BOUNDS OF CHAIN-OF-THOUGHT ROBUSTNESS: REASONING STEPS, EMBED NORMS, AND BEYOND

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing research indicates that the output of Chain-of-Thought (CoT) is significantly affected by input perturbations. Although many methods aim to mitigate such impact by optimizing prompts, a theoretical explanation of how these perturbations influence CoT outputs remains an open area of research. This gap limits our in-depth understanding of how input perturbations propagate during the reasoning process and hinders further improvements in prompt optimization methods. Therefore, in this paper, we theoretically analyze the effect of input perturbations on the fluctuation of CoT outputs. We first derive an upper bound for input perturbations under the condition that the output fluctuation is within an acceptable range, based on which we prove that: *(i)* This upper bound is positively correlated with the number of reasoning steps in the CoT; *(ii)* Even an infinitely long reasoning process cannot eliminate the impact of input perturbations. We then apply these conclusions to the Linear Self-Attention (LSA) model, which can be viewed as a simplified version of the Transformer. For the LSA model, we prove that the upper bound for input perturbation is negatively correlated with the norms of the input embedding and hidden state vectors. To validate this theoretical analysis, we conduct experiments on three mainstream datasets and four mainstream models. The experimental results align with our theoretical analysis, empirically demonstrating the correctness of our findings[1].

1 INTRODUCTION

Chain-of-Thought (CoT) is an effective method that enhances the performance of large language models (LLMs) by prompting the model to generate a step-by-step reasoning process, thereby improving the quality of the results (Wei et al., 2022). However, numerous studies have indicated that CoT is highly sensitive to input, where subtle input perturbations can lead to significant performance fluctuations (Zhao et al., 2024; Shi et al., 2024b). To address this issue, researchers have proposed prompt optimization methods to enhance the reasoning performance of LLMs by refining the input prompt, lowering the effect of the input perturbation (Vatsal & Dubey, 2024; Sahoo et al., 2025). For instance, TextGrad (Yuksekgonul et al., 2025) optimizes prompts by constructing textual gradients, while OPRO (Yang et al., 2024) utilizes the LLM itself to iteratively generate more suitable prompts.

Despite this progress, a key gap remains: most studies treat CoT robustness as an empirical phenomenon, with little theoretical understanding of *why* and *how* perturbations propagate through the reasoning process of LLMs, thereby affecting output fluctuation. Without such analysis, our understanding of CoT robustness remains incomplete, and prompt optimization risks being limited to ad-hoc techniques. This motivates a fundamental research question: **what governs the CoT robustness of LLMs to input perturbations?**

Following the previous work (Huang et al., 2025), we consider CoT as a multistep iterative process, with the output of each step serving as the input for the next. Our theoretical analysis shows that under the assumption of Lipschitz continuity (Qi et al., 2023; Collins et al., 2025), longer CoT reasoning indeed reduces the fluctuation of outputs to input perturbations, but it never fully eliminates them. Even with an infinite number of CoT steps, a non-zero robustness bound remains, suggesting that CoT inherently dampens but cannot completely neutralize perturbations.

[1] Our code is released in https://anonymous.4open.science/r/CoT-Robust-DF71

Table 1: The main findings and corresponding evidence and experiment of this paper.

Finding	Evidence	Experiment
More reasoning steps can reduce the effect of input perturbations.	Theorem 1	§4.3
The effect of input perturbations cannot be entirely eliminated by continuously increasing the number of CoT reasoning steps.	Equation 4	§4.3
CoT robustness is negatively correlated with the norms of the input embedding and hidden state vectors.	Theorem 2	§4.4

To further ground our analysis, we investigate robustness in the Linear Self-Attention (LSA) model (Wang et al., 2020a; Zhang et al., 2024a), which is commonly adopted as a simplified version of Transformer (Vaswani et al., 2017) for analysis without loss of generality. We prove that CoT robustness highly depends on model-level factors: the sensitivity to perturbations correlates negatively with the norm of the input vector and the hidden state vectors. Additionally, we discuss the impact of other factors in LSA on CoT robustness.

Finally, we validate our theory with experiments on four mainstream LLMs (Llama2, Llama3.1, Deepseek-R1-Distilled-Llama3.1, Qwen3) across three widely used reasoning datasets (MATH, MMLU-Pro, GPQA). The experimental results indicate that the variation in output fluctuation is consistent with the trends of the various factors identified in our theoretical analysis, thereby validating our findings. Furthermore, based on the analysis, we propose selecting the prompt by maximizing the upper bound of the input perturbation, which achieves consistent improvements over prior work, aiming to inspire future research in this area.

The main findings of our work are summarized in Table 1, and our main contributions are as follows:

- We provide an upper bound for the output fluctuation with respect to input perturbations under the assumption of Lipschitz continuity and prove that even an infinitely long CoT cannot completely counteract the impact of input perturbations.
- Taking the LSA model as a case study, we demonstrate that robustness to input perturbations is negatively correlated with the norms of the input and hidden state embedding vectors.
- Our experiments across multiple mainstream datasets and LLMs validate our theoretical analysis, and improvements based on our analysis also enhance performance compared to existing prompt optimization methods.

2 Robustness of Chain-of-Thought

In this section, we discuss the impact of input perturbations on the model output. We begin by providing some fundamental definitions. Then, we derive the upper bound for the output fluctuation given the input perturbation when the model satisfies Lipschitz continuity. Afterward, we determine the upper bound for the input perturbation when the output fluctuation is within an acceptable range. All the proof of this section is shown in Appendix C.

2.1 Preliminary

Let $x, y \in \mathbb{R}^d$ denote the embedding vectors of the user query and the corresponding output, where d is the dimension of the embedding space. Following previous works (Zhang et al., 2024a; Von Oswald et al., 2023), we directly use embedding vector instead of token embedding for analysis, as we consider the impact of user query as a whole on CoT robustness. We also prove that the conclusion is same with considering multiple tokens in Appendix D.5. Let $\delta \in \mathbb{R}^d$ represent the input perturbation, and $\tilde{x} = x + \delta$ represent the perturbed input. Following previous work (Huang et al., 2025), we model the CoT reasoning process as a multistep iterative procedure, where the output of each step serves as the input for the next step. Let $K \in \mathbb{N}^+$ be the total number of CoT reasoning steps, and let $h_{k,x} \in \mathbb{R}^d$ denote the hidden state at step k taking x as input[2]. Let $f(h, x) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$

[2]In practical models, x can be viewed as the input embedding vector, and h can be viewed as the encoded vector from the last layer of the model.

represent the mapping function to generate the hidden state corresponding to an arbitrary reasoning model. Thus, we have $h_{1,x} = f(0, x)$ and $h_{k,x} = f(h_{k-1,x}, x)$. We denote the output fluctuation caused by the perturbation δ on step k as $\varepsilon_k = h_{k,\tilde{x}} - h_{k,x}$.

2.2 Upper Bound of Output Fluctuation

We primarily discuss the impact of input perturbations on the model output under the assumption of Lipschitz continuity. Lipschitz continuity imposes a constraint on the growth rate of the model output, preventing an explosive increase. Considering that the output of current LLMs typically exhibits stable changes, many analytical works adopt this condition as a fundamental assumption (Qi et al., 2023; Collins et al., 2025). Specifically, for a given bivariate function $f(h, x)$, Lipschitz continuity requires the existence of constants $C, \gamma \in \mathbb{R}$ such that:

$$\|f(h_1, x_1) - f(h_2, x_2)\| \leq \gamma \|h_1 - h_2\| + C\|x_1 - x_2\| \tag{1}$$

Considering that the input h at each step is the output of the previous step, by substituting it and expanding recursively, we can derive the following theorem:

Theorem 1. *If f is Lipschitz continuous with respect to constants $C, \gamma \in \mathbb{R}$ as defined in Equation 1, then for a given input perturbation $\delta \in \mathbb{R}^d$, the upper bound of the corresponding output fluctuation ε_K of the final step K satisfies that:*

$$\|\varepsilon_K\| \leq \left(A\gamma^K + \frac{C}{1-\gamma}(1 - \gamma^K) \right) \|\delta\|$$

where $A = \max \frac{\|\varepsilon_1\|}{\|\delta\|}$.

From Theorem 1, we can observe that the propagation of the input perturbation can be mainly divided into two parts. *(i)* The part contained in the hidden state vector: since the hidden state vector is updated at each step, the coefficient of this part of the perturbation is continuously multiplied by the corresponding Lipschitz constant γ; *(ii)* The part contained in the input vector: since the input vector at each step does not change, this part of the perturbation gradually accumulates at each step, and thus the corresponding perturbation coefficient is $\sum_{k=1}^{K} C\gamma^k = \frac{C}{1-\gamma}(1 - \gamma^K)$.

Besides, considering that when the model is fixed, the corresponding parameters C and γ are also fixed. Therefore, based on Theorem 1, the upper bound of the output fluctuation primarily depends on two factors: the number of reasoning steps K and the magnitude of the perturbation $\|\delta\|$. Based on previous work (Zhou et al., 2020; Diehl Martinez et al., 2024), we assume that $\gamma < 1$, which implies that for a well-trained model, the output fluctuation gradually converges to a fixed value rather than diverging infinitely. We also fit the values of γ in Appendix F.1 using practical datasets and models. Consequently, as the number of reasoning steps K increases, the corresponding upper bound of output fluctuation decreases, indicating that the increment of CoT steps can mitigate the impact of input perturbations on the model output.

2.3 Upper Bound of Input Perturbation

In practical applications, a model can tolerate a certain degree of output fluctuation while maintaining the same final result. For example, in a classification task, as long as the probability of the same option remains the highest before and after the input perturbation, a certain level of fluctuation in the output probabilities can not affect the final answer. Therefore, we assume there exists an acceptable boundary $R \in \mathbb{R}^+$, such that we consider the output fluctuation to be acceptable when the following condition is met:

$$\|\varepsilon\| \leq R \tag{2}$$

To ensure that the norm of output fluctuation ε is less than R, we require the expression on the right-hand side of the inequality in Theorem 1 to be less than R, which yields:

$$\|\delta\| \leq \frac{R}{A\gamma^K + \frac{C}{1-\gamma}(1 - \gamma^K)} \tag{3}$$

It can be observed that the upper bound of the input perturbation is mainly influenced by R, C, and γ. A larger R indicates that a greater output fluctuation is acceptable, thus leading to a larger upper

bound for the input perturbation. Conversely, larger values of C and γ, according to Equation 1, suggest that the model is less capable of compressing the output fluctuation, implying a weaker ability to handle input perturbations, which results in a smaller upper bound for the input perturbation.

