Efficient Inference for Reasoning Model

密言

May 13, 2025

Overview

1. **Challenge and Class of Reasoning Model**

2. **Explicit Compact CoT**
   - 2.1 CoT Compression
   - 2.2 Early Exit

3. **Implicit Latent CoT**

4. **Future Direction**

Challenge

1. significant token consumption
2. highly memory overhead
3. increased inference time

- model compression, efficient model design, and system-level optimization, can alleviate problems 2 and 3.
- focus on token inefficiency

2 Classes

1. the explicit compact CoT
    - CoT compression
    - Early Exit
2. the implicit latent CoT

Train-free Methods

Chain of Draft: Thinking Faster by Writing Less[1]

Insights:

- 标准提示方法要求模型直接输出答案，没有推理过程，缺乏透明性，在没有中间步骤的情况下进行多步骤推理容易产生幻觉
- 标准 CoT 消耗 token 多，latency 大
- CoD 将推理过程浓缩为最小的抽象表述，减少 token 数量且推理过程具有透明性

Sketch-of-Thought: Efficient LLM Reasoning with Adaptive Cognitive-Inspired Sketching[2]

Methods:

- 输入问题 → 路由模型选择推理范式 → Few shot prompt → Sketched reasoning → Final answer

[1] Silei Xu et al. *Chain of Draft: Thinking Faster by Writing Less*. 2025. arXiv: 2502.18600 [cs.CL]. URL: https://arxiv.org/abs/2502.18600.

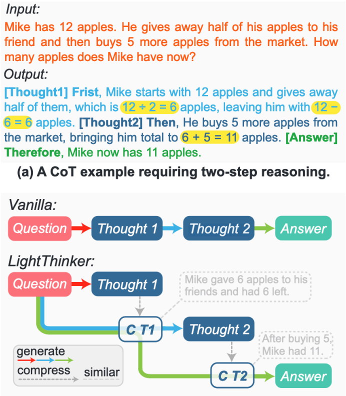
[2] Simon A. Aytes, Jinheon Baek, and Sung Ju Hwang. *Sketch-of-Thought: Efficient LLM Reasoning with Adaptive Cognitive-Inspired Sketching*. 2025. arXiv: 2503.05179 [cs.CL]. URL: https://arxiv.org/abs/2503.05179.

Finetune Methods

LightThinker: Thinking Step-by-Step Compression[3]

Insights:

- LLM 生成的 token 具有两个作用：保证语义连贯性和促进推理
- 人类解决复杂推理问题时只会写下关键步骤，将剩余思考过程保存在脑中



Input:

Mike has 12 apples. He gives away half of his apples to his friend and then buys 5 more apples from the market. How many apples does Mike have now?

Output:

[Thought1] Frist, Mike starts with 12 apples and gives away half of them, which is 12 ÷ 2 = 6 apples, leaving him with 12 − 6 = 6 apples. **[Thought2] Then**, He buys 5 more apples from the market, bringing him total to 6 + 5 = 11 apples. **[Answer] Therefore**, Mike now has 11 apples.

(a) A CoT example requiring two-step reasoning.

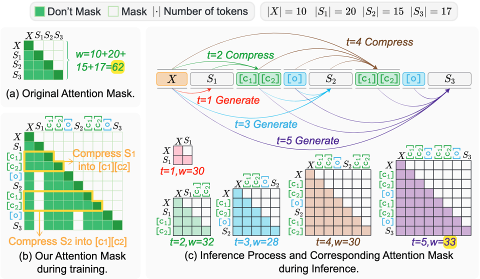
(b) Diagram of the reasoning process for Vanilla and LightThinker

[3] Jintian Zhang et al. *LightThinker: Thinking Step-by-Step Compression*. 2025. arXiv: 2502.15589 [cs.CL]. URL: https://arxiv.org/abs/2502.15589.

