Simple, Scalable Reasoning via Iterated Summarization

Anonymous Authors[1]

1. Introduction

Recent advances in optimizing language models for reasoning via reinforcement learning have yielded dramatic improvements on mathematical tasks (OpenAI, 2024; DeepSeek-AI et al., 2025). Much of this improvement comes from "thinking" longer, which has motivated test-time techniques to extend how long they think (Muennighoff et al., 2025). However, these techniques increase the length of the reasoning trace, leading to substantial computational and memory overhead. While various approaches reduce this cost through specialized training (Yan et al., 2025; Aggarwal & Welleck, 2025; Xia et al., 2025; Cheng & Durme, 2024), we ask: can simple test-time interventions manage expanding context windows without additional training?

We investigate *iterated summarization* (IS) as a practical framework where models alternate between generating and summarizing reasoning traces (Figure 2). This training-free approach leverages pretrained LLMs' summarization capabilities to enable extended mathematical reasoning within bounded contexts. The key challenge lies in determining what constitutes an effective summary of mathematical reasoning. Should we preserve intermediate calculations, proof strategies, or only critical insights? We systematically explore summarization strategies ranging from simple heuristics to LLM-based approaches. Our most effective method uses an LLM to summarize traces with emphasis on backtracking moments—points where the model reconsiders its approach—which are particularly valuable in mathematical problem-solving.

Our contributions:

- We introduce **Iterated Summarization** (IS), enabling models to "think longer" on mathematical problems without significantly increasing the context.
- We explore multiple summarization techniques and identify behaviors that unlock greater reasoning capabilities.
- On AIME 2024 and 2025, our best IS technique achieves an 11% performance boost over baselines.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2. Iterated Summarization

IS alternates between reasoning and summarization (Figure 2). In iteration t, a reasoning model R produces trace r_t, which a summarizer S compresses into summary s_t. The model then regenerates a solution using all prior summaries $s_1 \dots s_{t-1}$ as context. We use $T{=}5$ iterations in our experiments.

2.1. Summarization Methods

We compare four summarization approaches:

1. **Base Summary:** Prompts an LLM to summarize the reasoning trace without specific instructions to understand what the summarizer inherently preserves.
2. **Backtracking Summary:** Instructs the summarizer to highlight moments where the model changed approach or revised thinking, motivated by recent work identifying backtracking as crucial for successful reasoning (Gandhi et al., 2025; Venhoff et al., 2025).
3. **Post-Think:** Extracts the final answer after the </think> tag—typically a brief solution justifying the answer. Open-weights reasoning models structure responses with <think> tags enclosing their reasoning, followed by this "post-think" answer.
4. **Last-k truncation:** Takes the last $k{=}404$ tokens before </think>, capturing the most recent reasoning steps. We set k to match the average length of Base summaries for fair comparison.

2.2. Baselines

Self-consistency Samples multiple solutions and takes majority vote (Wang et al., 2023). We use 5 samples to match IS iterations.

Budget forcing (Wait") Extends reasoning by replacing </think> with Wait" (Muennighoff et al., 2025), forcing continued reasoning within context limits.

First-k truncation Negative control taking first k tokens after <think>.

Method	AIME 2024	AIME 2025	Combined
Pass@1	0.733 ± 0.019	0.525 ± 0.050	0.629 ± 0.026
Self-Consistency	0.753 ± 0.056	0.630 ± 0.059	0.692 ± 0.055
"Wait"	0.733 ± 0.024	0.533 ± 0.043	0.633 ± 0.018
Answer Only	0.808 ± 0.021	0.558 ± 0.042	0.683 ± 0.029
Post-Think	0.792 ± 0.028	0.658 ± 0.042	0.725 ± 0.020
First-k	0.658 ± 0.021	0.450 ± 0.029	0.554 ± 0.022
Last-k	0.783 ± 0.017	0.608 ± 0.021	0.696 ± 0.014
Base Summary	0.775 ± 0.016	0.642 ± 0.037	0.708 ± 0.020
Backtracking Summary	**0.817** ± 0.017	**0.667** ± 0.043	**0.742** ± 0.025

Table 1. Accuracy (± 1 SEM) across methods for AIME 2024, AIME 2025, and combined datasets. The best method is bold and the second-best is underlined.

Answer Only Provides only "The answer is `final answer`" as summary.

