Think Clearly: Improving Reasoning via Redundant Token Pruning

Paper

Daewon Choi[1] Jimin Lee[2] Jihoon Tack[1] Woomin Song[1, 3] Saket Dingliwal[3] Sai Muralidhar Jayanthi[3]
Bhavana Ganesh[3] Jinwoo Shin[1] Aram Galstyan[3] Sravan Babu Bodapati[3]

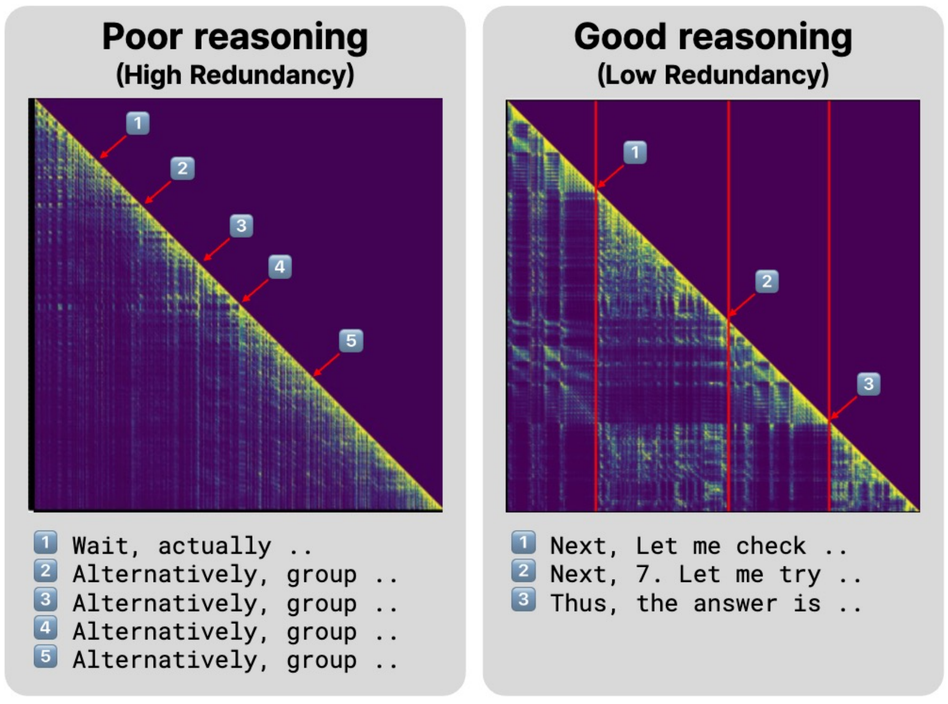
[1]KAIST [2]Korea University [3]Amazon AGI

Contact: daeone0920@kaist.ac.kr

TL;DR: Just think clearly-
Removing KV Cache from redundant tokens improves reasoning for free.

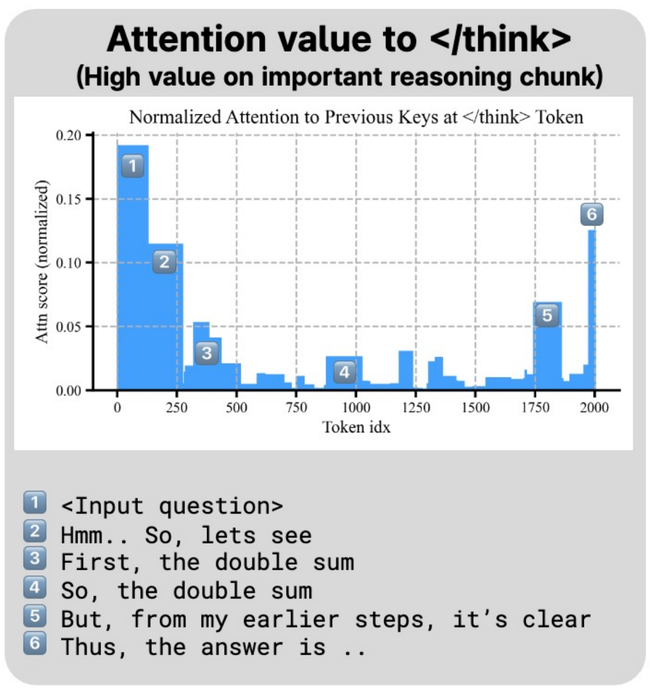
Not All Tokens Matter for Reasoning

1. Existence of redundant reasoning tokens



✓ Attention maps when the model fails to produce the correct answer (i.e., poor reasoning) and when it succeeds (i.e., good reasoning)
⟶ Poor reasoning leads to highly redundant attention patterns !

