CAN LARGE LANGUAGE MODELS REASON ROBUSTLY WITH NOISY RATIONALES?

Zhanke Zhou[1] **Rong Tao**[1] **Jianing Zhu**[1] **Yiwen Luo**[2] **Zengmao Wang**[2] **Bo Han**[1†]
[1]TMLR Group, Hong Kong Baptist University [2]Wuhan University

ABSTRACT

This paper investigates an under-explored challenge in large language models (LLMs): chain-of-thought prompts with *noisy rationales*—irrelevant or inaccurate reasoning steps—despite advancements in in-context learning. We construct the NoRa dataset, specifically designed to evaluate LLMs' robustness to noisy rationales, based on which we reveal a widespread vulnerability among LLMs to such noise, with limited efficacy from existing robust methods.

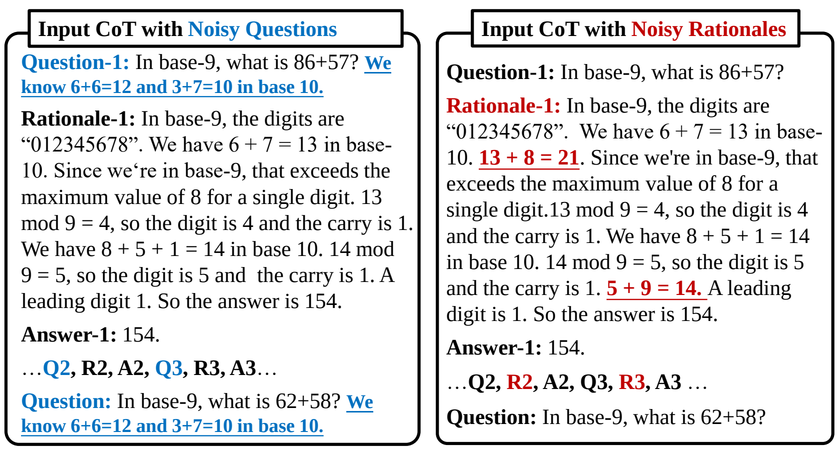

Figure 1: CoT with noisy questions or *noisy rationales (the new research problem)*. Each 3-shot input CoT includes three prompting examples (questions Q_1, Q_2, Q_3, rationales R_1, R_2, R_3, and answers A_1, A_2, A_3) and one test question.

1 INTRODUCTION

In-context learning (ICL) is a common approach in large language models (LLMs), enabling one to extrapolate from a few examples and adapt without fine-tuning Brown et al. (2020); Wei et al. (2022a); Dong et al. (2022). However, ICL's efficacy is tied to the quality and clarity of the input prompts, particularly in the prevailing chain-of-thought (CoT) strategy that provides rationales, i.e., intermediate reasoning steps to solve a question Wei et al. (2022b). Recent studies indicate that LLMs struggle with noisy questions: they are easily distracted by irrelevant context and exhibit instability with small input modifications Shi et al. (2023); Tian et al. (2023); Zheng & Saparov (2023).

Notably, this work shifts focus from the well-studied noisy questions (Noisy-Q) problem to the under-explored *noisy rationales* (Noisy-R) problem, wherein factually *inaccurate or irrelevant reasoning steps* are paired with valid question-answer prompts, as shown in Fig. 1. The emphasis on Noisy-R is due to its practical challenges, with examples drawn from diverse sources such as crowdsourced platforms, dialogue systems, and AI-generated data. LLMs' robustness against Noisy-R is unknown yet, urging us to conduct a comprehensive evaluation and devise the corresponding robust method. However, considering the lack of a desirable benchmark, a new dataset is necessary for the evaluation.

†Correspondence to Bo Han (bhanml@comp.hkbu.edu.hk).

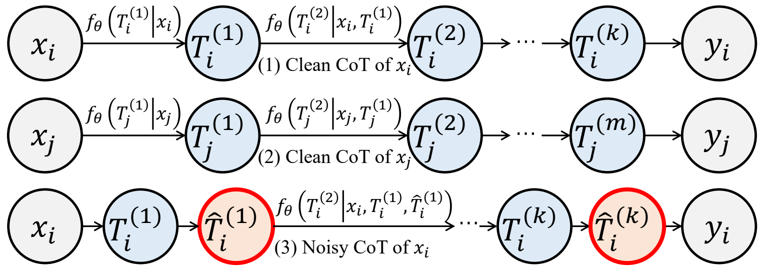

Figure 2: Chain modeling of the noisy rationale problem: For noisy chain (3) with question x_i, the rationale with clean thoughts $T_i^{(j)}$ and noisy ones $\hat{T}_i^{(j)}$ is to deduce answer y_i.

In this work, we construct the NoRa (**No**isy **Ra**tionales) dataset, a comprehensive testbed to evaluate the robust reasoning capability of LLMs with the Noisy-R problem across various reasoning domains. NoRa contains a total of 26391 questions, covering three types of reasoning tasks: mathematical, symbolic, and commonsense. Specifically, we uniformly formalize the noisy rationales by adding irrelevant or inaccurate thoughts, control the reasoning difficulty through different noise ratios, and guarantee overall prompting correctness without modifying the question or answer.

We evaluate various LLMs with NoRa and disclose that all of them are *intrinsically vulnerable* to noisy rationales. Therein, compared to the clean scenario, GPT-3.5 exhibits an average of 30.2% accuracy decrease with noisy rationales (see Appendix Fig. 3). Nonetheless, only limited improvements are achieved with existing robust methods based on the intrinsic denoising ability of LLMs. Hence, Noisy-R is much more challenging than Noisy-Q.

To the best of our knowledge, we are the *first* to investigate the problem of noisy rationales:

- We construct the NoRa for benchmarking LLMs' robustness against the noisy rationales (Sec. 2).
- We systematically evaluate LLMs with NoRa and reveal their unsatisfactory robustness (Sec. 3).

2 The NoRa Dataset

In this section, we construct the NoRa (**No**isy **Ra**tionales) dataset for benchmarking reasoning robustness with Noisy-R. NoRa contains 26391 questions and 5 subsets, covering mathematical, symbolic, and commonsense reasoning tasks, wherein ICL and CoT examples play an essential role.

2.1 Definition of Noisy Rationales

We start by formalizing the ICL and CoT demonstrations. Given a test question x_{test} and an LLM f_θ, one expects to get the correct answer y_{test} as $f_\theta(x_{\text{test}}) \mapsto y_{\text{test}}$. This zero-shot reasoning manner cannot guarantee reasoning effectiveness, especially when encountering unfamiliar contexts or scenarios. To boost this, the ICL techniques prompt the LLM with a few supporting examples $S_n = \{(x_i, y_i)\}_{i=1}^n$ collected in the current context, each composed of a question x_i and answer y_i, and then constructing the new input of x_{ICL} as

$$x_{\text{ICL}} = [S_n, x_{\text{test}}] = [x_1, y_1, \ldots, x_n, y_n,\ x_{\text{test}}] .$$

With S_n, the $f_\theta(x_{\text{ICL}}) \mapsto y_{\text{test}}$ is easier than $f_\theta(x_{\text{test}}) \mapsto y_{\text{test}}$. Then, the CoT further refines x_{ICL} by constructing the step-by-step *rationale* $\mathcal{T}_i$, consisting several thoughts $T_i^{(j)}$, i.e.,

$$x_{\text{CoT}} = [x_1, \mathcal{T}_1, y_1, \ldots, x_n, \mathcal{T}_n, y_n,\ x_{\text{test}}] ,$$
$$\text{where} \quad \mathcal{T}_i = \left[T_i^{(1)}, T_i^{(2)}, T_i^{(3)}, \ldots, T_i^{(k)}\right] . \tag{1}$$

However, as mentioned, the thoughts in CoT (Eqn. 1) can be noisy in practice. This noise can be attributed to (1) *irrelevant thoughts*, which are irrelevant but correct, or (2) *inaccurate thoughts*, which are relevant but factually wrong. Here, we uniformly formalize these two kinds of noise as

$$\hat{\mathcal{T}}_i = \left[T_i^{(1)}, \hat{T}_i^{(1)}, T_i^{(2)}, \hat{T}_i^{(2)}, \ldots, T_i^{(k)}, \hat{T}_i^{(k)}\right] , \tag{2}$$

where $\hat{T}_i^{(j)}$ represents a noisy thought (irrelevant or inaccurate) that is coherent with previous clean thought $T_i^{(j)}$ (relevant and correct) in Eqn. 1. In what follows, we elaborate on the definition and generation of these noisy thoughts.

Irrelevant thoughts refer to incorporating extraneous details unhelpful for solving the question, e.g., discussing the genetic overlap of siblings when the task is to deduce family roles in relationship reasoning. Redundant information may be introduced by the LLM's diverse response generation or by humans when clarifying concepts in examples Chandler & Sweller (1991); Zhao et al. (2023).

Inaccurate thoughts refer to factual errors in rationales that are common in mathematical calculation or transcription, e.g., "5+5=10" in base-9 calculation. The emergence of noise can be due to algorithmic limitations, errors in training data, misinterpretations of context or instructions, and logical fallacies Koo et al. (2023); Sambasivan et al. (2021).

Remark 2.1. Both types of noise only impact the finer details of the reasoning chain without affecting the correctness of question x_i and answer y_i. This distinction ensures that the reasoning based on the noisy demonstration is not fundamentally flawed; only the reasoning rationale $\hat{\mathcal{T}}_i$ is noisy.

2.2 Tasks and Statistics

The NoRa dataset covers the three reasoning tasks listed below. In noise generation, irrelevant thoughts, sourced from extraneous scientific or social facts, and inaccurate thoughts, arising from misguided reasoning, are both based on intermediate thoughts of Eqn. 1 (see examples in Tab. 8).

- **NoRa-Math.** This derives from the Base Calculation dataset Wu et al. (2023) for evaluating non-standard base arithmetic skills and features two addition tasks of *base-9* and *base-11*. The mastery of mathematical concepts and the rules of specific bases are the keys to solving these tasks.
- **NoRa-Symbolic.** We utilize the SCAN dataset Lake & Baroni (2018) here, which aims to transform natural language into symbolic, machine-understandable instructions. This transformation is learned from the prompting examples, comprising two subtasks: (1) *Equal*-length instructions are required for both the prompting examples S_n and the test question x_{test} and (2) *Longer*-length-instruction test question, prompted by the shorter-length-instruction examples.
- **NoRa-Commonsense.** This task is constructed with the CLUTRR dataset Sinha et al. (2019), which is geared towards family relation path reasoning, e.g., "who is aunt's sister's mother?" It requires the mastery and application of commonsense knowledge and cognitive skills for reasoning.

