distributes attention broadly across the reasoning region, suggesting that its initially considers the full reasoning race. However, as depth increases, the model's focus sharpens toward the end-of-reasoning boundary, particularly the </think> token. This transition implies a progressive information migration phenomenon, where reasoning content is gradually compressed toward the end of the span.



(a) Layer 1 (b) Layer 4 (c) Layer 6 (d) Layer 12

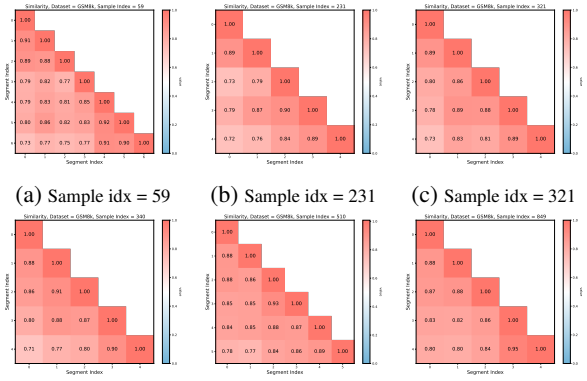
(e) Layer 21 (f) Layer 25 (g) Layer 29 (h) Layer 31

Figure 2: Attention heatmaps across different layers of DeepSeek-R1-Distill-LLaMA-8B on a GSM8K sample (Cobbe et al., 2021). Tokens within the <think>...</think> span receive uniform attention in early layers, but deeper layers gradually shift focus to the boundary tokens, indicating information migration and compression of reasoning content. Similar observations can be found in other models and datasets

We attribute this behavior to causal masking: under left-to-right generation, downstream tokens cannot access future context. As results, reasoning must be internally summarized and propagated forward token by token. This leads earlier reasoning tokens to fade from view, while later tokens—particularly </think>—accumulate and represent the distilled reasoning state. Similar phenomena have been explored by (Lin et al., 2025).

Analyzing Reasoning Redundancy. Building on the information migrration mechanism discussed above, we ask: *How early can useful reasoning be distilled during generation?* Since reasoning content is expected to progressively compress toward the end of the span (*e.g.*, </think>), we hypothesize that inserting this token at intermediate positions during decoding should yield hidden states that already approximate the final reasoning state. If true, this would suggest that the model has already internalized most of the reasoning content before completing the full chain.

To test this hypothesis, we conduct a similarity-based redundancy analysis. Specifically, we insert the </think> token at a fixed segment length of 16 tokens during the reasoning generation process using DeepSeek-R1-Distill-Qwen-7B. At each insertion point, we extract the last-layer hidden state of the </think> token, treating it as the representation of accumulated reasoning up to that step. We then compute pairwise cosine similarities between these intermediate hidden states.



(a) Sample idx = 59 (b) Sample idx = 231 (c) Sample idx = 321

(d) Sample idx = 340 (e) Sample idx = 510 (f) Sample idx = 849

Figure 3: We insert a </think> token every 16 tokens in DeepSeek-R1-Distill-Qwen-7B and extract last-layer hidden states. These states are highly similar (0.9) across segments, showing that reasoning adds little new information. The final state is also similar to earlier ones, indicating early convergence and redundancy in later reasoning. Similar observations can be found across other models and datasets. Best view with zooming in.

As shown in Figure 3, the similarity between adjacent reasoning segments remains consistently high (∼0.9), indicating that each additional segment introduces only marginal new information. Moreover, the similarity between the final </think> state and earlier ones increases steadily,

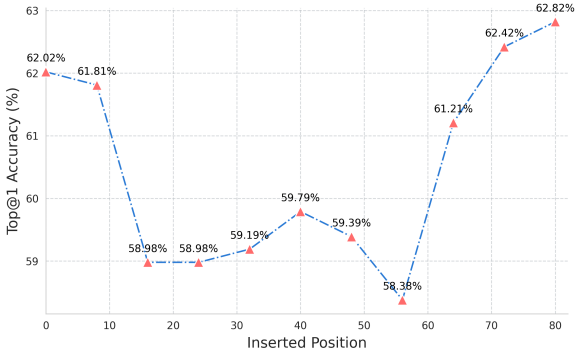


Figure 4: Accuracy of DeepSeek-R1-Distill-Qwen-7B *vs.* position where </think> is inserted. The benchmark is BBH dataset (Suzgun et al., 2022).

confirming the progressive nature of reasoning aggregation. Notably, even early inserted </think> tokens already yield hidden states highly similar to the final one—supporting the view that most useful reasoning content is distilled early, and extended CoT traces incur diminishing returns.