Taking $\gamma < 1$ and letting $K \to \infty$, we can obtain that:

$$\|\delta\| \leq \frac{R(1-\gamma)}{C} \tag{4}$$

Equation 4 indicates that the effect of extending the reasoning process to eliminate input perturbation is limited. Even with an infinitely long reasoning process, if the input perturbation exceeds a certain threshold, the model cannot eliminate the resulting output fluctuation. For example, if we "perturb" a numerical reasoning problem into a coding problem, the model cannot generate the answer to the original problem, regardless of the reasoning length.

3 Chain-of-Thought Robustness on Linear Self-Attention

According to the discussion in §2.3, the upper bound of input perturbation that a model can tolerate using CoT depends on the properties of the model itself. Therefore, in this section, we discuss the factors that influence the upper bound of input perturbation on the Linear Self-Attention model (LSA) (Wang et al., 2020a; Zhang et al., 2024a), which can be viewed as a simplified version of the current mainstream Transformer architecture. All the proofs of this section are shown in Appendix C. We also discuss the influence of various *non-linear factors* in the Transformer on the conclusions of this section in Appendix D.1 and we analyze none-linear attention in Appendix D.3.

3.1 Definition of Linear Self-Attention

We first define LSA following the previous work (Wang et al., 2020a; Zhang et al., 2024a). Let $W^{KQ}, W^{PV} \in \mathbb{R}^{b\times b}$ denote the combined query-key and projection-value matrices, and let $\rho \in \mathbb{R}^+$ be the normalization factor. We denote the parameters as $\theta = (W^{KQ}, W^{PV}, \rho)$. Let $E = [h, x]$. The LSA is then defined as:

$$f_{LSA}(h, x; \theta) = E + W^{PV} E \frac{E^\top W^{KQ} E}{\rho} \tag{5}$$

LSA can be viewed as replacing the non-linear softmax mapping in a single-layer Transformer with a linear mapping. Following prior work (Zhang et al., 2024a), we set $\rho = 1$ in this paper.

Based on Equation 5, Zhang et al. (2024a) proves that for a well-trained LSA on the training data $\{(h_i, x_i, y_i)_N\}$, its parameters θ must satisfy:

$$W^{KQ}_* = [\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{4}} \begin{pmatrix} \Gamma^{-1} & 0_d \\ 0_d^\top & 0 \end{pmatrix}, W^{PV}_* = [\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}} \begin{pmatrix} 0_{d\times d} & 0_d \\ 0_d^T & 1 \end{pmatrix} \tag{6}$$

where $\Gamma = \left(1 + \frac{1}{N}\right)\Lambda + \frac{1}{N}\mathrm{Tr}(\Lambda) I_d \in \mathbb{R}^{d\times d}$ and Λ denote the covariance matrix of the training data. Substituting these optimal parameters into the equation yields:

$$f_{LSA}(h, x; \theta_*) = E + \begin{pmatrix} 0_{d\times d} & 0_d \\ 0_d^T & 1 \end{pmatrix} E \frac{E^\top \begin{pmatrix} \Gamma^{-1} & 0_d \\ 0_d^\top & 0 \end{pmatrix} E}{\rho} \tag{7}$$

Considering the gradient explosion without the residual flow, we introduce the residual coefficient $\eta \in (0, 1)$ to LSA (Zhang et al., 2019; Bachlechner et al., 2020). The corresponding function is:

$$f_{LSA}(h, x; \theta_*) = \eta E + \begin{pmatrix} 0_{d\times d} & 0_d \\ 0_d^T & 1 \end{pmatrix} E \frac{E^\top \begin{pmatrix} \Gamma^{-1} & 0_d \\ 0_d^\top & 0 \end{pmatrix} E}{\rho} \tag{8}$$

Next, we use Equation 8 as the prediction function $f_{LSA}(h, x)$ to discuss the effect of input perturbations on the LSA output.

3.2 Input Robustness of Linear Self-Attention

Based on Equation 8, we provide the upper bounds for the two Lipschitz constants in Equation 1:

Lemma 1. *If $\|x\| \leq R_x$ and $\|h\| \leq R_h$, let $\alpha = (\mathrm{Tr}(\Gamma^{-2}))^{-\frac{1}{4}}$. Then we have:*

$$C \leq \eta + \alpha^{-1} \|\Gamma^{-1}\| R_h^2$$

$$\gamma \leq \sqrt{\eta^2 + 4 R_x^2 \alpha^{-2} \|\Gamma^{-1}\|^2 2^2 R_h^2}$$

The assumption of R_x and R_h in Lemma 1 bounds the norms of x and h. Considering that excessively large embedding vectors can lead to unstable inference, the embedding vector norms in mainstream LLMs are typically confined within a certain range (Fazlyab et al., 2019; Kim et al., 2021), making this assumption reasonable.

By substituting the upper bounds of C and γ from Lemma 1 into the Equation 3, we can obtain that:

Theorem 2. *If $\|x\| \leq R_x$ and $\|h\| \leq R_h$ and let*

$$\alpha = \left[\mathrm{Tr}(\Gamma^{-2})\right]^{\frac{1}{4}}, \qquad s = \|\Gamma^{-1}\|, \qquad \beta = \alpha^{-1} s R_h^2, \qquad \gamma = \sqrt{\eta^2 + 4 R_x^2 \alpha^{-2} s^2 R_h^2}\,.$$

With $A > 0$ such that $\|e_0\| \leq A\|\delta\|$, the certified tolerable input-perturbation radius of the LSA map for the reasoning step $K \in \mathbb{N}^+$ is:

$$\|\delta\| \leq \frac{(1-\gamma) R}{\big(\eta + \beta\big) \; + \; \big(A(1-\gamma)(1+\beta)\big) \gamma^K}$$

In particular, if $\gamma < 1$, as $K \to \infty$, we can derive that:

$$\|\delta\| \leq \frac{(1-\gamma) R}{\eta + \beta}$$

According to Theorem 2, the impact of input perturbations on model outputs primarily depends on:

- R: The range of output perturbation that is acceptable. *A larger range indicates a greater tolerance for perturbations*, leading to a higher upper bound for the input perturbation.
- R_x: The tolerable perturbation radius is negatively correlated with R_x, indicating that *a larger norm of the input lowers the model's robustness to input perturbations*. According to the proof, a larger R_x leads to a larger coefficient of the perturbation in the resulting bound.
- R_h: The tolerable perturbation radius is negatively correlated with R_h. This suggests that *a larger norm of the internal state makes the model more susceptible to being led astray* during the reasoning process, thus weakening its resistance to input perturbations.
- Γ: The covariance matrix of the training data. *More inconsistent training data leads to the model being more sensitive to input perturbations*.
- η: A larger residual coefficient indicates that the model retains more information from input, causing *the effects of input perturbations to be preserved across layers*.

The theoretical results suggest two main robustness levers at inference and training time. At inference, Theorem 1 shows that longer, more structured chains of thought reduce output fluctuations, while smaller norms of input embeddings and encoded representations decrease the effective Lipschitz constants in the bounds. At training time, Theorem 2 indicates that reducing representation norms and decreasing the training-data covariance (Γ) (i.e., making the data more consistent), increases the certified perturbation radius.

We further discuss the impact of vector norms on the CoT robustness in Appendix D.4. Considering that verifying the effects of Γ and η requires modifying the training data and the model architecture of LLMs, this work provides only a theoretical analysis of Γ and η to inspire future work on corresponding empirical studies, while focusing on verifying the effects of R, R_x, and R_h.

Table 2: Average exact match (EM) and output fluctuation (OF) of different models using various prompts on different datasets. The highest EM and lowest OF under each setting is marked in **bold**.

Model	MATH		MMLU-Pro		GPQA	
	EM	OF	EM	OF	EM	OF
Llama2-7b	14.2 ± 5.0	0.475	11.2 ± 5.7	0.622	17.5 ± 4.7	0.509
Llama3.1-8b	45.8 ± 7.2	0.366	41.0 ± 10.7	0.350	26.6 ± 5.7	0.467
Llama-R1-8b	64.8 ± 3.0	0.158	44.8 ± 8.3	0.292	28.5 ± 2.9	0.371
Qwen3-8b	$\mathbf{77.2 \pm 1.6}$	**0.097**	$\mathbf{46.9 \pm 5.2}$	**0.162**	$\mathbf{37.3 \pm 1.9}$	**0.214**

4 Experiment

4.1 Experiment Setup

Dataset Our experiments are conducted on three reasoning datasets: MATH (Hendrycks et al., 2021), MMLU-Pro (Wang et al., 2024c), and GPQA (Rein et al., 2024). Detailed descriptions of these three datasets are provided in Appendix E.1. Considering the high difficulty of these datasets, which require multistep reasoning processes for solutions, we suppose they can effectively reflect the influence of various factors on the model's ability to handle input perturbations. We also adapt experiments on more datasets in Appendix F.2.

Model We conduct experiments on four mainstream LLMs including: Llama2-7b (Touvron et al., 2023), Llama3.1-8b (Grattafiori et al., 2024), Deepseek-R1-Distilled-Llama3.1-8b (Llama-R1-8b) (DeepSeek-AI et al., 2025) and Qwen3-8b (Yang et al., 2025). These models cover a range of capabilities, allowing for a comprehensive evaluation of how model type and different factors affect the handling of input perturbations. For Llama2-7b and Llama3.1-8b, we employ the instruct version. For Qwen3-8b, we utilize its *Thinking Mode* to fully leverage its performance. We also experiment with the performance under different model scales in Appendix F.5.

Metric To evaluate both the performance and robustness, we adopt the following two metrics: ***(i) Exact Match (EM)***: Whether the predicted answer is the same as the correct answer to the question. A higher value for this metric indicates that the model is better at solving the given dataset, reflecting the overall performance in a specific setting. ***(ii) Output Fluctuation (OF)***: The normalized entropy (Friedrich, 2021) of the answers generated from different prompts for the question. A higher value for this metric indicates that the output on the given question is less consistent, reflecting the robustness of the specific setting. We detail how to calculate OF in Appendix E.4. We also evaluate with the other fluctuation metric in Appendix F.3.

Input Perturbations To fully reflect the robustness to input perturbations, for each model and dataset, we first generate multiple prompts. Then, for each question, we use these prompts to generate multiple answers. We evaluate the performance by analyzing the correctness and consistency of these answers. To ensure the reliability of our results, we collect all prompts during the optimization of three mainstream methods, including TextGrad (Yuksekgonul et al., 2025), OPRO (Yang et al., 2024), and CFPO (Liu et al., 2025). The base prompts used follow Grattafiori et al. (2024), which is shown in Appendix E.2. The number of prompts used for each dataset and model is detailed in Appendix E.3. We also adapt the experiments using the same prompts on different datasets and models in Appendix F.4.