Finetune Methods

Methods:

- 动态压缩原始思维链：生成每个思考步骤后会压缩成几个紧凑的 gist tokens，丢弃原始思考过程。
- 训练模型压缩时机：
  - 重构数据教会模型何时压缩，使用 special token，压缩触发符 <w>，缓存 token <c>，压缩结束符 <o>
  - 将思维步骤压缩到 gist tokens 中
  - 设计专门的 attention mask


🟩 Don't Mask □ Mask |·| Number of tokens
$|X| = 10$ $|S_1| = 20$ $|S_2| = 15$ $|S_3| = 17$
w=10+20+15+17=62
Compress S1 into [c1][c2]
Compress S2 into [c1][c2]
t=2 Compress
t=4 Compress
t=1 Generate
t=3 Generate
t=5 Generate
t=1,w=30
t=2,w=32
t=3,w=28
t=4,w=30
t=5,w=33

(a) Original Attention Mask.

(b) Our Attention Mask during training.

(c) Inference Process and Corresponding Attention Mask during Inference.

Figure 2: Overview of LightThinker, illustrated with an example requiring three-step reasoning. Fig. (a) shows the attention mask of Vanilla during both training and inference. Fig. (b) depicts the attention mask of LightThinker during the training. Fig. (c) presents the complete inference process of LightThinker along with the attention mask corresponding to each step. Here, 'w' denotes the size of the matrix.

Finetune Methods

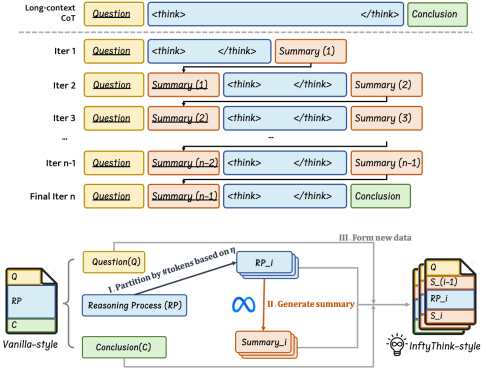
INFTYTHINK: Breaking the Length Limits of Long-Context Reasoning in Large Language Models[4]

Insights:

- 带中间推理过程摘要的推理迭代，打破推理长上下文限制和提高推理效率。

Methods:

- 迭代式推理
  - 生成推理片段后会生成一段推理摘要
  - 推理摘要作为下一次推理的输入历史
- 训练数据集构建
  - 推理分割 - 摘要生成 - 组合成 Inftythink 训练范式



[4] Yuchen Yan et al. *InftyThink: Breaking the Length Limits of Long-Context Reasoning in Large Language Models*. 2025. arXiv: 2503.06692 [cs.CL]. URL: https://arxiv.org/abs/2503.06692.