3.3 ThinkLess: Reasoning Termination and Output Regulation

Building on our earlier conclusion that the most useful reasoning content is distilled early, we ask: *can reasoning be safely truncated early without sacrificing answer quality?* Since the model gradually compresses reasoning into the </think> token, it may be possible to shorten the reasoning trace while still preserving essential information.

To verify this, we divide the full reasoning sequence into equal-length segments and insert the </think> token at varying cut-off points, thereby terminating reasoning at different locations. We then measure model accuracy across termination positions. Surprisingly, as shown in Figure 4, truncating reasoning early leads to a decreasing accuracy—despite our hypothesis that essential information should have already migrated toward the </think> token. This unexpected decline gradually recovers as the termination point moves later, forming a U-shaped performance curve.

A more detailed investigation in Sec. A shows that the observed decline in performance is not attributable to deficiencies in the model's reasoning process. Instead, the drop primarily arises from output formatting issues—such as the omission of the final answer or deviations from the expected response structure. These formatting errors can lead to incorrect evaluations, even when the model's internal reasoning is logically sound. Notably, after manually correcting these malformed outputs to align with the desired answer format, we find the underlying responses are indeed accurate, resulting in a substantial recovery in overall accuracy.

This confirms that the observed accuracy dip is a surface-level artifact: early termination disrupts output form, not semantic correctness. The model had already internalized the reasoning; it simply failed to express it in the expected format.

These results confirm that the model primarily relies on the </think> token to access reasoning information—rather than attending to every reasoning token individually. As a result, extending the reasoning span offers limited benefit, revealing substantial redundancy in long-form CoT.

ThinkLess Framework. We introduce **ThinkLess**, a simple, training-free framework to reduce CoT inference cost. The key idea is to insert the </think> token shortly after <think>, thereby skipping the majority of reasoning generation. This early termination substantially reduces decoding time and KV cache memory usage. However, such abrupt truncation may produce malformed answers that lack structural completeness.

To overcome this dilemma, ThinkLess employs a lightweight instruction-based output regulation step. For each task, we prepend a short instruction prompt (see Sec. B) to clarify output expectations. This approach leverages the strong instruction-following abilities of modern LLMs, enabling the model to produce well-structured responses—even in the absence of explicit reasoning. Since the added instruction is minimal, the overall inference cost remains low.

Clarification: ThinkLess Without Explicit Reasoning. ThinkLess inserts the </think> token right after <think>, thereby skipping the generation of any explicit CoT reasoning. At first glance, this appears to challenge the *information migration hypothesis*: if no intermediate reasoning tokens are produced, it is unclear what reasoning content, if any, is being transferred to inform the final answer.

We contend, however, that the </think> token serves a deeper function than a mere delimiter. It acts as a *semantic anchor*—a learned symbolic abstraction that implicitly encodes a compressed representation of the reasoning process. During pretraining, language models likely acquire the ability to internalize multi-step reasoning patterns and embed this abstracted knowledge into compact markers such as </think>. This hypothesis

is supported by our empirical observations: even when the reasoning trace is entirely omitted, the model frequently produces correct answers, indicating that the cognitive process of reasoning may have been executed internally and silently.

From this perspective, </think> does not denote the absence of reasoning, but rather the culmination of an *internalized* reasoning trajectory. It signals to the model that deliberation has concluded and that it should proceed to answer generation. This behavior can be interpreted as a form of *reasoning distillation*, in which the explicit explanatory steps are compressed into latent activations, allowing for both efficient inference and high-quality outputs without requiring full CoT generation.

4 Experiment

4.1 Datasets

To comprehensively evaluate our proposed method across diverse reasoning and knowledge-intensive scenarios, we conduct experiments on the following four benchmark datasets:

- **GSM8K** (Cobbe et al., 2021): A grade-school level math word problem dataset to assess arithmetic reasoning. Each question requires multi-step calculation and logical deduction.
- **MMLU** (Hendrycks et al., 2020): It covers 57 tasks across various domains including humanities, STEM, and social sciences, measuring general knowledge and reasoning ability.
- **GPQA** (Rein et al., 2024): A graduate-level physics question answering dataset targeting conceptual understanding. It tests model capability in high-level scientific reasoning.
- **BBH** (Suzgun et al., 2022): This subset focuses on difficult tasks that require multi-step, symbolic, or logical reasoning, offering a rigorous stress test for language models.

4.2 Metrics

We report three key evaluation metrics across all tasks to provide a comprehensive comparison of both performance and efficiency: Top-1 accuracy (Top@1↑), inference time (Time↓), and token usage (Tokens↓). Accuracy reflects the percentage of exact top-1 matches. All results are from a single run.