Noise Ratio. Given the noise ratio $\epsilon \in (0, 1)$, the expected number of added noisy thoughts for a k-length CoT demonstration is $\lfloor \epsilon \cdot k + 1/2 \rfloor$. Specifically, for an irrelevant thought $\hat{T}_i^{(j)}$ in j-th position of i-th example, a Bernoulli distribution Bern$(\epsilon) \in \{0, 1\}$ is adopted to indicate its binary existence.

Statistics. A categorization of task difficulties is provided as Easy, Medium, and Hard, with corresponding noise ratios of 0.3, 0.5, and 0.8. The detailed statistics are in Appendix 2. Regarding the number of thoughts in rationale, Math entails 8 thoughts, Symbolic varies from 2 to 12, and Commonsense requires 5 thoughts. Besides, a detailed introduction to NoRa is in Appendix B.2.

3 Evaluating LLMs on NoRa dataset

In this section, we provide a comprehensive evaluation of representative LLMs and robust methods on the newly constructed NoRa dataset. First, we introduce the basic evaluation setups. Second, we reveal the unreliable robustness of these LLMs and methods under noisy rationales in Sec. 3.2.

3.1 Evaluation Setups

To ensure a comprehensive assessment, we employ a variety of methods as baselines, encompassing the two traits of self-correction and self-consistency. (see Appendix A) Due to the space limit, we left the baseline details in Appendix C.

LLM Basis. We employ GPT-3.5-turbo-0613 (Floridi & Chiriatti, 2020) as our base LLM (denoted as Base) for the analyses presented in this study. In addition, we conduct evaluations on three supplementary models, including Gemini-Pro (Team et al., 2023), Llama2-70B (Touvron et al., 2023), and Mixtral-8x7B (Jiang et al., 2024). For all baselines, we consistently set the temperature parameter τ to default 1. We conduct evaluations on 300 questions for each task and repeat the reasoning 5 times for each question.

Evaluation Metric. Given a set of test question $\mathcal{Q} = \{(x_{\text{test}}, y_{\text{test}})\}$ and a set of CoT-prompting examples $\mathcal{P} = [x_1, \mathcal{T}_1, y_1, \ldots, x_n, \mathcal{T}_n, y_n]$, we define the accuracy of the denoising method $\mathcal{M}$ (with a specific LLM f_θ), namely,

Task	Method $\mathcal{M}$	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}})$	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{irrelevant}})$ Easy	Medium	Hard	Avg.	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{inaccurate}})$ Easy	Medium	Hard	Avg.
Math Base-9	Base	46.4	39.3	30.3	<u>26.6</u>	32.1	23.2	10.1	6.0	13.1
	w/ ISC (Huang et al., 2023a)	24.3	17.7	14.7	12.7	15.0	18.4	13.7	12.3	14.8
	w/ SP Xi et al. (2023)	26.2	25.5	25.5	21.9	24.3	20.0	<u>18.4</u>	**14.3**	17.6
	w/ SM (Robey et al., 2023)	42.0	32.7	25.7	22.0	26.8	27.0	**21.0**	<u>13.7</u>	<u>21.6</u>
	w/ SD (Zhang et al., 2023b)	<u>61.6</u>	<u>49.6</u>	**37.0**	23.7	<u>36.8</u>	<u>29.0</u>	12.7	7.0	16.2
	w/ SC Wang et al. (2023)	**62.3**	**52.0**	<u>36.6</u>	**38.6**	**42.4**	**41.7**	17.3	11.3	**23.4**
Math Base-11	Base	<u>23.9</u>	<u>19.1</u>	13.6	10.7	14.5	14.0	6.7	3.6	8.1
	w/ ISC (Huang et al., 2023a)	11.2	8.3	7.8	6.0	7.4	6.5	5.2	4.7	5.5
	w/ SP Xi et al. (2023)	20.7	17.5	**16.7**	<u>14.0</u>	<u>16.0</u>	<u>14.1</u>	**10.7**	**10.8**	**11.9**
	w/ SM (Robey et al., 2023)	16.3	12.0	6.0	5.7	7.9	12.0	<u>9.3</u>	<u>7.7</u>	9.7
	w/ SD (Zhang et al., 2023b)	17.9	12.3	12.0	13.3	12.5	17.0	8.7	5.3	10.3
	w/ SC Wang et al. (2023)	**33.7**	**25.3**	<u>16.3</u>	**15.0**	**18.9**	**19.7**	<u>9.3</u>	3.3	<u>10.8</u>
Symbolic Equal	Base	<u>32.7</u>	<u>28.1</u>	<u>25.1</u>	<u>23.0</u>	<u>25.4</u>	<u>29.1</u>	<u>26.1</u>	22.7	<u>26.0</u>
	w/ ISC (Huang et al., 2023a)	23.9	20.0	16.3	15.5	17.3	19.2	18.3	18.1	18.5
	w/ SP Xi et al. (2023)	23.2	23.0	22.6	22.7	22.8	23.7	22.5	<u>23.5</u>	23.2
	w/ SM (Robey et al., 2023)	25.0	20.7	19.7	16.7	19.0	21.0	20.3	20.0	20.4
	w/ SD (Zhang et al., 2023b)	9.9	10.1	10.9	10.3	10.4	10.1	10.9	10.4	10.5
	w/ SC Wang et al. (2023)	**35.3**	**31.0**	**28.3**	**27.0**	**28.8**	**33.3**	**30.7**	**26.0**	**30.0**
Symbolic Longer	Base	<u>9.2</u>	<u>6.3</u>	<u>7.2</u>	<u>6.0</u>	<u>6.5</u>	<u>7.0</u>	<u>6.8</u>	<u>6.0</u>	<u>6.6</u>
	w/ ISC (Huang et al., 2023a)	4.9	4.6	2.7	3.7	3.7	3.4	4.3	3.3	3.7
	w/ SP Xi et al. (2023)	5.1	4.3	4.1	3.9	4.1	4.9	4.0	4.5	4.5
	w/ SM (Robey et al., 2023)	1.7	0.7	0.7	1.3	1.0	1.3	0.7	0.3	0.8
	w/ SD (Zhang et al., 2023b)	0.1	0.1	0.1	0.2	0.1	0.1	0.3	0.0	0.1
	w/ SC Wang et al. (2023)	**13.0**	**7.7**	**9.0**	**6.3**	**7.7**	**8.0**	**8.0**	**8.7**	**8.2**
Commonsense	Base	45.7	44.3	42.3	41.4	42.7	36.7	33.4	28.3	32.8
	w/ ISC (Huang et al., 2023a)	21.8	24.3	22.5	21.4	22.7	23.3	26.5	24.0	24.6
	w/ SP Xi et al. (2023)	47.9	48.2	46.7	<u>48.1</u>	47.7	<u>49.6</u>	46.6	46.5	47.6
	w/ SM (Robey et al., 2023)	<u>53.3</u>	<u>50.3</u>	<u>50.0</u>	46.7	<u>49.0</u>	47.7	<u>49.0</u>	<u>49.3</u>	<u>48.7</u>
	w/ SD (Zhang et al., 2023b)	**54.0**	**58.3**	**57.3**	**57.7**	**57.8**	**57.0**	**58.3**	**53.7**	**56.3**
	w/ SC Wang et al. (2023)	52.0	46.3	45.0	44.7	45.3	44.7	44.7	38.0	42.5

Table 1: Evaluation results on NoRa dataset, quantified by the accuracy of reasoning on 3-shot prompting examples with clean, irrelevant, or inaccurate rationales. The **boldface** numbers mean the best results; <u>underlines</u> indicate the second-best.

$$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}) = \textstyle\sum_{(x_{\text{test}}, y_{\text{test}}) \in \mathcal{Q}} \mathbf{1}[\mathcal{M}(\mathcal{P}, x_{\text{test}}) = y_{\text{test}}] / |\mathcal{Q}|.$$

We report the results in percentage (%) with one decimal point. Therein, $\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}})$, $\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{irrelevant}})$, and $\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{inaccurate}})$ indicate accuracy with clean, irrelevant, and inaccurate rationales, respectively. When $\mathcal{P} = \emptyset$, then $\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \emptyset)$ represent the zero-shot result.

3.2 Unreliability Revealing with Noisy-R

We conduct the reasoning tasks on LLM with Noisy-R and summarize the results in Tab. 1. Overall, the base LLM with all the existing reasoning methods is severely affected by irrelevant or inaccurate noise, most showing a 15%-30% decrease with irrelevant noise and a more drastic 20%-80% decrease with inaccurate noise compared with clean rationales. While methods like SP and SD do exhibit resilience to noise on partial datasets, their performance remains inconsistent and is often declining. To further reveal the unreliability, we analyze the robust methods in the following two categories.

Observation 1 **Self-correction methods perform poorly on most tasks.**

Observation 2 **Self-consistency methods can improve reasoning instead of truly denoising.**

Observation 3 **Adjusting model temperature can help reasoning under Noisy-R.**

Observation 4 **Increasing prompting examples boosts noisy reasoning accuracy on most tasks.**

Observation 5 **Different LLMs generally suffer from Noisy-R.**

Detailed discussion on these observations can be found in Appendix. D.3.

4 Conclusion

In this work, we investigate the under-explored problem of noisy rationales in LLMs by introducing the NoRa dataset, which tests LLMs against irrelevant or inaccurate thoughts in question-answer scenarios. Our findings show LLMs' vulnerability to noisy rationales, inadequately mitigated by existing denoising methods.

The extension advocates for advancing LLMs by employing strategies, e.g., external knowledge bases with a retrieval-augmented framework, inductive reasoning to extract rules from noisy examples, and multi-modal data integration to enhance the LLMs' robustness and reasoning capabilities.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Paul Chandler and John Sweller. Cognitive load theory and the format of instruction. *Cognition and instruction*, 8(4):293–332, 1991.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. Quac: Question answering in context. *arXiv preprint arXiv:1808.07036*, 2018.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30:681–694, 2020.

C Daniel Freeman, Laura Culp, Aaron Parisi, Maxwell L Bileschi, Gamaleldin F Elsayed, Alex Rizkowsky, Isabelle Simpson, Alex Alemi, Azade Nova, Ben Adlam, et al. Frontier language models are not robust to adversarial arithmetic, or" what do i need to say so you agree 2+ 2= 5? *arXiv preprint arXiv:2311.07587*, 2023.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*, 2023a.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*, 2023b.

Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems. In *EMNLP*, 2017.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Seonmin Koo, Chanjun Park, Seolhwa Lee, Jaehyung Seo, Sugyeong Eo, Hyeonseok Moon, and Heuiseok Lim. Uncovering the risks and drawbacks associated with the use of synthetic data for grammatical error correction. *IEEE Access*, 2023.

Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*, 2018.

Fangyu Lei, Xiang Li, Yifan Wei, Shizhu He, Yiming Huang, Jun Zhao, and Kang Liu. S $\hat{3}$ hqa: A three-stage approach for multi-hop text-table hybrid question answering. *arXiv preprint arXiv:2305.11725*, 2023.

