Think-at-Hard (TaH)

Selective Latent Iterations to Improve Reasoning Language Models

Tianyu Fu*, Yichen You*, Zekai Chen, Guohao Dai, Huazhong Yang, Yu Wang†

**equal contribution*

†corresponding author (yu-wang@tsinghua.edu.cn)

presenter: Tianyu Fu (fuvty@outlook.com)

Tsinghua University

INFINIGENCE

Shanghai Jiao Tong University

Authors

Tianyu Fu*
Tsinghua University
Inference AI

Yichen You*
Tsinghua University

Zekai Chen
Tsinghua University

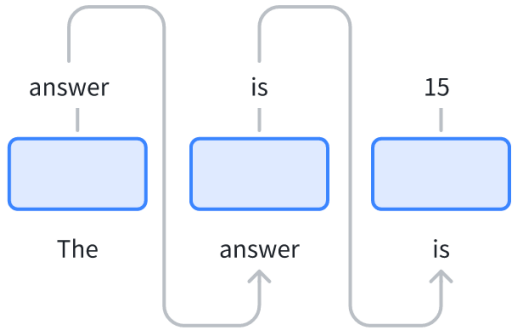
Guohao Dai
Shanghai Jiao Tong University
Inference AI

Huazhong Yang
Tsinghua University

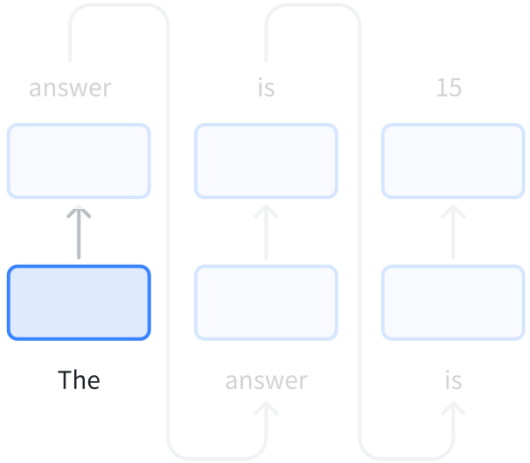
Yu Wang†
Tsinghua University

Background: Looped Transformer

- Looped transformers
  - After a standard forward, model feeds latent states back as next-iter. inputs.
  - After some internal iterations (loop), model then verbalizes the output token
  - During auto-regressive generation, interleave between latent and verbal space



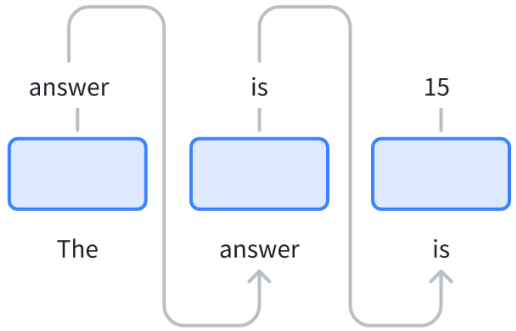
standard transformers



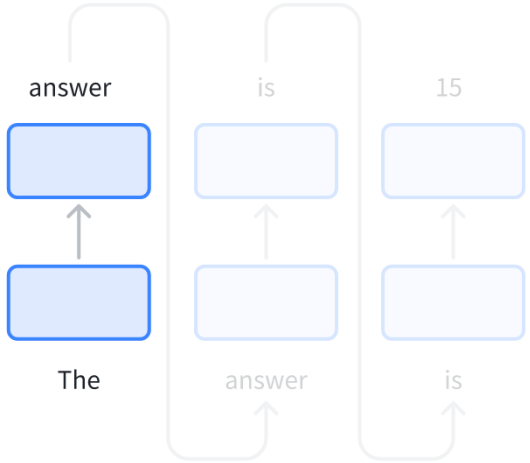
looped transformers

Background: Looped Transformer

- Looped transformers
  - After a standard forward, model feeds latent states back as next-iter. inputs.
  - After some internal iterations (loop), model then verbalizes the output token
  - During auto-regressive generation, interleave between latent and verbal space



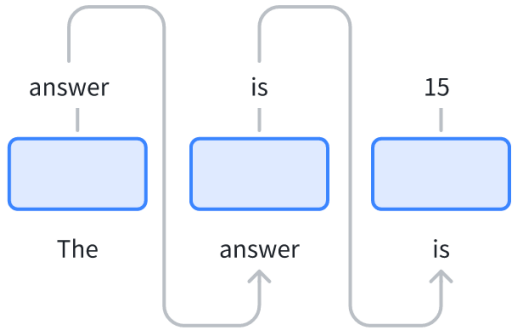
standard transformers



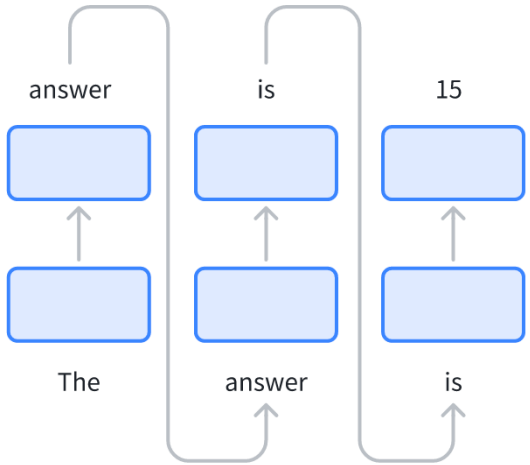
looped transformers

Background: Looped Transformer

- Looped transformers
  - After a standard forward, model feeds latent states back as next-iter. inputs.
  - After some internal iterations (loop), model then verbalizes the output token
  - During auto-regressive generation, interleave between latent and verbal space



