RHO-1: Not All Tokens Are What You Need

Zhenghao Lin[⋆χφ] **Zhibin Gou**[⋆πφ] **Yeyun Gong**[⋄φ] **Xiao Liu**[φ] **Yelong Shen**[φ]
Ruochen Xu[φ] **Chen Lin**[⋄χ] **Yujiu Yang**[⋄π] **Jian Jiao**[φ] **Nan Duan**[φ] **Weizhu Chen**[φ]
[χ]Xiamen University [π]Tsinghua University [φ]Microsoft
https://aka.ms/rho

Post Pretrain

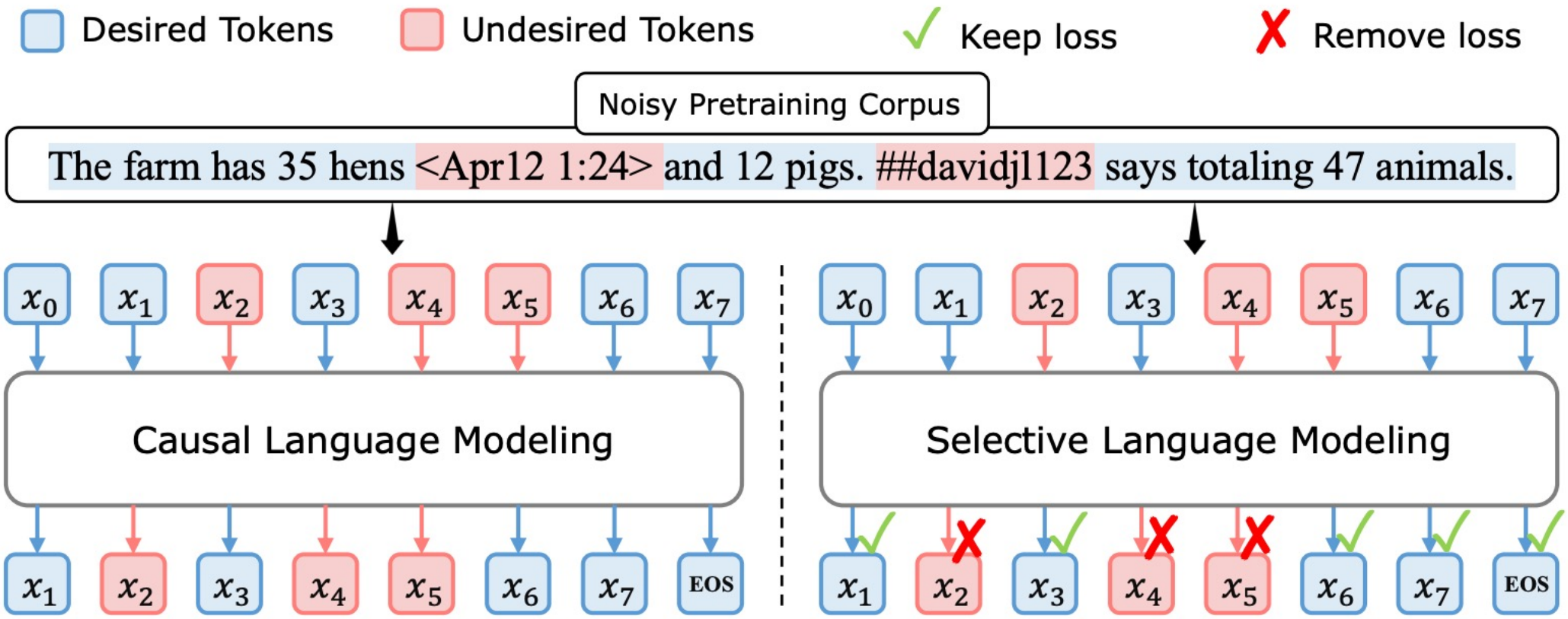
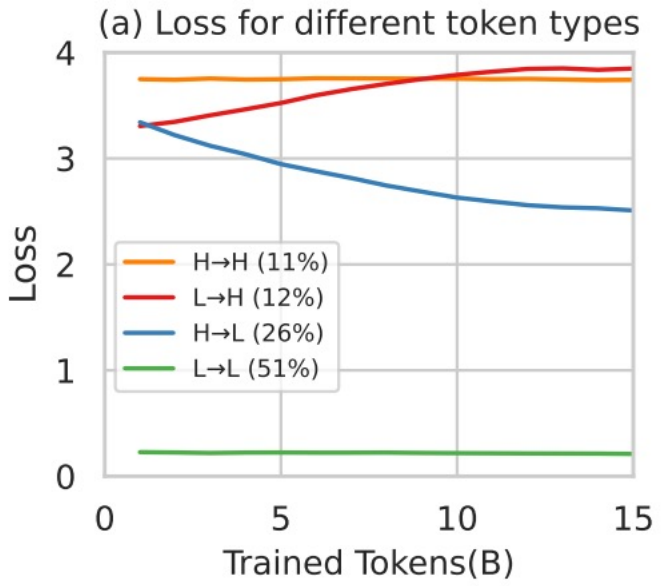


Figure 2: **Upper:** Even an extensively filtered pretraining corpus contains token-level noise. **Left:** Previous Causal Language Modeling (CLM) trains on all tokens. **Right:** Our proposed Selective Language Modeling (SLM) selectively applies loss on those useful and clean tokens.

