Invalid Logic, Equivalent Gains: The Bizarreness of Reasoning in Language Model Prompting

Rylan Schaeffer [*1] **Kateryna Pistunova** [*2] **Samar Khanna** [*1] **Sarthak Consul** [*1] **Sanmi Koyejo** [1]

Abstract

Language models can be prompted to reason through problems in a manner that significantly improves performance. However, *why* such prompting improves performance is unclear. Recent work showed that using logically *invalid* Chain-of-Thought (CoT) prompting improves performance almost as much as logically *valid* CoT prompting, and that editing CoT prompts to replace problem-specific information with abstract information or out-of-distribution information typically doesn't harm performance. Critics have responded that these findings are based on too few and too easily solved tasks to draw meaningful conclusions. To resolve this dispute, we test whether logically invalid CoT prompts offer the same level of performance gains as logically valid prompts on the hardest tasks in the BIG-Bench benchmark, termed BIG-Bench Hard (BBH). We find that the logically *invalid* reasoning prompts do indeed achieve similar performance gains on BBH tasks as logically valid reasoning prompts. We also discover that some CoT prompts used by previous works contain logical errors. This suggests that covariates beyond logically valid reasoning are responsible for performance improvements.

1. Introduction

Language models can perform significantly better when prompted in particular ways. For example, prompts that recommend or guide language models through step-by-step processing have been shown to significantly improve performance on question answering, conversational response generation and other tasks (Nye et al., 2021; Wei et al., 2022b; Jung et al., 2022; Kojima et al., 2022; Yao et al., 2023). These prompting techniques are especially effective on the hardest tasks (Suzgun et al., 2022) in the BIG-Bench benchmark (Srivastava et al., 2022), leading many to conclude that such techniques unlock emergent[1] human-like reasoning abilities in large language models (Wei et al., 2022a). However, *why* such prompting strategies improve performance is unclear. Madaan & Yazdanbakhsh (2022) showed replacing problem-specific information in Chain-of-Thought (CoT) prompts with either abstract information or out-of-distribution information typically doesn't harm CoT's performance gains, and Wang et al. (2022) showed that logically *invalid* CoT prompts (i.e., prompts which contain logically invalid chains of reasoning) achieve approximately the same gains as logically valid CoT prompts. These discoveries are puzzling, but critics warned against drawing confident or far-ranging conclusions results because (1) only a small number of tasks are considered (specifically GSM8K (Cobbe et al., 2021), Bamboogle (Press et al., 2022), and two commonsense tasks (Srivastava et al., 2022)) and (2) the tasks are relatively easy to solve.

In this work, we aim to clarify this dispute by asking whether logically invalid CoT prompts indeed offer comparable performance gains as logically valid CoT prompts in *more* and *harder* tasks. To answer this question, we evaluate the performance of logically invalid CoT prompts on the hardest tasks in BIG-Bench (Srivastava et al., 2022), the so-called BIG-Bench Hard (BBH) tasks (Suzgun et al., 2022). We find that logically invalid CoT prompts induce performance gains approaching logically valid CoT prompts on BBH tasks. We also discover that CoT prompts from previous work contain logical errors, a discovery that we confirmed with the original authors. Our findings support growing evidence that while prompting techniques can, without doubt, significantly improve language model performance on complex tasks, there is questionable evidence that these performance gains are attributable to logical reasoning skills. Covariates in the prompts other than logical reasoning may be responsible for the performance improvements and point to the need for continued investigation.

[*]Equal contribution [1]Department of Computer Science, Stanford University [2]Department of Physics, Stanford University. Correspondence to: Rylan Schaeffer <rschaef@cs.stanford.edu>, Sanmi Koyejo <sanmi@cs.stanford.edu>.

ICML 2023 Workshop Knowledge and Logical Reasoning in the Era of Data-driven Learning.

[1]Although see Schaeffer et al. (2023).