Richard E Mayer. *Thinking and problem solving: An introduction to human cognition and learning.* Scott, Foresman, 1977.

Lalchand Pandia and Allyson Ettinger. Sorting through the noise: Testing robustness of information processing in pre-trained language models. In *EMNLP*, 2021.

Alexander Robey, Eric Wong, Hamed Hassani, and George J Pappas. Smoothllm: Defending large language models against jailbreaking attacks. *arXiv preprint arXiv:2310.03684*, 2023.

Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. "everyone wants to do the model work, not the data work": Data cascades in high-stakes ai. In *CHI*, 2021.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *ICML*, 2023.

Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L Hamilton. Clutrr: A diagnostic benchmark for inductive reasoning from text. *arXiv preprint arXiv:1908.06177*, 2019.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Qingyuan Tian, Hanlun Zhu, Lei Wang, Yang Li, and Yunshi Lan. R3 prompting: Review, rephrase and resolve for chain-of-thought reasoning in large language models under noisy context prompting: Review, rephrase and resolve for chain-of-thought reasoning in large language models under noisy context. In *EMNLP*, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Gladys Tyen, Hassan Mansoor, Peter Chen, Tony Mak, and Victor Cărbune. Llms cannot find reasoning errors, but can correct them! *arXiv preprint arXiv:2311.08516*, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022b.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. *arXiv preprint arXiv:2307.02477*, 2023.

Zhiheng Xi, Senjie Jin, Yuhao Zhou, Rui Zheng, Songyang Gao, Tao Gui, Qi Zhang, and Xuanjing Huang. Self-polish: Enhance reasoning in large language models via problem refinement. *arXiv preprint arXiv:2305.14497*, 2023.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*, 2023.

Jiehang Zeng, Jianhan Xu, Xiaoqing Zheng, and Xuanjing Huang. Certified robustness to text adversarial attacks by randomized [mask]. *Computational Linguistics*, 49(2):395–427, 2023.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*, 2023a.

Zhen Zhang, Guanhua Zhang, Bairu Hou, Wenqi Fan, Qing Li, Sijia Liu, Yang Zhang, and Shiyu Chang. Certified robustness for large language models with self-denoising. *arXiv preprint arXiv:2307.07171*, 2023b.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. In *ICLR*, 2023c.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 2023.

Hongyi Zheng and Abulhair Saparov. Noisy exemplars make large language models more robust: A domain-agnostic behavioral analysis. In *EMNLP*, 2023.

APPENDIX

A Preliminaries **9**

B Benchmark **9**

B.1 A Further Discussion on Noisy Rationales in CoT Demonstrations 9

B.2 Noise Generation 12

C Implementation Details **13**

D Full Experiments **14**

D.1 Detailed Setups of the Experiments 14

D.2 Full Quantitative Results 14

D.3 Observations 15

E Full Examples of the NoRa Dataset **17**

A PRELIMINARIES

In this section, we systematically review the techniques of in-context learning, with a focus on the imperfect scenario.

In-context learning (ICL) Brown et al. (2020); Wei et al. (2022a) is a prevalent method employed in LLMs, where the generation of responses is contingent upon the immediate contextual discourse. Its principal advantage is its capacity to generalize from a few input-output examples and adapt to various applications without requiring the time-consuming fine-tuning of the model's parameters. As a crucial strategy in ICL, prompt engineering crafts effective prompts that guide the model's reasoning procedure. Notably, the Chain-of-Thought (CoT) Wei et al. (2022b) encourages the LLM to break down a complex question into intermediate steps by providing a few examples with rationales for prompting.

Limitations. Though effective, ICL suffers from the disadvantages of susceptibility to selected demonstrations and the intricacy of generating these demonstrations, where the ultimate performance is closely tied to the prompts' quality and clarity. Recent investigations on *noisy questions* have shown that (i) LLMs can be distracted by irrelevant or adversarial context, as they are designed to pay close attention to the context provided in the prompt Jia & Liang (2017); Pandia & Ettinger (2021); Shi et al. (2023); Tian et al. (2023) and (ii) the LLM reasoning is unstable, namely, small modifications to the prompt could potentially cause large variations in the model's output Zhang et al. (2023b); Zheng & Saparov (2023). Besides, another line of *noisy answers* Lei et al. (2023); Freeman et al. (2023) justifies the feasibility of misleading the LLM to agree factual errors such as "1+1=3".

Countermeasures. To combat these vulnerabilities, two traits are desirable with the LLM's intrinsic robustness, i.e.,

- *Self-correction*, wherein LLMs attempt to correct their initial responses based solely on their inherent capabilities without external feedback, e.g., by refining prompts through iterative corrections of responses or question trajectories Yang et al. (2023); Xi et al. (2023). Although LLMs can learn to ignore irrelevant information by examples or instructions Shi et al. (2023), they are proved to be still struggling to correct their responses without external feedback, and at times, their performance might even degrade after self-correction Huang et al. (2023a); Tyen et al. (2023).
- *Self-consistency*, on the other hand, aims to obtain a consistent answer against input perturbations. This is achieved by generating multiple samples via randomized smoothing on input questions Zeng et al. (2023) or diverse paths within the reasoning procedure Wang et al. (2023) with answer aggregation. Although effective for Noisy-Q, this strategy can be quite expensive because of its repeated reasoning. Besides, it cannot explicitly rectify questions or rationales.

Noisy-R problem is our focus (instead of Noisy-Q), where prompting examples are with *clean* question-answer pairs and *noisy* rationales. Here, the Noisy-R mainly originates from (1) the inherent imperfections, inconsistencies, and inaccuracy of humans' cognitive processes Mayer (1977); Choi et al. (2018) and (2) the diversity, unpredictability, and hallucination of the LLMs' generative mechanisms Zhang et al. (2023c); Huang et al. (2023b); Zhang et al. (2023a).

B BENCHMARK

B.1 A FURTHER DISCUSSION ON NOISY RATIONALES IN COT DEMONSTRATIONS

In this part, we further clarify the background and settings of the Noisy-R problem investigated in our work, i.e., Noisy Rationales in CoT Demonstrations. Having established the presence of irrelevant and inaccurate thoughts as primary noise types, we now turn to their origins and impact within human-model interactions.

Irrelevant and inaccurate thoughts can be generated by both the model and humans. On the one hand, irrelevant or inaccurate thoughts are likely to appear within the rationales of LLMs' answers. These rationales, drawn from historical dialogues between users and the model, can create a noisy context for new conversations. On the other hand, human demonstrators, to enhance the reasoning capabilities of language models, actively provide in-context demonstrations. However, these demonstrations, whether crafted by humans or sourced from datasets, can also contain noisy rationales.

Irrelevant thoughts - model perspective: LLMs tend to explain the concept of terms during reasoning. For instance, if you ask GPT-4 to debug an error related to the 'concurrent' package, it tends to start by explaining what the 'concurrent' package is rather than immediately addressing the debug request. Such explanations, while informative, may introduce irrelevant noise into the conversation. Here are some key reasons for their emergence:

- **Cognitive bias of models**: When addressing complex queries, LLMs tend to include explanations for terms or concepts mentioned in the prompt that are unnecessary for solving the specific problem presented, as a kind of irrelevant thought. This behavior exhibits a form of cognitive bias where the model, unable to assess the inquirer's level of understanding, leads to explicating background information. This is comparable to a lecturer explaining the basics of a subject to a class without first assessing their students' existing knowledge, potentially leading to irrelevant elaborations.
- **Lack of precise contextual understanding**: Despite LLMs' proficiency in processing language and recognizing patterns, they cannot always fully grasp the precise context or specific requirements of a problem. This shortfall can result in the production of thoughts that, although seemingly related, do not directly contribute to resolving the question at hand. Their responses might align more with the broader theme of the discussion rather than the specific, critical details needed for a precise solution.
- **User query ambiguity of the dialogue mechanism**: Ambiguities in user queries can stem from the use of vague or multifaceted language, prompting the model to generate a wide array of responses. This situation is similar to a search engine returning a variety of results for a query that lacks specificity. The model, attempting to cover all potential meanings of the query, may produce responses that contain thoughts unrelated to the user's actual intent. For instance, if someone asks, "What is force calculation?", the model might provide information on both Newtonian mechanics and quantum mechanics. These responses, encompassing a broad range of topics, could influence the answers to subsequent physics questions, leading to a continuation of the ambiguity and further complicating the conversation.
- **Progressive disclosure bias of the dialogue mechanism**: When engaging with LLMs, individuals often initiate the dialogue by describing simpler concepts and progressively work their way toward articulating the complex question at hand because of human limitations in language or comprehension abilities, which prevent a person from presenting the entire complexity of an issue in one go. This step-by-step approach, while natural for humans who struggle to directly convey intricate problems, can introduce extraneous content that contributes to noise within the model's contextual reasoning. As the conversation builds, the LLM will factor in these initial, possibly tangential, explanations into its understanding of the context, potentially leading to a dilution of the focus necessary for solving the specific issue. This phenomenon reflects a human cognitive strategy in communication that may not be optimally aligned with the operational mechanisms of LLMs for efficient problem-solving.

Irrelevant thoughts - human perspective: When a human is actively demonstrating CoT reasoning, the introduction of irrelevant thoughts could be due to a variety of reasons:

- **Cognitive overload of humans**: Humans may introduce irrelevant information when they are trying to process too much information at once, which can lead to a loss of focus and the inclusion of tangential thoughts. For instance, a programmer is struggling with a bug in a complex piece of software and asks an LLM for help. To provide context, the programmer starts explaining the issue with a CoT rationale approach, intending to walk the LLM through their thought process. However, due to the complexity of the code and the stress of finding the bug, the programmer gets sidetracked. They include unnecessary details about the different error messages encountered in the past, unrelated functions in the code, and general thoughts on software development.
- **Associative thinking of humans**: Humans naturally think in an associative manner, where one thought may lead to another that is only loosely related to the task at hand. This can result in straying from the main point during a CoT explanation. For example, while a programmer is outlining the steps to diagnose a software issue for an LLM, they might recall a similar problem they encountered on a different project. This memory could lead them to mention troubleshooting strategies, tools, or anecdotes from that past experience, which, although related to the broader theme of problem-solving, do not directly contribute to resolving the current issue.

- **Irrelevant content in datasets**: In the future, it is likely that companies or professional organizations will increasingly utilize databases to assemble CoT prompts. However, these databases, whether privately maintained or publicly accessible, can contain irrelevant reasoning processes. This is especially true for databases sourced from crowdsourcing platforms or open forums, where the information is contributed by a diverse set of individuals with varying levels of expertise and focus. When these datasets are used to provide in-context information for CoT reasoning, the noise can originate from the inclusion of off-topic discussions, personal opinions, or overly verbose explanations that do not directly address the problem at hand. Such noise can be inadvertently introduced into the CoT process when humans, drawing from these databases, provide explanations that contain unnecessary or tangential information.