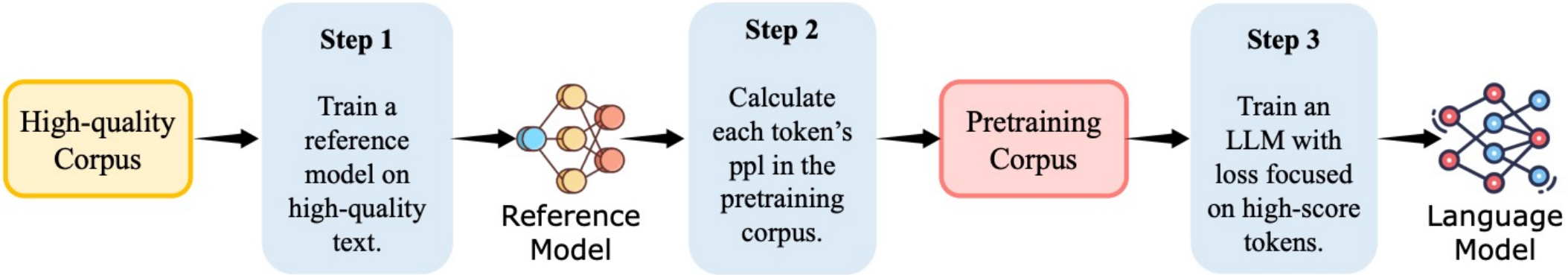
Post Pretrain



» Sulphuric Acid in Australia Select A Forum Fundamentals » Chemistry in General » Organic Chemistry » Reagents and Apparatus Acquisition » Beginnings » Responsible Practices » Miscellaneous » The Wiki Special topics » Technochemistry » Energetic Materials » Biochemistry » Radiochemistry » Computational Models and Techniques » Prepublication Non-chemistry » Forum Matters » Legal and Societal Issues \n \n Pages: 1 2 \n Author: Subject: Sulphuric Acid in Australia \n hissingnoise \n International Hazard \n \n Posts: 3939 \n Registered: 26-12-2002 \n Member Is Offline \n \n Mood: Pulverulescent! \n \n I've stated several times on various threads, that SO₃ produces a practically incondensable acid mist when led to water and, BTW, at 700°C the decomposition rate of SO₃ is ~87% . . . \n Cracking Na₂S₂O₇ proceeds at ~466°C and the issuing gasses are readily absorbed by conc. H₂SO₄ to form oleum! \n \n Phthalic Acid \n Harmless \n \n Posts: 19 \n Registered: 7-8-2011 \n Location: Australia \n Member Is Offline \n \n Mood: No Mood \n \n That's a good idea Neil, I'll be sure to try that next time (probably for H2O2). Just went to Tradelink and asked if they sold Moflo drain cleaner. The guy said yeah and I asked for a liter of it. No problems whatsoever, he just said "be careful with it". It was $45 but a liter will last me a while and making it myself would've been vastly more expensive I imagine. Success! MeSynth Hazard to Others Posts: 107 Registered: 29-7-2011 Member Is Offline Mood: http://www.youtube.com/watch?v=5ZltqlVuDIo Sulfuric acid can be produced in the laboratory by burning sulfur in air and dissolving the gas produced in a hydrogen peroxide solution. SO2 + H2O2 → H2SO4 this was found on wikipedia... did you not look through the sullfuric acid wiki before boiling down batery acid? anyways... There are some good videos on youtube that demonstrate how to synthesize sulfuric acid using different methods. The drain cleaner you get from the store will be impure and may contain organic matter that discolors the acid.

Figure 11: **Sample text containing four categories of tokens.** Among them, blue represents tokens of categorie H→L, green indicates tokens of categorie L→L, yellow signifies tokens of categorie H→H, and red denotes tokens of categorie L→H.

Selective Language



$$\mathcal{L}_{\text{CLM}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\log P(x_i|x_{<i};\theta)$$

$$\mathcal{L}_{\Delta}(x_i) = \mathcal{L}_{\theta}(x_i) - \mathcal{L}_{\text{RM}}(x_i)$$

$$\mathcal{L}_{\text{SLM}}(\theta) = -\frac{1}{N * k\%}\sum_{i=1}^{N} I_{k\%}(x_i)\cdot\log P(x_i|x_{<i};\theta)$$