2. Methods

2.1. Tasks: BIG-Bench Hard ⊂ BIG-Bench

Beyond-the-Imitation Game Benchmark (BIG-Bench) is a benchmark of over 200 natural language tasks created to evaluate the capabilities and limitations of language models (Srivastava et al., 2022). BIG-Bench Hard (BBH) is a subset of BIG-Bench consisting of 23 tasks specifically identified as extremely difficult for language models (Suzgun et al., 2022). BBH was constructed by excluding BIG-Bench tasks using the following criteria: tasks with more than three subtasks, tasks with fewer than 103 examples, tasks without human-rater baselines, tasks that do not use multiple-choice or exact match as the evaluation metric, and tasks on which the best reported model beats average reported human-rater score; the remaining 23 BIG-Bench tasks comprise the BBH tasks. The BBH tasks cover a variety of categories, including traditional NLP, mathematics, commonsense reasoning, and question-answering. In general, BBH tasks typically fall into one of two categories: more traditional NLP tasks (e.g., Date Understanding) or more algorithmic tasks (e.g., Boolean Expressions). Many state of the art language models, including GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2022), as well as internal dense and sparse Google models, were shown to be incapable of solving BBH tasks above average human rater performance when asked directly.

2.2. Prompt Types

In this work, we compare three prompt types applied to the BBH tasks. Examples of task problems and prompts are shown in Table 1.

Chain-of-Thought (CoT) Prompting In (logically valid) Chain-of-Thought (CoT) prompting, each question in each task is prepended with three high-quality, human hand-written, topical examples of Question→Reasoning→Answer. The reasoning is a sequence of natural language steps that derive the answer from information in the question (Nye et al., 2021; Wei et al., 2022b). For each BBH task, Suzgun et al. (2022) released three examples of Question→Reasoning→Answer that we adapt. We do not use their prompts directly because we want CoT prompts to contain logically valid chains of reasoning, and as we discovered (Sec. 3.1), some of Suzgun et al.'s CoT prompts contain logical errors.

Logically Invalid Chain-of-Thought (Invalid CoT) Prompting In logically invalid Chain-of-Thought (Invalid CoT) prompting, we edit the reasoning in the three examples for all three tasks to become logically invalid. To do this, we modify the reasoning to contain nonsensical or invalid steps. For instance, on the Boolean Expressions task, we provided the reasoning, "Because English does not permit multiple negatives, the expression "(not not True)" evaluates to "(not True)", and on the Date Understanding task, we provided the reasoning, "If tomorrow is 11/12/2019, then today is 11/12/2018. The date one year ago from today is 11/11/2018." In all cases, the modified BBH exemplars were logically invalid, but reached the correct answers. We wrote each example once and did not modify the examples whatsoever to improve task performance.

Answer-Only (AO) Prompting To establish baseline performance, we follow previous work and prompt language models to answer each question in each task directly. We call this "Answer Only" (AO) prompting.

Prompt Type	**Example Query**
	Evaluate the result of a random Boolean expression. Q: not ((not not True)) is
Answer Only	False
Scratchpad/Chain-of-Thought (CoT)	A: Let's think step by step. Remember that (i) expressions inside brackets are always evaluated first and that (ii) the order of operations from highest priority to lowest priority is "not", "and", "or", respectively. We first simplify this expression "Z" as follows: "Z = not ((not not True)) = not ((A))" where "A = not not True". Let's evaluate A: A = not not True = not (not True) = not False = True. Plugging in A, we get: Z = not ((A)) = not ((True)) = not True = False. So the answer is False.
Logically-Invalid Chain-of-Thought (Invalid CoT)	A: Let's think step by step. Remember that (i) expressions inside brackets are always evaluated first and that (ii) the order of operations from highest priority to lowest priority is "not", "and", "or", respectively. Because English does not permit multiple negatives, the expression "(not not True)" evaluates to "(not True)". The expression "not ((not not True))" therefore evaluates to "not ((not True))". By the same logic, the expression "not ((not True))" simplifies to "not True". In Boolean logic, "not True" is False. So the answer is False.