Given that ThinkLess omits the explicit reasoning, we also report Top-k accuracy (Top@k) ($k \geq 2$) for ThinkLess variants. In this setup, the model is allowed to generate k candidate answers for each question, and the response is considered correct if any of them is accurate. This allows us to assess ThinkLess under a relaxed evaluation regime, which reflects its ability to retain answer quality even when reasoning tokens are suppressed.

To ensure fair comparison, we constrain the total number of generated tokens in the Top@k setting to remain comparable to the token budget used by standard CoT decoding (*i.e.*, Top@1 with full reasoning). This enables an apples-to-apples evaluation of accuracy under equivalent costs of tokens.

4.3 Backbones and Baselines

To ensure a comprehensive and fair evaluation, we conduct experiments on publicly available LLMs within the 7B to 14B parameter scale. This range reflects the practical constraints imposed by our available GPU resources, while still covering models with strong reasoning capabilities.

Backbone. *Qwen2.5-7B/14B* (Yang et al., 2024): A family of powerful open-source instruction-tuned models known for their strong general reasoning abilities. *LLaMA3.1-8B* (Grattafiori et al., 2024): A well-balanced model from the LLaMA series that combines efficient inference with competitive instruction-following performance. All backbones are evaluated under identical decoding settings to ensure a consistent comparison.

Baselines. We compare ThinkLess against a single, strong baseline: the full CoT distilled variant. This model is obtained by distilling reasoning capabilities from a more powerful DeepSeek-R1, and it represents a high-performance upper bound.

ThinkLess requires no fine-tuning, no auxiliary data, and no changes to the underlying model weights. To our best knowledge, we are the first to offer such efficient CoT reasoning compression in a fully training-free manner. Given this setting, the distilled full CoT model provides the most appropriate and meaningful baseline for comparison.

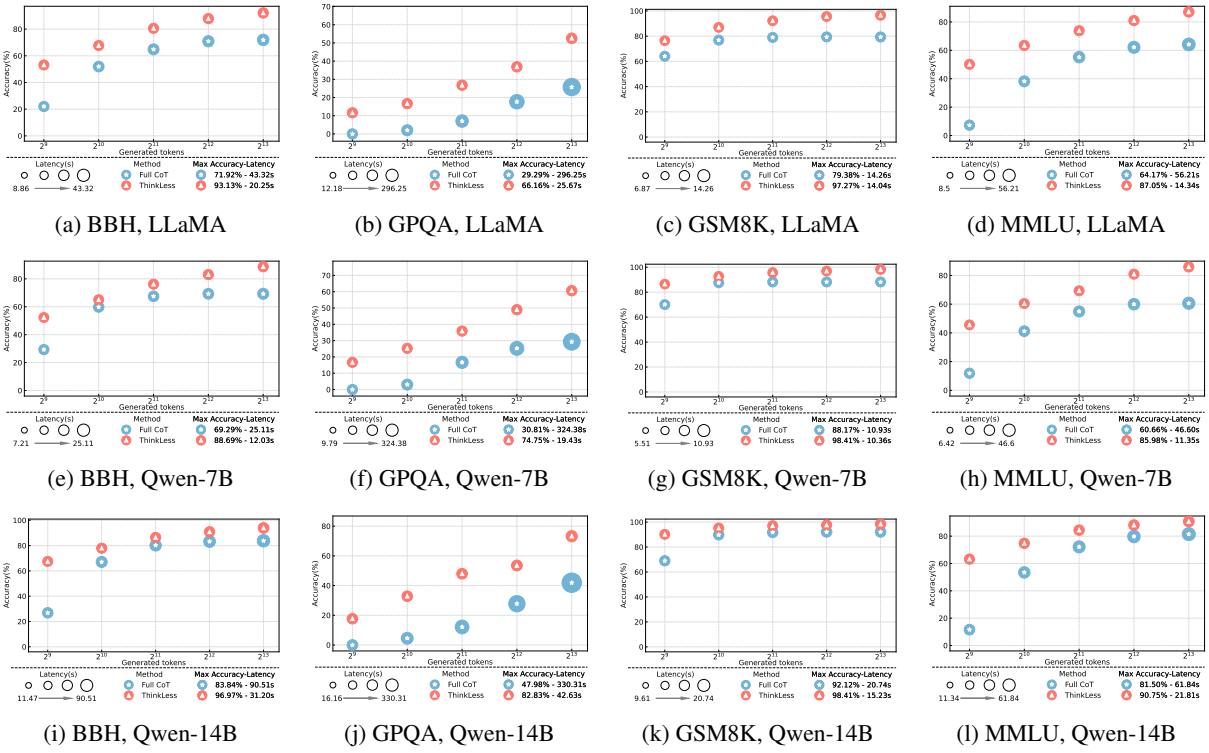
4.4 How Effective is ThinkLess?