Experiment

Model	$\|\theta\|$	Data	Uniq. Toks*	Train Toks	GSM8K	MATH†	SVAMP	ASDiv	MAWPS	TAB	MQA	MMLU STEM	SAT‡	AVG
1-2B Base Models														
Tinyllama	1.1B	-	-	-	2.9	3.2	11.0	18.1	20.4	12.5	14.6	16.1	21.9	13.4
Phi-1.5	1.3B	-	-	-	32.4	4.2	43.4	53.1	66.2	24.4	14.3	21.8	18.8	31.0
Qwen1.5	1.8B	-	-	-	36.1	6.8	48.5	63.6	79.0	29.2	25.1	31.3	40.6	40.0
Gemma	2.0B	-	-	-	18.8	11.4	38.0	56.6	72.5	36.9	26.8	34.4	50.0	38.4
DeepSeekLLM	1.3B	OWM	14B	150B	11.5	8.9	-	-	-	-	-	29.6	31.3	-
DeepSeekMath	1.3B	-	120B	150B	23.8	13.6	-	-	-	-	-	33.1	56.3	-
Continual Pretraining on Tinyllama-1B														
Tinyllama-CT	1.1B	OWM	14B	15B	6.4	2.4	21.7	36.7	47.7	17.9	13.9	23.0	25.0	21.6
RHO-1-Math	1.1B	OWM	14B	9B	29.8	14.0	49.2	61.4	79.8	25.8	30.4	24.7	28.1	38.1
Δ				-40%	+23.4	+11.6	+27.5	+24.7	+32.1	+7.9	+16.5	+1.7	+3.1	**+16.5**
RHO-1-Math	1.1B	OWM	14B	30B	36.2	15.6	52.1	67.0	83.9	29.0	32.5	23.3	28.1	40.9
≥ 7B Base Models														
LLaMA-2	7B		-	-	14.0	3.6	39.5	51.7	63.5	30.9	12.4	32.7	34.4	31.4
Mistral	7B		-	-	41.2	11.6	64.7	68.5	87.5	52.9	33.0	49.5	59.4	52.0
Minerva	8B	-	39B	164B	16.2	14.1	-	-	-	-	-	35.6	-	-
Minerva	62B	-	39B	109B	52.4	27.6	-	-	-	-	-	53.9	-	-
Minerva	540B	-	39B	26B	58.8	33.6	-	-	-	-	-	63.9	-	-
LLemma	7B	PPile	55B	200B	38.8	17.2	56.1	69.1	82.4	48.7	41.0	45.4	59.4	50.9
LLemma	34B	PPile	55B	50B	54.2	23.0	67.9	75.7	90.1	57.0	49.8	54.7	68.8	60.1
Intern-Math	7B	-	31B	125B	41.8	14.4	61.6	66.8	83.7	50.0	57.3	24.8	37.5	48.7
Intern-Math	20B	-	31B	125B	65.4	30.0	75.7	79.3	94.0	50.9	38.5	53.1	71.9	62.1
DeepSeekMath	7B	-	120B	500B	64.1	34.2	74.0	83.9	92.4	63.4	62.4	56.4	84.4	68.4
Continual Pretraining on Mistral-7B														
Mistral-CT	7B	OWM	14B	15B	42.9	22.2	68.6	71.0	86.1	45.1	47.7	52.6	65.6	55.8
RHO-1-Math	7B	OWM	14B	10.5B	66.9	31.0	77.8	79.0	93.9	49.9	58.7	54.6	84.4	66.2
Δ				-30%	+24.0	+8.8	+9.2	+8.0	+7.8	+4.8	+11.0	+2.0	+18.8	**+10.4**

Experiment

Table 2: **Tool-integrated reasoning results of math pretraining.**

Model	Size	Tools	SFT Data	GSM8k	MATH	SVAMP	ASDiv	MAWPS	TAB	GSM-H	AVG
Used for SFT?				✓	✓	✗	✗	✗	✗	✗	
Previous Models											
GPT4-0314	-	✗	-	92.0	42.5	93.1	91.3	97.6	67.1	64.7	78.3
GPT4-0314 (PAL)	-	✓	-	94.2	51.8	94.8	92.6	97.7	95.9	77.6	86.4
MAmmoTH	70B	✓	MI-260k	76.9	41.8	82.4	-	-	-	-	-
ToRA	7B	✓	ToRA-69k	68.8	40.1	68.2	73.9	88.8	42.4	54.6	62.4
ToRA	70B	✓	ToRA-69k	84.3	49.7	82.7	86.8	93.8	74.0	67.2	76.9
DeepSeekMath	7B	✓	ToRA-69k	79.8	52.0	80.1	87.1	93.8	85.8	63.1	77.4
Our Pretrained Models											
TinyLlama-CT	1B	✓	ToRA-69k	51.4	38.4	53.4	66.7	81.7	20.5	42.8	50.7
RHO-1-Math	1B	✓	ToRA-69k	59.4	40.6	60.7	74.2	88.6	26.7	48.1	56.9
Δ				+8.0	+2.2	+7.3	+7.5	+6.9	+6.2	+5.3	**+6.2**
Mistral-CT	7B	✓	ToRA-69k	77.5	48.4	76.9	83.8	93.4	67.5	60.4	72.6
RHO-1-Math	7B	✓	ToRA-69k	81.3	51.8	80.8	85.5	94.5	70.1	63.1	75.3
Δ				+3.8	+3.4	+3.9	+1.7	+1.1	+2.6	+2.7	**+2.7**

Table 3: **Self-Reference results.** We use OpenWebMath (OWM) to train the reference model.

Model	Score Function	Data	Uniq. Toks	Train Toks	GSM8K	MATH	SVAMP	ASDiv	MAWPS	MQA	AVG
Tinyllama-CT (RM)	-	OWM	14B	15B	6.3	2.6	21.7	36.7	47.7	13.9	21.5
Tinyllama-SLM	$\mathcal{L}_{\text{RM}}$	OWM	14B	10.5B	6.7	4.6	23.3	40.0	54.5	14.3	23.9
Tinyllama-SLM	$\mathcal{H}_{\text{RM}}$	OWM	14B	10.5B	7.0	4.8	23.0	39.3	50.5	13.5	23.0
Tinyllama-SLM	$\mathcal{L}_{\text{RM}} \cap \mathcal{H}_{\text{RM}}$	OWM	14B	9B	7.1	5.0	23.5	41.2	53.8	18.0	24.8
Tinyllama-CT	-	PPile	55B	52B	8.0	6.6	23.8	41.0	54.7	14.2	24.7
Tinyllama-SLM	$\mathcal{L}_{\text{RM}} \cap \mathcal{H}_{\text{RM}}$	PPile	55B	36B	8.6	8.4	24.4	43.6	57.9	16.1	26.5