More experimental setups are detailed in Appendix E.5.

4.2 Overall Evaluation

CoT Robustness Scales with Model Capability The average performance and corresponding fluctuations for different prompts across various datasets and models are shown in Table 2. Results show that across all models, as their capabilities increase, not only does the average EM improve, but the corresponding output fluctuation also decreases. Regarding different tasks, multiple-choice sets (MMLU-Pro, GPQA) exhibit larger fluctuation than MATH, where small logit shifts can flip the selected option (Pezeshkpour & Hruschka, 2024; Wang et al., 2024a). Yet on GPQA, despite lower EM, fluctuation is not excessive, suggesting *difficulty* alone does not significantly affect the CoT robustness. Interpreted through our bounds, stronger models typically (i) train on *data with higher consistency* (better cleaning and synthesis) which increases the upper bound of input perturbations,

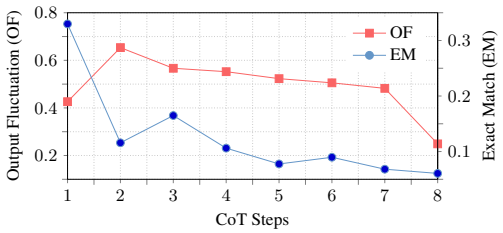


Figure 2: The change in OF (left Y-axis) and EM (right Y-axis) with the reasoning steps of the generated CoT, averaged across all experimental datasets and models. The X-axis denotes the CoT step and Y-axis denotes the value of each metric. The curves at X and Y axes illustrate the data distribution. The CoT steps are segmented using ROSCOE (Golovneva et al., 2023).

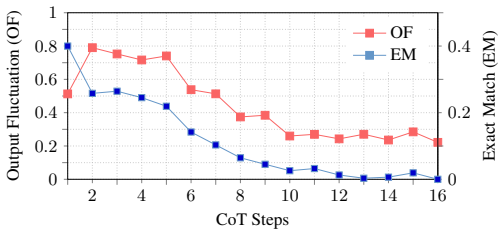


Figure 3: The change in OF (left Y-axis) and EM (right Y-axis) with the reasoning steps on all datasets and models under the reasoning steps from 1 to 16. The X-axis denotes the CoT step and Y-axis denotes each metric. The curves at X and Y axes illustrate the data distribution. The CoT steps are segmented using ROSCOE (Golovneva et al., 2023).

which is governed by the data-consistency constant Γ in Theorem 2, and (ii) yield *longer, more structured reasoning steps*, increasing K in Theorem 1 and thereby tightening the fluctuation bound. Models supporting Long-CoT (Chen et al., 2025) (e.g., Llama-R1, Qwen3) exemplify this effect. Notably, some settings exhibit larger fluctuations in EM despite having smaller OF. This occurs because the average EM differs across settings, where a setting with a high average EM, even a minor output fluctuation can result in a large absolute EM fluctuation. In contrast, OF directly measures the consistency of the outputs, thus offering a more faithful representation of output robustness.

Greater Input Perturbation Makes Output Less Robust To observe the effect of input perturbation on the model output, we analyze the change in output fluctuation with respect to the input perturbation across all datasets and models. For each question, the input perturbation is calculated as the average distance of input embedding vectors from their mean vector. The results are shown in Figure 1. From the figure, we can find that: *(i)* As the input perturbation increases, the output fluctuation also increases (Pearson Correlation Coefficient $= 0.619$), which supports the conclusion of Theorem 1. *(ii)* The majority of input perturbations are concentrated in the range of less than 0.1, yet the corresponding change in output fluctuation is quite large, which indicates that even slight fluctuations in the input can lead to significant fluctuations in the output, which is consistent with the findings of previous studies (Zhao et al., 2024; Bigelow et al., 2024). *(iii)* When the input perturbation exceeds 0.2, the output fluctuation becomes relatively robust as the input perturbation increases. This is because the output fluctuation is measured using normalized entropy, whose maximum possible value depends on how many prompts are used to generate answers. This means that even when input changes become larger, the maximum possible fluctuation in the output stays roughly constant.

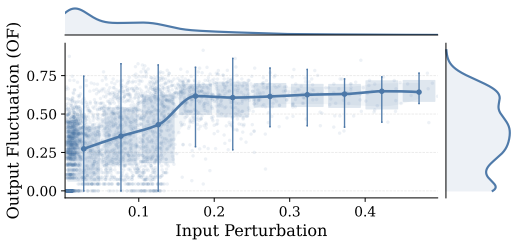


Figure 1: The output fluctuation across input perturbation on all datasets and models. Each point denotes one question, where X-axis denotes the input perturbation as the average distance of embedding vectors from their mean vector, and Y-axis denotes OF. The curves at X and Y axes illustrate the data distribution. The Pearson correlation coefficient is 0.619.

4.3 Impact of Reasoning Step Length on CoT Robustness

Robustness is Positively Correlated with Reasoning Step Length To verify the impact of reasoning steps, we analyze performance as a function of CoT steps (steps computed following ROSCOE (Golovneva et al., 2023)). The experimental results are presented in Figure 2. Figure 2 reveals the following trends: *(i)* Output fluctuation generally *decreases* as steps increase, matching Theorem 1: larger K tightens the robustness bound. *(ii)* The model output fluctuation is relatively low for one-step CoT cases because trivially solvable items need little reasoning and are stable even with short chains. *(iii)* The performance (*i.e.*, EM) does *not* not necessarily increase with K. More steps often correlate with harder items, so accuracy can drop as K rises.

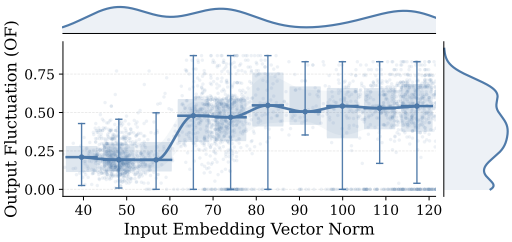


Figure 4: The change in output fluctuation with the norm of the input embedding vector across all experimental datasets and models. Each point denotes the result of one question, where X-axis denotes the input vector norm and Y-axis denotes OF of this question. The Pearson correlation coefficient is 0.506.

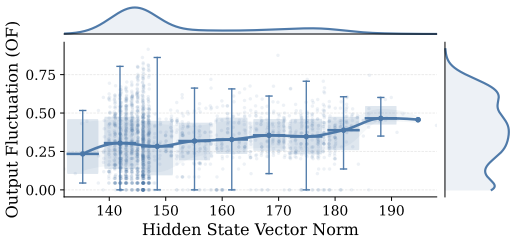


Figure 5: The change in output fluctuation with the norm of the hidden state vector across all experimental datasets and models. Each point denotes the result of one question, where X-axis denotes the hidden state vector norm and Y-axis denotes OF of this question. The Pearson correlation coefficient is 0.229.

Infinity Reasoning Steps Cannot Eliminate the Impact of Input Perturbation To further verify the conclusion from Equation 4 that even infinitely long reasoning steps cannot completely eliminate the impact of input perturbations, we conduct experiments with an extended number of reasoning steps. We add the instruction "`You MUST reason in exactly` K `steps`" to the prompt to guide the model in generating longer reasoning processes, requiring the model to generate outputs for $K = 1, ..., 16$ steps. Considering that the model could not strictly follow the instruction to generate the specified steps, we still use ROSCOE to calculate the actual steps. The results are presented in Figure 3. From the figure, we observe that as the number of reasoning steps increases, the output fluctuation decreases but eventually converges to a relatively stable level. This indicates that the role of reasoning steps in eliminating perturbations is limited, thereby empirically validating the conclusion of Equation 4. Since OF begins to fluctuate, we suppose that current experimental steps have supported our conclusion and do not conduct experiments over 16 steps.

4.4 Impact of Embedding Norms on CoT Robustness

Larger Input Embedding Norm Makes Output Less Robust To verify the relationship between output fluctuation and the norm of the input embedding vector, we analyze the experimental results across all datasets and models, as shown in Figure 4. From the figure, we can observe that: *(i)* As the norm of the input embedding vector increases, the model output fluctuation shows a general upward trend, which confirms the related conclusions in Theorem 2. *(ii)* As the input embedding norm grows, output fluctuation saturates, since a normalized entropy capped by the number of prompts, its maximum stays roughly constant even under larger input perturbations. *(iii)* When the norm of the input embedding vector increases from 60 to 70, the output fluctuation exhibits a sudden jump, which indicates that a threshold exists for the vector norm that the model can handle stably. Once this threshold is surpassed, most input perturbations exceed the upper bound defined in Theorem 2, causing significant fluctuations in the output.

Larger Hidden State Norm Makes Output Less Robust To verify the relationship between the norm of the hidden state vector and the output fluctuation, we analyze the results across all datasets and models. The hidden state vector is extracted from the last layer of the last CoT step. The results are shown in Figure 5. From the figure, we can find that: *(i)* As the norm of the hidden state increases, the output fluctuation shows a general upward trend, which confirms the positive correlation between the two as stated in Theorem 2. *(ii)* The vector norms for the majority of data points are concentrated on the $(140, 150)$ range, which indicates that a well-trained model tends to encode data into a specific and relatively small norm interval to mitigate the impact of input perturbations. *(iii)* Overall, the change in output fluctuation with the hidden state norm is not significant. We suppose the reasons for this are that the constant γ is determined by the upper bound of the hidden state norm rather than its specific value, and that the various normalization structures like LayerNorm (Xiong et al., 2020) within the Transformer architecture mitigate the output fluctuation to some extent.