Table 1 and Figure 5 present a detailed comparison between our proposed **ThinkLess** framework and the **Distill** baseline. The maximum token budget is set as 8k in Table 1. We detail accuracy, inference time, and token consumption below.

Comparable Accuracy Despite Omitting Reasoning. While ThinkLess entirely skips the visible CoT reasoning trace, its Top@1 accuracy re-

Method	GSM8K			MMLU			GPQA			BBH			AVG.		
	Top@1 ↑	Time↓	Tokens↓	Top@1 ↑	Time↓	Tokens↓	Top@1 ↑	Time↓	Tokens↓	Top@1 ↑	Time↓	Tokens↓	Top@1 ↑	Time↓	Tokens↓
Qwen2.5-7B															
Distill	88.17	10.62	438.92	60.86	47.01	1817.84	30.81	148.82	5523.17	69.29	24.79	976.08	62.28	57.81	2189.00
ThinkLess w/o Instruct	87.79	6.57	274.20	54.04	6.77	279.50	31.31	15.39	631.87	62.02	8.57	341.91	58.79	9.33	381.87
ThinkLess	88.40	5.46	235.41	57.06	9.07	370.34	40.91	14.59	591.17	65.25	9.34	379.32	62.91	9.62	394.06
Qwen2.5-14B															
Distill	92.12	20.37	508.40	81.40	62.20	1516.46	41.92	217.62	5205.02	83.84	55.48	1349.88	74.82	88.92	2144.94
ThinkLess w/o Instruct	92.42	9.94	252.49	75.95	12.06	300.33	39.39	24.69	612.79	76.36	11.04	275.33	71.03	14.43	360.24
ThinkLess	92.49	9.05	235.32	76.44	14.84	361.92	44.95	22.34	547.43	78.38	14.73	351.00	73.07	15.24	373.92
LLaMA3.1-8B															
Distill	79.38	12.95	493.70	64.07	56.69	2119.48	25.76	162.79	6094.77	71.92	33.21	1252.02	60.28	66.41	2489.99
ThinkLess w/o Instruct	79.76	6.99	270.19	57.55	7.85	298.14	30.30	15.75	600.88	65.45	8.28	315.91	58.27	9.72	371.28
ThinkLess	78.92	6.73	260.74	60.27	10.23	384.55	31.31	48.81	1817.93	71.92	11.45	430.89	60.61	19.31	723.53

Table 1: Comparison of our ThinkLess and DeepSeek-R1 distilled models.



(a) BBH, LLaMA (b) GPQA, LLaMA (c) GSM8K, LLaMA (d) MMLU, LLaMA

(e) BBH, Qwen-7B (f) GPQA, Qwen-7B (g) GSM8K, Qwen-7B (h) MMLU, Qwen-7B

(i) BBH, Qwen-14B (j) GPQA, Qwen-14B (k) GSM8K, Qwen-14B (l) MMLU, Qwen-14B

Figure 5: Top@k accuracy of ThinkLess *vs.* Top@1 accuracy of DeepSeek-distilled models across datasets and models. We set $k = \frac{\text{Token Budget}}{512}$ to match the token usage on par with distilled models. Legends follow Figure 1.

mains consistently close to that of the full CoT baseline. For example, with Qwen2.5-7B, ThinkLess achieves an average accuracy of 62.91%, compared to 62.28% from Distill. With Qwen2.5-14B, ThinkLess reaches 73.07% *vs.* 74.82%. These small differences—within 1–2 points—demonstrate that ThinkLess retains most of the reasoning quality, validating our core hypothesis: reasoning can be effectively compressed into latent representations without explicit CoT generation.

Enhanced Accuracy under Comparable Token Budgets. Figure 5 presents the Top@k accuracy of ThinkLess compared against the Top@1 accuracy of the full CoT Distill baseline, under an equal token budget. The results show that ThinkLess significantly outperforms the distilled counterpart across various datasets and model backbones. Notably, beyond accuracy improvements, ThinkLess also achieves lower inference latency. This is because the k candidate answers in ThinkLess can be generated in parallel, whereas the distilled baseline must generate a long CoT sequence token by token in an inherently sequential manner.

Massive Reductions in Token Usage and Inference Time. ThinkLess achieves substantial efficiency gains across all settings. On average: *1. Token usage is reduced by 60–70%*, dropping from 2189 tokens (Qwen2.5-7B, Distill) to just 904 with ThinkLess. *2. Inference time is reduced by 50% or more*, *e.g.*, from 21.89s to 9.64s with Qwen2.5-14B.