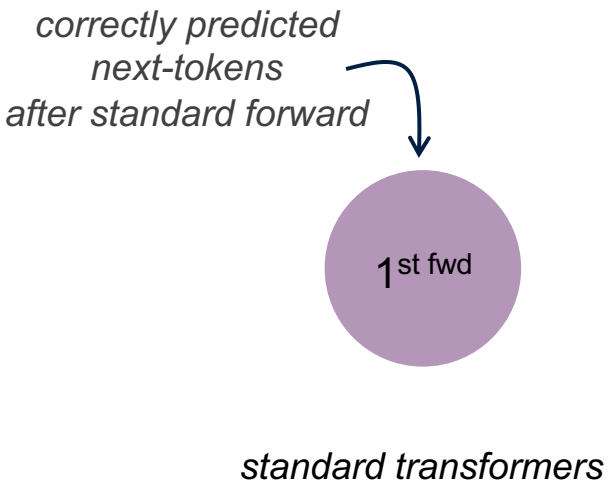
standard transformers



looped transformers

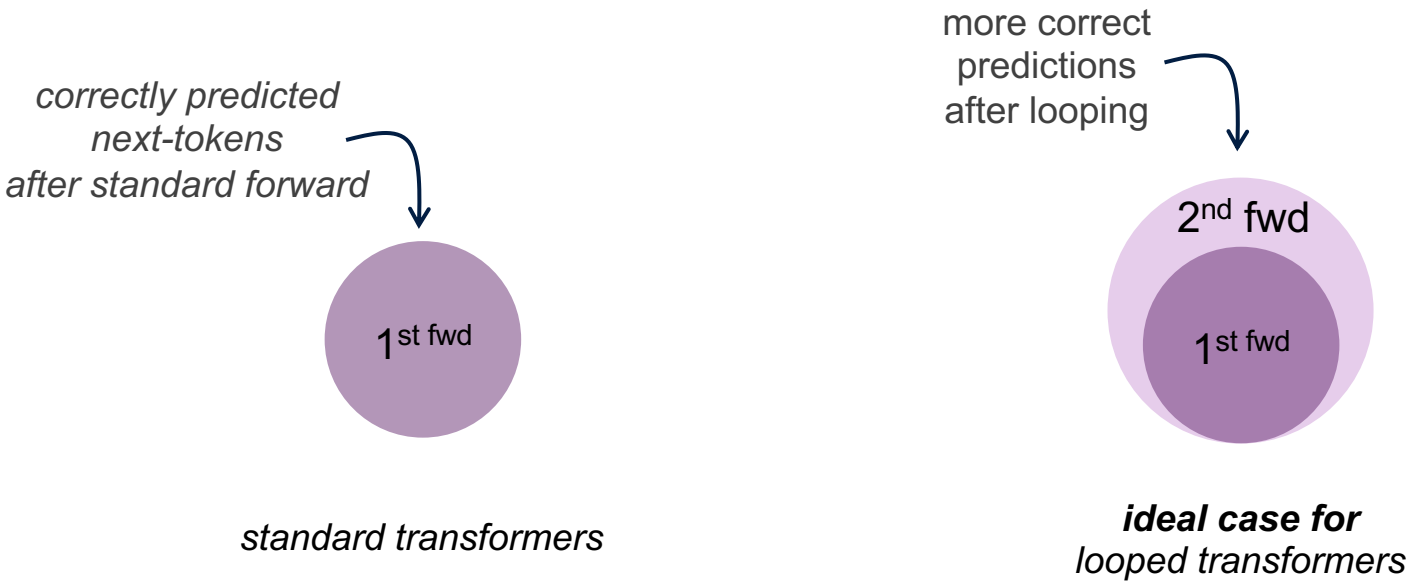
Background: Looped Transformer

- Prediction accuracy during looping



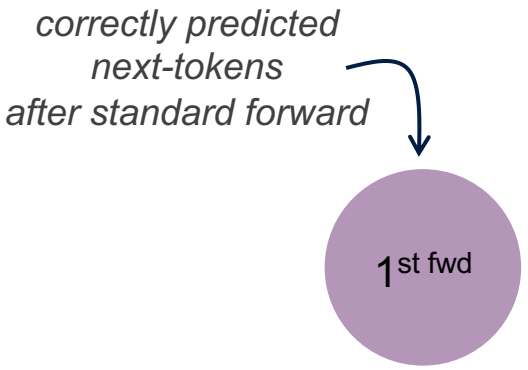
Background: Looped Transformer

- Prediction accuracy during looping

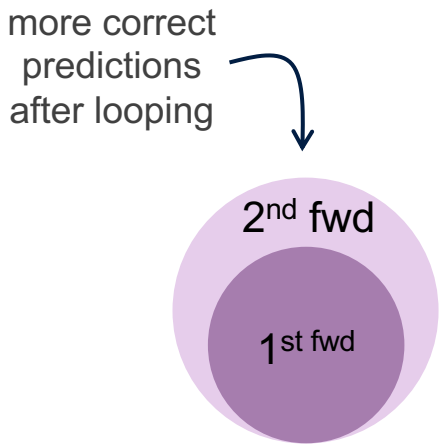


Background: Looped Transformer

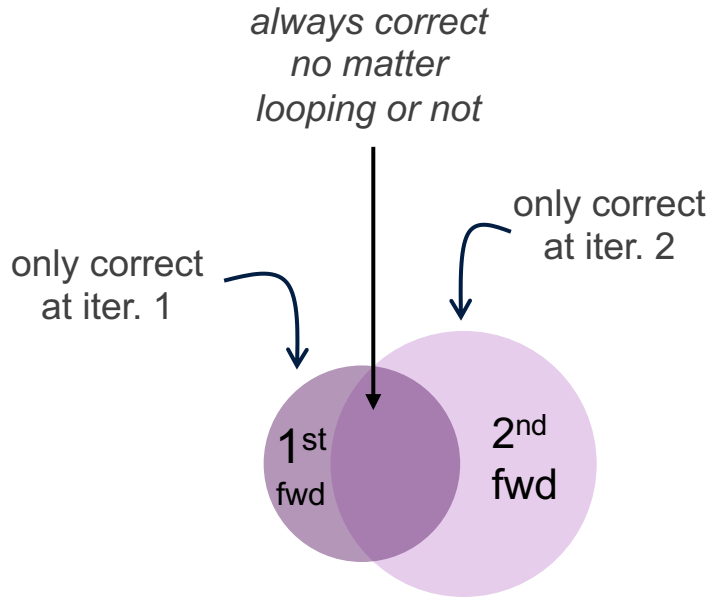
- Prediction accuracy during looping