Table 1. **Examples of different prompt types.**

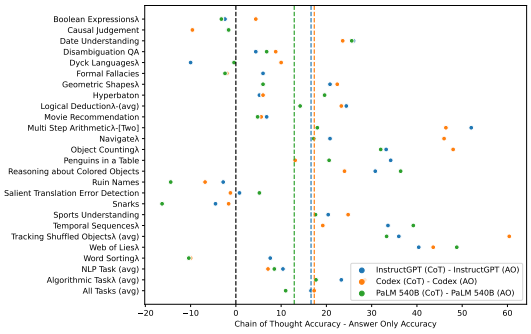

Figure 1. **Codex benefits more from Chain-of-Thought Prompting than InstructGPT or PaLM 540B on BIG-Bench Hard Tasks.** Dashed vertical lines indicate average model performance across all BIG-Bench Hard Tasks. Data from Suzgun et al. (2022).

2.3. Metrics

To evaluate the performance of different prompting types, we used Accuracy (i.e. Exact String Match) to evaluate model outputs. We acknowledge that recent work showed such a choice of metric may reach misleading scientific conclusions (Schaeffer et al., 2023), but made the choice to ensure fair comparison with the relevant preceding work.

2.4. Choice of Language Model

Due to limited resources, we could only evaluate one model. Only Codex (Chen et al., 2021) and InstructGPT (Ouyang et al., 2022) were publicly queryable, thus ruling out PaLM 540B (Chowdhery et al., 2022). We chose to evaluate Codex because our visualizing Suzgun et al's data revealed Codex outperformed both InstructGPT and PaLM 540B; this point was not made by the authors, but visualizing their data anew reveals this conclusion (Fig. 1). Although this finding may seem surprising, independent work has found that language models reason better if pretrained heavily on code (Madaan et al., 2022), as Codex was. Perhaps unsurprisingly, we found that Codex's high average performance with CoT prompting is driven by large improvements on BBH algorithmic tasks, overriding mediocre improvements (or regressions) on natural language tasks.

2.5. Choice of BBH Tasks

We evaluate AO, CoT, and Invalid CoT prompting strategies on all BBH tasks.

3. Experimental Results

3.1. Previous CoT Prompts Contain Logical Errors

While converting the logically valid CoT prompts written for BBH tasks by Suzgun et al. (2022) to logically invalid CoT prompts, we discovered that at least three of the BBH

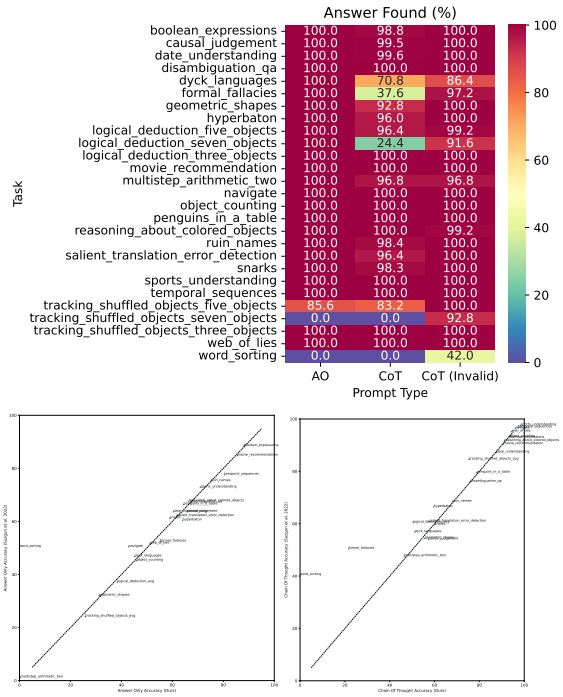

Figure 2. **Sanity checks.** Top: All prompting strategies almost always produced an answer across tasks. Bottom: Our accuracies for Answer Only (AO) and Chain-of-Thought (CoT) prompting closely matched published values from Suzgun et al. (2022) except on one task: word_sorting.