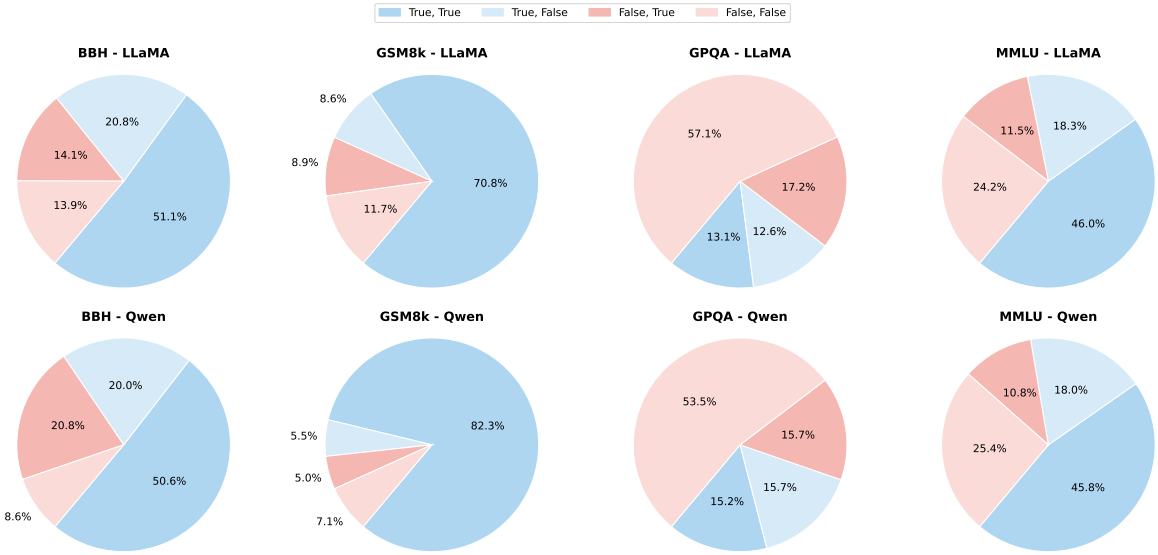


Figure 6: Answer overlap between Distill and ThinkLess w/o Instruct. Each pie shows the proportion of "a, b" cases, where "a" is Distill's results (True or False) and "b" is ThinkLess w/o Instruct's (True or False).

These savings stem from truncating long reasoning sequences early via </think>, which eliminates most of the token generation and KV cache accumulation that typically burdens autoregressive inference. Crucially, these gains come without any fine-tuning, distillation, or prompt engineering, making ThinkLess easy to deploy.

Robustness Across Models and Tasks. Although ThinkLess occasionally underperforms on specific datasets (*e.g.*, slightly lower on BBH with Qwen2.5-14B), its average accuracy is remarkably stable across all backbones. This consistency indicates that our method generalizes well across diverse reasoning tasks and model families.

The Role of Output Regulation. Comparing ThinkLess to its ablated version **ThinkLess w/o Instruct** highlights the impact of our lightweight instruction-based output regulation. Across all settings, ThinkLess consistently outperforms the w/o Instruct variant in Top@1 accuracy, often by a significant margin. For instance: On MMLU with Qwen2.5-14B: ThinkLess achieves 76.44% *vs.* 75.22%. On BBH with LLaMA3-8B: 71.92% *vs.* 65.45%, a gap of over 6 points.

Figure 6 illustrates the answer agreement between Distill and ThinkLess w/o Instruct across datasets and backbones. Across most of the datasets, over 70% of predictions remain consistent (*i.e.*, <True, True> or <False, False>), demonstrating ThinkLess can well preserve ability of the Distill model despite its early termination.

This confirms that without output regulation, the model—though internally sound—frequently fails to produce well-structured answers (*e.g.*, missing final choice or wrong format). The addition of a short task-specific instruction guides the model to produce answers in a predictable and scorable format, which is critical for maintaining accuracy in the absence of full reasoning traces.

Summary. ThinkLess achieves comparable Top@1 accuracy to full CoT reasoning while halving inference time and reducing token usage by up to 70%, all in a training-free and model-agnostic manner. These results demonstrate that ThinkLess offers a highly practical trade-off between reasoning fidelity and computational efficiency.

5 Conclusion

This paper presents **ThinkLess**, an inference-efficient framework that reduces the overhead of CoT reasoning without any model modification or additional training. By analyzing attention patterns, we find that final answers rely little on early reasoning steps—enabling safe early termination via a reasoning terminator token. To preserve answer completeness and format, a lightweight output regulation step is introduced, leveraging the model's instruction-following ability. Experimental Results show that ThinkLess achieves comparable accuracy to full CoT decoding while significantly lowering token usage and latency, making it a practical and generalizable solution for real-world deployment.