Experiment

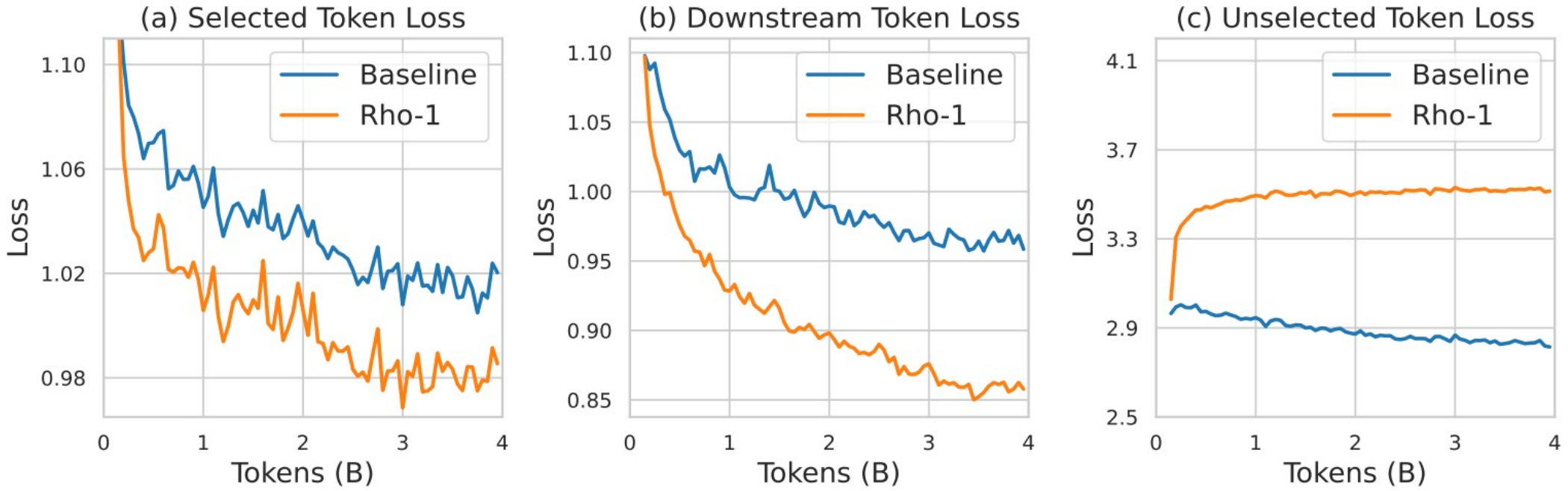


Figure 6: **The dynamics of pretraining loss and downstream loss.** (a) and (c) represent the loss of tokens selected/unselected by SLM during pretraining in both SLM and CLM methods, while (b) represents the loss of the SLM and CLM methods on MetaMath [Yu et al., 2024]. We tested the above results through the process of pretraining with a total of 4 billion tokens.

Experiment

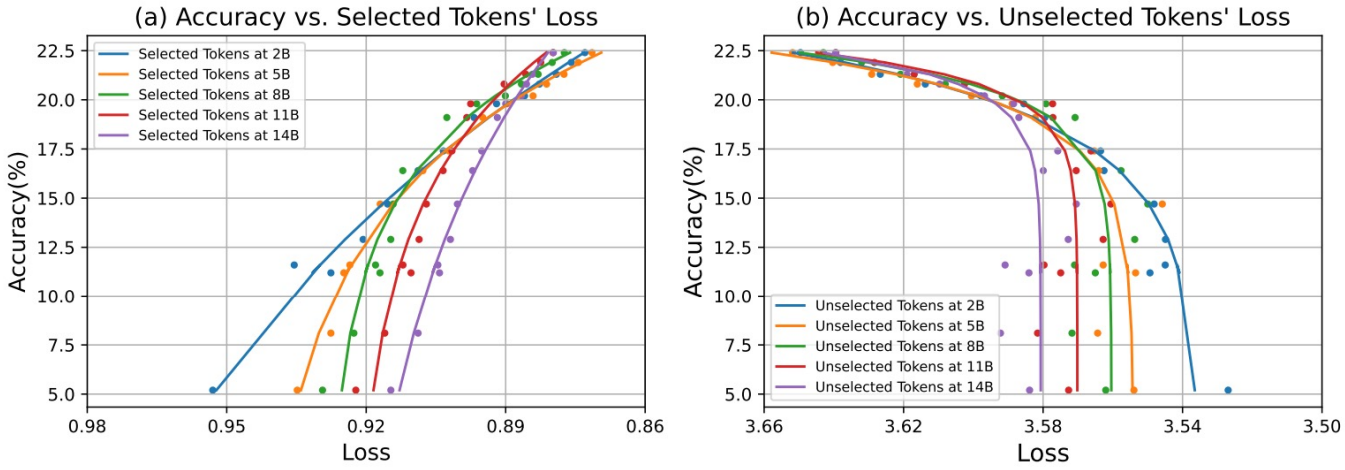


Figure 7: **The relationship between the selected tokens / unselected tokens loss in SLM and downstream task performance.** The y-axis represents the average few-shot accuracy on GSM8k and MATH. The x-axis represents the average loss on selected tokens / unselected tokens at corresponding checkpoint (2B, 5B, 8B, 11B, and 14B).

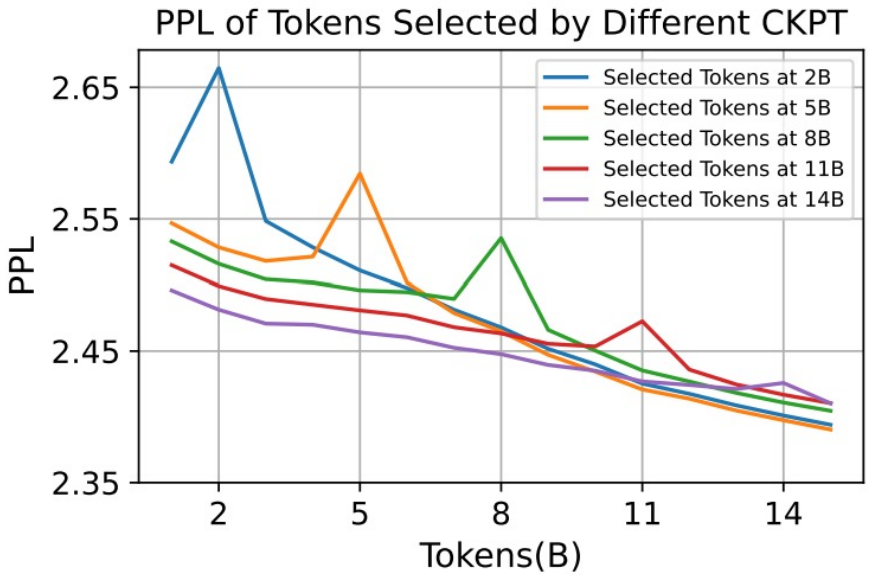


Figure 8: **The PPL of tokens selected by different checkpoint.** We test the PPL of the tokens selected at 2B, 5B, 8B, 11B, and 14B.