Computational complexity: LighterThinker vs InftyThink

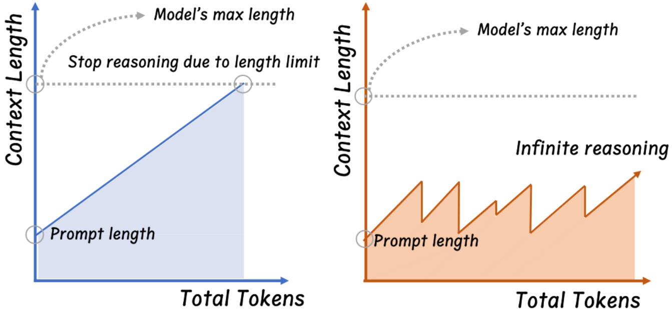


Figure: vanilla and InftyThink

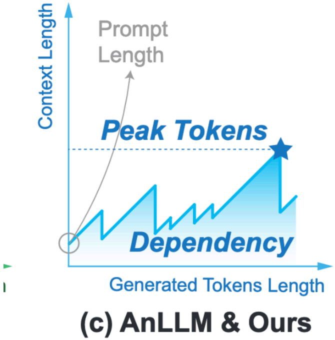
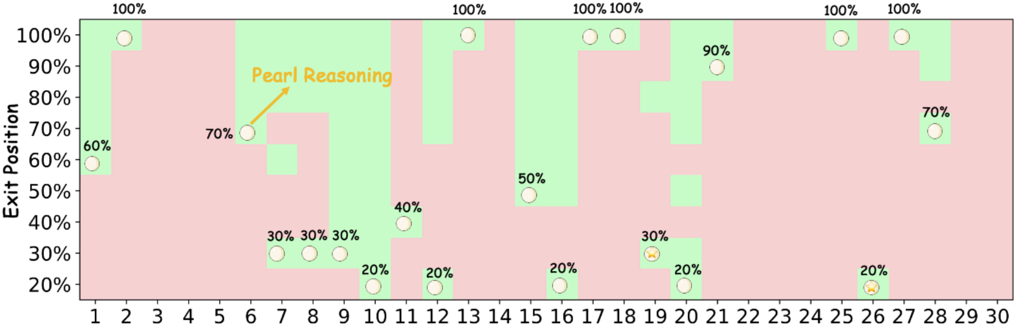


Figure: LighterThinker

DYNAMIC EARLY EXIT IN REASONING MODELS[5]

Insights:

- Overthinking 不仅降低了效率，还引入冗余的推理步骤（比如过度详细的思考、过多尝试路径）而导致准确性下降。
- 75% 的推理样本存在推理信息刚好足够正确解答问题的 critical point（称为Pearl Reasoning），如何找到这样的 Pearl Reasoning 对于高效性和准确性非常重要。

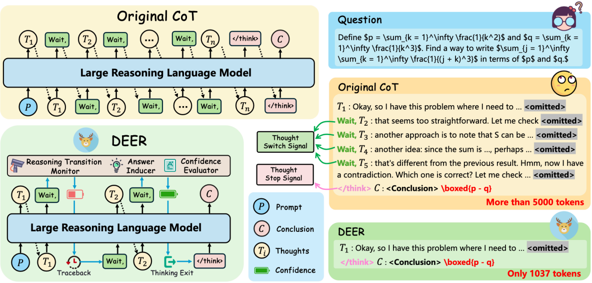


[5] Chenxu Yang et al. *Dynamic Early Exit in Reasoning Models*. 2025. arXiv: 2504.15895 [cs.CL]. URL: https://arxiv.org/abs/2504.15895.

DYNAMIC EARLY EXIT IN REASONING MODELS

Methods:

- 推理过程中遇到” wait”，产生中间答案，计算 confidence，大于阈值时结束推理，进行 Final Answer
- 分支并行化：wait 之后利用 kv-cache 新建分支产生 Final Answer，评估结果 confidence



Question

Define $p = \sum_{k = 1}^\infty \frac{1}{k^2}$ and $q = \sum_{k = 1}^\infty \frac{1}{k^3}$. Find a way to write $\sum_{j = 1}^\infty \sum_{k = 1}^\infty \frac{1}{(j + k)^3}$ in terms of p and $q.$

Original CoT

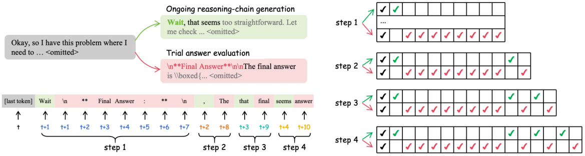
T_1 : Okay, so I have this problem where I need to ... <omitted>
Wait, T_2 : that seems too straightforward. Let me check <omitted>
Wait, T_3 : another approach is to note that S can be ... <omitted>
Wait, T_4 : another idea: since the sum is ..., perhaps ... <omitted>
Wait, T_5 : that's different from the previous result. Hmm, now I have a contradiction. Which one is correct? Let me check ... <omitted>
</think> C : <Conclusion> \boxed{p - q}

More than 5000 tokens

DEER

T_1 : Okay, so I have this problem where I need to ... <omitted>
</think> C : <Conclusion> \boxed{p - q}