standard transformers



ideal case for
looped transformers

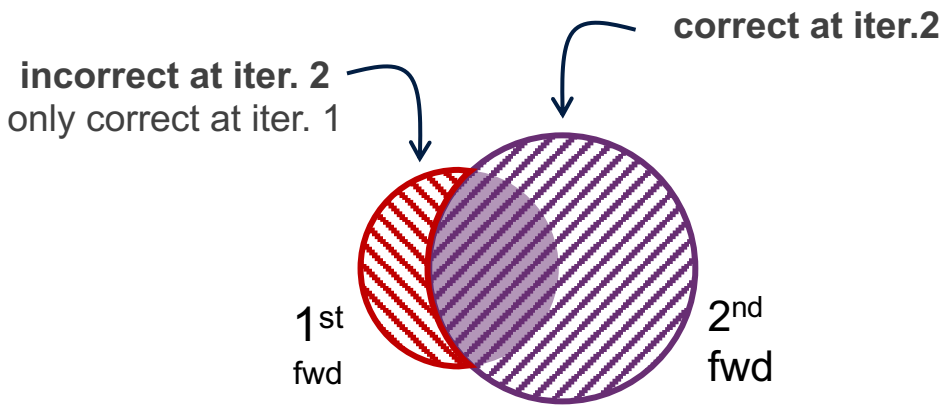


reality for
looped transformers

Latent Overthinking Phenomenon

When always iterating twice (Always2),
the second iteration sometimes revise correct answers into wrong ones

NTP accuracy illustration
for *Always2* looped transformers



example
for looped transformer prediction

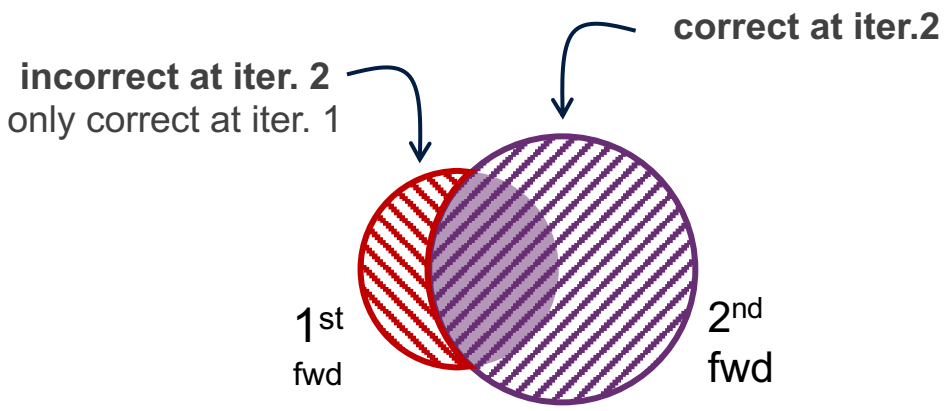
user query: How many numbers from 1–100 contain the digit “7”?

