Escape Sky-high Cost: Early-stopping Self-Consistency for Multi-step Reasoning

Yiwei Li[1*], Peiwen Yuan [1*], Shaoxiong Feng [2], Boyuan Pan [2], Xinglin Wang [1], Bin Sun [1], Heda Wang [2] and Kan Li [1]

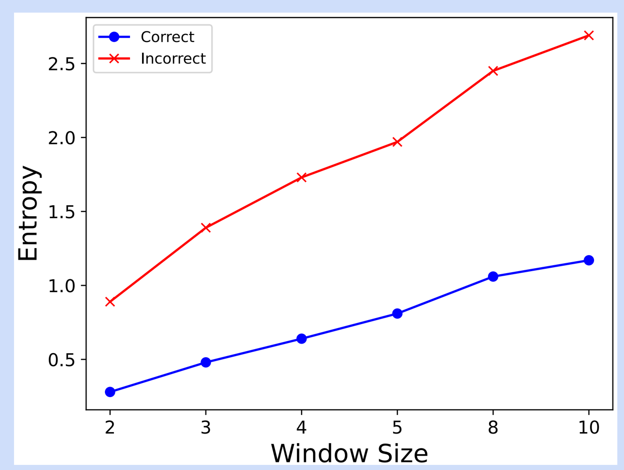
[1] School of Computer Science, Beijing Institute of Technology