6 Limitations

While ThinkLess demonstrates strong efficiency and accuracy trade-offs, several limitations remain:

Reliance on Instruction Quality. The success of ThinkLess depends on the effectiveness of lightweight output regulation instructions. Poorly phrased or overly generic instructions may fail to guide the model toward well-structured outputs, especially for complex or ambiguous tasks. Designing effective instructions for new tasks may require manual tuning or domain-specific insights.

Lack of Dynamic Truncation Strategy. ThinkLess currently inserts the </think> token at fixed positions, without dynamically adapting to the complexity of individual questions. For harder tasks requiring deeper reasoning, premature truncation may omit essential content. Developing an adaptive termination policy that tailors reasoning length to question difficulty remains an open direction.

Assumption of Internal Reasoning Compression. ThinkLess assumes that LLMs internally compress reasoning into the </think> token, which may hold for certain instruction-tuned models but not all. Models without strong instruction-following capabilities or those trained with different prompting formats may not benefit from early termination in the same way, limiting the generalizability of our method.

Limited Scalability Validation. Due to computational resource constraints, we only evaluate ThinkLess on mid-sized models (7B–14B) and a limited set of reasoning benchmarks. Its performance on larger foundation models or broader tasks remains to be validated.

These limitations also highlight important directions for future work. In particular, extending ThinkLess to larger-scale models, more diverse task types, and dynamic truncation policies remains a key focus of our ongoing efforts.

7 Ethical Considerations

We use publicly available datasets and model checkpoints under licenses that permit research use. Details about the license terms and usage restrictions are provided in Section 4.1. We ensured that all artifacts were used in accordance with their intended purpose as stated by the original providers.

References

Simon A Aytes, Jinheon Baek, and Sung Ju Hwang. 2025. Sketch-of-thought: Efficient llm reasoning with adaptive cognitive-inspired sketching. *arXiv preprint arXiv:2503.05179*.

Zhenni Bi, Kai Han, Chuanjian Liu, Yehui Tang, and Yunhe Wang. 2024. Forest-of-thought: Scaling test-time compute for enhancing llm reasoning. *arXiv preprint arXiv:2412.09078*.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and 1 others. 2024. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*.

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

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. 2023. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36:70757–70798.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. 2024. Token-budget-aware llm reasoning. *arXiv preprint arXiv:2412.18547*.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Zhihang Lin, Mingbao Lin, Luxi Lin, and Rongrong Ji. 2025. Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5334–5342.

Zichen Liu, Changyu Chen, Wenjun Li, Tianyu Pang, Chao Du, and Min Lin. 2025a. There may not be aha moment in r1-zero-like training—a pilot study.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025b. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *The 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL 2023)*.

Shuofei Qiao, Yixin Ou, Ningyu Zhang, Xiang Chen, Yunzhi Yao, Shumin Deng, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022. Reasoning with language model prompting: A survey. *arXiv preprint arXiv:2212.09597*.

Ziran Qin, Yuchen Cao, Mingbao Lin, Wen Hu, Shixuan Fan, Ke Cheng, Weiyao Lin, and Jianguo Li. 2025. Cake: Cascading and adaptive kv cache eviction with layer preferences. *arXiv preprint arXiv:2503.12491*.

Xiaoye Qu, Yafu Li, Zhaochen Su, Weigao Sun, Jianhao Yan, Dongrui Liu, Ganqu Cui, Daizong Liu, Shuxian Liang, Junxian He, and 1 others. 2025a. A survey of efficient reasoning for large reasoning models: Language, multimodality, and beyond. *arXiv preprint arXiv:2503.21614*.

Yuxiao Qu, Matthew YR Yang, Amrith Setlur, Lewis Tunstall, Edward Emanuel Beeching, Ruslan Salakhutdinov, and Aviral Kumar. 2025b. Optimizing test-time compute via meta reinforcement fine-tuning. *arXiv preprint arXiv:2503.07572*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, and Shiguo Lian. 2025. Dast: Difficulty-adaptive slow-thinking for large reasoning models. *arXiv preprint arXiv:2503.04472*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Hanjie Chen, Xia Hu, and 1 others. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, and 1 others. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

Qwen Team. 2025. Qwq-32b: Embracing the power of reinforcement learning.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Siwei Wu, Zhongyuan Peng, Xinrun Du, Tuney Zheng, Minghao Liu, Jialong Wu, Jiachen Ma, Yizhi Li, Jian Yang, Wangchunshu Zhou, and 1 others. 2024. A comparative study on reasoning patterns of openai's o1 model. *arXiv preprint arXiv:2410.13639*.

Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. 2025. Tokenskip: Controllable chain-of-thought compression in llms. *arXiv preprint arXiv:2502.12067*.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. 2025. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Hai Ye, Mingbao Lin, Hwee Tou Ng, and Shuicheng Yan. 2024. Multi-agent sampling: Scaling inference compute for data synthesis with tree search-based agentic collaboration. *arXiv preprint arXiv:2412.17061*.