Only 1037 tokens



Okay, so I have this problem where I need to … <omitted>

Ongoing reasoning-chain generation

Wait, that seems too straightforward. Let me check ... <omitted>

Trial answer evaluation

\n**Final Answer**\n\nThe final answer is \\boxed{... <omitted>

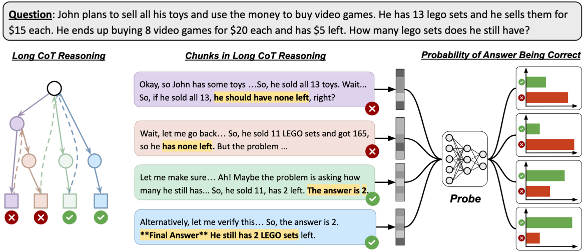
Reasoning Models Know When They’re Right: Probing Hidden States for Self-Verification[6]

Insights:

- 推理过程的隐藏状态中存在正确性信息

Methods:

- 推理过程分 chunk，每个 chunk 的 last token 的 last layer 的 hidden states 和该 chunk 的正确性
- 使用数据训练 probe
- 根据 probe 阈值 Early Exit 推理过程



Question: John plans to sell all his toys and use the money to buy video games. He has 13 lego sets and he sells them for $15 each. He ends up buying 8 video games for $20 each and has $5 left. How many lego sets does he still have?

Okay, so John has some toys …So, he sold all 13 toys. Wait... So, if he sold all 13, he should have none left, right?

Wait, let me go back… So, he sold 11 LEGO sets and got 165, so he has none left. But the problem ...

Let me make sure… Ah! Maybe the problem is asking how many he still has... So, he sold 11, has 2 left. The answer is 2.

Alternatively, let me verify this… So, the answer is 2. **Final Answer** He still has 2 LEGO sets left.

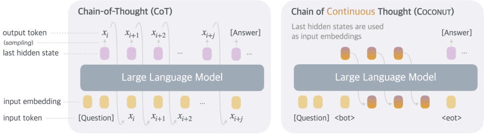
Long CoT Reasoning
Chunks in Long CoT Reasoning
Probability of Answer Being Correct
Probe

[6] Anqi Zhang et al. *Reasoning Models Know When They’re Right: Probing Hidden States for Self-Verification*. 2025. arXiv: 2504.05419 [cs.AI]. URL: https://arxiv.org/abs/2504.05419.

COCONUT (Chain of Continuous Thought)[7]

Insights:

- 语言空间并非推理的最优选择
  - 文本连贯性的冗余
  - 计算预算分配不均
- 潜在空间推理的潜力
  - 理想情况下，LLM 应该能够在没有语言约束的情况下自由推理，仅在必要时将其发现转化为语言。
- 连续思维的新兴高级推理模式
  - COCONUT 范式下的“连续思维”可以编码多个潜在的下一步推理选择，使得模型能够执行类似于广度优先搜索（BFS）的过程来解决问题。

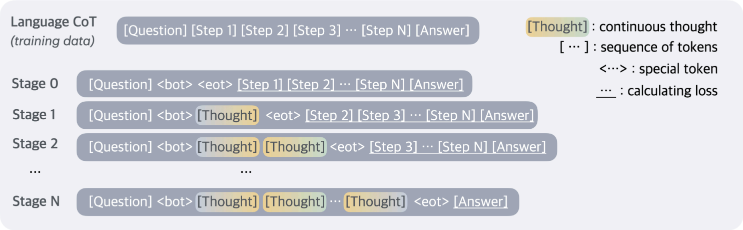


[7] Shibo Hao et al. *Training Large Language Models to Reason in a Continuous Latent Space*. 2024. arXiv: 2412.06769 [cs.CL]. URL: https://arxiv.org/abs/2412.06769.