[2] Xiaohongshu Inc

*Equal Contributions {liyiwei, peiwenyuan}@bit.edu.cn

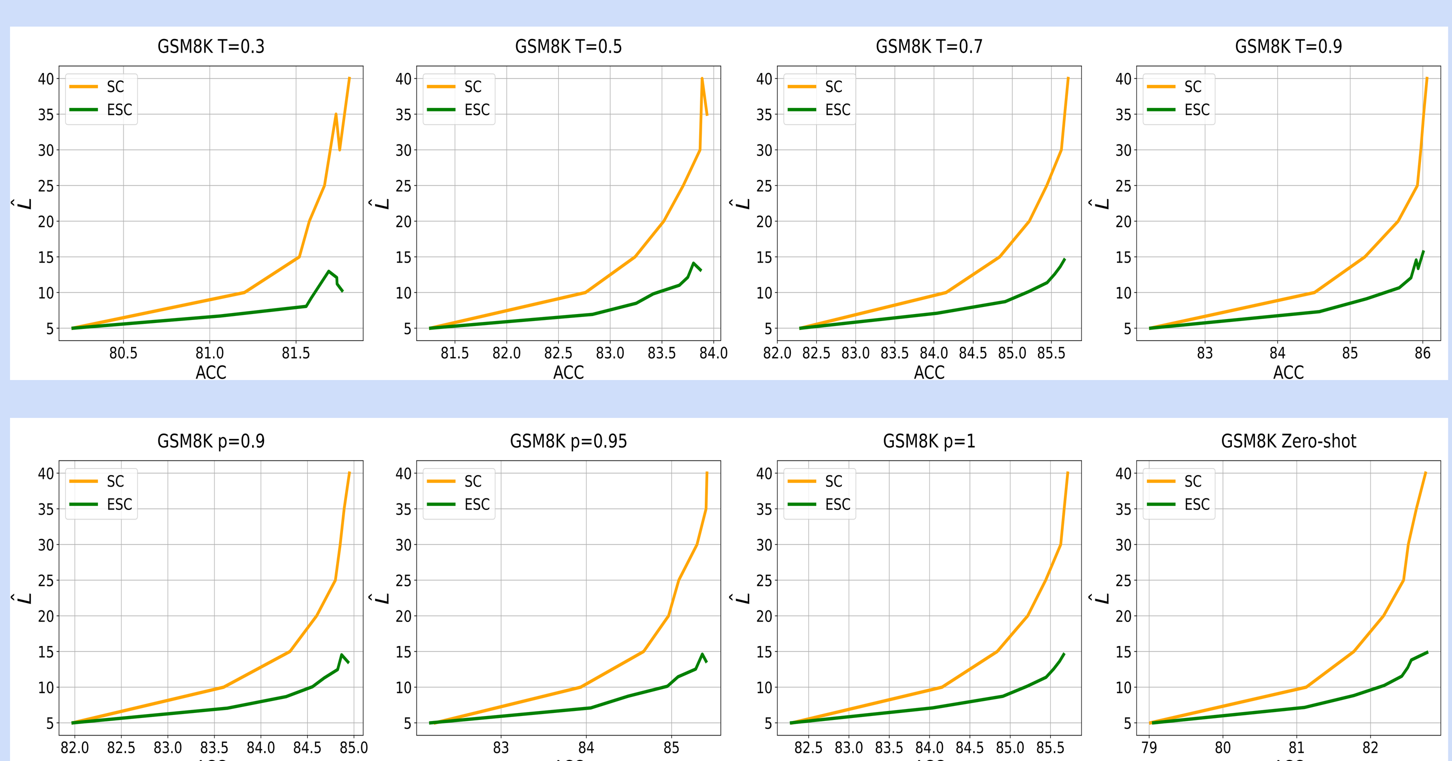
Motivation

- Self-consistency (SC) has been a widely used decoding strategy for chain-of-thought reasoning. Despite bringing significant performance improvements across a variety of multi-step reasoning tasks, it is a high-cost method that requires multiple sampling with the preset size. Taking MATH dataset as an example, evaluating the entire test set with SC (sampling size as 64) costs about **2000$** through GPT-4 API !



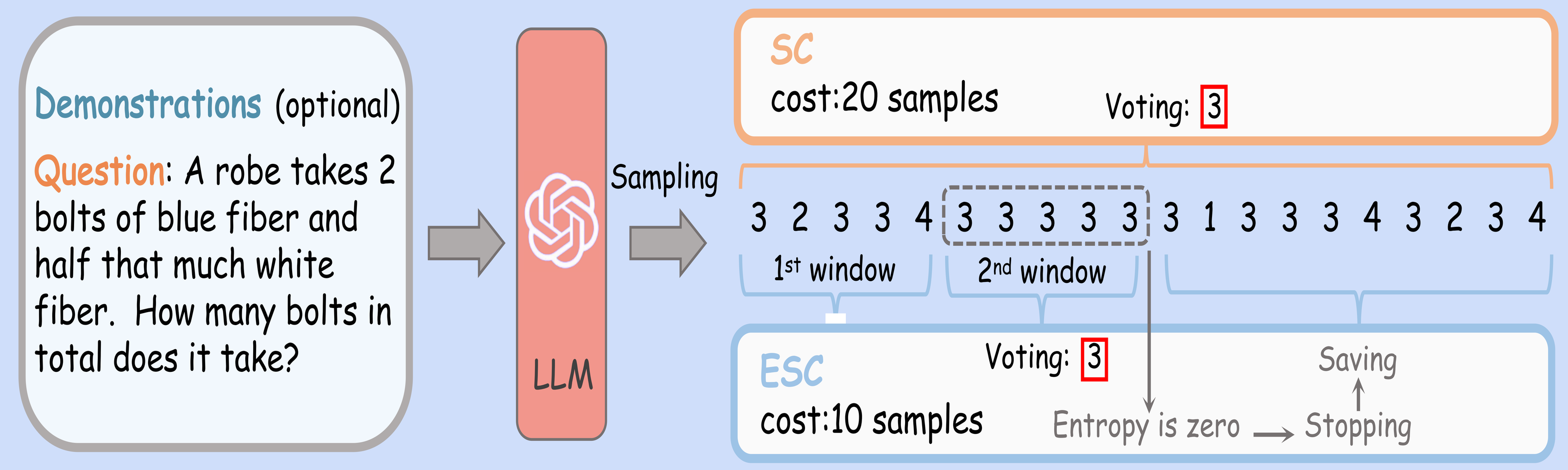
- We employ entropy as a representation of the answer distribution shape. Figure shows the mean entropy value of correct and incorrect voting answer within a window respectively, showing that distributions with correct one as highest probability answer typically have much lower entropy values. It can be a indicator to determine whether sampling should continue. Based on this, we propose early-stopping self-consistency (ESC), truncating the sampling process with low entropy window.

Robustness



- We show how ESC behaves for GSM8K as the decoding sampling temperature increases. Savings are consistent across different generation temperatures.
- ESC is robust to p values for top-p sampling.
- ESC can generalize to zero-shot manner.
- The accuracy of ESC and SC with different groups of demonstrations. We can see that ESC is robust to various demonstrations.

Early-stopping Self-Consistency (ESC)



Full process of ESC compared with original SC. We divide the large sample size into several sequential small windows. Stop sampling when answers within a window are all the same, i.e., the entropy score of predicted answer distribution is zero.