tasks' CoT prompts were *already* logically invalid. On the Multistep Arithmetic Task, a human handwritten CoT prompt reads: "Then, the final equation is A * B = -41 * -3 = (-61) * (-3) = 123. So the answer is 123." To clarify, -41 was substituted with -61 without justification, and -61 * -3 is not 123, even though 123 is the correct answer. On the Navigate task, a human handwritten CoT prompt reads: "(4) Take 7 steps right: (0, 7), facing the positive y-axis." should be "(4) Take 7 steps right: (0, 0), facing the positive y-axis." On the Web of Lies task, "(3) Vina says Jerry tells the truth. Since we know from (2) that Jerry tells the truth, if Vine says Jerry tells the truth, then Vina tells the truth." should be "(3) Vina says Jerry tells the truth. Since we know from (2) that Jerry tells the truth, if Vina says Jerry tells the truth, then Vina tells the truth." All three errors were raised to the authors, who confirmed our discoveries and accepted our suggested corrections as well. This discovery already hints that CoT prompts need not be logically correct for the model to output the correct answer. It also hints that previous CoT prompts from earlier papers, e.g., (Wei et al., 2022b), may need to be investigated further.

3.2. Sanity Checks

We found that Codex produced an answer under all prompting strategies on almost all tasks (Fig. 2 top), and the accura-

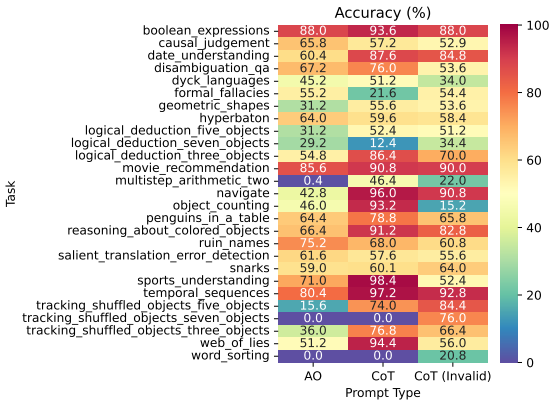

Figure 3. **Accuracy per task per prompt type on BIG-Bench Hard.** Prompt Types: AO = Answer Only, CoT = Chain-of-Thought, CoT (Invalid) = Logically Invalid Chain-of-Thought. Approximately 200-250 questions per BIG-Bench Hard task.

cies under Answer Only (AO) and Chain-of-Thought (CoT) prompting closely matched published values (Suzgun et al., 2022). The one exception was a single task: word_sorting.

3.3. Accuracy by Prompt Type by Task

The Accuracy of Codex on each of the BIG-Bench Hard (BBH) tasks under each of the three prompt types (Answer Only, Chain-of-Thought, Logically Invalid Chain-of-Thought) is displayed in Fig. 3. Despite using the same model and the same decoding hyperparameters (e.g., temperature), we notice that there is tremendous variation in the accuracies. To better compare the performance across the three prompt types, we visualized each prompt type's average accuracy over all BBH tasks. We found that Chain-of-Thought prompting beats Answer Only prompting, but Logically Invalid Chain-of-Thought is close behind Chain-of-Thought and better than Answer Only (Fig. 4 top). Pairwise comparisons between prompt types across different BBH tasks reveal that different prompt types do not dominate one another (Fig. 4 bottom), but rather display rich structure. We note that we do not optimize logically invalid CoT prompts whatsoever, meaning the Invalid CoT performance here is likely below what tuning could achieve.

4. Conclusion

On the diverse and challenging BIG-Bench Hard tasks, we find that Chain-of-Thought prompting performs best on average, but logically invalid Chain-of-Thought prompting is close behind and outperforms Answer Only prompting. This demonstrates that completely illogical reasoning in the CoT prompts do not significantly harm the performance of the language model. Our findings suggest that valid reasoning in prompting is not the chief driver of performance