4.5 Prompt Optimization with Higher Input Robustness

To shed light on future research, we discuss how to optimize the performance of prompt optimization based on Theorem 2. Let $\tau = \alpha^{-1}s$ and F denote the expression on the right-hand side of Theorem 2. We hope to select the prompt that makes F as large as possible, thereby increasing the upper bound of the input perturbation. Let $A = (R_x R_h)^2$, we can derive that:

$$\frac{\partial F}{\partial A} = -\frac{R\tau^2}{2(\eta + \tau R_h^2)\sqrt{\eta^2 + \tau^2 A}} < 0 \quad (9)$$

This shows that F is negatively correlated with A, where a larger value of A^{-1} corresponds to a larger upper bound for the input perturbation, meaning the model can tolerate greater input perturbations. Therefore, for each question, we first construct inputs using all obtained prompts and extract the corresponding embedding layer vectors, as well as the vectors from the final layer to serve as hidden state vectors. We then calculate the norms of both to obtain A. Subsequently, for each question, we select the prompt with the highest value of A^{-1} as the designated prompt for inference. The experimental results are shown in Table 3. We calculate only the Exact Match (EM) for each method and not the Output Fluctuation (OF), since each method selects a single optimal prompt for each question to perform inference, and consequently yields only one output as the final answer. From the table, we can see that our method brings performance improvements across all settings, which demonstrates the effectiveness of our method. We also discuss the efficiency of our method in Appendix D.2. Since the primary objective of this paper is to analyze the factors affecting input robustness, rather than to optimize prompt optimization methods, we leave the investigation into how to better effectiveness and efficiency, and a more extensive comparison with additional baselines for future work.

Table 3: EM on each model and dataset using different prompt optimization methods. "-" denotes using the single base prompt directly. The best result under each setting is marked in **bold**.

Model	Method	MATH	MMLU-Pro	GPQA
Llama3.1-8b	-	46.8	45.7	23.7
	TextGrad	45.2	47.4	27.6
	OPRO	44.6	47.1	27.1
	CFPO	47.0	48.1	27.6
	Ours	**47.2**	**49.0**	**32.3**
Qwen3-8b	-	77.4	42.3	37.1
	TextGrad	77.6	44.9	38.4
	OPRO	77.2	45.9	37.4
	CFPO	77.0	45.8	38.4
	Ours	**77.6**	**49.2**	**38.4**

5 Related Works

Robustness of Chain-of-Thought. Numerous studies show that slight perturbations in the input can lead to drastic changes in the output of CoT (Zhao et al., 2024; Shi et al., 2024b). Therefore, to enhance the performance of CoT, a variety of works are proposed to improve and analyze the CoT robustness. For example, noisy or off-task rationales reliably degrade CoT performance. Contrastive denoising, including CD-CoT and NoRa, mitigates these effects (Zhou et al., 2024). Break-The-Chain applies semantics-preserving rewrites (narrativization, mild constraint changes, reordering, numeric tweaks) to reveal sensitivity in code generation (Roh et al., 2025). Character-level perturbations (R^2ATA) likewise disrupt reasoning (Gan et al., 2024). Chain-of-Defensive-Thought structures defensive rationales that resist corruption or injection and reduce collapse (Wang et al., 2025). Post-hoc Self-Correction Reflection repairs errors under perturbations (Wu et al., 2025). Self-Consistency reduces single-path brittleness through voting (Wang et al., 2023b). CoT is sensitive to step order and exemplar relevance (Wang et al., 2023a). Theory indicates that more coherent chains aid error correction but increase vulnerability to noise in intermediate steps (Cui et al., 2024). Evidence also suggests that CoT often functions as constrained imitation rather than genuine reasoning (Shao & Cheng, 2025). Generalization analyses for nonlinear Transformers identify robustness conditions under noise and distribution shift (Li et al., 2024).

Despite these advances, the mechanism by which input perturbations induce output changes remains under-specified. We derive upper bounds that link input perturbations to output fluctuations and analyze the factors that govern CoT robustness, extending prior research.

Prompt Optimization. Prompt optimization methods primarily focus on how to optimize prompts based on the given model and task to enhance the performance. Work on prompt optimization spans RL and gradient-free edit search (Deng et al., 2022; Prasad et al., 2023), influential-token clustering to shrink the search space (Zhou et al., 2023a), and ensemble-style boosting to avoid single-prompt failure (Hou et al., 2023). Refinements include genetic and actor–critic editing, localized zeroth-order updates, and exemplar-ordering optimization (Xu et al., 2022; Dong et al., 2024; Hu et al., 2024; Wu et al., 2024). APE and OPRO iteratively propose and select improved instructions (Zhou

et al., 2023b; Yang et al., 2024). ProTeGi and APO implement textual "gradient descent" with beam or bandit search (Pryzant et al., 2023). TextGrad generalizes to "automatic differentiation via text" (Yuksekgonul et al., 2025). Data-driven pipelines such as Self-Instruct and Auto-Instruct bootstrap and rank prompt sets (Wang et al., 2023c; Zhang et al., 2023). Search strategies include MCTS with reflective error analysis (Wang et al., 2024b). Budgeted best-arm identification supports selection under tight evaluation budgets (Shi et al., 2024a). Preference-based black-box optimization aligns prompts with user goals (Cheng et al., 2024). RL improves textual-prompt stability (Kwon et al., 2024). Compiler-style systems such as DSPy learn prompts for multi-stage LM pipelines (Khattab et al., 2024). OPRO-like gains may attenuate on smaller open models (Zhang et al., 2024b).

Despite strong empirical progress, the mechanism pathway from input perturbations to output fluctuations remains poorly understood. We analyze this affect and its determinants to guide principled designs for the future prompt optimization works.

6 Conclusion

In this paper, we theoretically analyze the influence of various factors on the input robustness of CoT. We first prove that the impact of input perturbations on the CoT output is negatively correlated with the number of CoT reasoning steps, and that even an infinite number of steps cannot completely eliminate the effects of input perturbations. We then apply these findings to LSA, demonstrating that its input robustness is negatively correlated with the norms of the input embedding and hidden state vectors. To validate these conclusions, we conduct experiments on four mainstream LLMs and three mainstream datasets. Experimental results reveal that output fluctuations vary with different factors in line with our expectations, supporting the validity of our findings. Furthermore, guided by this analysis, we propose to select the prompt by raising the upper bound of input perturbation, which yields consistent performance gains over previous works. Moving forward, our work opens several promising avenues for advancing robust chain-of-thought reasoning. In particular, a key next step is to systematically examine how the parameters Γ and η in Theorem 2 influence input robustness, which could also inform the design of more resilient large reasoning models.

7 Reproducibility

We have provided all proofs of this paper in Appendix C. We will release the experimental and pre-processed data and code upon the paper being accepted.

References

Thomas C. Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, G. Cottrell, and Julian McAuley. Rezero is all you need: Fast convergence at large depth. In *Conference on Uncertainty in Artificial Intelligence*, 2020. URL https://api.semanticscholar.org/CorpusID:212644626.

Eric J Bigelow, Ekdeep Singh Lubana, Robert P. Dick, Hidenori Tanaka, and Tomer Ullman. In-context learning dynamics with random binary sequences. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=62K7mALO2q.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models, 2025. URL https://arxiv.org/abs/2503.09567.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. Finqa: A dataset of numerical reasoning over financial data. *Proceedings of EMNLP 2021*, 2021.

Jiale Cheng, Xiao Liu, Kehan Zheng, Pei Ke, Hongning Wang, Yuxiao Dong, Jie Tang, and Minlie Huang. Black-box prompt optimization: Aligning large language models without model training. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*

(Long Papers), pp. 3201–3219, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.176. URL https://aclanthology.org/2024.acl-long.176/.

Prateek Chhikara. Mind the confidence gap: Overconfidence, calibration, and distractor effects in large language models, 2025. URL https://arxiv.org/abs/2502.11028.

Liam Collins, Advait Parulekar, Aryan Mokhtari, Sujay Sanghavi, and Sanjay Shakkottai. In-context learning with transformers: softmax attention adapts to function lipschitzness. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2025. Curran Associates Inc. ISBN 9798331314385.

Yingqian Cui, Pengfei He, Xianfeng Tang, Qi He, Chen Luo, Jiliang Tang, and Yue Xing. A theoretical understanding of chain-of-thought: Coherent reasoning and error-aware demonstration. *arXiv preprint arXiv:2410.16540*, 2024. URL https://arxiv.org/abs/2410.16540.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric Xing, and Zhiting Hu. Rlprompt: Optimizing discrete text prompts with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3369–3391, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.222. URL https://aclanthology.org/2022.emnlp-main.222/.

Richard Diehl Martinez, Pietro Lesci, and Paula Buttery. Tending towards stability: Convergence challenges in small language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3275–3286, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.187. URL https://aclanthology.org/2024.findings-emnlp.187/.

Yihong Dong, Kangcheng Luo, Xue Jiang, Zhi Jin, and Ge Li. Pace: Improving prompt with actor-critic editing for large language model. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 7304–7323, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.436. URL https://aclanthology.org/2024.findings-acl.436/.

Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George J. Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc.

Roland Friedrich. Complexity and entropy in legal language. *Frontiers in Physics*, 9:671882, jun 2021. doi: 10.3389/fphy.2021.671882.

Esther Gan, Yiran Zhao, Liying Cheng, Yancan Mao, Anirudh Goyal, Kenji Kawaguchi, Min-Yen Kan, and Michael Shieh. Reasoning robustness of llms to adversarial typographical errors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 10449–10459, Miami, Florida, USA, 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.emnlp-main.584/.

Olga Golovneva, Moya Peng Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. ROSCOE: A suite of metrics for scoring step-by-step reasoning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=xYlJRpzZtsY.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor

Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Bairu Hou, Joe O'Connor, Jacob Andreas, Shiyu Chang, and Yang Zhang. Promptboosting: Black-box text classification with ten forward passes. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 13463–13488. PMLR, 2023. URL https://proceedings.mlr.press/v202/hou23b.html.

Wenyang Hu, Yao Shu, Zongmin Yu, Zhaoxuan Wu, Xiangqiang Lin, Zhongxiang Dai, See-Kiong Ng, and Bryan Kian Hsiang Low. Localized zeroth-order prompt optimization. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024. URL https://arxiv.org/abs/2403.02993.

Jianhao Huang, Zixuan Wang, and Jason D. Lee. Transformers learn to implement multi-step gradient descent with chain of thought. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=r3DF5sOo5B.

Yue Huang, Jiawen Shi, Yuan Li, Chenrui Fan, Siyuan Wu, Qihui Zhang, Yixin Liu, Pan Zhou, Yao Wan, Neil Zhenqiang Gong, and Lichao Sun. Metatool benchmark for large language models: Deciding whether to use tools and which to use. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=R0c2qtalgG.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. Dspy: Compiling declarative language model calls into state-of-the-art pipelines. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sY5N0zY5Od. ICLR 2024 (spotlight).

Hyunjik Kim, George Papamakarios, and Andriy Mnih. The lipschitz constant of self-attention, 2021. URL https://openreview.net/forum?id=DHSNrGhAY7W.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=e2TBb5y0yFf.