iter. 1 reply:	I’ll **reason** step by step. ✓	We can **list** all the numbers. ✗
latent iteration:	***latent overthink*** *Maybe directly answer this time.*	***selective latent think*** *List all? Too many cases.*
iter. 2 reply:	I’ll **directly** answer 10. ✗	We can **split** tens and ones. ✓

Latent Overthinking Phenomenon

When always iterating twice (Always2),
the second iteration sometimes revise correct answers into wrong ones

accuracy illustration
for *Always2* looped transformers



NTP accuracy
for post-trained Ouro [1] on OpenR1 validation set

2.1% correct only at iter.1

71.0% correct at both iters

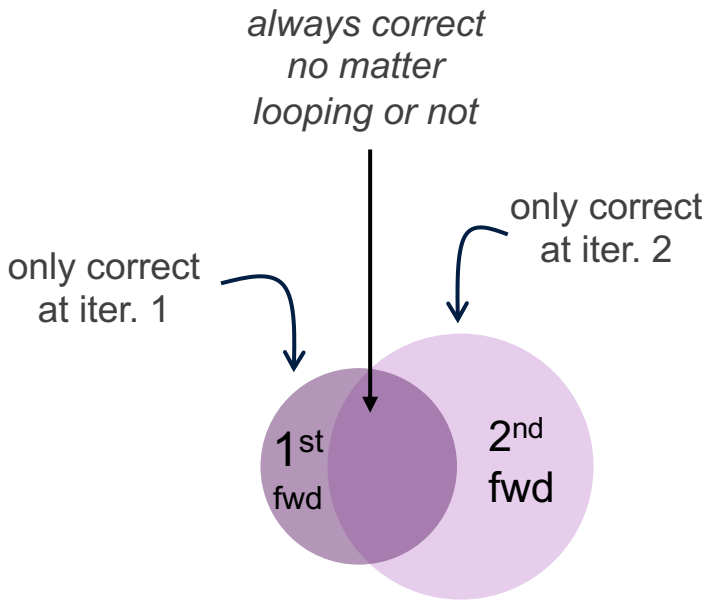
8.7% correct only at iter.2

18.2% incorrect at both iters

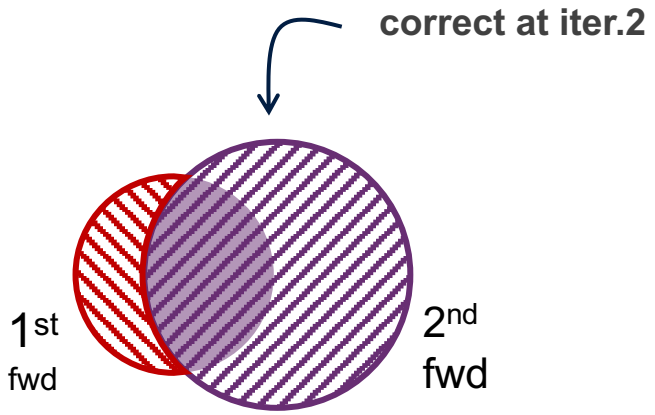
[1] Zhu, Rui-Jie, et al. "Scaling latent reasoning via looped language models." arXiv preprint arXiv:2510.25741 (2025).

