Cognitive Compression: Recursive Note-Taking for Efficient Chain-of-Thought in LLMs

Stanford CS224N Custom Project

Anuj Jamwal
Department of Computer Science
Stanford University
anujjam@stanford.edu

Abstract

Chain-of-Thought (CoT) prompting has unlocked significant reasoning capabilities in Large Language Models (LLMs) but incurs a linear growth in context memory and compute costs. Existing efficiency methods, such as PENCIL, address this by aggressively deleting intermediate reasoning steps, which prevents the model from backtracking or referencing prior partial results. We propose "Cognitive Compression," a method that fine-tunes Small Language Models (SLMs) to recursively summarize completed reasoning steps into concise natural language "notes" rather than deleting them. This approach aims to reduce token consumption and KV-cache usage while preserving the logical state required for complex problem-solving. We will evaluate this method on the GSM8K and AIME24 benchmarks, measuring both accuracy and token efficiency.

1 Key Information to include

- External collaborators (if you have any): None
- Mentor (custom project only): [Insert Mentor Name if known, or leave blank]
- Sharing project: No

2 Research paper summary (max 2 pages)

Title	PENCIL: Long Thoughts with Short Memory
Venue	International Conference on Machine Learning (ICML)
Year	2025
URL	https://arxiv.org/pdf/2503.14337

Table 1: Sample table for bibliographical information [1].

Background. Large Reasoning models have demonstrated emergent capabilities in complex domains like mathematics, programming, and logical reasoning by scaling up the length of Chain-of-Thought (CoT). This scaling, however, comes at a prohibitive cost: reasoning spans thousands of tokens, leading to memory overhead and substantial latency. Crucially, these lengthy traces often contain logical redundancy, such as over-explaining simple problems or superficially exploring multiple paths. There are at times branches that are explored but subsequently rejected, yet continue to linger in the CoT, processed repeatedly by the LLM. Prior work on context compression includes token-level methods like TokenSkip (Xia et al., 2025) [2], step-level pruning like SPIRIT (Cui et al., 2025) [3], and anchor guider pruning like ASAP (Zeng et al., 2026) [4].

Summary of contributions. The paper introduces PENCIL, which incorporates a novel reduction mechanism into the autoregressive generation process that recursively cleans up intermediate thoughts based on patterns learned during training. While the methods mentioned in the background focus on training the model to produce compressed CoT initially, PENCIL acknowledges that verbose reasoning is sometimes necessary to solve a sub-problem. By iteratively generating and erasing thoughts as part of the autoregressive loop at inference time, PENCIL can "think deeper" to solve harder problems using shorter context and less compute.

More concretely, the paper focuses on a simple yet universal reduction rule motivated by the function call stack in modern computers:

$$\mathbf{C}\ \texttt{[CALL]}\ \mathbf{T}\ \texttt{[SEP]}\ \mathbf{A}\ \texttt{[RETURN]} \quad \Rightarrow \quad \mathbf{C}\ \mathbf{A} \tag{1}$$

where `[CALL]`, `[SEP]`, and `[RETURN]` are special tokens that separate the context (**C**), thoughts (**T**), and answer (**A**) in the sequence. Once a computation completes (marked by `[RETURN]`), all intermediate reasoning steps (those between `[CALL]` and `[SEP]`) are removed, merging the answer back into the context. Importantly, this process can be applied recursively, allowing for hierarchical reasoning structures similar to nested function calls.

Using PENCIL, the authors demonstrated that a small 25M parameter transformer with a 2048 context length could solve Einstein's puzzle—a task that challenges much larger models like GPT-4.

Limitations and discussion. First, the model is trained and evaluated on SAT, QBF, and Einstein's puzzle, all of which require long chain following but are distinct from the complex numerical reasoning found in mathematics or coding. Whether the methodology generalizes to these domains is left unexplored.

Second, the compression mechanism is effectively a deletion of the context fragment. While this works when the model is correct in its chosen branch of reasoning, there is little to no context left for future reference. If the chosen line of reasoning proves wrong, the model cannot backtrack, as that part of the "memory" has been erased.

Why this paper? While techniques like KV-caching and sparse attention address architectural efficiency, PENCIL is unique in addressing *logical* efficiency at the token generation level. Unlike standard context compression (e.g., LLMLingua [5]) which filters tokens post-hoc, or summary-tokens (e.g., recurrent models), PENCIL introduces a mechanism to modify the history during generation. However, we argue that PENCIL's "delete-only" approach is too aggressive for complex math, motivating our "compress/note-taking" approach.

Wider research context. This work addresses the challenges posed by long Chain-of-Thought and long context windows. Long context suffers from two broad categories of redundancies:

1. **Structural Redundancy:** Digressive branches that are no longer relevant.
2. **Logical Redundancy:** Explaining trivial actions or repeating established facts.

The area of context compression is active. Approaches like Selective Context (Li et al., 2023) [6], LLMLingua (Jiang et al., 2023) [5], and CodeZip (Shi et al., 2025) [7] employ information-theoretic metrics or external models to filter tokens. Other efforts focus on efficient reasoning, training models to produce naturally efficient chains, such as TokenSkip (Xia et al., 2025) [2] or ASAP (Zeng et al., 2026) [4].

3 Project description (1-2 pages)

Goal. Current Chain-of-Thought (CoT) methods treat reasoning as an append-only log, leading to linear memory growth. PENCIL addresses this by deleting past thoughts, but this prevents the model from backtracking or referencing prior intermediate results. Our goal is to train a model to perform "Iterative Note-Taking": converting verbose reasoning blocks into concise state representations (notes) dynamically during generation. This shrinks the context window while preserving the logical "state" of the solution.