Similarly, we analyze the two sources of inaccurate thoughts as follows.

Inaccurate thought - model perspective: For models, LLMs may produce erroneous thoughts during the reasoning process, especially when dealing with complex problems. For example, when tackling a base-9 math problem without prior examples (zero-shot), GPT-3.5 may generate some inaccurate reasoning steps. The former dialogue will become inaccurate and noisy in the context of subsequent dialogues. Here are some key reasons for their emergence:

- **Outdated or incomplete training data of the model**: Language models are built upon datasets that may not be current or fully comprehensive. When faced with problems that require up-to-date knowledge or complete understanding, which are absent in their training data, models may rely on outdated or incomplete information, resulting in inaccurate outputs. For example, in the field of medicine, if new research suggests a change in treatment protocol after the model's last update, it wouldn't be able to advise on or reason with the new information.
- **Adaptation to novel reasoning contexts of the model**: New challenges may require models to reason within contexts that slightly or significantly differ from their training data. For instance, a model extensively trained on base-10 arithmetic might struggle with a base-9 math problem because it requires a shift in the underlying numerical framework. This kind of scenario demands on-the-fly adaptation to a novel reasoning context, which can lead to generating thoughts that do not accurately apply the learned principles from the base-10 system to the newly introduced base-9 system.
- **Misinterpretation of complex subjects of dialogue mechanism**: Users often fail to clearly articulate their complete requirements at the outset of an inquiry, leading to LLMs generating misunderstandings and inaccurate thoughts that do not align with user expectations. The process of correcting these thoughts is inherently a reasoning process laden with noisy contexts. As users provide feedback to refine the model's output, the iterative nature of this interaction can introduce additional inaccuracies as the model attempts to reconcile the new information with the previously misunderstood context.

Inaccurate thought - human perspective: Inaccurate thoughts in CoT can stem from the information provided by humans, whether it is self-made on the spot or sourced from a database for in-context learning by LLMs. These CoT demos can include inaccurate noise due to various factors:

- **Personal knowledge limitations of human**: Individuals may possess incomplete or outdated knowledge on a given subject, leading to the provision of incorrect information when creating a CoT. For instance, a person without expertise in mathematics might attempt to construct a CoT for a complex math problem and inadvertently introduce incorrect steps or conclusions. Their understanding may be based on heuristics or educational background that hasn't been updated to reflect more recent methodologies or discoveries in the field.
- **Cognitive biases of human**: Human reasoning can be influenced by a range of cognitive biases, such as confirmation bias, where an individual tends to search for, interpret, and remember information in a way that confirms their preconceptions, neglecting contrary information. Or the oversimplification of complex issues might lead to inaccurate reasoning steps within a CoT. These biases can skew the logic flow and result in conclusions that do not hold up under scrutiny or are based on flawed premises.
- **Data quality issues of database**: The databases that humans rely on for creating CoTs might contain errors or biases introduced during data collection and processing. If this

flawed data is used for in-context learning by LLMs, it can impart incorrect patterns of thought or factual inaccuracies. For example, a dataset with biased sampling methods might lead to generalizations that do not accurately represent the broader population or situation.

- **Contextual misplacement of database**: Information from databases may be stripped of its original context, leading to misinterpretation when reused. When humans include such decontextualized information in a CoT, they might not properly align it with the new context, introducing misunderstandings or inaccuracies. This is particularly problematic in nuanced fields where context heavily influences the meaning and applicability of information, such as legal precedents or cultural studies.

Given the convenience and adaptability of CoT reasoning, broader adoption in LLM applications is expected in the future. This structured approach enables LLMs to break down complex problems and explain their reasoning in a way that resembles how humans think, proving essential for sophisticated problem-solving and decision-making. Nonetheless, we are bound to face the noisy reasoning challenges inherent in CoT, stemming from both model-generated and human-contributed contexts, as mentioned above.

To address these challenges, we must focus on continuously improving training methods, keeping models updated with the latest information, enhancing their ability to parse context and ambiguity, and refining algorithms to diminish biases and logical inaccuracies.

B.2 Noise Generation

As detailed in Sec. 2, we have introduced both irrelevant and inaccurate noises into our dataset as insertions. These insertional noises are carefully integrated into the rationales, ensuring they neither modify the existing reasoning pathways nor affect the final answers. Furthermore, they are purposefully crafted based on the intermediate steps of the reasoning sequence, which guarantees that each piece of noise is contextually related to the stage of reasoning it accompanies. To maintain consistency, we treat each sentence in the reasoning sequence as a single thought; accordingly, each noise we introduce is also fashioned as only one sentence. For irrelevant noise, we draw from unrelated scientific or social facts. Alternatively, inaccurate noise involves the extension of an intermediate thought, which is redundant and incorrect.

- **NoRa-Math.** In NoRa-Math problems, an intermediate reasoning result typically manifests as a numerical value. For instance, from the equation "We have 4 + 2 = 6 in base-10. ", we take the derived number 6 and craft a sentence of noise to follow this particular reasoning step. Take the numeral "6"; we might introduce an unrelated fact such as "According to the Standard Model of physics, there are six types of quarks, the fundamental constituents of matter." For inaccurate thoughts, we generate a decimal addition related to 6, such as "6 + 5 = 11. ", which is inaccurate in base-9 representation.
- **NoRa-Symbolic.** NoRa-Symbolic problems are handled similarly; here, intermediate results are individual elements of a navigation instruction. We insert a sentence of noise that aligns with these specific components. For example, if the element is the directive "right," we fabricate a direction-related noise. One irrelevant thought can be "Turning right in countries that drive on the right side of the road typically does not intersect with oncoming traffic.". For inaccurate thoughts, to ensure that the noise does not contradict the previous reasoning, we select another related instruction on the same term domain. For example, when explaining the instruction "right," we might generate noise related to the instruction "left", such as "left means I_TURN_RIGHT.", which is incorrect and does not contradict the previous thought, and this thought itself is redundant.
- **NoRa-Commonsense.** When it comes to NoRa-Commonsense problems, we introduce noise that mirrors the relationships discerned during the reasoning process. For example, after establishing that "mother's sister is aunt," we craft a noise sentence associated with the concept of "aunt." An irrelevant noise insertion could be: "Aunts often play pivotal roles in the social development of primates, akin to their influence in human societies." To generate inaccurate thoughts, we might produce a statement like "an aunt's mother is also an aunt," which is logically inaccurate.

Table 2 shows detailed statistics of NoRa dataset, including the average number of thoughts per shot.

Difficulty	Noise Ratio	#total thoughts (#noisy thoughts) of prompting rationales (Avg.) Math Base-9	Math Base-11	Sym. Equal	Sym. Longer	Com.
Easy	0.3	10 (2)	10 (2)	11.5 (2.7)	11.0 (2.5)	7 (2)
Medium	0.5	12 (4)	12 (4)	13.3 (4.5)	12.7 (4.2)	8 (3)
Hard	0.8	14 (6)	14 (6)	16.0 (7.1)	15.2 (6.8)	9 (4)
#questions		4024	9269	4182	3920	4996

Table 2: Statistics of NoRa dataset.

In addition to inserting a fixed number of noisy thoughts per chain of thought shot, as detailed in the main text, we further introduce variability in noise addition by randomly inserting noise following each thought. This randomness follows a Bernoulli distribution $\text{Bern}(\epsilon) \in 0, 1$, where noise is added after a thought only if the Bernoulli trial results in 1. Consequently, while the fixed number approach guarantees a set amount of noise within a CoT shot, the random addition allows for the possibility of varying amounts of noise in each shot, dictated by the probability parameter ϵ.

C Implementation Details

In this part, we detail the implementation details of the baselines.

Self-correction Methods.

- **Intrinsic self-correction (ISC)** Huang et al. (2023a) asks the model to endeavor to rectify its initial responses based solely on its inherent capabilities. We employ the prompts from the paper, instructing LLMs to review and revise their answers to NoRa tasks with "Review your previous answer and find problems with your answer," followed by "Based on the problems you found, improve your answer. Please reiterate your answer.
- **Self-polish (SP)** Xi et al. (2023) teaches the model to eliminate noisy information, rearrange the logic structure, and organize local conditions into new ones in parallel. We implement this method by (1) prompting LLMs to individually refine each noisy CoT demonstration without additional information, repeating the process three times, and (2) combining these rephrased demos to form the context for the task reasoning.

Self-consistency Methods.

- **SmoothLLM (SM)** Robey et al. (2023) enhances robustness by injecting perturbations into the prompts and utilizing self-consistency to mitigate these effects. We apply the described disturbance methods to noisy rationale demonstrations and feed them into LLMs for reasoning tasks. This process is repeated five times, with the most common answer across iterations selected as the voted answer.
- **Selfdenoise (SD)** Zhang et al. (2023b) improves LLM robustness by preprocessing prompts with random masks; the LLMs then work to reconstruct the masked content, reducing noise and aiding incoherent reasoning. Our implementation involves (1) applying the masking method to the noisy rationales on each shot, (2) prompting the LLMs to infer and fill the mask sections of each demonstration, and (3) using the reconstructed CoT demonstrations for task reasoning. This process is also repeated five times, and the most common answer is selected.
- **Self-Consistency(SC)** Wang et al. (2023) boosts reasoning performance by sampling multiple outputs and conducting majority voting without engaging in any input processing. To apply this method, we run the same task 5 times and vote for the maximum number of the same answers.

D Full Experiments

D.1 Detailed Setups of the Experiments

We employ GPT-3.5-turbo-0613 (Floridi & Chiriatti, 2020) as our base LLM (denoted as Base) for the analyses presented in this study. In addition, we conduct evaluations on three supplementary models, including Gemini-Pro (Team et al., 2023), Llama2-70B (Touvron et al., 2023), and Mixtral-8x7B (Jiang et al., 2024). While evaluating baseline methods on various, we consistently keep the temperature parameter τ and the top-p setting at their default value of 1, along with all other hyperparameters of models set to defaults. We conduct experiments on the first 300 questions for each task and repeat reasoning 5 times for each question.

When we need LLMs to perform a reasoning task for a prompt repeatedly, using the OpenAI API, we set the "n" parameter to N to obtain N different responses in a single query. If we're not using the OpenAI API, we would instead make N separate queries to get the results. We assume all CoT experiments with clean rationales or noisy rationales are conducted in a 3-shot setting unless specified otherwise. Furthermore, all CoT examples are constructed by randomly drawing from all available questions, except for Symbolic-Longer, which has predefined demonstrating and testing scopes.

D.2 Full Quantitative Results

In this section, we supplement our discussion with additional experimental data.