Control Scheme for ESC

$$\mathbb{E}(Q) \leq \mathbb{E}_{\hat{P} \in \mathcal{M}(\mathbb{D})} (1 - \mathrm{pow}(1 - \hat{P}_{stop}, L//w)) \times Q_w(\hat{P}) + \mathrm{pow}(1 - \hat{P}_{stop}, L//w) \times Q_o(\hat{P})$$

$$\mathbb{E}(\hat{L}) = \mathbb{E}_{\hat{P} \in \mathcal{M}(\mathbb{D})} \sum_{j=0}^{L//w-1} [(\hat{P}_{stop} \times \mathrm{pow}(1 - \hat{P}_{stop}, j) \times j \times w) + \mathrm{pow}(1 - \hat{P}_{stop}, L//w) \times L] + w_0$$

First, we sample w_0 times on the whole dataset. Based on the results of the first observation window, we calculate the expected sampling cost and performance under different settings of (w, L). Finally, considering the sampling budget and performance requirements, we choose appropriate values of (w, L) based on the respective expected values to execute ESC.

Analysis

Method	10	15	20	25	30
SC	61.96	61.96	62.04	62.15	62.18
ESC	61.96 (0.00)	61.96 (0.00)	62.02 (-0.02)	62.11 (-0.04)	62.15 (-0.03)
$\hat{L}$	5.62 (-4.38)	6.02 (-8.98)	6.32 (-13.68)	6.57 (-18.43)	6.79 (-23.21)

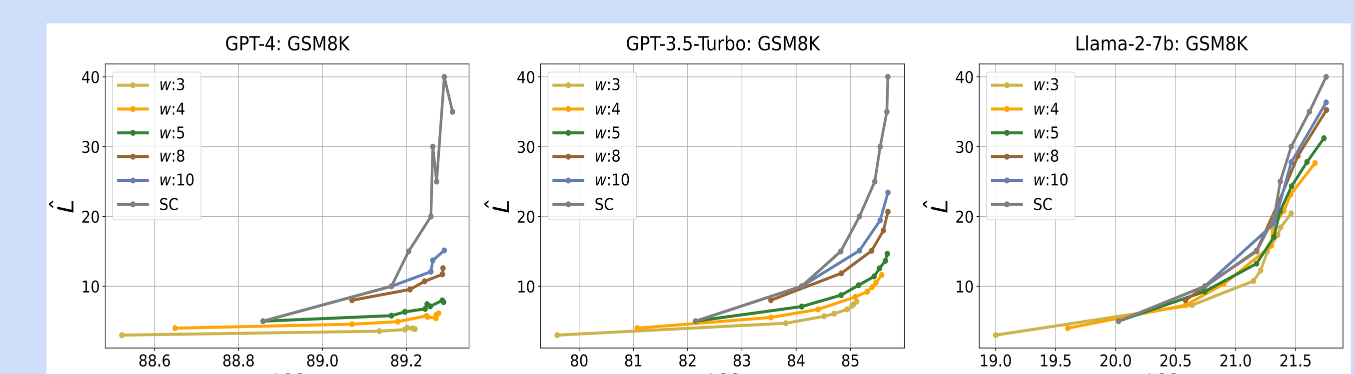
Max sampling size	10	20	30	40
PHP w. SC	86.32	86.64	86.76	87.00
PHP w. ESC	86.32 (0.00)	86.62 (-0.02)	86.77 (+0.01)	86.98 (-0.02)
$\hat{L}$	6.15 (-3.85)	7.83 (-12.17)	9.15 (-20.85)	10.26 (-29.74)
$\hat{L}$-PHP w. SC	86.02 (-0.30)	86.29 (-0.35)	86.32 (-0.44)	86.35 (-0.65)

ESC is suitable for PHP and open-ended generation tasks.