3. Results & Analysis

3.1. Experimental Setup

We use `DeepSeek-R1-Distill-Qwen-14B` as our reasoner R and `Qwen2.5-14B-Instruct` as our summarizer S, selecting models with room for improvement to demonstrate benefits from extended thinking. We evaluate on AIME 2024 and AIME 2025, comprising 60 high-school mathematics competition problems (Mathematical Association of America, 2024; 2025). Following DeepSeek-AI et al. (2025), we use temperature 0.6, top-p 0.95, and maximum generation length of 32,768 tokens. We report mean accuracy ± SEM across four independent runs per condition.

3.2. Iterated Summarization Boosts Reasoning

Table 1 shows all IS methods (except First-k) outperform Pass@1 on both AIME datasets. Our best method, Backtracking Summary, achieves **+11.3%** overall improvement over Pass@1 (+8.4% AIME 2024, +14.2% AIME 2025) and +5.0% over Self-Consistency. Figure 1 shows Backtracking Summary and Post-Think accuracy increasing nearly monotonically across iterations.

The "Wait" baseline shows minimal improvement (+0.4%), and generates only 1,345 tokens per continuation versus 9,018 tokens per IS iteration. This reveals a key difference: budget forcing merely extends existing reasoning, while IS enables complete solution re-attempts. Even Answer Only (67.1%) outperforms "Wait", showing that minimal prior information aids subsequent attempts.

We examined reasoning traces and their summaries to gain insights into the properties of Iterated Summarization. Unlike methods like self-consistency that generate independent solutions, IS enables the model to retain relevant portions of the previous attempts and focus effort on trying something new. This progressive improvement is evident when early iterations establish correct foundations (such as setting up equations or coordinate systems) but make errors in later steps. Subsequent iterations often preserve these foundations, spending compute on exploring new approaches. In contrast to other summarization methods, backtracking summaries often describe abandoned approaches, allowing the model to learn from these attempts.

Method	Improved (%) $\text{wrong}_{t=1} \rightarrow \text{correct}_{t=5}$	Regressed (%) $\text{correct}_{t=1} \rightarrow \text{wrong}_{t=5}$
"Wait"	2.00 ± 2.00	**0.60** ± 0.60
Answer-Only	30.26 ± 6.76	8.86 ± 3.55
Post-Think	30.11 ± 4.00	2.57 ± 1.86
First-k	21.85 ± 3.64	24.99 ± 3.70
Last-k	23.19 ± 2.71	3.23 ± 2.50
Base Summary	27.88 ± 2.90	4.03 ± 0.85
Backtracking Summary	**31.81** ± 3.49	0.66 ± 0.66

Table 2. Stability of methods for extending test-time compute (± 1 SEM) The best method is bold and the second-best is underlined.

3.3. Stability of Iterated Summarization

We evaluate whether IS enables solving initially unsolved problems (**improvement**) while minimizing regression on initially solved problems. Tracking problems from iteration 0 to 5, our best method (backtracking) achieves the highest improvement-to-regression ratio: solving **31.81%** of initially incorrect problems while regressing on only **0.66%** of initially correct ones. This demonstrates IS as a **stable** method for scaling inference-time compute.

Conclusion

Iterated Summarization is a framework that alternates between reasoning and summarization of reasoning traces to extend a model's thinking time while managing the challenges that come with longer reasoning. On the AIME 2024 & 2025 benchmarks, our best variant, Backtracking Summary, boosts accuracy by over 11% compared to Pass@1, while also outperforming self-consistency and "Wait" baselines. Crucially, these gains are stable: later iterations correct 31.81% of previously unsolved problems while regressing on only 0.66% of solved ones.

Impact Statement

This paper presents IS, a lightweight, model-agnostic framework that empirically extends the reasoning capabilities of LLMs at inference-time. This method can be used in a variety of applications that require a boost in performance without additional post-training or compute requirements. While it is possible that powerful reasoning models could present certain risks when applied to sensitive domains, our proposed framework is unlikely to introduce or magnify these risks. IS enhances model reasoning through efficient context management and iterative thinking, allowing models to operate within existing safety frameworks and constraints.

References

Aggarwal, P. and Welleck, S. L1: Controlling how long a reasoning model thinks with reinforcement learning, 2025. URL https://arxiv.org/abs/2503.04697.

Cheng, J. and Durme, B. V. Compressed chain of thought: Efficient reasoning through dense representations, 2024.

DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Ding, H., Xin, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Wang, J., Chen, J., Yuan, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Ye, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Zhao, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Xu, Y., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Gandhi, K., Chakravarthy, A., Singh, A., Lile, N., and Goodman, N. D. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars, 2025. URL https://arxiv.org/abs/2503.01307.