Training Large Language Models to Reason in a Continuous Latent Space

Shibo Hao[1,2,*], **Sainbayar Sukhbaatar**[1], **DiJia Su**[1], **Xian Li**[1], **Zhiting Hu**[2], **Jason Weston**[1], **Yuandong Tian**[1]

[1]FAIR at Meta, [2]UC San Diego

*Work done at Meta

CoT

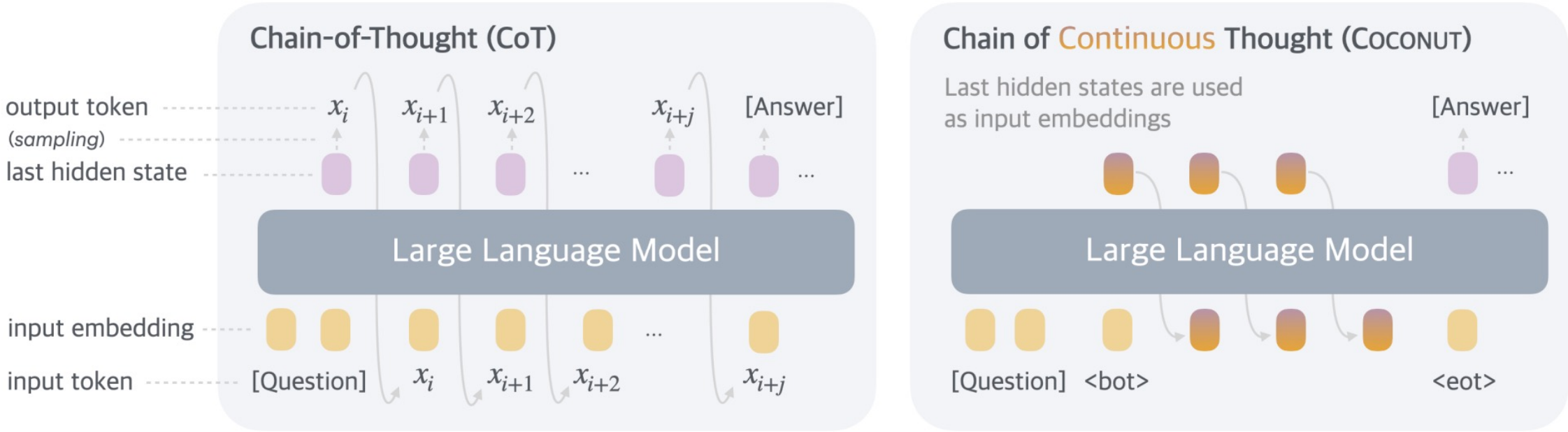


Figure 1 A comparison of Chain of Continuous Thought (COCONUT) with Chain-of-Thought (CoT). In CoT, the model generates the reasoning process as a word token sequence (e.g., $[x_i, x_{i+1}, ..., x_{i+j}]$ in the figure). COCONUT regards the last hidden state as a representation of the reasoning state (termed "continuous thought"), and directly uses it as the next input embedding. This allows the LLM to reason in an unrestricted latent space instead of a language space.