Different LLMs. Fig. 3 displays the result of the GPT-3.5-Turbo model's evaluation on the NoRa Dataset. It corresponds to base model results in Tab. 1. We have also conducted comprehensive experiments on the Gemini model to evaluate various types of noise. Fig. 4 shows the full performance evaluation of Gemini on the NoRa dataset.

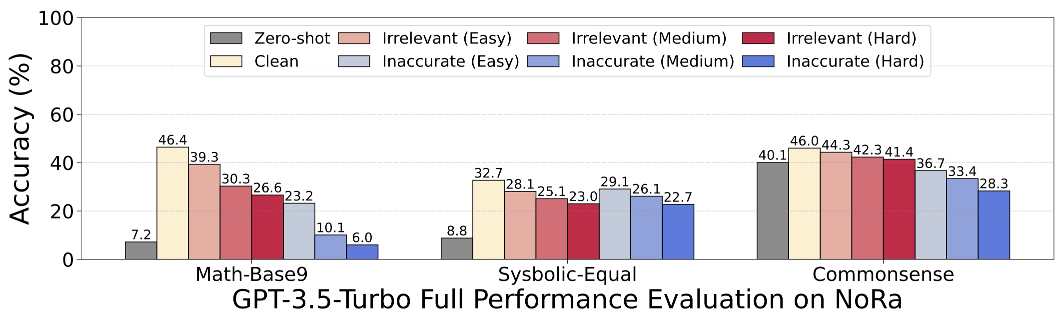

Figure 3: **GPT-3.5-Turbo** Full Performance Evaluation on the NoRa Dataset.

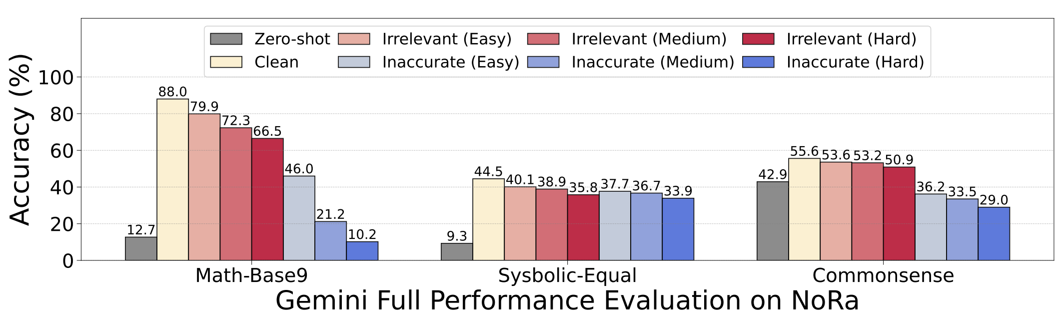

Figure 4: **Gemini** Full Performance Evaluation on the NoRa Dataset.

Computation Cost. Tab. 3 shows the cost of different baselines.

The Normalized Difference in Accuracy (NDA) Metric. we propose a new evaluation score to quantify the efficacy of $\mathcal{M}$ under the noisy scenario, namely, Normalized Difference in Accuracy

Task	Method $\mathcal{M}$	#Tokens per clean sample	#Tokens per irrelevant sample				#Tokens in inaccurate sample			
			Easy	Medium	Hard	Avg.	Easy	Medium	Hard	Avg.
Math Base-9	Base	702.9	858.2	1027.9	1209.5	1031.9	763.9	826.5	893.8	828.1
	w/ ISC (Huang et al., 2023a)	1375.9	1635.8	1922.6	2224.8	1927.7	1522.1	1633.2	1735.9	1630.4
	w/ SP Xi et al. (2023)	1664.9	1739.6	1815.9	1900.1	1818.5	1742.4	1793.2	1824.4	1786.7
	w/ SM (Robey et al., 2023)	3872.4	5029.2	5434.0	6197.6	5553.6	4226.8	4530.7	4831.1	4529.5
	w/ SD (Zhang et al., 2023b)	5882.6	7365.2	9453.7	11123.8	9314.2	6033.3	10819.6	11625.9	9492.9
	w/ SC Wang et al. (2023)	2002.2	2317.4	2783.7	3362.3	2821.1	2089.0	2228.4	2416.3	2244.6
Math Base-11	Base	710.1	877.3	1057.9	1226.0	1053.7	773.1	834.4	897.3	834.9
	w/ ISC (Huang et al., 2023a)	2498.2	2988.3	3531.7	4010.4	3510.1	2687.9	2872.6	3026.4	2862.3
	w/ SP Xi et al. (2023)	2837.6	2972.3	3136.5	3199.0	3102.6	2971.2	3055.2	3119.4	3048.6
	w/ SM (Robey et al., 2023)	3945.6	5170.4	6406.1	7480.0	6352.2	4272.4	4567.2	4847.2	4562.3
	w/ SD (Zhang et al., 2023b)	9578.2	11613.6	13848.5	15856.9	13773.0	10094.5	10842.7	11631.1	10856.1
	w/ SC Wang et al. (2023)	2054.5	2385.7	2872.0	3405.4	2887.7	2161.0	2289.5	2433.8	2294.8
Symbolic Equal	Base	1623.7	1834.9	1991.0	2221.4	2015.8	1765.3	1868.8	2022.4	1885.5
	w/ ISC (Huang et al., 2023a)	5246.0	5868.1	6345.2	7007.1	6406.8	5691.5	5998.2	6449.5	6046.4
	w/ SP Xi et al. (2023)	5343.0	5539.8	5661.8	5858.0	5686.5	5526.3	5635.7	5800.3	5654.1
	w/ SM (Robey et al., 2023)	10272.9	11936.8	13156.2	14860.2	13317.7	11315.3	12029.4	13093.6	12146.1
	w/ SD (Zhang et al., 2023b)	6267.1	6782.5	6965.8	7004.4	6917.6	6763.1	7006.6	7135.4	6968.4
	w/ SC Wang et al. (2023)	4089.3	4429.2	4724.0	5267.5	4806.9	4262.6	4404.0	4691.1	4452.6
Symbolic Longer	Base	1687.1	1826.1	1862.4	2017.9	1902.1	1802.9	1832.4	1952.3	1862.5
	w/ ISC (Huang et al., 2023a)	5601.1	5957.7	6052.7	6466.0	6158.8	5932.0	6073.3	6382.8	6129.4
	w/ SP Xi et al. (2023)	5687.9	5765.3	5815.2	5893.1	5824.5	5907.2	5911.1	6051.5	5956.6
	w/ SM (Robey et al., 2023)	10487.3	11492.5	11838.6	12922.2	12084.4	11277.9	11566.7	12346.6	11730.4
	w/ SD (Zhang et al., 2023b)	3087.1	3194.6	3277.5	3351.7	3274.6	3214.7	3416.0	3417.6	3349.4
	w/ SC Wang et al. (2023)	4934.9	5153.9	5175.4	5441.8	5257.0	5146.2	5146.8	5524.5	5272.5
Commonsense	Base	553.2	712.0	789.5	867.6	789.7	605.9	635.6	669.9	637.1
	w/ ISC (Huang et al., 2023a)	1873.2	2334.8	2559.2	2786.7	2560.2	2021.0	2110.4	2207.5	2113.0
	w/ SP Xi et al. (2023)	4728.7	5502.0	5907.6	6308.6	5906.1	4998.5	5131.9	5265.1	5131.8
	w/ SM (Robey et al., 2023)	3532.2	4755.0	5377.2	5999.0	5377.1	3943.7	4153.7	4352.8	4150.1
	w/ SD (Zhang et al., 2023b)	5007.8	5882.5	6357.3	6816.1	6352.0	5285.0	5430.4	5561.9	5425.8
	w/ SC Wang et al. (2023)	853.5	1125.7	1252.5	1382.0	1253.4	900.2	941.2	1007.3	949.6

Table 3: Computation cost (#tokens) of all methods.

(NDA),

$$\text{NDA}(\mathcal{M}, \mathcal{Q}, \mathcal{P}) = \frac{\text{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{noisy}}) - \text{Acc}(\mathcal{M}, \mathcal{Q}, \emptyset)}{\text{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}}) - \text{Acc}(\mathcal{M};, \mathcal{Q}, \emptyset)}, \tag{3}$$

where $\text{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}})$, $\text{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{noisy}})$ and $\text{Acc}(\mathcal{M};, \mathcal{Q}, \emptyset)$ represent the accuracy of method $\mathcal{M}$ with clean rationales, noisy rationales (irrelevant or inaccurate), and without CoT demos. Fig. 5 is the illustration of the NDA metric.

Tab. 4 presents a comparison of the accuracy and NDA across all methods. A negative value in NDA indicates that the accuracy of noisy rationales falls below that of 0-CoT.

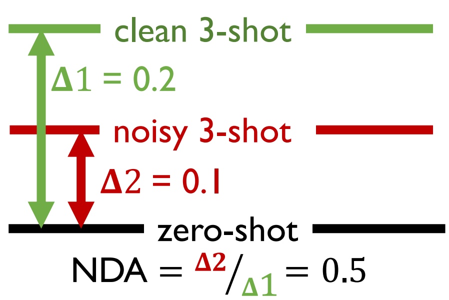

Figure 5: Illustration of the NDA metric

D.3 Observations

Observation 1. **Self-correction methods perform poorly on most tasks.** Therein, ISC (Huang et al., 2023a) and SP Xi et al. (2023) rely on the inherent capabilities of LLMs to enhance the quality of generated responses. However, in the absence of external feedback, the model's self-correction ability