Mathematical Association of America. Maa invitational competitions - American Invitational Mathematics Examination. Mathematical Association of America, February 2024. URL https://maa.org/maa-invitational-competitions/. Competition examination.

Mathematical Association of America. Maa invitational competitions - American Invitational Mathematics Examination. Mathematical Association of America, February 2025. URL https://maa.org/maa-invitational-competitions/. Competition examination.

Muennighoff, N., Yang, Z., Shi, W., Li, X. L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candès, E., and Hashimoto, T. s1: Simple test-time scaling, 2025. URL https://arxiv.org/abs/2501.19393.

OpenAI. Openai o1 system card. Technical report, OpenAI, September 2024.

Venhoff, C., Arcuschin, I., Torr, P., Conmy, A., and Nanda, N. Understanding reasoning in thinking language models via steering vectors. In *Workshop on Reasoning and Planning for Large Language Models*, 2025. URL https://openreview.net/forum?id=OwhVWNOBcz.

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Huggingface's transformers: State-of-the-art natural language processing, 2020. URL https://arxiv.org/abs/1910.03771.

Xia, H., Li, Y., Leong, C. T., Wang, W., and Li, W. Tokenskip: Controllable chain-of-thought compression in llms, 2025.

Yan, Y., Shen, Y., Liu, Y., Jiang, J., Zhang, M., Shao, J., and Zhuang, Y. Inftythink: Breaking the length limits of long-context reasoning in large language models. *arXiv preprint arXiv:2503.06692*, 2025.

A. Appendix

A.1. Artifact License Details

- **Model weights.** We rely on two open-weight LLMs: `DeepSeek-R1-Distill-Qwen-14B`[1] and `Qwen2.5-14B-Instruct`[2], both released under the MIT license.

- **Benchmark data.** The AIME 2024 & 2025 problem sets are in the public domain (problems reproduced from the Art of Problem Solving archive).

- **Code and prompts.** Our implementation, prompts, and evaluation scripts will be released on GitHub under the permissive MIT license.

A.2. Hyperparameter & Experiment Details

We experimented with using `DeepSeek-R1-Distill-Qwen-14B` itself as the summarizer but found that it would often attempt to solve the problem instead of summarizing.

Both DeepSeek-R1-Distill-Qwen-14B and Qwen-2.5-14B-Instruct have 14.7 billion parameters each. Running the main experiments (4 seeds × 60 problems × 5 iterations) consumed approximately 45 GPU-hours on a single NVIDIA A100-80GB.

We use Hugging Face models and tokenizers for running models and tokenization (Wolf et al., 2020).

We use these sampling parameters for experiments:

Parameter	Reasoning Model	Summarizer
max_tokens	32768	32768
temperature	0.6	0.6
top_p	0.95	0.95
top_k	40	40
presence_penalty	0	0
frequency_penalty	0	0

Table 3. Sampling parameters for reasoning and summarization models.

These parameters were chosen to maintain consistency with the original DeepSeek-R1 paper (DeepSeek-AI et al., 2025). The max_tokens value was set high enough to accommodate the longest reasoning traces while avoiding truncation.

Algorithm 1 Iterated Summarization (IS)

Require: question q, reasoning model R, summarizer S, iterations T
$\Sigma \leftarrow [\,]$ {list of summaries}
for $t = 1$ **to** T **do**
  $r_t \leftarrow R\left(q,\ \texttt{summaries} = \Sigma\right)$
  if $t < T$ **then**
    $s_t \leftarrow S(r_t)$ {compress trace}
    Σ.append(s_t)
  end if
end for
return final answer extracted from r_T =0