Results

		MATH	GSM8K	CSQA	SQA	Letter	Coinflip
GPT-4	CoT	50.44	87.70	83.71	78.63	93.12	100.00
	SC	60.32	89.29	87.18	81.67	95.00	/
	ESC	60.32 (0.00)	89.29 (0.00)	87.18 (0.00)	81.70 (+0.03)	94.98 (-0.02)	/
	$\hat{L}$	42.40 (-21.60)	7.98 (-32.02)	9.29 (-30.71)	7.19 (-31.39)	6.32 (-33.68)	/
	$\hat{L}$-SC	59.98 (-0.34)	89.07 (-0.22)	86.49 (-0.69)	81.40 (-0.27)	94.59 (-0.39)	/
GPT-3.5 Turbo	CoT	35.53	75.83	74.17	67.66	80.50	83.74
	SC	49.97	85.69	78.10	75.90	83.21	99.54
	ESC	49.96 (-0.01)	85.67 (-0.02)	78.10 (0.00)	75.71 (-0.19)	83.15 (-0.06)	99.49 (-0.05)
	$\hat{L}$	52.37 (-11.63)	14.65 (-25.35)	11.70 (-28.30)	8.51 (-27.93)	8.82 (-31.18)	13.03 (-26.97)
	$\hat{L}$-SC	49.79 (-0.13)	84.82 (-0.85)	77.67 (-0.43)	75.07 (-0.83)	82.74 (-0.41)	98.67 (-0.82)
Llama-2 7B	CoT	5.09	18.07	65.28	46.23	14.87	54.74
	SC	7.68	21.75	67.70	63.15	23.32	59.13
	ESC	7.68 (0.00)	21.74 (-0.01)	67.68 (-0.02)	63.01 (-0.14)	23.32 (0.00)	58.99 (-0.14)
	$\hat{L}$	62.48 (-1.52)	31.21 (-8.79)	11.82 (-28.18)	11.00 (-23.96)	34.73 (-5.27)	14.87 (-25.13)
	$\hat{L}$-SC	7.68 (0.00)	21.52 (-0.22)	66.97 (-0.71)	61.19 (-1.96)	23.11 (-0.21)	58.11 (-0.88)

Model	Method	16	24	32	40	48	64
GPT-4	SC	58.92	59.40	59.77	59.95	60.07	60.31
	ESC	58.92 (0.00)	59.40 (0.00)	59.77 (0.00)	59.95 (0.00)	60.07 (0.00)	60.31 (0.00)
	$\hat{L}$	13.56 (-2.44)	18.72 (-5.28)	23.67(-8.33)	28.49 (-11.51)	33.21 (-14.79)	42.41 (-21.59)
GPT-3.5 Turbo	SC	47.34	48.48	49.02	49.40	49.65	49.96
	ESC	47.33 (-0.01)	48.49 (+0.01)	49.02 (0.00)	49.41 (+0.01)	49.64 (-0.01)	49.96 (0.00)
	$\hat{L}$	14.84 (-1.16)	21.38 (-2.62)	27.76 (-4.24)	34.02 (-5.98)	40.20 (-7.80)	52.37 (-11.63)
Llama-2 7B	SC	7.10	7.28	7.40	7.45	7.54	7.70
	ESC	7.10 (0.00)	7.28 (0.00)	7.40 (0.00)	7.45 (0.00)	7.54 (0.00)	7.70 (0.00)
	$\hat{L}$	15.88 (-0.12)	23.72 (-0.28)	31.52 (-0.48)	39.29 (-0.71)	47.04 (-0.96)	62.48 (-1.52)



There findings:

- ESC significantly reduces costs while barely affecting performance.
- ESC is a scalable decoding process across sampling and window size.
- Cost savings are positively correlated with performance.

Conclusion

- We introduced a simple yet effective sampling process called early-stopping self-consistency (ESC). By stopping the decoding process with high confident window, ESC greatly reduce the cost of SC **without sacrificing performance**.
- A control scheme for ESC is further derived to dynamically select the performance-cost balance for different tasks and models, which **requires no extra prior knowledge of model capabilities and task difficulty**.
- The empirical results show that ESC reduces the actual number of samples of chain-of-thought reasoning **by a significant margin** on six popular benchmarks, while attaining comparable performances. We also show control scheme for ESC can predict the performance-cost trade-off accurately across various tasks and models. The additional evaluations indicate that ESC can robustly save cost considering different decoding settings and prompts, and even on open-ended generation tasks.

References

- Wei J, Wang X, Schuurmans D, et al. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 2022, 35: 24824-24837.
- Wang X, Wei J, Schuurmans D, et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. The Eleventh International Conference on Learning Representations. 2023.
- Aggarwal P, Madaan A, Yang Y. Let's Sample Step by Step: Adaptive-Consistency for Efficient Reasoning and Coding with LLMs. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023: 12375-12396.

Contact

Email: liyiwei@bit.edu.cn

Tel: (+86) 18810516898

Wechat: 18810516898