Jintian Zhang, Yuqi Zhu, Mengshu Sun, Yujie Luo, Shuofei Qiao, Lun Du, Da Zheng, Huajun Chen, and Ningyu Zhang. 2025a. Lightthinker: Thinking step-by-step compression. *arXiv preprint arXiv:2502.15589*.

Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yueting Zhuang, and Weiming Lu. 2024. Self-contrast: Better reflection through inconsistent solving perspectives. *arXiv preprint arXiv:2401.02009*.

Xiaoying Zhang, Da Peng, Yipeng Zhang, Zonghao Guo, Chengyue Wu, Chi Chen, Wei Ke, Helen Meng, and Maosong Sun. 2025b. Will pre-training ever end? a first step toward next-generation foundation mllms via self-improving systematic cognition. *arXiv preprint arXiv:2503.12303*.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

A Analysis of Output Formatting Issues from Early Termination

While ThinkLess is designed to terminate reasoning early and rely on internalized representations for answer generation, this can occasionally result in malformed outputs—particularly when the model is not explicitly instructed on how to format its final answer. Tables 2 and 3 illustrate common failure cases across different datasets, caused not by flawed reasoning, but by formatting deviations that hinder correct evaluation.

GPQA: Outputting the Answer Directly Instead of the Option. In multiple GPQA examples, the model correctly computes the numerical answer (*e.g.*, "18", "$\frac{1+nv}{n+v}$", or "3536"), but fails to select the corresponding multiple-choice option letter (*e.g.*, "A", "B", *etc.*). This is problematic because the task requires choosing from a list, and direct numeric answers—though logically correct—are treated as incorrect under automatic evaluation scripts. This issue is a direct consequence of skipping the reasoning trace, which would otherwise reinforce the expected answer format (*e.g.*, "The answer is A").

BBH: Verbose or Misaligned Natural Language Outputs. In BBH, early termination sometimes causes the model to output full explanations (*e.g.*, "No, Tamika does not tell the truth") instead of a concise boolean token like "False" or a lettered choice. In one example, the model responds with an overly verbose phrase: "The statement 'Return your final response within boxed {}' is True"—a hallucination likely caused by partial instruction remnants seen during pretraining. These cases reveal how early truncation may weaken task compliance, especially for boolean or classification-based tasks that expect minimal final output.

MMLU: Misformatting Algebraic Expressions. For math-heavy tasks like MMLU, the model sometimes produces an exactly correct symbolic expression (*e.g.*, "$(x+1)(x-2)(x+4)$") that does not match any of the provided answer options verbatim. Though mathematically equivalent to one of the choices, the mismatch in formatting or token order causes the model to be penalized. This highlights the fragility of matching-based evaluation when outputs are not explicitly aligned with options.

Key Insight: Output Regulation is Essential. These examples demonstrate that output formatting errors—not reasoning failures—are the dominant cause of performance drop in ThinkLess without instruction-based regulation. The missing or misaligned final answers occur because the model lacks an explicit signal about how to conclude the response after <$/$think> is triggered.

By contrast, ThinkLess with output regulation prepends a short, task-specific instruction (*e.g.*, "Select the best option (A, B, C, D):") that helps the model map internal reasoning to a valid and scorable final output—without increasing token length significantly. This regulation mechanism is crucial for ensuring compatibility with automatic scorers and maintaining downstream performance.

B Instruction-based Output Regulation

Tables 4 and 5 provides instructions details that regulate the output formatting across different datasets and their subtasks.

Dataset	Input	Output
GPQA	If an equimolar mixture X of two liquids, which decolorizes bromine water, is treated with platinum when heated, then an equimolar mixture Y of two other liquids is formed as a result of disproportionation, which does not decolorize bromine water. Hydrogenation of both mixture X and mixture Y in the presence of platinum under rather severe conditions gives only one substance, a certain hydrocarbon Z (mass fraction of hydrogen is 14.28%), which is a constituent of mixture Y and widely used as a solvent. Substance Z does not react further with hydrogen. There are no conjugated multiple bonds in the molecules of the compounds of mixture X. Indicate the total number of hydrogen atoms in two liquids of mixture X. A. 18 B. 22 C. 16 D. 12	18
	A light beam is propagating through a glass with index of refraction n. The glass is moving at constant velocity v in the same direction as the beam and toward the observer in laboratory. What is the speed of light in glass relative to the observer in laboratory? Take the speed of light in vacuum c=1. A. 1 B. $\frac{1-nv}{n+v}$ C. $\frac{1+nv}{n+v}$ D. $\frac{1+nv}{n-v}$	$\frac{1+nv}{n+v}$
	A synchrocyclotron is a special type of cyclotron. A proton is injected at $t = 0$ with negligible kinetic energy, i.e., $T \simeq 0$, and undergoes two accelerations in the accelerator to reach the kinetic energy $$T_1 = 950\,\text{MeV}.$$ The alternating voltage applied in the accelerator is given by $$U(t) = U_0 \cos\left[2\pi\nu(\alpha)\, t + \Phi_0\right].$$ How many revolutions does the proton make in the synchrocyclotron? A. 5300 B. 2500 C. 3536 D. 1864	3536

Table 2: Input and output cases of early termination from GPQA.