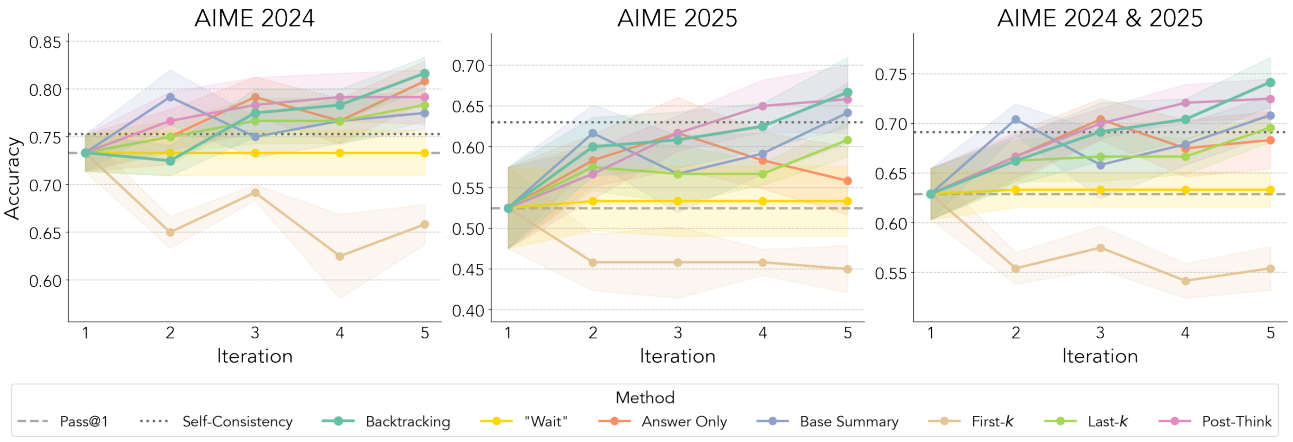

Figure 1. Accuracy on AIME 2024 and 2025 problems by iteration for different methods of extending test-time compute. Shaded regions represent ± 1 SEM.

A.3. Additional Figures

Approach	Iter 1	Iter 2	Iter 3	Iter 4	Average
Backtracking	2.14	2.09	2.00	1.94	2.04
Base Summarization	0.45	0.44	0.39	0.50	0.45

Table 4. Average Backtracking Behavior Counts For Summaries Across Iterations

Method	Iter 1	Iter 2	Iter 3	Iter 4	Iter 5	Overall
Backtracking	1.28	1.79	2.00	1.88	1.98	1.78
Base Sum	1.46	2.00	2.14	1.87	2.16	1.93

Table 5. Average Backtracking Behavior Counts For Reasoning Across Iterations

AIME Problems Prompt

Solve the following AIME problem. All answers are integers ranging from 0 to 999, inclusive. Report your answer in `\boxed{}` format.

PROBLEM:
{question}

Figure 3. AIME Problem Prompt Template

[1] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-14B
[2] https://huggingface.co/Qwen/Qwen2.5-14B-Instruct

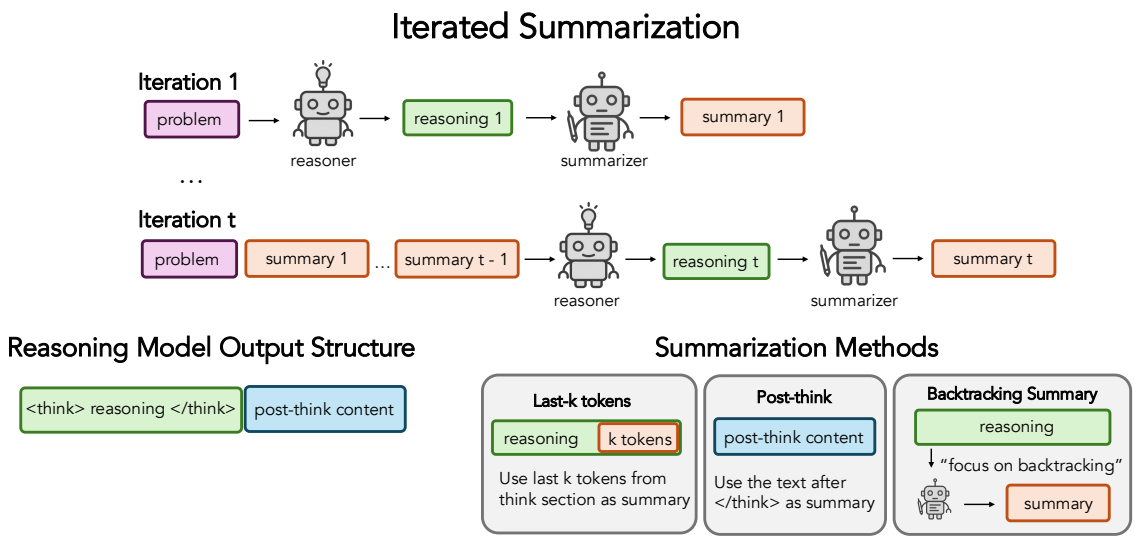

Figure 2. Iterated Summarization Overview. **Top:** Iterated summarization alternates between summarizing lengthy reasoning traces and using those summaries as context for a reasoning model's next attempt at solving a problem. **Left:** structure of reasoning model outputs, divided into reasoning and post-think content. **Right:** Illustration of three summarization methods: Last-k, Post-Think, and Backtracking summary.