Dataset	Method $\mathcal{M}$	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \emptyset)$	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}})$	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{Irrelevant}})$ (NDA) Easy	Medium	Hard	Avg.	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{Inaccurate}})$ (NDA) Easy	Medium	Hard	Avg.
Math Base-9	Base	7.2	46.4	39.3 (81.9)	30.3 (58.9)	26.6 (49.5)	32.1 (63.5)	23.2 (40.8)	10.1 (7.4)	6.0 (-3.0)	13.1 (15.1)
	w/ ISC	8.7	24.3	17.7 (57.7)	14.7 (38.5)	12.7 (25.6)	15.0 (40.4)	18.4 (62.2)	13.7 (32.1)	12.3 (23.1)	14.8 (39.1)
	w/ SP	7.2	26.2	25.5 (**96.3**)	25.5 (**96.3**)	21.9 (**77.4**)	24.3 (**90.0**)	20.0 (**67.4**)	18.4 (**58.9**)	**14.3** (**37.4**)	17.6 (**54.7**)
	w/ SM	12.3	42.0	32.7 (68.7)	25.7 (45.1)	22.0 (32.7)	26.8 (48.8)	27.0 (49.5)	**21.0** (29.3)	13.7 (4.7)	21.6 (31.3)
	w/ SD	12.3	61.6	49.6 (75.7)	**37.0** (50.1)	23.7 (23.1)	36.8 (49.7)	29.0 (33.9)	12.7 (0.8)	7.0 (-10.8)	16.2 (7.9)
	w/ SC	12.3	**62.3**	**52.0** (79.4)	36.6 (48.6)	**38.6** (52.6)	**42.4** (60.2)	**41.7** (58.8)	17.3 (10.0)	11.3 (-2.0)	**23.4** (22.2)
Math Base-11	Base	5.5	23.9	19.1 (73.9)	13.6 (44.0)	10.7 (28.3)	14.5 (48.9)	14.0 (46.2)	6.7 (6.5)	3.6 (-10.3)	8.1 (14.1)
	w/ ISC	7.4	11.2	8.3 (23.7)	7.8 (10.5)	6.0 (-36.8)	7.4 (0.0)	6.5 (-23.7)	5.2 (-57.9)	4.7 (-71.1)	5.5 (-50.0)
	w/ SP	5.5	20.7	17.5 (**78.9**)	**16.7** (**73.7**)	14.0 (**55.9**)	16.0 (**69.1**)	14.1 (56.6)	**10.7** (**34.2**)	**10.8** (**34.9**)	**11.9** (**42.1**)
	w/ SM	8.0	16.3	12.0 (48.2)	6.0 (-24.1)	5.7 (-27.7)	7.9 (-1.2)	12.0 (48.2)	9.3 (15.7)	7.7 (-3.6)	9.7 (20.5)
	w/ SD	8.0	17.9	12.3 (43.4)	12.0 (40.4)	13.3 (53.5)	12.5 (45.5)	17.0 (**90.9**)	8.7 (7.1)	5.3 (-27.3)	10.3 (23.2)
	w/ SC	8.0	**33.7**	**25.3** (67.3)	16.3 (32.3)	**15.0** (27.2)	**18.9** (42.4)	**19.7** (45.5)	9.3 (5.1)	3.3 (-18.3)	10.8 (10.9)
Symbolic Equal	Base	8.8	32.7	28.1 (80.8)	25.1 (68.2)	23.0 (59.4)	25.4 (69.5)	29.1 (84.9)	26.1 (72.4)	22.7 (58.2)	26.0 (72.0)
	w/ ISC	5.7	23.9	20.0 (78.6)	16.3 (58.2)	15.5 (53.8)	17.3 (63.7)	19.2 (74.2)	18.3 (69.2)	18.1 (68.1)	18.5 (70.3)
	w/ SP	8.8	23.2	23.0 (**98.6**)	22.6 (**95.8**)	22.7 (**96.5**)	22.8 (**97.2**)	23.7 (**103.5**)	22.5 (**95.1**)	23.5 (**102.1**)	23.2 (**100.0**)
	w/ SM	9.7	25.0	20.7 (71.9)	19.7 (65.4)	16.7 (45.8)	19.0 (60.8)	21.0 (73.9)	20.3 (69.3)	20.0 (67.3)	20.4 (69.9)
	w/ SD	9.7	9.9	10.1 (—)	10.9 (—)	10.3 (—)	10.4 (—)	10.1 (—)	10.9 (—)	10.4 (—)	10.5 (—)
	w/ SC	9.7	**35.3**	**31.0** (83.2)	**28.3** (72.7)	**27.0** (67.6)	**28.8** (74.6)	**33.3** (92.2)	**30.7** (82.0)	**26.0** (63.7)	**30.0** (79.3)
Symbolic Longer	Base	0.0	9.2	6.3 (68.5)	7.2 (78.3)	6.0 (65.2)	6.5 (70.7)	7.0 (76.1)	6.8 (73.9)	6.0 (65.2)	6.6 (71.7)
	w/ ISC	0.1	4.9	4.6 (**93.7**)	2.7 (54.2)	3.7 (75.0)	3.7 (75.0)	3.4 (68.7)	4.3 (**87.5**)	3.3 (66.7)	3.7 (75.0)
	w/ SP	0.0	5.1	4.3 (84.3)	4.1 (**80.4**)	3.9 (**76.5**)	4.1 (**80.4**)	4.9 (**96.1**)	4.0 (78.4)	4.5 (**88.2**)	4.5 (**88.2**)
	w/ SM	0.0	1.7	0.7 (—)	0.7 (—)	1.3 (—)	1.0 (—)	1.3 (—)	0.7 (—)	0.3 (—)	0.8 (—)
	w/ SD	0.0	0.1	0.1 (—)	0.1 (—)	0.2 (—)	0.1 (—)	0.1 (—)	0.3 (—)	0.0 (—)	0.1 (—)
	w/ SC	0.0	**13.0**	**7.7** (59.2)	**9.0** (69.2)	**6.3** (48.5)	**7.7** (59.2)	**8.0** (61.5)	**8.0** (61.5)	**8.7** (66.9)	**8.2** (63.1)

Table 4: Comparing Accuracy and NDA Metrics Across All Methods. The **boldface** numbers mean the best results; underlines indicate the second-best. "—" denotes methods with poor results, where zero-shot performance closely approximates clean 3-shot outcomes, rendering NDA calculation meaningless.

Task	Setting	Temperature 0	0.3	0.5	0.7	1
Base-9	clean	61.0	60.9	57.5	55.3	46.4
	inaccurate-easy	29.7	28.0	27.2	26.6	21.7
	inaccurate-hard	5.0	5.1	5.5	4.6	5.0
Base-11	clean	34.0	33.8	31.6	29.8	23.9
	irrelevant-easy	21.7	23.1	21.3	23.3	19.1
	irrelevant-hard	17.0	17.5	15.5	14.1	10.7
Sym.(E)	clean	34.2	35.8	35.7	34.6	32.7
	irrelevant-easy	28.6	31.5	29.8	29.1	28.1
	irrelevant-hard	27.0	26.1	26.2	24.0	23.0
Sym.(L)	clean	6.3	8.3	8.9	8.9	9.3
	inaccurate-easy	5.0	7.3	8.6	8.3	7.0
	inaccurate-hard	4.0	6.1	6.3	6.2	6.0

Table 5: Comparing performances of the base model with different temperatures. Sym. is the symbolic task.

Task	Setting	#Prompting Examples 1	2	3	4	5
Base-9	clean	24.8	38.3	46.4	50.8	50.5
	inaccurate-easy	17.5	22.2	23.2	25.4	25.6
	inaccurate-hard	11.3	6.3	6.0	5.7	5.7
Base-11	clean	11.8	20.4	23.9	29.9	32.1
	irrelevant-easy	8.9	15.9	19.1	21.7	26.3
	irrelevant-hard	7.7	10.0	10.7	15.2	16.1
Sym.(E)	clean	18.0	26.5	32.7	39.8	—
	inaccurate-easy	17.3	23.6	29.1	34.7	—
	inaccurate-hard	15.0	21.0	22.7	—	—
Sym.(L)	clean	2.7	7.7	9.3	11.3	12.2
	irrelevant-easy	2.3	5.4	7.0	8.8	8.9
	irrelevant-hard	1.9	4.0	6.0	6.3	—

Table 6: Comparing performances of base model with a varying number of examples ("—" denotes over token limit).

Model	Task	Setting 0-shot	clean	irr.	ina.
GPT3.5	Base-9	7.2	46.4	30.3	10.1
	Sym.(E)	8.8	32.7	25.1	26.1
	Com.	40.0	45.7	42.3	33.4
Gemini	Base-9	12.7	88.0	72.3	21.2
	Sym.(E)	9.3	44.5	38.9	36.7
	Com.	42.9	55.6	53.2	33.5
Llama2	Base-9	1.7	4.9	2.9	2.7
	Sym.(E)	4.7	10.1	8.7	9.1
	Com.	35.0	42.3	41.9	40.2
Mixtral	Base-9	3.9	27.5	16.3	3.7
	Sym.(E)	8.3	19.3	17.9	15.1
	Com.	24.2	37.5	34.9	31.1

Table 7: Comparing LLMs with 0-shot, 3-shot clean, and 3-shot medium irrelevant (irr.) / inaccurate (ina.) rationales.

in reasoning tasks is limited, often resulting in the miscorrection of the given content. SP can only slightly improve the accuracy of commonsense tasks, while ISC performs poorly across all tasks. As in Tab. 1, these methods can even perform worse than the base model.

***Observation* 2. Self-consistency methods can improve reasoning instead of truly denoising.** Two self-consistency approaches, SM (Robey et al., 2023) and SD (Zhang et al., 2023b)s are originally proposed to address Noisy-Q issues. When applied to our Noisy-R scenarios, they tend to easily disrupt the intrinsic logical coherence within the thought chain. Although these methods utilizing smooth strategies (e.g., random smoothing or masking) perform well on the commonsense dataset, they can hardly handle the more difficult reasoning tasks and achieve even close to 0%, e.g., in the Symbolic Longer task. Another method, SC (Wang et al., 2023), performs well in all tasks compared to the base model, improving both clean and noisy reasoning performance. However, SC does not conduct explicit denoising on rationales during its reasoning procedure, but it requires a high computation cost of #tokens (details in Appendix D.2).

Beyond the unreliability of existing methods, we analyze the potential of the LLM's intrinsic properties (e.g., temperature, example shot number, and different LLMs) for reasoning under Noisy-R. The results are presented in Tabs. 5 & 6 & 7.

***Observation* 3. Adjusting model temperature can help reasoning under Noisy-R.** In Tab. 5, we evaluate the base LLM using different temperatures on 3-shot demonstrations. Overall, reducing temperature can enhance the model's accuracy under both noisy and clean rationale reasoning, compared to the default temperature of 1. However, the relationship between temperature and accuracy is not linear for noisy reasoning; instead, there are multiple peaks in accuracy within the temperature range of 0 to 1. Additionally, excessively low temperatures (e.g., 0) tend to result in verbosity and repetition, which cause the model to exceed token limits up to 30% in symbolic tasks where the length of expected answers is quite shifty among different questions.

***Observation* 4. Increasing prompting examples boosts noisy reasoning accuracy on most tasks.** In Tab. 6, we evaluate the model using different numbers of exemplars while keeping the temperature at 1. In general, the accuracy of LLM will improve as the number of noisy examples increases in the clean and most noisy rationales. However, it should be noted that in tasks with high-level noise from NoRa-Math, increasing prompting examples can result in reduced accuracy, as observed in the base-9 inaccurate-hard dataset, even falling lower than the 0-shot accuracy of 7.2%.

***Observation* 5. Different LLMs generally suffer from Noisy-R.** In Tab. 7, we evaluate different LLMs on the NoRa dataset across three different settings: 0-shot CoT, 3-shot clean rationales, and 3-shot medium-level noisy rationales. Gemini-Pro outperforms GPT-3.5 in overall performance; however, it demonstrates a similar degree of sensitivity to noise, with a 4.3%-17.8% performance decline with irrelevant rationales and a 17.5%-76.1% decline with inaccurate rationales compared to clean rationales. While Mixtral 8x7B shows a slight underperformance compared to GPT-3.5, it also manifests a vulnerability to noise, incurring a 6.9%-40.7% loss with irrelevant rationales and a 17.0%-86.5% loss with inaccurate rationales. By contrast, Llama2 70B performs suboptimal, leading to a 1.0%-40.8% drop with irr. rationales and a 11.0%-44.9% drop with ina. rationales.