We also identify the following stretch goals:

1. Analyze the emergence of stronger reasoning capabilities compared to the baseline model.
2. Empower the model with the ability to "course correct" over compressed CoT. We will explore training the model to retrieve specific segments of the previous chain of thought, allowing it to review past actions without permanently exploding the context.
3. Expand training and evaluate performance on the AIMO-2 Kaggle dataset [8].

Task. We will fine-tune a Small Language Model (SLM), such as Qwen-2.5-Math-1.5B or 7B. The model will be trained to:

1. Recognize when a reasoning step is complete.
2. Generate a compressed "note" of that step.
3. Discard the raw tokens of the step from its context, retaining only the note in the context.
4. Continue reasoning based on the note.

We will measure the performance of the model on its ability to successfully solve the problems with a reduction is token count.

Data. We will use the OpenMathReasoning [9] dataset from nvidia available on Hugging Face. OpenMathReasoning is a large-scale math reasoning dataset containing step-by-step solutions for about 306K unique mathematical problems.

To prepare the dataset, we will select a subset of problems with long CoTs and use a Large LLM (e.g., GPT-4o or DeepSeek-V3) to annotate the data. We will prompt the teacher model to identify logical break-points in the chain of thought and generate a concise 'memory summary' of the preceding steps. We aim to process approximately 1,000 to 2,000 examples to ensure sufficient diversity for the fine-tuning process.

Methods. We will employ a Teacher-Student distillation approach:

- **Data Synthesis:** As described above, we will generate a parallel dataset where verbose reasoning steps are replaced by summary tokens using a teacher model.
- **Training:** We will fine-tune the student model on this augmented dataset to learn the distribution $P(Summary|Context, Reasoning)$ and $P(NextStep|Context, Summary)$. This finetuning is similar to other CoT training and only differs in the training data.
- **Inference:** We will implement a custom generation loop that will detect the completion of a summary phase and physically remove the preceding verbose tokens from memory, retaining only the summary.

Baselines. We will establish a baseline by executing the original model on the selected problems. The baseline will capture:

1. Token Count (Total tokens processed)
2. Outcome Accuracy (Correct/Wrong)
3. Peak Memory Usage (to demonstrate efficiency gains)

We will also capture these metrics for the evaluation benchmarks (proposed below) to draw a baseline.

Evaluation. We will evaluate on **GSM8K** (to ensure that basic reasoning is retained) and **AIME24** (to test complex long-context handling).

- **Primary Metric:** Accuracy (%) compared to baseline.
- **Efficiency Metric:** Average Tokens Processed and KV Cache reduction.

Ethical Implications. The primary ethical risk in compressing reasoning chains is the potential for *loss of nuance* and *hallucinated summaries*. If the model learns to compress context aggressively, it may summarize away critical safety constraints or edge cases present in the original prompt, leading to unsafe outputs that appear logically sound but ignore initial instructions. Furthermore, "note-taking" models obfuscate the reasoning process; unlike full CoT, where a user can verify every step, a compressed state is a "black box" of the model's internal memory. This lack of interpretability makes it harder for humans to audit why a model made a specific decision.

In order to mitigate this, we propose the following options

1. The client can retain the CoT and log operations which can be presented to the human on demand
2. Monitoring the semantic similarity between the full CoT and the compressed notes to ensure fidelity is maintained.

References

[1] Chenxiao Yang, Nati Srebro, David McAllester, and Zhiyuan Li. Pencil: Long thoughts with short memory. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

[2] Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. TokenSkip: Controllable chain-of-thought compression in LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 3351–3363, Suzhou, China, November 2025. Association for Computational Linguistics.

[3] Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, Yue Xing, Jiliang Tang, and Qi He. Stepwise perplexity-guided refinement for efficient chain-of-thought reasoning in large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics: ACL 2025*, pages 18581–18597, Vienna, Austria, July 2025. Association for Computational Linguistics.

[4] Wenhao Zeng, Yaoning Wang, Chao Hu, Yuling Shi, Chengcheng Wan, Hongyu Zhang, and Xiaodong Gu. Pruning the unsurprising: Efficient llm reasoning via first-token surprisal. *arXiv preprint arXiv:2508.05988*, 2025.

[5] Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LLMLingua: Compressing prompts for accelerated inference of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13358–13376, Singapore, November 2023. Association for Computational Linguistics.

[6] Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. Compressing context to enhance inference efficiency of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6342–6353, Singapore, November 2023. Association for Computational Linguistics.

[7] Yuling Shi, Yichun Qian, Hongyu Zhang, Beijun Shen, and Xiaodong Gu. Longcodezip: Compress long context for code language models. *arXiv preprint arXiv:2510.00446*, 2025.

[8] Simon Frieder, Sam Bealing, Arsenii Nikolaiev, Geoff C. Smith, Kevin Buzzard, Timothy Gowers, Peter J. Liu, Po-Shen Loh, Lester Mackey, Leonardo de Moura, Dan Roberts, D. Sculley,

Terence Tao, David Balduzzi, Simon Coyle, Alex Gerko, Ryan Holbrook, Addison Howard, and XTX Markets. Ai mathematical olympiad - progress prize 2. https://kaggle.com/competitions/ai-mathematical-olympiad-progress-prize-2, 2024. Kaggle.

[9] Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset, 2025.