Base Summarization Prompt

Summarize the following attempted solution to the problem:

PROBLEM:
`{question}`
ATTEMPTED SOLUTION:
`{reasoning}`
SUMMARY:

Figure 4. Base Summarization Prompt Template

Backtracking Summarization Prompt

Summarize the following attempted solution to the problem, emphasizing the instances where the model changed its strategy, revised a previous decision, or explicitly backtracked from a prior line of reasoning.

PROBLEM:
`{question}`
ATTEMPTED SOLUTION:
`{reasoning}`
SUMMARY:

Figure 5. Backtracking Summarization Prompt Template

Improve Using Summaries Prompt

Solve the following AIME problem. All answers are integers ranging from 0 to 999, inclusive. Report your answer in `\boxed{}` format.

PROBLEM:
`{question}`
Here are summaries of your previous solution attempts:
`{summaries}`

Based on your previous solution attempts, evaluate whether the most recent approach and answer are correct. If not, consider a different approach.

Figure 6. Prompt Template for later iterations to use and build from previous summaries

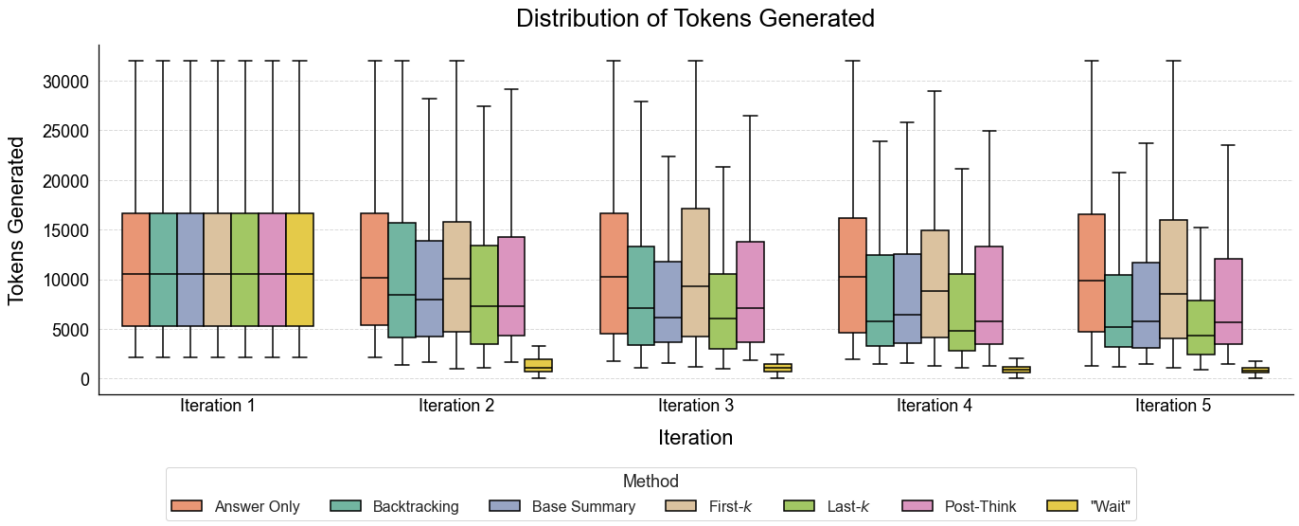

Figure 7. Token Count Distribution

Method	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
Answer Only	12438.47 ± 521.66	11839.93 ± 488.61	11821.38 ± 517.14	11672.89 ± 503.48	11939.12 ± 535.31
Backtracking	12438.47 ± 521.66	10609.80 ± 509.23	9036.55 ± 424.25	8444.58 ± 425.09	7979.73 ± 438.79
Base Summary	12438.47 ± 521.66	9596.15 ± 401.03	8427.26 ± 393.72	9116.83 ± 484.48	8246.03 ± 421.15
First-k	12438.47 ± 521.66	11517.05 ± 514.59	11451.91 ± 543.41	10740.58 ± 496.92	10849.08 ± 522.41
Last-k	12438.47 ± 521.66	9225.58 ± 439.45	7627.92 ± 372.73	7438.36 ± 417.94	5919.89 ± 320.07
Post-Think	12438.47 ± 521.66	9759.13 ± 476.39	9240.95 ± 419.28	8854.25 ± 465.18	8341.21 ± 419.39
“Wait”	12438.47 ± 521.66	1635.50 ± 104.94	1390.48 ± 92.66	1243.05 ± 97.79	1109.97 ± 83.68

Table 6. Summary statistics (± SEM) for each method across iterations.