Training & Inference

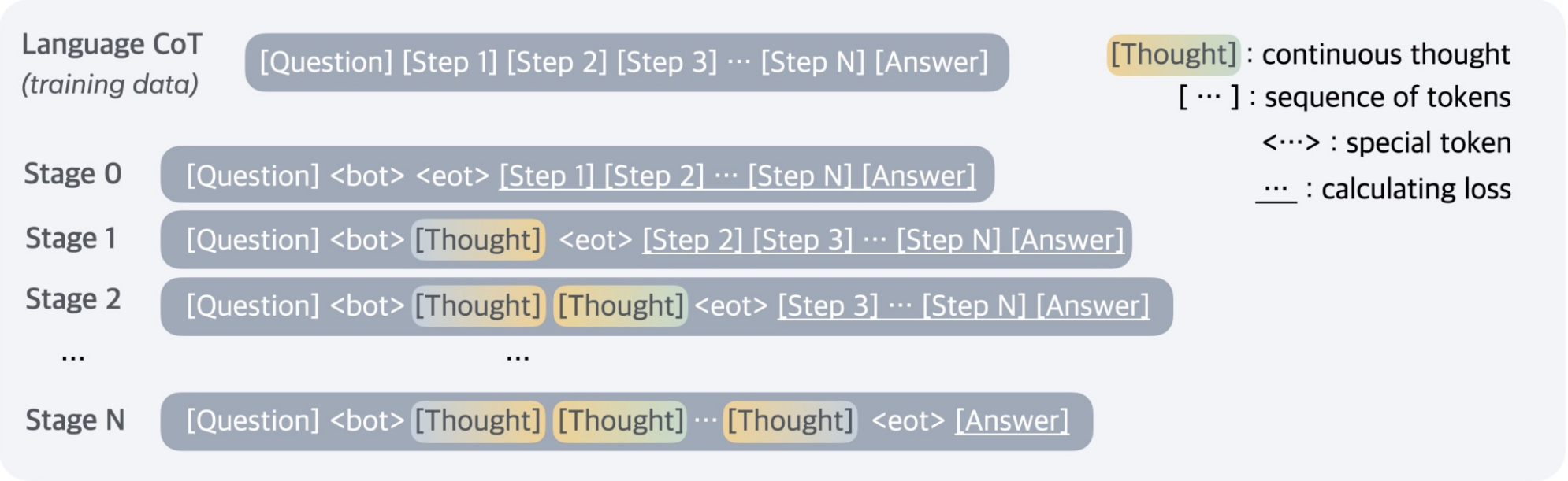


Figure 2 Training procedure of Chain of Continuous Thought (COCONUT). Given training data with language reasoning steps, at each training stage we integrate c additional continuous thoughts ($c = 1$ in this example), and remove one language reasoning step. The cross-entropy loss is then used on the remaining tokens after continuous thoughts.

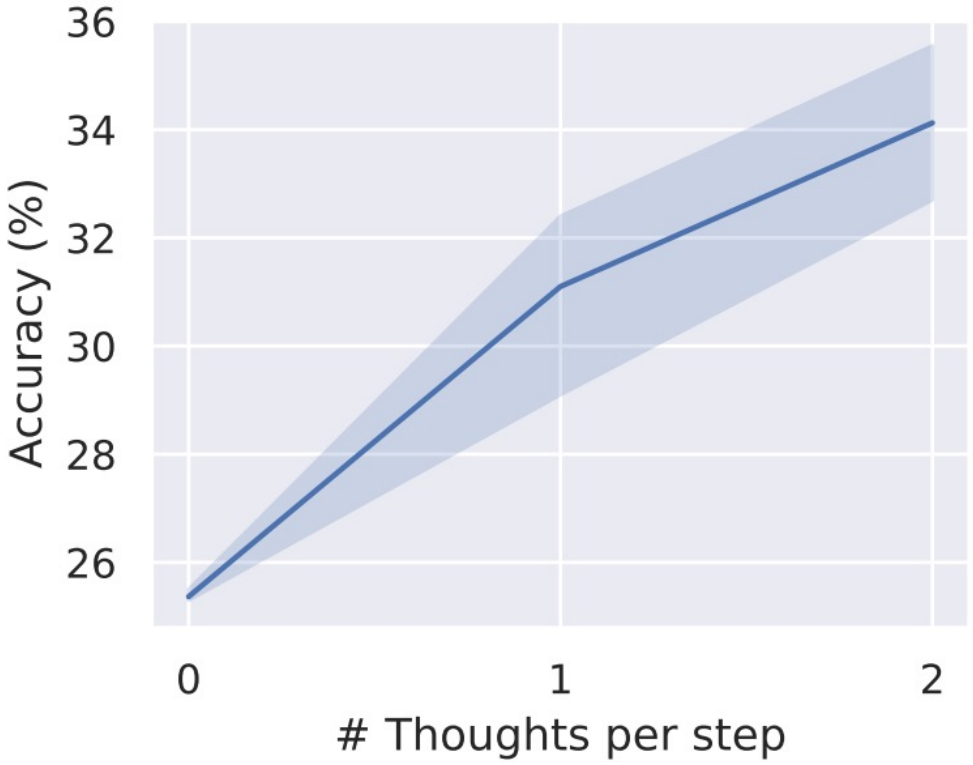
potential strategies: a) train a binary classifier on latent thoughts to enable the model to autonomously decide when to terminate the latent reasoning, or b) always pad the latent thoughts to a constant length. We found that both approaches work comparably well. Therefore, we use the second option in our experiment for simplicity, unless specified otherwise.