Minchan Kwon, Gaeun Kim, Jongsuk Kim, Haeil Lee, and Junmo Kim. Stableprompt: Automatic prompt tuning using reinforcement learning for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9868–9884, November 2024. URL https://aclanthology.org/2024.emnlp-main.551.pdf.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Hongkang Li, Meng Wang, Songtao Lu, Xiaodong Cui, and Pin-Yu Chen. Training nonlinear transformers for chain-of-thought inference: A theoretical generalization analysis. *arXiv preprint arXiv:2410.02167*, 2024. URL https://arxiv.org/abs/2410.02167. OpenReview ID: n7n8McETXw.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Yuanye Liu, Jiahang Xu, Li Lyna Zhang, Qi Chen, Xuan Feng, Yang Chen, Zhongxin Guo, Yuqing Yang, and Peng Cheng. Beyond prompt content: Enhancing llm performance via content-format integrated prompt optimization, 2025. URL https://arxiv.org/abs/2502.04295.

Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 188–197, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1018. URL https://aclanthology.org/D19-1018/.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: an imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2006–2017, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.130. URL https://aclanthology.org/2024.findings-naacl.130/.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. Grips: Gradient-free, edit-based instruction search for prompting large language models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 3845–3864, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.277. URL https://aclanthology.org/2023.eacl-main.277/.

Reid Pryzant, Dan Iter, Jerry Li, Yin Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7957–7968, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.494. URL https://aclanthology.org/2023.emnlp-main.494/.

Xianbiao Qi, Jianan Wang, Yihao Chen, Yukai Shi, and Lei Zhang. Lipsformer: Introducing lipschitz continuity to vision transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=cHf1DcCwcH3.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Jaechul Roh, Varun Gandhi, Shivani Anilkumar, and Arin Garg. Break-the-chain: Reasoning failures in llms via adversarial prompting in code generation. *arXiv preprint arXiv:2506.06971*, 2025. URL https://arxiv.org/abs/2506.06971.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications, 2025. URL https://arxiv.org/abs/2402.07927.

Jintian Shao and Yiming Cheng. Cot is not true reasoning, it is just a tight constraint to imitate: A theory perspective. *arXiv preprint arXiv:2506.02878*, 2025. URL https://arxiv.org/abs/2506.02878.

Chengshuai Shi, Kun Yang, Zihan Chen, Jundong Li, Jing Yang, and Cong Shen. Efficient prompt optimization through the lens of best arm identification. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/b46bc1449205888e1883f692aff1a252-Abstract-Conference.html.

Zhenmei Shi, Junyi Wei, Zhuoyan Xu, and Yingyu Liang. Why larger language models do in-context learning differently? In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024b.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Shubham Vatsal and Harsh Dubey. A survey of prompt engineering methods in large language models for different nlp tasks, 2024. URL https://arxiv.org/abs/2407.12994.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2717–2739, Toronto, Canada, 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.153. URL https://aclanthology.org/2023.acl-long.153/.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *CoRR*, abs/2006.04768, 2020a. URL https://arxiv.org/abs/2006.04768.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020b. Curran Associates Inc. ISBN 9781713829546.

Wenxiao Wang, Parsa Hosseini, and Soheil Feizi. Chain-of-defensive-thought: Structured reasoning elicits robustness in large language models against reference corruption. *arXiv preprint arXiv:2504.20769*, 2025. URL https://arxiv.org/abs/2504.20769.

Xinpeng Wang, Chengzhi Hu, Bolei Ma, Paul Rottger, and Barbara Plank. Look at the text: Instruction-tuned language models are more robust multiple choice selectors than you think. In *First Conference on Language Modeling*, 2024a. URL https://openreview.net/forum?id=qHdSA85GyZ.

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P. Xing, and Zhiting Hu. Promptagent: Strategic planning with language models enables expert-level prompt optimization. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=22pyNMuIoa. ICLR 2024.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023b. URL https://openreview.net/forum?id=1PL1NIMMrw.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13484–13508, Toronto, Canada, July 2023c. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.754. URL https://aclanthology.org/2023.acl-long.754/.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024c. URL https://openreview.net/forum?id=y10DM6R2r3.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6/.

Hua Wu, Haotian Hong, Jiayu Mao, Zhexun Yin, Yanxiong Wu, Xiaojing Bai, Li Sun, Mengyang Pu, Juncheng Liu, and Yihuan Li. Forging robust cognition resilience in large language models: The self-correction reflection paradigm against input perturbations. *Applied Sciences*, 15(9):5041, 2025. URL https://www.mdpi.com/2076-3417/15/9/5041.

Zhaoxuan Wu, Xiaoqiang Lin, Zhongxiang Dai, Wenyang Hu, Yao Shu, See-Kiong Ng, Patrick Jaillet, and Bryan Kian Hsiang Low. Prompt optimization with ease? efficient ordering-aware automated selection of exemplars. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/dd8e7dae18cecd7c9137840161e1bf62-Abstract-Conference.html.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10524–10533. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/xiong20b.html.

Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. Gps: Genetic prompt search for efficient few-shot learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8162–8171, December 2022. URL https://aclanthology.org/2022.emnlp-main.559.pdf.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Bb4VGOWELI.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative ai by backpropagating language model feedback. *Nature*, 639(8055):609–616, mar 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-08661-4. URL https://doi.org/10.1038/s41586-025-08661-4.

Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Residual learning without normalization via better initialization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1gsz30cKX.

Ruiqi Zhang, Spencer Frei, and Peter L. Bartlett. Trained transformers learn linear models in-context. *J. Mach. Learn. Res.*, 25(1), January 2024a. ISSN 1532-4435.

Tuo Zhang, Jinyue Yuan, and Salman Avestimehr. Revisiting opro: The limitations of small-scale llms as optimizers. In *Findings of the Association for Computational Linguistics: ACL 2024*, Bangkok, Thailand, 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.100. URL https://aclanthology.org/2024.findings-acl.100/.

Zhihan Zhang, Shuohang Wang, Wenhao Yu, Yichong Xu, Dan Iter, Qingkai Zeng, Yang Liu, Chenguang Zhu, and Meng Jiang. Auto-instruct: Automatic instruction generation and ranking for black-box language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Singapore, 2023.

Siyan Zhao, Tung Nguyen, and Aditya Grover. Probing the decision boundaries of in-context learning in large language models. In *First Workshop on Long-Context Foundation Models @ ICML 2024*, 2024. URL https://openreview.net/forum?id=t90UB9wvUZ.

Han Zhou, Xingchen Wan, Ivan Vulić, and Anna Korhonen. Survival of the most influential prompts: Efficient black-box prompt search via clustering and pruning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 13064–13077, Singapore, December 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.870. URL https://aclanthology.org/2023.findings-emnlp.870/.

Wangchunshu Zhou, Canwen Xu, Tao Ge, Julian McAuley, Ke Xu, and Furu Wei. Bert loses patience: fast and robust inference with early exit. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=92gvk82DE-. ICLR 2023 (poster).

Zhanke Zhou, Rong Tao, Jianing Zhu, Yiwen Luo, Zengmao Wang, and Bo Han. Can language models perform robust reasoning in chain-of-thought prompting with noisy rationales? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://arxiv.org/abs/2410.23856. arXiv:2410.23856.

Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. On the calibration of large language models and alignment. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 9778–9795, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.654. URL https://aclanthology.org/2023.findings-emnlp.654/.

A Ethics Statement

All datasets and models used in this paper are publicly available, and our usage follows their licenses and terms.

B LLM Usage

We have employed the AI tool for coding and writing polishing.

C Proofs

Proof of Theorem 1.

$$\varepsilon_k := h_k(x+\delta) - h_k(x), \qquad k \in \mathbb{N}^+.$$
$$h_k(x) = f(h_{k-1}(x), x), \qquad h_k(x+\delta) = f\big(h_{k-1}(x+\delta), x+\delta\big).$$
$$\|\varepsilon_k\| = \big\|f(h_{k-1}(x+\delta), x+\delta) - f(h_{k-1}(x), x)\big\|.$$
$$\|f(h_1, x_1) - f(h, x)\| \le \gamma\|h_1 - h\| + C\|x_1 - x\|.$$
$$\Rightarrow \quad \|\varepsilon_k\| \le \gamma\|\varepsilon_{k-1}\| + C\|\delta\|.$$
$$\Rightarrow \quad \|\varepsilon_K\| \le \gamma^K\|\varepsilon_0\| + C\|\delta\| \sum_{i=0}^{K-1} \gamma^i.$$
$$\sum_{i=0}^{K-1} \gamma^i = \frac{1-\gamma^K}{1-\gamma} \quad (\gamma \in [0,1)).$$
$$\Rightarrow \quad \|\varepsilon_K\| \le \gamma^K\|\varepsilon_1\| + \frac{C}{1-\gamma}(1-\gamma^K)\|\delta\|.$$
$$A := \max \frac{\|\varepsilon_1\|}{\|\delta\|}.$$
$$\Rightarrow \quad \boxed{\|\varepsilon_K\| \le \Big(A\gamma^K + \frac{C}{1-\gamma}(1-\gamma^K)\Big)\|\delta\|}$$

□

Proof of Lemma 1.

$$E = \begin{bmatrix} h \\ x \end{bmatrix}, \qquad f(E) = \eta E + (PE)\, s(E), \qquad s(E) = E^\top K E.$$
$$P = W_*^{PV} = [\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \qquad K = W_*^{KQ} = [\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{4}} \begin{bmatrix} \Gamma^{-1} & 0 \\ 0 & 0 \end{bmatrix}.$$
$$K_s := \tfrac{1}{2}(K + K^\top) = K, \qquad \nabla s(E) = 2K_s E.$$
$$\nabla f(E) = \eta I + s(E)P + (PE)\,(2K_s E)^\top. \tag{$\star$}$$

Bound for C.

$$\frac{\partial f}{\partial x}(E) = \eta\,\Pi_x + s(E)\,P_x + (PE)\,\big(2K_s E\big)_x^\top.$$
$$K = \begin{bmatrix} * & 0 \\ 0 & 0 \end{bmatrix} \;\Rightarrow\; (K_s E)_x = 0 \;\Rightarrow\; (2K_s E)_x = 0.$$
$$\Rightarrow \quad \frac{\partial f}{\partial x}(E) = \eta\,\Pi_x + s(E)\,P_x.$$
$$\Big\|\tfrac{\partial f}{\partial x}(E)\Big\| \le \eta + \|P_x\|\,|s(E)| \le \eta + \|P\|\,\|K\|\,\|E_h\|^2.$$
$$E_h = \begin{bmatrix} h \\ 0 \end{bmatrix}, \quad \|E_h\| = \|h\| \le R_h, \quad \|P\| = [\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}}, \quad \|K\| = [\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{4}}\|\Gamma^{-1}\|.$$
$$\Rightarrow \quad \boxed{C \;\le\; \eta + \|\Gamma^{-1}\|\,R_h^2} \quad \text{(i.e., } C \le \eta + [\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{4}}\|\Gamma^{-1}\|\,R_h^2\text{)}.$$