2. Attention score to </think>



✓ Attention scores associated with the end-of-thinking token </think>.
⟶ </think> attends to key reasoning chunks that contain crucial information for deriving the final answer !

Improving Reasoning with Redundant Token Pruning

Step 1. Identifying redundant tokens via self-summarization

During an intermediate step of decoding, forward the following summarization prompt:

"Time is up. Given the time I've spent and the approaches I've tried, I should stop thinking and now write summarization in one sentence.</think>"

Use end of thinking token !

Token importance score

$$s_t^{(\ell,h)} = \alpha_{\text{</think>}\to t}^{(\ell,h)}$$

Attention weight from the </think> !

Step 2. Step-aware eviction with hierarchical budget allocation

Aggregate importance score per reasoning step

$$c_r^{(\ell)} = \frac{1}{|H \cdot r|} \sum_{h \in H} \sum_{t \in r} s_t^{(\ell,h)}$$

Given a token eviction budget k, Evict KV Cache of tokens from redundant steps

$$e_{\tilde{r}_i}^{(\ell)} = \min\left(|\tilde{r}_i^{(\ell)}|,\ k - \sum_{j=1}^{i-1} e_{\tilde{r}_j^{(\ell)}} \right)$$

# of evicted tokens per step

Experiments

Removing redundant tokens improves overall accuracy across mathematical benchmarks without any training ! * Parenthesis: response length

Model	Method	Dataset						
		MATH	Minerva	GaoKao	AIME2024	AIME2025	AMC2023	Average
Qwen2.5-7B	FullKV	87.0 (3397)	59.9 (3391)	65.8 (3845)	36.7 (7060)	23.3 (7133)	75.0 (5004)	57.9 (4971)
	Ours	**87.2** (2926)	**60.5** (3471)	**67.1** (4219)	**46.7** (6841)	**36.7** (6905)	**82.5** (4488)	**63.4** (4808)
Llama3.1-8B	FullKV	81.0 (3389)	45.9 (4060)	67.1 (4689)	**33.3** (7067)	13.3 (7088)	75.0 (4986)	52.6 (5213)
	Ours	**83.8** (3345)	**48.1** (3941)	**69.8** (4532)	**33.3** (7210)	**23.3** (7375)	**77.5** (4700)	**55.9** (5183)

(Qwen2.5-7B / Llama3.1-8B are reasoning LLMs distilled from DeepSeek-R1)
* We perform eviction per every 200 / 300 generation steps

Does token eviction itself leads to improved performance?

Score	AIME2024	AIME2025
FullKV	36.7	23.3
Random	36.7	33.3
H2O	40.0	26.7
Ours	**46.7**	**36.7**

(Random: uniformly at random
H2O: lowest accumulated attention)

Our method can be effective under aggressive compression.

	Compression ratio	
Method	25%	50%
FullKV	42.6	
Streaming-LLM	35.4	39.4
H2O	34.6	39.2
Pyramid-Infer	30.6	40.0
Ours	**36.0**	**40.2**

Component analysis

Summ	Step	AIME2024	AMC2023
✗	✗	40.0	70.0
✓	✗	36.7	77.5
✓	✓	**46.7**	**82.5**

(Summ: self-summarization
Step: step-aware token eviction)

Non-mathematical benchmark

Method	GPQA
FullKV	32.0 (6418)
Ours	**36.4** (6277)

(GPQA: Multiple-choice science)

Discussion points

Q1. Segment of reasoning steps

"Wait" "Alternatively" "Another angle" "Another approach" "But wait" "Hold on" "Hmm" "Maybe" "Looking back" "Okay" "Let me" "First" "Then" "Alright" "Compute" "Correct" "Good" "Got it" "I don't see any errors" "I think" "Let me double-check" "Let's see" "Now" "Remember" "Seems solid" "Similarly" "So" "Starting" "That's correct" "That seems right" "Therefore" "Thus"

Q2. Connection to overthinking

	AIME24	AMC23
FullKV	36.7	75.0
Chain of Draft	23.3	72.5
Break the Chain	23.3	72.5
Ours	**46.7**	**82.5**

Overthinking methods hurt performance by removing output redundancy, while we improve accuracy by targeting internal redundancy.

Q3. Which tokens are frequently evicted?

Token	Frequency (normalized)
, (comma)	1.00
2	0.97
" " (blank quote)	0.84
1	0.62
4	0.49
. (full stop)	0.46

They are punctuation marks / numbers that are contextually redundant.