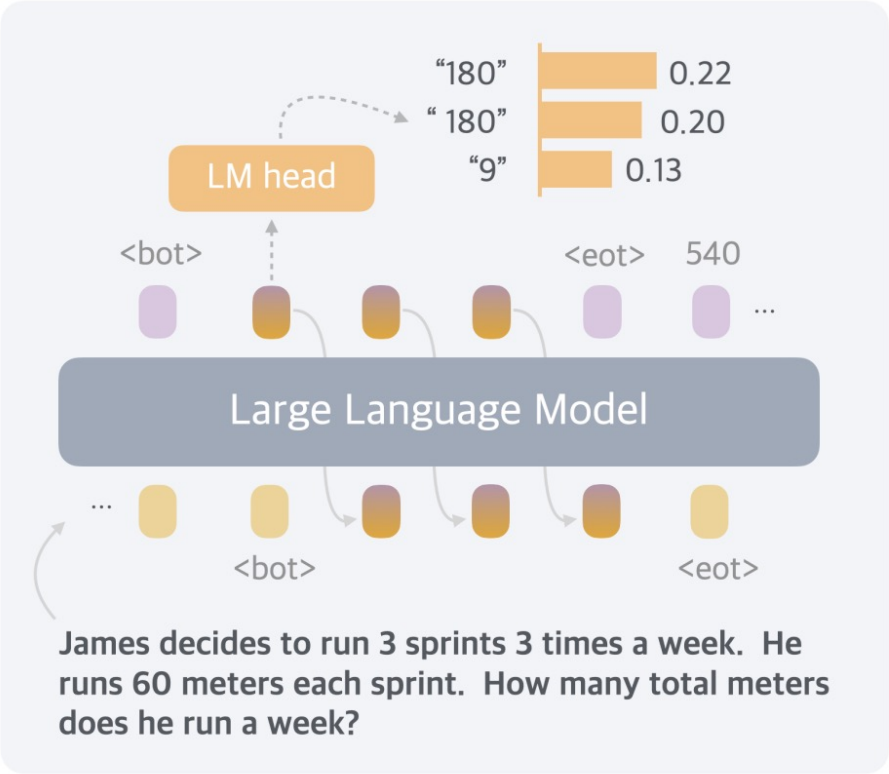
Experiments

Method	GSM8k		ProntoQA		ProsQA	
	Acc. (%)	# Tokens	Acc. (%)	# Tokens	Acc. (%)	# Tokens
CoT	42.9 ±0.2	25.0	98.8 ±0.8	92.5	77.5 ±1.9	49.4
No-CoT	16.5 ±0.5	2.2	93.8 ±0.7	3.0	76.7 ±1.0	8.2
iCoT	30.0*	2.2	99.8 ±0.3	3.0	98.2 ±0.3	8.2
Pause Token	16.4 ±1.8	2.2	77.7 ±21.0	3.0	75.9 ±0.7	8.2
COCONUT (Ours)	34.1 ±1.5	8.2	99.8 ±0.2	9.0	97.0 ±0.3	14.2
- *w/o curriculum*	14.4 ±0.8	8.2	52.4 ±0.4	9.0	76.1 ±0.2	14.2
- *w/o thought*	21.6 ±0.5	2.3	99.9 ±0.1	3.0	95.5 ±1.1	8.2
- *pause as thought*	24.1 ±0.7	2.2	100.0 ±0.1	3.0	96.6 ±0.8	8.2

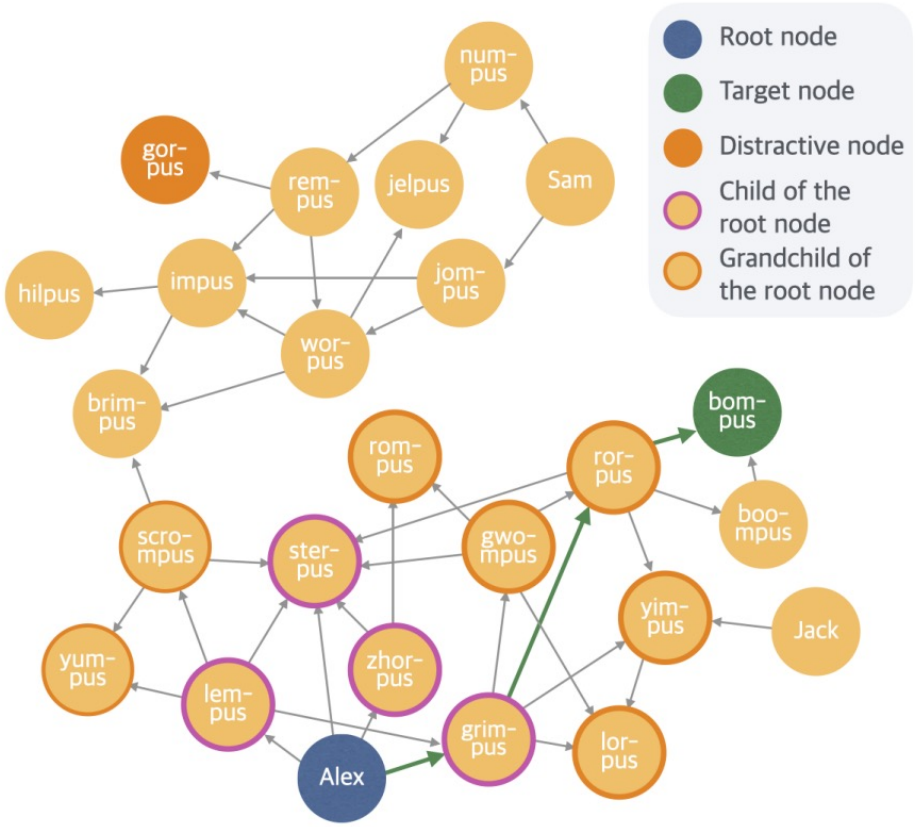
Table 1 Results on three datasets: GSM8l, ProntoQA and ProsQA. Higher accuracy indicates stronger reasoning ability, while generating fewer tokens indicates better efficiency. *The result is from Deng et al. (2024).

Experiments





Understanding



Question:

Every grimpus is a yimpus. Every worpus is a jelpus. Every zhorpus is a sterpus. Alex is a grimpus ⋯ Every lumps is a yumpus.
Question: **Is Alex a gorpus or bompus?**

Ground Truth Solution

Alex is a grimpus.
Every grimpus is a rorpus.
Every rorpus is a bompus.
Alex is a bompus

CoT

Alex is a lempus.
Every lempus is a scrompus.
Every scrompus is a yumpus.
Every yumpus is a rempus.
Every rempus is a gorpus.
Alex is a gorpus ✗

(Hallucination)

COCONUT (k=1)

<bot> [Thought] <eot>
Every lempus is a scrompus.
Every scrompus is a brimpus.
Alex is a brimpus ✗

(Wrong Target)

COCONUT (k=2)

<bot> [Thought] [Thought] <eot>
Every rorpus is a bompus.
Alex is a bompus ✓

(Correct Path)

Figure 6 A case study of ProsQA. The model trained with *CoT* hallucinates an edge (*Every yumpus is a rempus*) after getting stuck in a dead end. COCONUT (k=1) outputs a path that ends with an irrelevant node. COCONUT (k=2) solves the problem correctly.