Bound for γ.

$$\frac{\partial f}{\partial h}(E) = \eta\,\Pi_h + s(E)\,P_h + (PE)\left(2K_sE\right)_h^{\top}.$$
$$P_h = 0 \quad (\text{since } PE = [0;\,[\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}}x]),$$
$$(2K_sE)_h = 2\,[\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{4}}\Gamma^{-1}h.$$

For any $v \in \mathbb{R}^d$,

$$\frac{\partial f}{\partial h}(E)v = \begin{bmatrix}\eta v\\ 0\end{bmatrix} + \begin{bmatrix}0\\ \|PE\|\cdot\frac{2\,\|\Gamma^{-1}h\|}{[\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}}}\,\frac{(h^{\top}\Gamma^{-1}v)}{\|\Gamma^{-1}h\|}\,\frac{PE}{\|PE\|}\end{bmatrix}.$$

$$\|PE\| = [\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}}\|x\| \le [\mathrm{Tr}(\Gamma^{-2})]^{\frac{1}{4}}R_x, \qquad \|\Gamma^{-1}h\| \le \|\Gamma^{-1}\|\,\|h\| \le \|\Gamma^{-1}\|R_h.$$

$$\text{(orthogonal blocks)} \quad \Rightarrow \quad \left\|\frac{\partial f}{\partial h}(E)v\right\|^2 \le \eta^2\|v\|^2 + \Big(2\,R_x\,\|\Gamma^{-1}\|\,R_h\Big)^2\|v\|^2.$$

$$\Rightarrow \quad \boxed{\gamma \;\le\; \sqrt{\eta^2 + 4\,R_x^2\,\|\Gamma^{-1}\|^{\,2}\,R_h^2}} \quad (\text{i.e., } \gamma \le \sqrt{\eta^2 + 4\,R_x^2[\mathrm{Tr}(\Gamma^{-2})]^{-\frac{1}{2}}\|\Gamma^{-1}\|^{\,2}\,R_h^2}\,).$$

□

D Additional Discussion

D.1 Influence of Non-Linear Factors of Transformer

In this section, we discuss the influence of different non-linear factors within the Transformer architecture on the conclusions of Theorem 2. Overall, most non-linear factors contribute to enhancing the model's input robustness. Due to the complexity of theoretically proving the effects of these non-linear factors, we only provide an intuitive analysis and leave rigorous mathematical proofs for future work.

Attention Non-linearity (Softmax) The exponential normalization of Softmax produces sharp distributions at low temperatures or with large logit scaling, leading to a "winner-takes-all" switching behavior among highly competitive keys. Intuitively, this amplifies the sensitivity to perturbations in the input and intermediate states, which is equivalent to increasing the effective Lipschitz constant (γ) and the input channel coefficient (C). It also causes locally quasi-discrete transitions in attention weights. Therefore, within the framework of Theorem 2, sharper attention typically reduces the tolerable perturbation radius. Conversely, smoother attention (achieved with high temperature or small scaling factors) mitigates this sensitivity, thereby increasing the robustness radius.

Non-linearity of Normalization Layers (LayerNorm/RMSNorm) Normalization explicitly constrains the norm of hidden states through demeaning and scaling by variance. When statistics are stable, this effectively suppresses R_h and weakens the amplification chain across layers, manifesting as smaller effective values for γ and C. This aligns with the monotonic relationship described in Theorem 2, where a smaller norm corresponds to stronger robustness. However, it is important to note that when the intra-layer variance becomes abnormally small (close to zero), the scaling factor can locally amplify noise, creating transient high-gain regions and leading to edge cases where robustness decreases. Therefore, stable statistics and moderate pre-scaling (such as layer scaling during training) help ensure the positive impact of normalization on the robustness radius.

Non-linearity of Feed-Forward Network Activations (GELU/ReLU/SwiGLU) The activation function determines the gain of the local Jacobian. In saturated regions (such as the left tail of GELU), the local slope approaches zero, which suppresses noise propagation and limits the norm of intermediate representations, thereby increasing the tolerable perturbation radius. In contrast, high-gain regions (resulting from large weights or strong inputs) amplify the norm of intermediate states and the output sensitivity, which translates to larger effective values for γ and C. Gated variants (such as SwiGLU/MoE) can also trigger discrete switching of channels or experts near their thresholds, causing the output to undergo abrupt transitions in response to small perturbations. Overall, operating the activations in low-to-medium gain regions and controlling the scale of the weights helps to reduce the effective sensitivity and decrease R_h, which aligns with the monotonic properties described in Theorem 2.

Residual Paths and Layer Scaling (Semantics of η) The residual path directly injects the representation from the previous layer into the next, scaled by a coefficient, which can be viewed as the η in Theorem 2. A larger η allows more input and intermediate perturbations to pass through without attenuation and accumulate in deeper layers, leading to an increase in the effective γ and a decrease in the tolerable perturbation radius. Conversely, a smaller residual coefficient or layer scaling techniques (such as the ideas behind ReZero/LayerScale) can suppress this long-chain amplification and enhance robustness. A trade-off exists, as an overly small η can limit feature reuse and gradient flow. In practice, a moderate but non-zero layer scaling is often adopted to achieve a better compromise between expressive power and robustness under the constraints of Theorem 2.

D.2 Efficiency of Our Method

In this section, we discuss the computational efficiency of the improved method we propose in §4.5. Let M be the total number of candidate prompts, and let $T(\mathcal{M}, D)$ denote the time it takes for the model $\mathcal{M}$ to run once on the evaluation dataset D. Then, because calculating each candidate requires the hidden state vector for every data point, which necessitates a full inference pass, the total running time is:

$$O(M \cdot T(\mathcal{M}, D)) \tag{10}$$

Compared to other prompt optimization methods, many approaches also need to run each generated prompt on an evaluation dataset to assess its quality (e.g., TextGrad, OPRO). Therefore, the efficiency of our method is considered comparable to that of previous work. Furthermore, since this paper primarily focuses on theoretical analysis rather than methodological improvements, we leave further enhancements to the effectiveness and efficiency as future work.

D.3 Input Robustness of Non-Linear Self-Attention

According to the discussion in §2.3, the certified input-perturbation radius obtainable via CoT depends on the model's Lipschitz properties. In this section, we replace the LSA with a *non-linear* (softmax) attention and derive a counterpart of Theorem 2.

We adopt a standard single-head attention mechanism with a residual flow. Let $W^Q, W^K, W^V, W^{PV} \in \mathbb{R}^{b\times b}$ be the projection matrices, and let $\tau > 0$ be the temperature. Denote the parameters by $\theta = (W^Q, W^K, W^V, W^{PV}, \tau)$ and let $E = [h, x]$. Define

$$Q = E\,W^Q, \qquad K = E\,W^K, \qquad V = E\,W^V, \qquad A(E) = \mathrm{softmax}\Big(\tfrac{1}{\tau} Q K^\top\Big),$$

where $\mathrm{softmax}$ is applied row-wise. To mitigate gradient explosion, we introduce a residual coefficient $\eta \in (0, 1)$ as in the LSA case. The non-linear attention map is

$$f_{\text{Attn}}(h, x; \theta) \;=\; \eta\, E \;+\; W^{PV}\big(A(E)\, V\big). \tag{11}$$

In what follows, we analyze the input robustness of equation 11 under the same Lipschitz framework as in §2.3.

We first upper bound the two Lipschitz constants in equation 1. Throughout, $\|\cdot\|$ denotes the operator (spectral) norm.

Lemma 2. *Suppose $\|x\| \le R_x$ and $\|h\| \le R_h$. Let*

$$s_Q = \|W^Q\|, \quad s_K = \|W^K\|, \quad s_V = \|W^V\|, \quad s_{PV} = \|W^{PV}\|, \quad s_{QK} = s_Q s_K.$$

Let $L_\sigma(\tau)$ be the (row-wise) Lipschitz constant of the $\mathrm{softmax}$ *map with temperature τ under the chosen norm. Then the constants C and γ in equation 1 admit the bounds*

$$C \;\le\; \eta \;+\; s_{PV}\, s_V\, L_\sigma(\tau)\, s_{QK}\, R_h^2, \qquad \gamma \;\le\; \sqrt{\eta^2 \;+\; 4\, R_x^2 \left(s_{PV}\, s_V\, L_\sigma(\tau)\, s_{QK}\right)^2 R_h^2}\,.$$

Plugging Lemma 2 into the general input-perturbation bound equation 3, we obtain the following certified radius for non-linear attention.

Theorem 3 (Certified Input-Perturbation Radius of Softmax Attention)**.** *If $\|x\| \le R_x$ and $\|h\| \le R_h$, define*

$$\tilde{\beta} \;=\; s_{PV}\, s_V\, L_\sigma(\tau)\, s_{QK}\, R_h^2, \qquad \tilde{\gamma} \;=\; \sqrt{\eta^2 \;+\; 4\, R_x^2 \left(s_{PV}\, s_V\, L_\sigma(\tau)\, s_{QK}\right)^2 R_h^2}\,.$$

With $A > 0$ such that $\|e_0\| \le A\|\delta\|$, the certified tolerable input-perturbation radius of the map equation 11 at CoT step $K \in \mathbb{N}^+$ is

$$\|\delta\| \;\le\; \frac{(1-\tilde{\gamma})\,R}{\big(\eta+\tilde{\beta}\big) \;+\; \big(A(1-\tilde{\gamma})(1+\tilde{\beta})\big)\,\tilde{\gamma}^K}.$$

In particular, if $\tilde{\gamma} < 1$, as $K \to \infty$,

$$\|\delta\| \;\le\; \frac{(1-\tilde{\gamma})\,R}{\eta+\tilde{\beta}}.$$

It can be seen that Theorem 3 has a similar format to Theorem 2, showing that they have the same conclusion regarding the CoT robustness. The above discussion shows the effectiveness of our conclusion under the non-linear scenario.