Oracle Policy for Looping

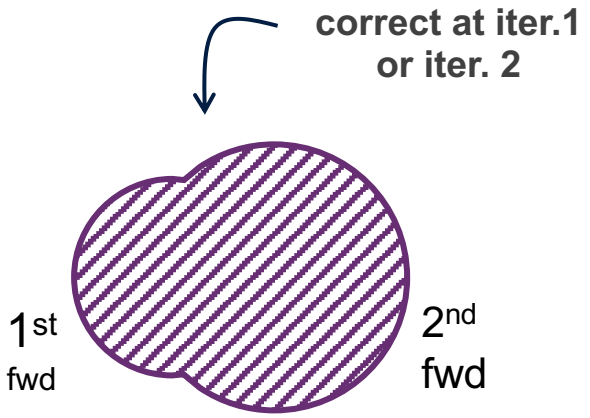
- Prediction accuracy during looping



reality for
looped transformers



Always2
loop twice at all tokens

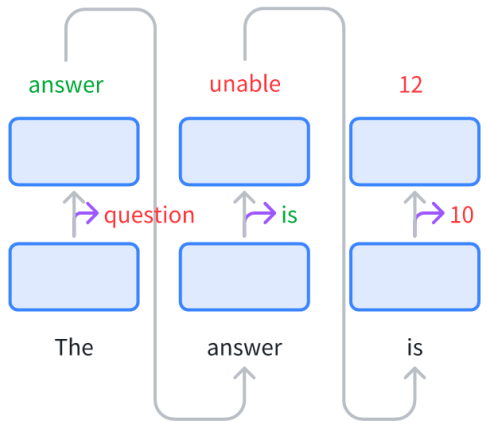


Oracle
loop only when
previous iteration is wrong

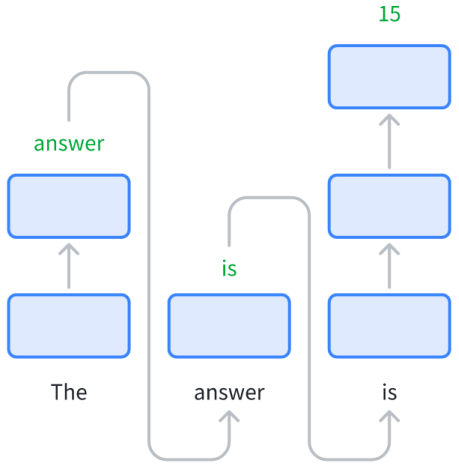
Oracle Policy for Looping

If we have a **perfect dynamic loop policy (oracle policy)**,
what is the maximum performance gain we can get?

design an oracle policy
to test performance ceiling



compare predictions over larger LLM across all iterations



only loop when the current iteration makes wrong prediction

huge performance gain
when trained and tested with oracle looping

Iter. Policy	NTP	AMC23	MMLU100	HE++
Always1	73.1	38.1	56.0	39.6
Always2	79.7	40.6	60.0	40.9
Orcl.	81.8/ + 2.1	47.9/ + 7.3	62.0/ + 2.0	43.3/ + 2.4
Orcl. w.TaH	89.3/ + 9.6	68.8/ + 28.2	85.0/ + 25.0	72.9/ + 32.0

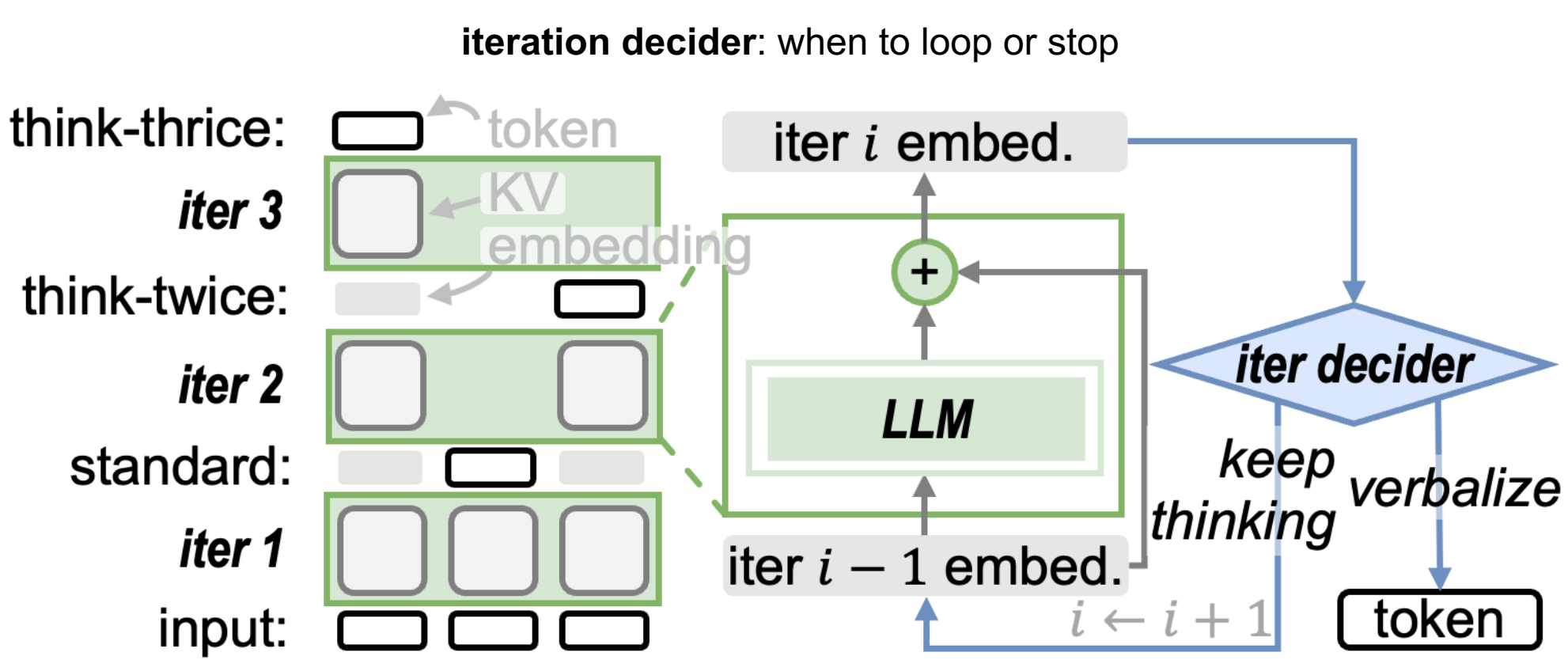
oracle policy ***only loop on 12-19% tokens but increase accuracy by up to 32%***

Selectively skipping latent iterations on most tokens can further increase reasoning quality.