Dataset	Input	Output
BBH	Fidel lies. Millicent says Fidel tells the truth. Crista says Millicent lies. Leda says Crista lies. Tamika says Leda tells the truth. Does Tamika tell the truth?	No, Tamika does not tell the truth.
	False or not False or True or True is	The statement "Return your final response within `\boxed {}`." is **True**.
	The following paragraph describes a set of seven objects arranged in a fixed order. The statements are logically consistent within the paragraph. On a branch, there are seven birds: a robin, a hummingbird, a raven, a blue jay, a crow, a cardinal, and a quail. The hummingbird is to the left of the crow. The cardinal is to the right of the quail. The blue jay is the fourth from the left. The robin is the second from the left. The quail is to the right of the crow. The raven is to the left of the robin. Which bird is the leftmost? A. The crow is the leftmost B. The raven is the leftmost C. The blue jay is the leftmost D. The robin is the leftmost E. The hummingbird is the leftmost F. The cardinal is the leftmost G. The quail is the leftmost	The **raven** is the leftmost
MMLU	Find all zeros in the indicated finite field of the given polynomial with coefficients in that field: $x^5 + 3x^3 + x^2 + 2x \quad \text{in } \mathbb{Z}_5$ A. $0, 1$ B. $0, 4$ C. 0 D. 1	0, 4
	The polynomial $x^3 + 2x^2 + 2x + 1$ can be factored into linear factors in $\mathbb{Z}_7[x]$. Find this factorization. A. $(x-2)(x+2)(x-1)$ B. $(x+1)(x+4)(x-2)$ C. $(x+1)(x-4)(x-2)$ D. $(x-1)(x-4)(x-2)$	$\boxed{(x+1)(x-2)(x+4)}$

Table 3: Input and output cases of early termination from BBH and MMLU.

Dataset	Sub-task	Instruction
BBH	boolean expression	Evaluate the given Boolean expression step by step, carefully analyzing each operation and verifying the logic at every stage. Ensure the reasoning process is accurate and consistent. Return the final result as either "True" or "False".
	causal judgement	Assess whether the stated causal relationship between two events or phenomena is logically valid. Analyze the connection step by step, verify your reasoning at each stage, and base your judgment on evidence, logic, and plausibility. Conclude by providing your final answer as "Yes" or "No".
	formal fallacies	Analyze the given argument to determine whether it is deductively valid. Start by identifying and formalizing the premises and conclusion. Reflect on each step of your evaluation, ensuring the conclusion follows logically and necessarily from the premises without relying on external information or assumptions. Finally, respond with either "valid" or "invalid".
	web of lies	Based on the statements made by the characters, determine whether the specified character is telling the truth. Analyze the relationships and consistency between the statements step by step, reflect on your reasoning at each stage, and ensure your judgment is logically sound. The final answer should be "Yes" or "No"'
	navigate	Given the navigation instructions, determine whether you can reach the destination. You can learn to analyze, but the final answer should be "Yes" or "No".
	logical deduction seven objects	Solve the following logic puzzle to determine the correct order of seven objects based on the given clues. Analyze the clues step by step, reflect on your reasoning at each stage, and systematically eliminate incorrect possibilities. Finally, evaluate all the options (A-G) and select the one that represents the correct answer.
	ruin names	Analyze each option for its humor, creativity, and resemblance to the original name step by step. Reflect on the reasoning process to determine the best choice for each question. Output your answers as a sequence of four letters (A-D), one for each question.
	temporal sequences	Determine the correct order of events from the given choices. For each item, select the correct option (A-D) and output them in order.

Table 4: Instruction regulations on BBH Subtasks

Dataset	Instruction
GSM8K	Solve the math problem step by step. Give only the final numerical answer.
MMLU	Given the multiple-choice question above drawn from different academic disciplines, think step by step, self-check your reasoning, and output only the single final option (A, B, C, or D).
GPQA	You will be given a graduate-level multiple-choice science question. Think step-by-step (LaTeX allowed), self-check, then output one line with only the letter A, B, C, or D.

Table 5: Instruction regulations on GSM8K, MMLU and GPQA.