E FULL EXAMPLES OF THE NORA DATASET

Dataset	Irrelevant Thoughts	Inaccurate Thoughts
Math Base-9	In base-9, digits run from 0 to 8. We have 3 + 2 = 5 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 5 mod 9 = 5, so the digit is 5 and the carry is 0. There are five oceans on Earth: the Atlantic, Pacific, Indian, Arctic, and Southern. We have 8 + 6 + 0 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 155. Answer: 155	In base-9, digits run from 0 to 8. We have 3 + 2 = 5 in base-10. 5 + 3 = 8. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 5 mod 9 = 5, so the digit is 5 and the carry is 0. 5 + 9 = 14. We have 8 + 6 + 0 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 155. Answer: 155
Symbolic Equal	... "turn around right" means the agent needs to turn right, and repeat this action sequence four times to complete a 360-degree loop. Many GPS navigation systems will issue a 'turn around' command if the driver deviates from the planned route. So, in action sequence is I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT. ...	... "turn around right" means the agent needs to turn right, and repeat this action sequence four times to complete a 360-degree loop. Turn opposite is I_TURN_RIGHT I_TURN_LEFT. So, in action sequence is I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT. ...
Commonsense	The relations path are son, sister, uncle, which means Francisco is David's son's sister's uncle. For son's sister, we have son's sister is daughter. So the relations path are reduced to daughter, uncle. In genetics, mitochondrial DNA is always inherited from the mother, making the mother-daughter genetic link unique. For daughter's uncle, we have daughter's uncle is brother. So the relations path are reduced to brother. Therefore, the answer is brother. Answer:brother	The relations path are son, sister, uncle, which means Francisco is David's son's sister's uncle. For son's sister, we have son's sister is daughter. So the relations path are reduced to daughter, uncle. For daughter's uncle, we have daughter's uncle is brother. We have brother' sister is brother. So the relations path are reduced to brother. Therefore, the answer is brother. Answer:brother

Table 8: Exampler noisy rationales (consisting noisy thoughts) from the NoRa dataset.

In this section, we show the full examples of the NoRa dataset.

Nora-Math:

- Base-9 Irrelevant (Tab. 9)
- Base-9 Inaccurate (Tab. 10)
- Base-11 Irrelevant (Tab. 11)
- Base-11 Inaccurate (Tab. 12)

NoRa-Symbolic:

- Symbolic-Equal Irrelevant (Tab. 13)
- Symbolic-Equal Inaccurate (Tab. 14)
- Symbolic-Longer Irrelevant (Tab. 15)
- Symbolic-Longer Inaccurate (Tab. 16)

NoRa-Commonsense:

- Commonsense Irrelevant (Tab. 17)
- Commonsense Inaccurate (Tab. 18) .

Easy Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-9, the digits are "012345678". We have 4 + 2 = 6 in base-10. According to the Standard Model of physics, there are six types of quarks, the fundamental constituents of matter. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. In music, an octave consists of eight notes in the Western musical scale. 6 mod 9 = 6, so the digit is 6 and the carry is 0. We have 6 + 6 + 0 = 12 in base 10. 12 mod 9 = 3, so the digit is 3 and the carry is 1. A leading digit is 1. So the answer is 136. Answer:\box{136}
Medium Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\box{result}". Assistant: In base-9, the digits are "012345678". We have 4 + 2 = 6 in base-10. The standard guitar has six strings. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. In mathematics, eight is a Fibonacci number and a cube number. 6 mod 9 = 6, so the digit is 6 and the carry is 0. Six is the atomic number of carbon in chemistry, a key element for life as we know it. We have 6 + 6 + 0 = 12 in base 10. 12 mod 9 = 3, so the digit is 3 and the carry is 1. A leading digit is 1. So the answer is 136. In astronomy, Saturn is the sixth planet from the Sun in our solar system. Answer:\box{136}
Hard Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\box{result}". Assistant: In base-9, the digits are "012345678". We have 4 + 2 = 6 in base-10. The hexadecimal system in computer science is base-16, from 0-F, and is often used in programming. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. Spiders, part of the arachnid family in biology, typically have eight legs. 6 mod 9 = 6, so the digit is 6 and the carry is 0. In astronomy, Saturn is the sixth planet from the Sun in our solar system. We have 6 + 6 + 0 = 12 in base 10. Twelve is the atomic number of magnesium in chemistry, a crucial element for life known to be a cofactor in over 300 enzymatic reactions. 12 mod 9 = 3, so the digit is 3 and the carry is 1. In computer science, a tri-state boolean can have 3 states: true, false, and unknown. A leading digit is 1. The first law of thermodynamics states that energy cannot be created or destroyed, only transferred or transformed. So the answer is 136. Answer:\box{136}

Table 9: Examples of NoRa Base-9 with Irrelevant Rationales

Easy Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\box{result}". Assistant: In base-9, the digits are "012345678". 9 + 8 = 17. We have 4 + 2 = 6 in base-10. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 8 + 3 = 11. 6 mod 9 = 6, so the digit is 6 and the carry is 0. We have 6 + 6 + 0 = 12 in base 10. 12 mod 9 = 3, so the digit is 3 and the carry is 1. A leading digit is 1. So the answer is 136. Answer:\box{136}.
Medium Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\box{result}". Assistant: In base-9, the digits are "012345678". We have 4 + 2 = 6 in base-10. 6 + 1 = 7. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 8 + 3 = 11. 6 mod 9 = 6, so the digit is 6 and the carry is 0. 6 + 9 = 15. We have 6 + 6 + 0 = 12 in base 10. 12 mod 9 = 3, so the digit is 3 and the carry is 1. 3 + 9 = 12. A leading digit is 1. So the answer is 136. Answer:\box{136}.
Hard Example: User: In base-9, what is 64+62? Please reason it step by step. End the response with the result in "Answer:\box{result}". Assistant: In base-9, the digits are "012345678". We have 4 + 2 = 6 in base-10. 6 + 6 = 12. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 8 + 6 = 14. 6 mod 9 = 6, so the digit is 6 and the carry is 0. We have 6 + 6 + 0 = 12 in base 10. 12 + 1 = 13. 12 mod 9 = 3, so the digit is 3 and the carry is 1. 3 + 9 = 12. A leading digit is 1. 1 + 9 = 10. So the answer is 136. 6 + 9 = 15. Answer:\box{136}.

Table 10: Examples of NoRa Base-9 with Inaccurate Rationales

Easy Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". We have 7 + 4 = 11 in base-10. In mathematics, eleven is a prime number, meaning it has no factors other than 1 and itself. Since we're in base-11, that exceeds the maximum value of A for a single digit. 11 mod 11 = 0, so the digit is 0 and the carry is 1. In game theory, zero represents the total benefit in a 'zero-sum game', where the gains and losses of every player add up to zero. We have 9 + 7 + 1 = 17 in base 10. 17 mod 11 = 6, so the digit is 6 and the carry is 1. A leading digit is 1. So the answer is 160. Answer:\box{160}
Medium Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". We have 7 + 4 = 11 in base-10. Since we're in base-11, that exceeds the maximum value of A for a single digit.11 mod 11 = 0, so the digit is 0 and the carry is 1. We have 9 + 7 + 1 = 17 in base 10. In human biology, the kidneys filter about 17 liters of blood every hour. 17 mod 11 = 6, so the digit is 6 and the carry is 1. In geometry, a hexagon is a six-sided polygon. A leading digit is 1. In mathematics, one is the first counting number and the starting point of the natural numbers. So the answer is 160. In astronomy, a zero-age main sequence star (ZAMS) is a star that has just started nuclear fusion of hydrogen in its core. Answer:\box{160}
Hard Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". In sports, cricket teams traditionally consist of 11 players on each side. We have 7 + 4 = 11 in base-10. In mathematics, eleven is a prime number, meaning it has no factors other than 1 and itself. Since we're in base-11, that exceeds the maximum value of A for a single digit. In computer science, the 'Big 10' is a group of ten core subjects that are fundamental in the field, including algorithms and data structures, computer organization and architecture, and software engineering. 11 mod 11 = 0, so the digit is 0 and the carry is 1. In chemistry, pH 0 on the pH scale represents extreme acidity. We have 9 + 7 + 1 = 17 in base 10. 17 mod 11 = 6, so the digit is 6 and the carry is 1. A leading digit is 1. In chemistry, hydrogen, which has an atomic number of 1, is the most abundant element in the universe. So the answer is 160. In astronomy, a zero-age main sequence star (ZAMS) is a star that has just started nuclear fusion of hydrogen in its core. Answer:\box{160}

Table 11: Examples of NoRa Base-11 with Irrelevant Rationales

Easy Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". We have 7 + 4 = 11 in base-10. Since we're in base-11, that exceeds the maximum value of A for a single digit. 10 + 5 = 15. 11 mod 11 = 0, so the digit is 0 and the carry is 1. We have 9 + 7 + 1 = 17 in base 10. 17 mod 11 = 6, so the digit is 6 and the carry is 1. A leading digit is 1. So the answer is 160. 0 + 11 = 11. Answer:\box{160}.
Medium Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". 11 + 8 = 19. We have 7 + 4 = 11 in base-10. 11 + 4 = 15. Since we're in base-11, that exceeds the maximum value of A for a single digit.11 mod 11 = 0, so the digit is 0 and the carry is 1. We have 9 + 7 + 1 = 17 in base 10. 17 + 7 = 24. 17 mod 11 = 6, so the digit is 6 and the carry is 1. A leading digit is 1. So the answer is 160. 0 + 11 = 11. Answer:\box{160}.
Hard Example: User: In base-11, what is 97+74? Please reason it step by step. End the response with the result in "Answer:\boxed{result}". Assistant: In base-11, the digits are "0123456789A". 11 + 8 = 19. We have 7 + 4 = 11 in base-10. Since we're in base-11, that exceeds the maximum value of A for a single digit. 10 + 7 = 17. 11 mod 11 = 0, so the digit is 0 and the carry is 1. 0 + 11 = 11. We have 9 + 7 + 1 = 17 in base 10. 17 + 2 = 19. 17 mod 11 = 6, so the digit is 6 and the carry is 1. 6 + 11 = 17. A leading digit is 1. So the answer is 160. 0 + 11 = 11. Answer:\box{160}.