key research opportunity

TaH Model Architecture

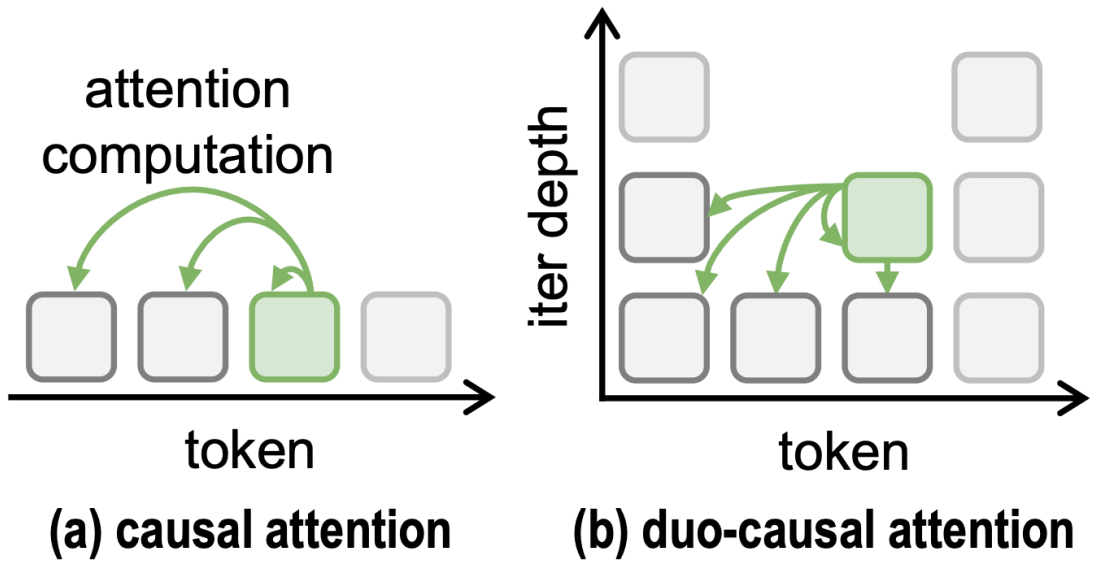
Think-at-Hard's architecture design: iteration decider
optimized for efficient dynamic looping



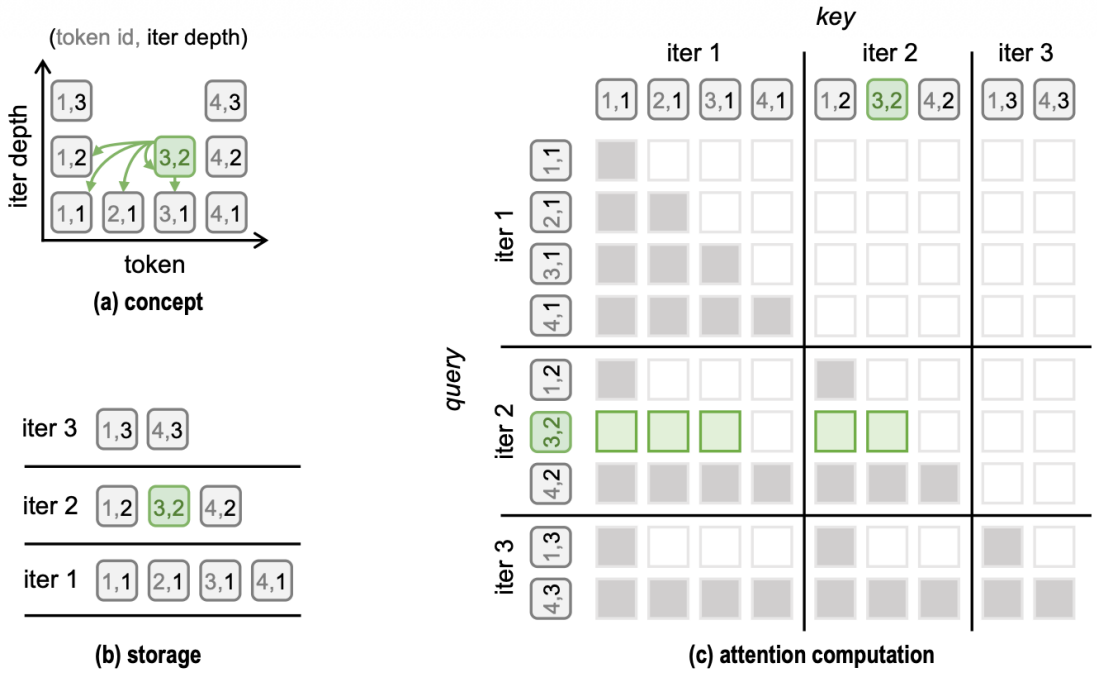
TaH Model Architecture

Think-at-Hard's architecture design: duo-causal attention
optimized for efficient dynamic looping