Proof. Write $f_{\text{Attn}}(E) = \eta E + \Phi(E)$ with $\Phi(E) = W^{PV}\big(\mathrm{softmax}(\frac{1}{\tau}EW^Q(EW^K)^\top)\,(EW^V)\big)$. By composing Lipschitz bounds of the bilinear map $E \mapsto EW^Q(EW^K)^\top$, the row-wise softmax (with constant $L_\sigma(\tau)$), and the linear maps W^V, W^{PV}, we obtain the stated bounds on C and γ. Substituting them into equation 3 yields Theorem 3. □

D.4 The Impact of Vector Norms on the CoT Robustness

While very small weight norms can indeed destabilize optimization during training, their analysis specifically targets inference-time robustness to input perturbations, so there is no contradiction—practical models must balance norm size for both stable training and robust inference. We further argue that robustness is better captured by absolute perturbations, i.e., the raw change in the output, rather than relative perturbations that normalize by the input magnitude, because the model ultimately makes decisions based on the absolute output vector. For example, in a multiple-choice setting, a substantial absolute shift in logits can change the predicted option even if the relative change is small. Therefore, we frame their conclusions in terms of absolute perturbation as a more faithful indicator of decision instability.

D.5 Analysis with Multiple Tokens

In our analysis, we treat the entire user query as a single embedding vector $x \in \mathbb{R}^d$, and the perturbation $\delta \in \mathbb{R}^d$ acts on this query-level representation rather than on individual token embeddings. The bivariate map $f(h, x)$ therefore models the interaction between (i) the current hidden state h and (ii) a fixed embedding of the full input query, not between two literal tokens.

We now show that this is mathematically equivalent to starting from the usual multi-token transformer input.

Corollary 1 (Equivalence of sequence input and bivariate model)**.** *Let the model at a given CoT step take as input*

- *a hidden state $h \in \mathbb{R}^{d_h}$, summarizing all past reasoning tokens; and*
- *a sequence of question tokens with embeddings $(e_1, \ldots, e_T)$, each $e_t \in \mathbb{R}^{d_e}$.*

Assume its next-step hidden state is given by some deterministic map

$$F_{full} : \mathbb{R}^{d_h} \times (\mathbb{R}^{d_e})^T \to \mathbb{R}^{d_h}.$$

Then there exist

1. *a linear isomorphism $U : (\mathbb{R}^{d_e})^T \to \mathbb{R}^d$ (vectorization / padding), and*
2. *a bivariate function $f : \mathbb{R}^{d_h} \times \mathbb{R}^d \to \mathbb{R}^{d_h}$,*

such that the dynamics can be written exactly as

$$h_{k+1} \;=\; f(h_k, x), \qquad x = U(e_1, \ldots, e_T).$$

Moreover, if F_{full} is Lipschitz in $(h, (e_t)_t)$, then f satisfies a Lipschitz condition of the form

$$\|f(h_1, x_1) - f(h_2, x_2)\| \le \gamma\|h_1 - h_2\| + C\|x_1 - x_2\| \tag{12}$$

for some constants $\gamma, C \ge 0$.

Proof. **Step 1: Reparameterizing the input sequence as a single vector.** Fix a maximum sequence length $T_{\max} \ge T$ and pad $(e_1, \dots, e_T)$ with a distinguished padding embedding so that every input can be regarded as an element of $(\mathbb{R}^{d_e})^{T_{\max}}$. This space is linearly isomorphic to $\mathbb{R}^d$ with $d = d_e T_{\max}$.

Let

$$U : (\mathbb{R}^{d_e})^{T_{\max}} \to \mathbb{R}^d$$

be any fixed linear bijection (e.g., concatenation followed by a permutation of coordinates). Define

$$x = U(e_1, \dots, e_T, \text{pad}, \dots, \text{pad}) \in \mathbb{R}^d.$$

Conversely, U^{-1} recovers the full token-level embedding tuple from x.

Step 2: Defining the bivariate function. Define

$$f(h, x) := F_{\text{full}}\big(h, U^{-1}(x)\big).$$

By construction,

$$h_{k+1} = F_{\text{full}}\big(h_k, (e_1, \dots, e_T)\big) = f\big(h_k, U(e_1, \dots, e_T)\big) = f(h_k, x),$$

so the original multi-token dynamics can be written as a bivariate map in (h, x).

Step 3: Preservation of Lipschitz continuity. Suppose the model is Lipschitz in $(h, (e_t)_t)$, i.e., there exist constants $\gamma \ge 0$, $C_{\text{tok}} \ge 0$ such that for all h_1, h_2 and token sequences (e_t), (e'_t),

$$\big\|F_{\text{full}}(h_1, (e_t)) - F_{\text{full}}(h_2, (e'_t))\big\| \le \gamma\|h_1 - h_2\| + C_{\text{tok}} \Big\|(e_t) - (e'_t)\Big\|_{\text{seq}},$$

where $\|\cdot\|_{\text{seq}}$ is any norm on $(\mathbb{R}^{d_e})^{T_{\max}}$.

Using the linear isomorphism U, equip $\mathbb{R}^d$ with the induced norm

$$\|x\| := \big\|U^{-1}(x)\big\|_{\text{seq}}.$$

Then for any (h_1, x_1), (h_2, x_2),

$$\begin{aligned}\|f(h_1, x_1) - f(h_2, x_2)\| &= \big\|F_{\text{full}}(h_1, U^{-1}x_1) - F_{\text{full}}(h_2, U^{-1}x_2)\big\| \\ &\le \gamma\|h_1 - h_2\| + C_{\text{tok}} \big\|U^{-1}(x_1) - U^{-1}(x_2)\big\|_{\text{seq}} \\ &= \gamma\|h_1 - h_2\| + C_{\text{tok}} \|x_1 - x_2\|.\end{aligned}$$

Thus f satisfies the Lipschitz condition with constants γ and $C = C_{\text{tok}}$. □

E Additional Information

E.1 Experimental Dataset

MATH (Hendrycks et al., 2021) is a benchmark for competition-level mathematical reasoning, comprising $12,500$ problems with full step-by-step solutions ($7,500$ training and $5,000$ test). It spans diverse subfields (e.g., algebra, geometry, number theory, combinatorics, probability, and calculus) and is widely used to evaluate and distill chain-of-thought style reasoning in mathematics. In this paper, we evaluate our conclusions with the subset of MATH, which contains 500 data following Lightman et al. (2024).

MMLU-Pro (Wang et al., 2024c) It is a strengthened successor to MMLU that emphasizes higher question quality and robustness. It contains over $12,000$ multiple-choice questions drawn from textbooks and exams across 14 academic domains (e.g., biology, business, chemistry, computer science, economics, engineering, health, history, law, mathematics, philosophy, physics, psychology, and others). Each item offers 10 options, which reduces guessability and increases discrimination among strong models.

GPQA (Rein et al., 2024) targets graduate-level, "Google-proof" scientific reasoning. The test set includes 448 expert-authored multiple-choice questions in biology, physics, and chemistry, designed such that even with open-web access, non-experts struggle while domain experts achieve only modest accuracy. GPQA thus probes high-level knowledge, multistep reasoning, and model reliability under stringent oversight conditions.

E.2 PROMPT

Prompt of MATH
Solve the following math problem efficiently and clearly. Regardless of the approach, always conclude with: Therefore, the final answer is: $boxed{answer}$. I hope it is correct.

Prompt of MMLU-Pro
The following are multiple choice questions (with answers) about domain. Think step by step and then finish your answer with the answer is (X) where X is the correct letter choice.

Prompt of GPQA
Given the following question and four candidate answers (A, B, C and D), choose the best answer. - For simple problems: Directly provide the answer with minimal explanation. - For complex problems: Use this step-by-step format: ## Step 1: [Concise description] (Brief explanation) ## Step 2: [Concise description] (Brief explanation) Regardless of the approach, always conclude with: The best answer is [the_answer_letter]. where the [the_answer_letter] is one of A, B, C or D. Let's think step by step.

Table 4: The prompt used in this paper.

In this section, we list the prompt we used in Table 4.

E.3 PROMPT NUMBER OF EACH SETTING

In this section, we present the number of prompts used for each dataset and model, as shown in Table 5. From the table, we can observe that the number of prompts is not consistent across different settings. This is because, during prompt optimization, the suitable prompts vary for different models and datasets. To ensure that optimal performance is achieved for each setting, we use a different set of prompts for each setting.

Dataset	**Llama2-7b**	**Llama3.1-8b**	**Llama-R1-8b**	**Qwen3-8b**
MATH	14	29	18	13
MMLU-Pro	20	29	20	16
GPQA	11	20	16	12

Table 5: The total number of generated prompts using TextGrad, OPRO, and CFPO under each setting.

E.4 Calculation of Output Fluctuation

Consider a collection of model outputs produced for the same input, represented as a multiset of strings of size M. Let p_i denote the empirical frequency of the i-th distinct string. The metric computes the Shannon entropy:

$$H \;=\; -\sum_i p_i \log_2 p_i, \tag{13}$$

and normalizes it by the maximal entropy achievable with M samples, namely $\log_2 M$. The resulting index:

$$\widehat{H} \;=\; \frac{H}{\log_2 M} \in [0,1] \tag{14}$$

is scale-free and directly comparable across different sample sizes. By construction, $\widehat{H} = 0$ when all outputs are identical (complete consensus, no fluctuation) and $\widehat{H} = 1$ when all M outputs are distinct (maximal dispersion, each outcome occurs once). For empty or singleton sets, the metric is defined to be 0, reflecting the absence of observable variability.

Output fluctuation manifests as dispersion in the empirical outcome distribution. Greater variability spreads probability mass more evenly across distinct strings, driving H toward its maximum and increasing $\widehat{H}$. Greater stability concentrates mass on a single outcome, driving H toward 0 and decreasing $\widehat{H}$. Normalization by $\log_2 M$ ensures that the same qualitative level of dispersion yields comparable scores even when the number of samples differs, while preserving the desired extremes ("all same" $\to 0$, "all different" $\to 1$).

E.5 Implementation Details

The input and hidden state vectors used in our experiments are the encoded vectors from the embedding layer and the final layer of the respective LLMs for the corresponding inputs. For each input, we set the model to generate a single output, with the temperature set to 0, top_p to 1.0, and the random seed fixed at 42. Our experiments are run on a single A100-80G GPU, with the average experiment time for each setting being approximately one hour. All our codes are implemented with PyTorch (Paszke et al., 2019), Transformers (Wolf et al., 2020), and VLLM (Kwon et al., 2023) using Python3.10. We detail how to plot the analysis figure in Appendix E.6.