Table 12: Examples of NoRa Base-11 with Inaccurate Rationales

Easy Example:
User: With IN:walk right twice and look right twice, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. The 'right' corresponds to the command I_TURN_RIGHT. Turning right in countries that drive on the right side of the road typically does not intersect with oncoming traffic. Subsequently, 'walk' translates to I_WALK. Therefore, the action sequence is I_TURN_RIGHT I_WALK. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK.
Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. Motorists in many areas are allowed to turn right at a red traffic signal after coming to a complete stop and ensuring the road is clear. The 'right' corresponds to the command I_TURN_RIGHT. Certain traffic circles or roundabouts require all entering traffic to turn right regardless of the direction they intend to take. Subsequently, 'look' translates to I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.
Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.

Medium Example:
User: With IN:walk right twice and look right twice, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. Certain traffic circles or roundabouts require all entering traffic to turn right regardless of the direction they intend to take. The 'right' corresponds to the command I_TURN_RIGHT. Subsequently, 'walk' translates to I_WALK. The average human walking speed is about 3.1 miles per hour, or 5 kilometers per hour. Therefore, the action sequence is I_TURN_RIGHT I_WALK. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK. Cooking recipes may instruct to fold or mix ingredients 'twice' to ensure even distribution and consistency.
Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. The 'right' corresponds to the command I_TURN_RIGHT. Motorists in many areas are allowed to turn right at a red traffic signal after coming to a complete stop and ensuring the road is clear. Subsequently, 'look' translates to I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. Action sequences in user experience design are the steps a user follows to interact with a system to achieve a goal. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.
Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.

Hard Example:
User: With IN:walk right twice and look right twice, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. In left-hand traffic jurisdictions, such as the UK, turning right is analogous to turning left in right-hand traffic jurisdictions, crossing the path of oncoming vehicles. The 'right' corresponds to the command I_TURN_RIGHT. Turning right in countries that drive on the right side of the road typically does not intersect with oncoming traffic. Subsequently, 'walk' translates to I_WALK. The longest unbroken walkable distance on Earth stretches from Cape Town, South Africa, to Magadan, Russia, a distance of over 22,000 km. Therefore, the action sequence is I_TURN_RIGHT I_WALK. In robotics, an action sequence involves a set of commands that direct a robot to perform a series of tasks. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK. The term 'twice' is numerically equivalent to two times or doubling the quantity or occurrence of something.
Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. The 'right' corresponds to the command I_TURN_RIGHT. Certain traffic circles or roundabouts require all entering traffic to turn right regardless of the direction they intend to take. Subsequently, 'look' translates to I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. The action sequence of a chemical reaction involves a series of steps that lead to the formation of products from reactants. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK. In geometry, if an object is scaled up 'twice' its original size, each of its dimensions is increased by a factor of two.
Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.

Table 13: Examples of NoRa Symbolic-Equal with Irrelevant Rationales

Easy Example: User: With IN:walk right twice and look right twice, what is OUT? Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>" Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. The 'right' corresponds to the command I_TURN_RIGHT. Subsequently, 'walk' translates to I_WALK. The term 'jump' corresponds to the command I_RUN. Therefore, the action sequence is I_TURN_RIGHT I_WALK. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK. Thrice means the entire sequence is repeated 6 times. Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. The 'right' corresponds to the command I_TURN_RIGHT. The term 'left' translates to I_TURN_RIGHT. Subsequently, 'look' translates to I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK. Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.
Medium Example: User: With IN:walk right twice and look right twice, what is OUT? Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>" Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. The term 'left' translates to I_TURN_RIGHT. The 'right' corresponds to the command I_TURN_RIGHT. The term 'left' translates to I_TURN_RIGHT. Subsequently, 'walk' translates to I_WALK. Therefore, the action sequence is I_TURN_RIGHT I_WALK. The term 'run' corresponds to the command I_JUMP. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK. Thrice means the entire sequence is repeated 7 times. Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. The 'right' corresponds to the command I_TURN_RIGHT. Subsequently, 'look' translates to I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK. Thrice means the entire sequence is repeated 6 times. Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.
Hard Example: User: With IN:walk right twice and look right twice, what is OUT? Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>" Assistant: Since command is walk right twice and look right twice, we should consider Step1: "walk right twice" firstly, "walk right" means the agent needs to turn right and walk. The term 'left' translates to I_TURN_RIGHT. The 'right' corresponds to the command I_TURN_RIGHT. The term 'left' translates to I_TURN_RIGHT. Subsequently, 'walk' translates to I_WALK. The term 'look' corresponds to the command I_RUN. Therefore, the action sequence is I_TURN_RIGHT I_WALK. The term 'walk' corresponds to the command I_RUN. Since we need do twice in command "walk right twice", this entire sequence is repeated 2 times, so the action sequence to "walk right twice" is :I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK. Thrice means the entire sequence is repeated 5 times. Now, we consider Step2:"look right twice", "look right" means the agent needs to turn right and look. The term 'left' translates to I_TURN_RIGHT. The 'right' corresponds to the command I_TURN_RIGHT. Subsequently, 'look' translates to I_LOOK. The term 'jump' corresponds to the command I_LOOK. Therefore, the action sequence is I_TURN_RIGHT I_LOOK. Since we need do twice in command "look right twice", this entire sequence is repeated 2 times, so the action sequence to "look right twice" is :I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK. Thrice means the entire sequence is repeated 5 times. Above all – So, final answer is OUT:I_TURN_RIGHT I_WALK I_TURN_RIGHT I_WALK I_TURN_RIGHT I_LOOK I_TURN_RIGHT I_LOOK.

Table 14: Examples of NoRa Symbolic-Equal with Inaccurate Rationales

Easy Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. In cinematography, an action sequence is a part of the film where intense physical actions such as fights or chases take place.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Medium Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Many GPS navigation systems will issue a 'turn around' command if the driver deviates from the planned route. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. In animation, an action sequence is a series of frames showing a character performing a continuous movement.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Hard Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. In sports coaching, action sequences are used to describe a set of movements that athletes must perform correctly.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. The phrase 'turn around' is often used metaphorically to signify a complete change in situation or attitude. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. Action sequences in user experience design are the steps a user follows to interact with a system to achieve a goal.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Table 15: Examples of NoRa Symbolic-Longer with Irrelevant Rationales

Symbolic Longer Inaccurate
Easy Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. The term 'opposite' implies a 180-degree turn, which requires I_TURN_RIGHT, I_TURN_LEFT. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Medium Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. The term 'opposite' implies a 180-degree turn, which requires I_TURN_RIGHT. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. The term 'opposite' implies a 180-degree turn, which requires I_TURN_LEFT.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Hard Example:
User: With IN:jump around left and jump around left, what is OUT?
Please reason it step by step, and provide the final action sequence as the answer. End the response with "So, final answer is OUT: <action sequence>"
Assistant: Since command is jump around left and jump around left, we should consider Step1: "jump around left" firstly, "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. The term 'opposite' implies a 180-degree turn, which requires I_TURN_RIGHT, I_TURN_LEFT. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. The term 'opposite' implies a 180-degree turn, which requires I_TURN_LEFT.
Now, we consider Step2:"jump around left", "jump around left" means the agent needs to turn left and jump, and repeat this action sequence four times to complete a 360-degree loop. Therefore, the action sequence is I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP. The term 'opposite' implies a 180-degree turn, which requires I_TURN_RIGHT.
Above all – So, final answer is OUT:I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP I_TURN_LEFT I_JUMP.

Table 16: Examples of NoRa Symbolic-Longer with Inaccurate Rationales

Easy Example:
User: In a family tree, if Chris is Frances's husband's son's son. Question: Chris is Frances's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Chris is Frances's husband's son's son, so the relations path is husband, son, son. For son's son, we have son's son is grandson. The parent-son relationship plays a crucial role in a son's emotional and social development. So the relations path are reduced to husband, grandson. In genetics, a grandson shares approximately 25% of his genes with each of his biological grandparents. For husband's grandson, we have husband's grandson is grandson. So the relations path are reduced to grandson. Therefore, Answer: grandson.

Medium Example:
User: In a family tree, if Chris is Frances's husband's son's son. Question: Chris is Frances's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Chris is Frances's husband's son's son, so the relations path is husband, son, son. In family relations, caregiving roles often fall to relatives, affecting their work-life balance and economic status. For son's son, we have son's son is grandson. From an evolutionary perspective, investing in the well-being of sons can contribute to an individual's inclusive fitness. So the relations path are reduced to husband, grandson. For husband's grandson, we have husband's grandson is grandson. So the relations path are reduced to grandson. In many cultures, grandsons often take on the role of carrying forward the family name and lineage. Therefore, Answer: grandson.

Hard Example:
User: In a family tree, if Chris is Frances's husband's son's son. Question: Chris is Frances's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Chris is Frances's husband's son's son, so the relations path is husband, son, son. The concept of family is legally recognized and protected, influencing policies around taxation, welfare, and rights. For son's son, we have son's son is grandson. Research has shown that a strong parent-son relationship can have a positive impact on a son's mental health. So the relations path are reduced to husband, grandson. According to the 'Grandmother Hypothesis' in evolutionary biology, grandmothers contribute to the survival of grandchildren, leading to increased human lifespan. For husband's grandson, we have husband's grandson is grandson. So the relations path are reduced to grandson. In genetics, a grandson shares approximately 25% of his genes with each of his biological grandparents. Therefore, Answer: grandson.

Table 17: Examples of NoRa Commonsense with Irrelevant Rationales

Easy Example:
User: In a family tree, if Chris is Frances's husband's son's son. Question: Chris is Frances's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Chris is Frances's husband's son's son, so the relations path is husband, son, son. For son's son, we have son's son is grandson. So the relations path are reduced to husband, grandson. We have grandson's daughter is brother. For husband's grandson, we have husband's grandson is grandson. So the relations path are reduced to grandson. We have grandson's mother is grandfather Therefore, Answer: grandson.

Medium Example:
User: In a family tree, if Chris is Frances's husband's son's son. Question: Chris is Frances's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Chris is Frances's husband's son's son, so the relations path is husband, son, son. We have son's uncle is daughter-in-law. For son's son, we have son's son is grandson. We have son's father is grandfather. So the relations path are reduced to husband, grandson. For husband's grandson, we have husband's grandson is grandson. So the relations path are reduced to grandson. We have grandson's daughter-in-law is father. Therefore, Answer: grandson.

Hard Example:
User: In a family tree, if Cindy is Pamela's mother's mother's daughter. Question: Cindy is Pamela's what? Please reason it step by step, and provide a single word answer describing the relationship. End the response in the format "Answer: relation "
Assistant: Cindy is Pamela's mother's mother's daughter, so the relations path is mother, mother, daughter. For mother's daughter, we have mother's daughter is sister. We have daughter's brother is father. So the relations path are reduced to mother, sister. We have sister's brother is sister. For mother's sister, we have mother's sister is aunt. We have sister's aunt is son. So the relations path are reduced to aunt. We have aunt's granddaughter is granddaughter. Therefore, Answer: aunt.

Table 18: Examples of NoRa Commonsense with Inaccurate Rationales