Duo-causal attention
cross-iter. info flow with full token parallelism



implemented with
attention with masking



TaH: Dynamic Looped Transformer

Think-at-Hard's training method
stabilize post-training of dynamic looping

stabilize training
with oracle policy

- two stage training
  - **stage 1: backbone supervision under π**
    - optimize the LLM by executing iterations according to π
    - compute the standard next-token prediction loss, but applied at the oracle-determined depth for each token
  - **stage 2: decider imitation under frozen backbone**
    - freeze the backbone model
    - binary cross-entropy, imitate the continuation labels from the oracle policy

training dynamics of the backbone
converges faster than standard

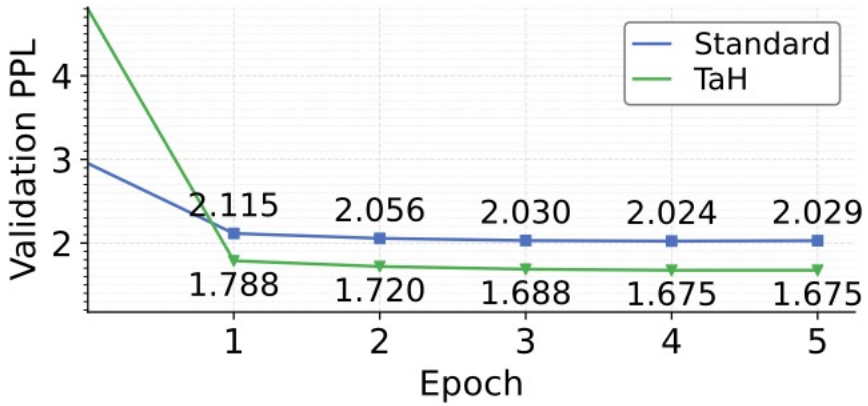
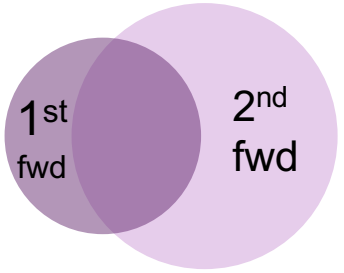


Figure 3. Training dynamics of the LLM backbone on Qwen3-0.6B-Base. TaH converges rapidly and achieves lower perplexity.

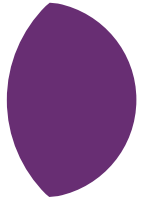
Accuracy Landscape Change

- Reducing the overlap to increase the coverage



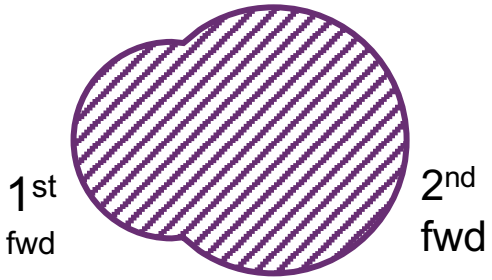
Train with always loop

71.0%

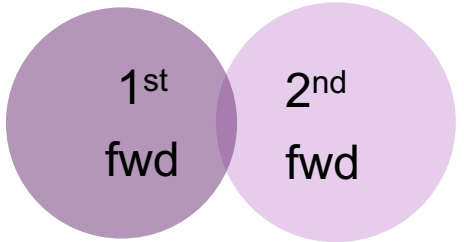


correct at either depth

81.8%



inference with oracle

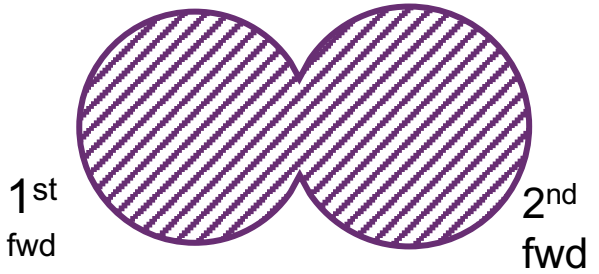


Train with oracle

37.0%

correct at either depth

89.3%



inference with oracle

TaH: Dynamic Looped Transformer

Setup

- Base model:
  - Qwen3-0.6B-Base
  - Qwen3-1.7B-Base
  - Qwen3-4B-Base
- Baselines
  - Standard: always verbalizes directly and reduces to the original Qwen model;
  - AlwaysThink: applies the maximum number of latent iterations to all tokens;
  - SoftThink: latent space thinking;
  - Ouro: looped transformer that scales iteration depths
- Method
  - we prune one layer from the base model so that TaH matches the parameter count of baselines. The layer is chosen to minimize the increase in validation loss.
  - we also report results for an unpruned variant, TaH+, which adds less than 3% extra parameters

[1] Tianyu Fu*, Yichen You*, Zekai Chen et al. "Think-at-Hard: Selective Latent Iterations to Improve Reasoning Language Models" *Submitted to ICML'26. [Under Review]*