COCONUT (Chain of Continuous Thought)

Methods:

- COCONUT 将 LLM 的 last hidden state 作为推理状态的表示，称之为“连续思维”
- 训练过程
  - 逐步替换：训练从标准的语言 CoT 实例开始。在后续的第 k 个阶段，原始 CoT 中的前 k 个语言推理步骤会被替换为 k 个连续思维。
  - 损失计算：使用标准的负对数似然损失进行优化，但在问题和潜在思维部分不计算损失。目标不是让连续思维压缩被移除的语言思维，而是促进对未来推理的预测。



Language CoT *(training data)* [Question] [Step 1] [Step 2] [Step 3] ··· [Step N] [Answer]

Stage 0 [Question] <bot> <eot> [Step 1] [Step 2] ··· [Step N] [Answer]

Stage 1 [Question] <bot> [Thought] <eot> [Step 2] [Step 3] ··· [Step N] [Answer]

Stage 2 [Question] <bot> [Thought] [Thought] <eot> [Step 3] ··· [Step N] [Answer]

··· ···

Stage N [Question] <bot> [Thought] [Thought] ··· [Thought] <eot> [Answer]

[Thought] : continuous thought

[···] : sequence of tokens

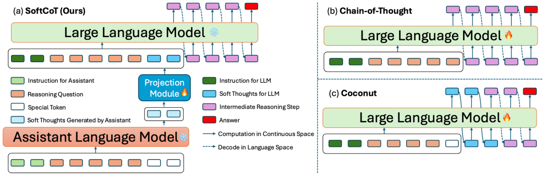
<···> : special token

··· : calculating loss

SoftCoT: Soft Chain-of-Thought for Efficient Reasoning with LLMs[8]

Insights：连续空间推理方法的限制

- 需要全参数微调
- 获得连续推理能力后，损失通用能力
- 不适用于最新的调优 LLM



Method:

- 使用一个轻量级的、固定的 assistant model 来推测性地生成实例特定的“软思维 token”（soft thought tokens）
- 通过一个可训练的投影模块（projection module）将这些软思维映射到主 LLM 的表示空间中，从而引导主 LLM 进行推理

[8] Yige Xu et al. *SoftCoT: Soft Chain-of-Thought for Efficient Reasoning with LLMs*. 2025. arXiv: 2502.12134 [cs.CL]. URL: https://arxiv.org/abs/2502.12134.

Future Direction

- Efficient Multimodal Reasoning
- Efficient Test-time Scaling
- Efficient and Trustworthy Reasoning
    - 安全性方面：推理过程产生一些不安全的内容
    - 可信性方面：
        - 推理链事实性和忠实性不足，推理路径延长会产生累积效应
        - 推理链不能完全反映模型的思维过程

References I

[1] Silei Xu et al. *Chain of Draft: Thinking Faster by Writing Less*. 2025. arXiv: 2502.18600 [cs.CL]. URL: https://arxiv.org/abs/2502.18600.

[2] Simon A. Aytes, Jinheon Baek, and Sung Ju Hwang. *Sketch-of-Thought: Efficient LLM Reasoning with Adaptive Cognitive-Inspired Sketching*. 2025. arXiv: 2503.05179 [cs.CL]. URL: https://arxiv.org/abs/2503.05179.

[3] Jintian Zhang et al. *LightThinker: Thinking Step-by-Step Compression*. 2025. arXiv: 2502.15589 [cs.CL]. URL: https://arxiv.org/abs/2502.15589.

[4] Yuchen Yan et al. *InftyThink: Breaking the Length Limits of Long-Context Reasoning in Large Language Models*. 2025. arXiv: 2503.06692 [cs.CL]. URL: https://arxiv.org/abs/2503.06692.

[5] Chenxu Yang et al. *Dynamic Early Exit in Reasoning Models*. 2025. arXiv: 2504.15895 [cs.CL]. URL: https://arxiv.org/abs/2504.15895.

References II

[6] Anqi Zhang et al. *Reasoning Models Know When They're Right: Probing Hidden States for Self-Verification*. 2025. arXiv: 2504.05419 [cs.AI]. URL: https://arxiv.org/abs/2504.05419.

[7] Shibo Hao et al. *Training Large Language Models to Reason in a Continuous Latent Space*. 2024. arXiv: 2412.06769 [cs.CL]. URL: https://arxiv.org/abs/2412.06769.

[8] Yige Xu et al. *SoftCoT: Soft Chain-of-Thought for Efficient Reasoning with LLMs*. 2025. arXiv: 2502.12134 [cs.CL]. URL: https://arxiv.org/abs/2502.12134.

The End