E.6 Plot of Analysis Figure

We visualize conditional distributions with an x–binned box–plot design. The x–range is uniformly partitioned into $K = 10$ equal–width bins; within each bin we compute the first quartile (Q_1), median, and third quartile (Q_3). Whiskers follow Tukey's rule and extend to the most extreme observations within $[\,Q_1 - 1.5\ \mathrm{IQR},\ Q_3 + 1.5\ \mathrm{IQR}\,]$, where $\mathrm{IQR} = Q_3 - Q_1$; bins with very few points are shown by a median marker only.

To convey the trend across bins, the binwise medians are connected by a shape–preserving piecewise cubic Hermite interpolant (PCHIP). An optional interquartile ribbon is drawn by interpolating Q_1 and Q_3 with the same scheme. For context, we overlay lightly jittered raw points in the background and add marginal density curves along the top (for x) and the right (for y), estimated via Gaussian KDE with Silverman's bandwidth; the right–hand marginal can be computed from an alternative y sample when provided. Box widths adapt to local bin spacing to prevent overlap in narrow x–ranges, and a unified low–saturation color palette is used for visual consistency.

F Additional Experiment

F.1 Fitting γ of Theorem 1

In this section, we verify that $\gamma < 1$ to ensure the reliability of the conclusions derived from Equation 4. Since the right-hand side of the inequality in Theorem 1 is positively correlated with γ, we consider the extreme case by replacing the inequality with an equality, which gives:

$$\|\varepsilon_K\| = \left(A\gamma^K + \frac{C}{1-\gamma}(1-\gamma^K) \right) \|\delta\| \tag{15}$$

Model	MATH	MMLU-Pro	GPQA
Llama2-7b	0.662	0.892	0.671
Llama3.1-8b	0.476	0.218	0.014
Llama-R1-8b	0.879	0.896	0.871
Qwen3-8b	0.754	0.744	0.015

Table 6: The fitted γ on different models and datasets.

Model	Amazon		FinQA		ToolE	
	EM	OF	EM	OF	EM	OF
Llama2-7b	17.4 ± 11.9	0.711	5.9 ± 3.7	0.479	27.1 ± 17.6	0.383
Llama3.1-8b	61.1 ± 35.6	0.271	37.6 ± 6.1	0.377	49.8 ± 18.0	0.365
Llama-R1-8b	60.3 ± 39.8	0.242	45.7 ± 7.4	0.276	51.5 ± 4.9	0.162
Qwen3-8b	61.1 ± 18.4	0.201	54.9 ± 5.9	0.246	56.0 ± 6.6	0.121

Table 7: The performance on Amazon Rview (Amazon), FinQA, and ToolE.

Then, for each question across all datasets, we compute the corresponding $\|\delta\|$ and $\|\varepsilon_K\|$ for different CoT steps K among all generated answers. We use this data to fit the parameter γ in Equation 15 using the least squares method. The fitting results are shown in Table 6. From the table, we can observe that the value of γ is less than 1 in all settings, which validates the reliability of the assumption made in our analysis.

F.2 Performance on More Datasets

To more comprehensively validate the changes in output fluctuation across different datasets, we conduct experiments on a broader range of datasets. We conduct experiments on the Amazon Review (Ni et al., 2019) (sentiment analysis), FinQA (Chen et al., 2021) (financial question answering), and ToolE (Huang et al., 2024) (tool use) datasets to verify our conclusions in scenarios that more closely resemble real-world applications. The experimental results are presented in Table 7. From the table, we can observe that as the model performance improves, the output fluctuation shows an overall downward trend. This is consistent with the conclusions we draw in Table 2.

F.3 Output Fluctuation with Other Metric

To more comprehensively evaluate fluctuations in model outputs, this section quantifies semantic variability (SV) across models and datasets. For each question, we first compute an embedding vector for every answer using all-MiniLM-L6-v2 (Wang et al., 2020b). We then take the average distance from these vectors to their centroid (mean vector) as the metric of output variability. As shown in Table 8, this metric exhibits strong agreement with OF, and the experimental findings are consistent with those in Table 8, thereby corroborating the correctness of our theoretical analysis.

F.4 Performance with Same Prompts

To ablate the effect of prompt differences on the evaluation, we conduct experiments using the same prompts across all models and datasets. For all models, we employ the prompts generated

Model	MATH		MMLU-Pro		GPQA	
	SV	OF	SV	OF	SV	OF
Llama2-7b	0.851	0.475	0.706	0.622	0.760	0.509
Llama3.1-8b	0.777	0.366	0.631	0.350	0.667	0.467
Llama-R1-8b	0.669	0.158	0.601	0.292	0.575	0.371
Qwen3-8b	0.653	0.097	0.579	0.162	0.554	0.214

Table 8: The output fluctuation using different metrics.

Model	MATH		MMLU-Pro		GPQA	
	EM	OF	EM	OF	EM	OF
Llama2-7b	12.2	0.653	13.8	0.578	15.9	0.523
Llama3.1-8b	48.9	0.375	35.1	0.347	28.1	0.470
Llama-R1-8b	65.4	0.147	40.2	0.303	30.0	0.404
Qwen3-8b	77.2	0.097	46.9	0.162	37.3	0.214

Table 9: The average EM and OF of different models and datasets. For the certain dataset, the prompts of each model are all same with Qwen3-8b.

Model	Scale	MATH		MMLU-Pro		GPQA	
		EM	OF	EM	OF	EM	OF
Llama3.1	8b	45.8 ± 7.2	0.366	41.0 ± 10.7	0.350	26.6 ± 5.7	0.467
	70b	56.0 ± 12.8	0.284	63.0 ± 14.4	0.186	42.6 ± 9.7	0.232
Qwen3	8b	77.2 ± 1.6	0.097	46.9 ± 5.2	0.162	37.3 ± 1.9	0.214
	34b	80.8 ± 4.0	0.075	67.8 ± 5.1	0.104	43.6 ± 3.2	0.177

Table 10: The performance of Llama3.1 and Qwen3 on each dataset with different model scales.

with Qwen3-8b. The results, as shown in Table 9, indicate that the conclusions drawn from using identical prompts are consistent with those in Table 2. Therefore, in our main experiments, to ensure a consistent methodology, we use each model to generate the prompts for its own inference.

F.5 Performance cross Different Model Scale

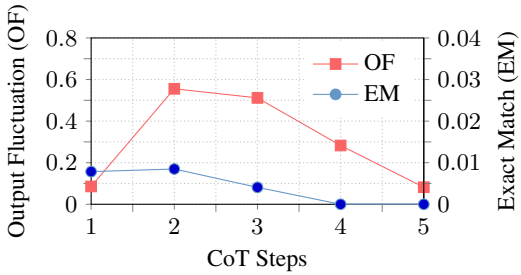
To verify how output fluctuations change with input perturbations on models of different scales, we measure the performance of models of varying scales on each dataset. The experimental results are shown in Table 10. From the table, we can find that although the EM of larger-scale models could exhibit greater fluctuations, from the perspective of OF, larger-scale models generally demonstrate better input robustness. This is because larger-scale models tend to generate a greater number of reasoning steps K (Wei et al., 2022; Kojima et al., 2022) and possess a stronger ability to widen the confidence gap between correct and incorrect answers, which in turn increases the acceptable perturbation threshold R (Zhu et al., 2023; Chhikara, 2025). Consequently, according to Theorem 3, larger-scale models exhibit better robustness.

F.6 Performance with CoT Steps under Each Setting

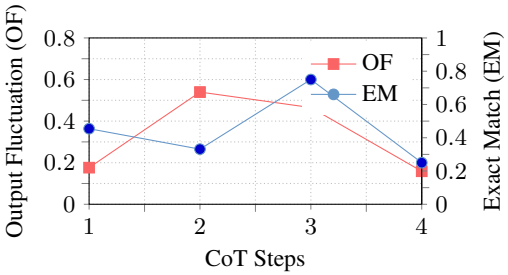
In this section, we list how performance varies with CoT steps under different models of MATH in Figure 6.

F.7 Perturbation with Embedding Norm under Different Settings

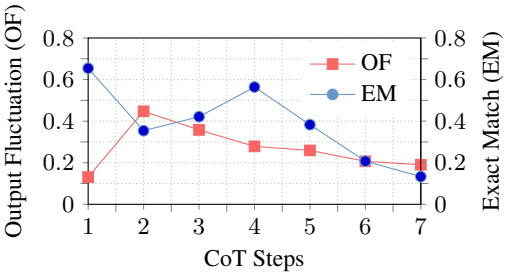
The variation of output perturbation with respect to embedding norm for all models on various datasets is illustrated in Figure 7 to Figure 9.



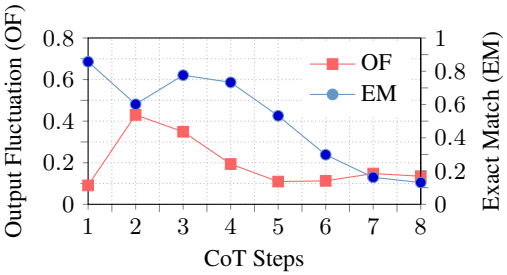
(a) MATH, Llama2-7b.



(b) MATH, Llama3.1-8b.



(c) MATH, Llama-R1-8b.



(d) MATH, Qwen3-8b.

Figure 6: EM and OF on MATH cross CoT steps with different models.

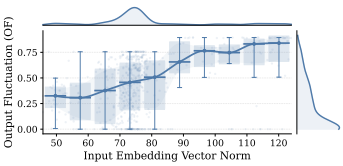


Figure 7: The change in output fluctuation with the norm of the input embedding vector across all experimental models on MATH. Each point denotes the result of one question, where X-axis denotes the input vector norm and Y-axis denotes OF of this question. The Pearson coefficient is 0.415.

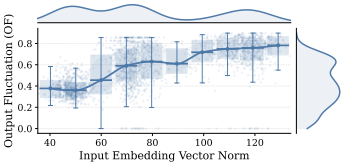


Figure 8: The change in output fluctuation with the norm of the input embedding vector across all experimental models on MMLU-Pro. Each point denotes the result of one question, where X-axis denotes the input vector norm and Y-axis denotes OF of this question. The Pearson coefficient is 0.634.

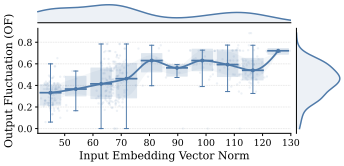


Figure 9: The change in output fluctuation with the norm of the input embedding vector across all experimental models on GPQA. Each point denotes the result of one question, where X-axis denotes the input vector norm and Y-axis denotes OF of this question. The Pearson coefficient is 0.541.