TaH: Dynamic Looped Transformer

Results

	AIME25	Olympiad	AMC23	MATH500	GSM8K	GPQA	MMLU	HE++	MBPP++	Average
					0.6B					
Standard	1.9	15.4	22.7	39.9	58.2	31.1	54.2	16.8	28.8	29.9
SoftThink	2.9	14.0	22.2	39.6	55.9	24.7	53.0	14.3	29.5	28.5
Ouro	2.1	14.2	19.7	37.4	56.6	**35.4**	54.0	18.9	23.5	29.1
AlwaysThink	1.3	12.6	21.9	37.8	52.6	30.8	51.4	9.1	13.8	25.7
TaH	2.1	19.1	24.1	46.2	63.6	29.0	56.4	21.6	33.9	32.9/ + 3.0
TaH+	**4.6**	**20.6**	**24.7**	**51.8**	**67.6**	31.3	**59.0**	**22.0**	**35.1**	**35.2**/ + 5.3
					1.7B					
Standard	10.8	33.8	39.7	67.8	80.2	30.3	74.1	39.0	51.9	47.5
SoftThink	5.4	30.7	40.3	64.8	80.0	33.3	73.5	43.3	49.1	46.7
Ouro	10.8	31.3	40.6	68.2	79.8	32.8	72.4	40.9	45.5	46.9
AlwaysThink	7.5	31.0	40.9	63.2	74.2	30.5	69.6	16.4	25.6	39.9
TaH	13.8	37.2	40.9	71.4	**84.8**	33.3	74.8	50.0	55.3	51.3/ + 3.8
TaH+	**15.4**	**37.6**	**48.4**	**72.6**	84.5	**39.4**	**76.6**	**51.5**	**57.5**	**53.7**/ + 6.2
					*4B**					
Standard	24.2	50.4	64.2	84.2	91.7	46.2	85.4	69.2	65.9	64.6
SoftThink	25.8	50.2	63.1	85.0	**92.5**	**50.3**	86.0	69.2	66.7	65.4
Ouro	25.0	51.4	64.4	83.8	90.7	50.0	85.9	70.7	66.7	65.4
TaH	27.1	50.5	**69.7**	**85.8**	91.0	48.5	86.2	70.1	67.7	66.3/ + 1.7
TaH+	**27.9**	**52.6**	68.1	85.6	91.7	49.0	**86.6**	**72.0**	**68.1**	**66.8**/ + 2.2

TaH: Dynamic Looped Transformer

Ablations

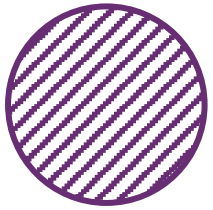
Variant	MATH500	AMC23	Olympiad	Average
TaH	**51.2**	**32.5**	**23.9**	**35.9**
Model Architecture				
Iteration Depth (TaH: Neural decider)				
Always-1	47.2	23.4	18.8	29.8/−6.1
Always-2	32.8	15.6	10.2	19.5/−16.4
Attention (TaH: Duo-causal)				
Causal@iter1	47.8	24.4	19.4	30.5/−5.4
Causal@current	42.0	23.8	16.4	27.4/−8.5
Depth Adapters (TaH: LoRA + Residual)				
w/o LoRA	51.6	29.7	22.4	34.6/−1.3
w/o LoRA & Res.	49.2	22.5	21.2	31.0/−4.9

Variant	MATH500	AMC23	Olympiad	Average
TaH	**51.2**	**32.5**	**23.9**	**35.9**
Training Scheme				
Supervision Type (TaH: Token-only)				
Token+latent	49.4	29.6	15.9	31.6/−4.3
Iteration Policy during LLM training (TaH: Oracle policy)				
Decider-based	44.8	24.1	17.3	28.7/−7.2
Dynamic	11.0	2.8	2.7	5.5/−30.4
Discrepancy metric in π (TaH: Top1 mismatch)				
Cross-entropy	47.4	21.2	20.4	29.7/−6.2
Entropy	42.0	21.9	16.9	26.9/−9.0

Takeaway: Dynamic Looping for Better Performance

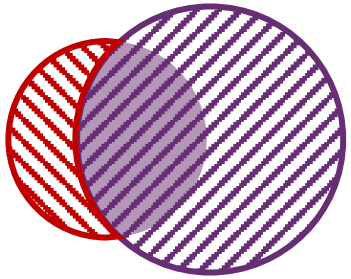
- Accuracy landscape and closing the gap for oracle

73.1%



standard
transformers

79.7%

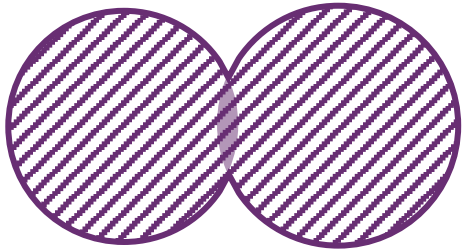


always2
looped transformers

81.8%

TaH oracle at inference
looped transformers

89.3%



TaH oracle at train & inference
looped transformers

** figures are used for illustrative purposes, not reflecting the real proportion*

Thank you

Think-at-Hard: Selective Latent Iterations to Improve Reasoning Language Models

Tianyu Fu*, Yichen You*, Zekai Chen, Guohao Dai, Huazhong Yang, Yu Wang†

*equal contribution †corresponding author (yu-wang@tsinghua.edu.cn)

presenter: Tianyu Fu (fuvty@outlook.com)

Tsinghua University INFINIGENCE SHANGHAI JIAO TONG UNIVERSITY