Understanding

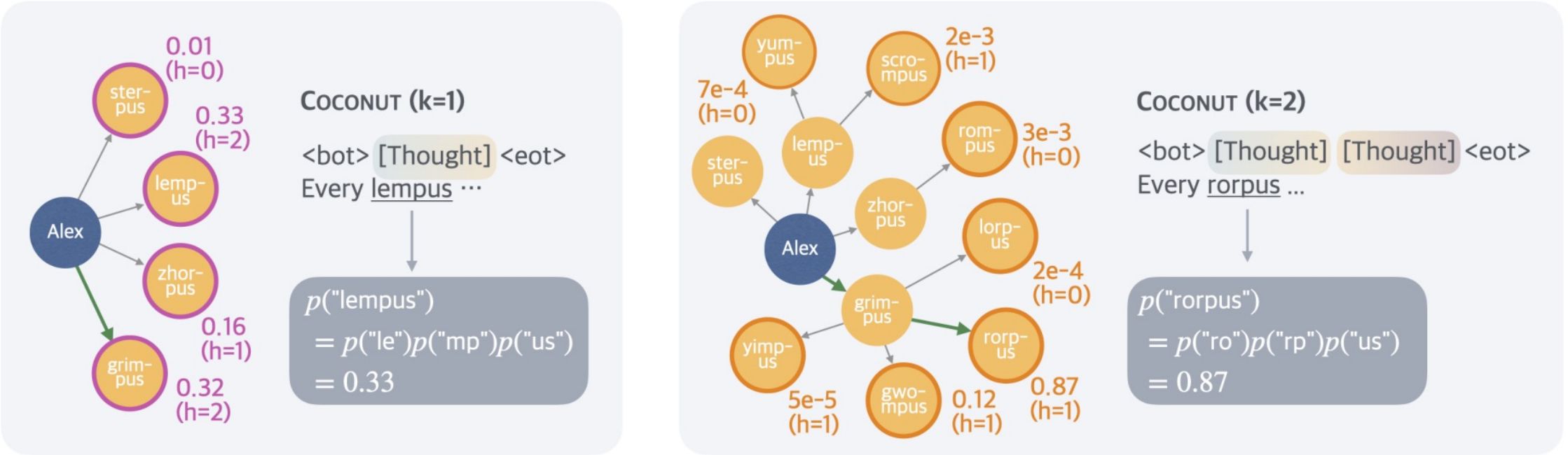


Figure 7 An illustration of the latent search trees. The example is the same test case as in Figure 6. The height of a node (denoted as h in the figure) is defined as the longest distance to any leaf nodes in the graph. We show the probability of the first concept predicted by the model following latent thoughts (e.g., "lempus" in the left figure). It is calculated as the multiplication of the probability of all tokens within the concept conditioned on previous context (omitted in the figure for brevity). This metric can be interpreted as an implicit value function estimated by the model, assessing the potential of each node leading to the correct answer.

Understanding

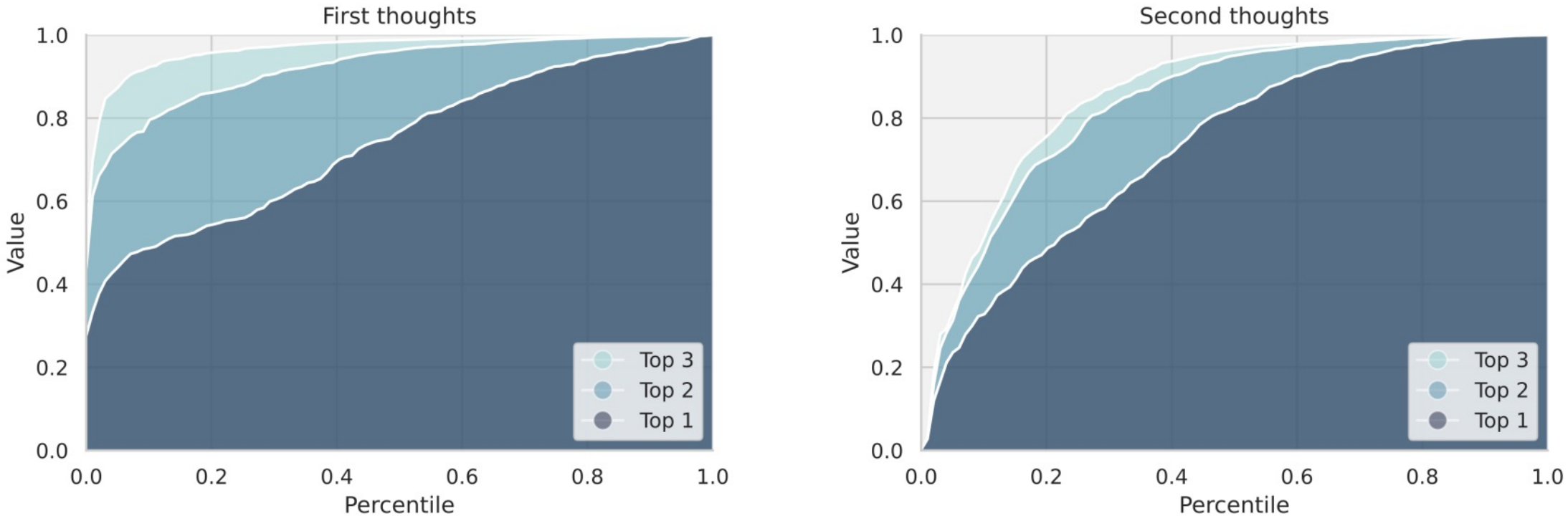


Figure 8 Analysis of parallelism in latent tree search. The left plot depicts the cumulative value of the top-1, top-2, and top-3 candidate nodes for the first thoughts, calculated across test cases and ranked by percentile. The significant gaps between the lines reflect the model's ability to explore alternative latent thoughts in parallel. The right plot shows the corresponding analysis for the second thoughts, where the gaps between lines are narrower, indicating reduced parallelism and increased certainty in reasoning as the search tree develops. This shift highlights the model's transition toward more focused exploration in later stages.

Thanks