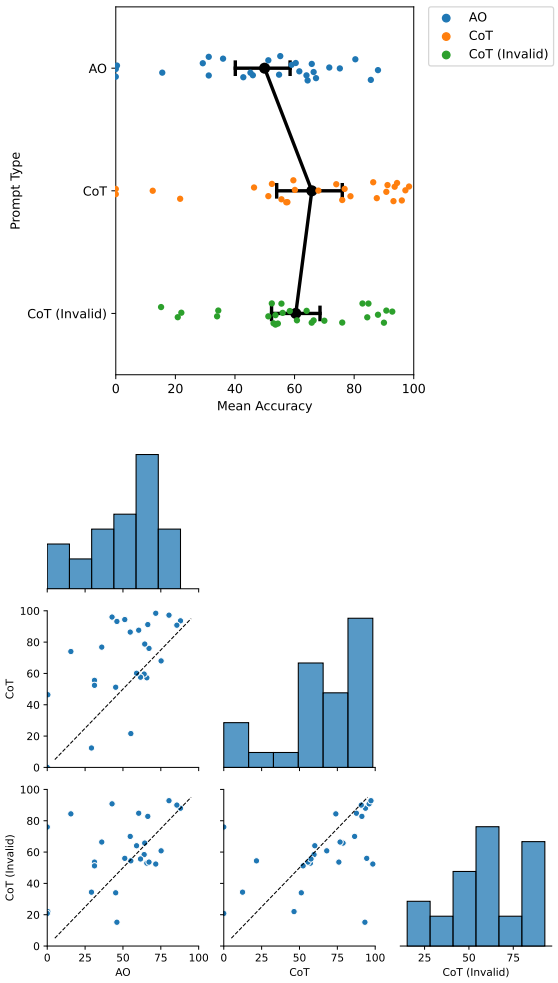

Figure 4. **Logically invalid Chain-of-Thought outperforms Answer Only and nearly matches Chain-of-Thought.** Top: Accuracy per BBH task, averaged over all BBH tasks. Bottom: Pairwise accuracy plots comparing the three different prompt types.

gains, raising the question of what is. We note that there are complementary approaches towards achieving reasoning in language models such as enforcing valid reasoning in the model architecture (Creswell et al., 2022; Creswell & Shanahan, 2022) or via autoformalization (Wu et al., 2022; Azerbayev et al., 2023).

Our work raises important questions for future work. Why are models robust to invalid CoT prompts? What features of the data or prompts result in the model outputting inconsistent or invalid outputs? Does increasing the degree of "incorrectness" or the number of incorrect prompts affect the model's sensitivity to invalid CoT? What other properties of the valid prompts is the model sensitive to? Answering these questions can yield useful insights into prompt engineering for LLMs, as well as a deeper understanding of when models output inconsistent or "hallucinated" answers.

References

Azerbayev, Z., Piotrowski, B., Schoelkopf, H., Ayers, E. W., Radev, D., and Avigad, J. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics. *arXiv preprint arXiv:2302.12433*, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. d. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Creswell, A. and Shanahan, M. Faithful reasoning using large language models. *arXiv preprint arXiv:2208.14271*, 2022.

Creswell, A., Shanahan, M., and Higgins, I. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv preprint arXiv:2205.09712*, 2022.

Jung, J., Qin, L., Welleck, S., Brahman, F., Bhagavatula, C., Bras, R. L., and Choi, Y. Maieutic prompting: Logically consistent reasoning with recursive explanations. *arXiv preprint arXiv:2205.11822*, 2022.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022.

Madaan, A. and Yazdanbakhsh, A. Text and patterns: For effective chain of thought, it takes two to tango. *arXiv preprint arXiv:2209.07686*, 2022.

Madaan, A., Zhou, S., Alon, U., Yang, Y., and Neubig, G. Language models of code are few-shot commonsense learners. *arXiv preprint arXiv:2210.07128*, 2022.

Nye, M., Andreassen, A. J., Gur-Ari, G., Michalewski, H., Austin, J., Bieber, D., Dohan, D., Lewkowycz, A., Bosma, M., Luan, D., et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Press, O., Zhang, M., Min, S., Schmidt, L., Smith, N. A., and Lewis, M. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*, 2022.

Schaeffer, R., Miranda, B., and Koyejo, S. Are emergent abilities of large language models a mirage? *arXiv preprint arXiv:2304.15004*, 2023.

Srivastava, A., Rastogi, A., Rao, A., Shoeb, A. A. M., Abid, A., Fisch, A., Brown, A. R., Santoro, A., Gupta, A., Garriga-Alonso, A., et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*, 2022.

Suzgun, M., Scales, N., Schärli, N., Gehrmann, S., Tay, Y., Chung, H. W., Chowdhery, A., Le, Q. V., Chi, E. H., Zhou, D., et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

Wang, B., Min, S., Deng, X., Shen, J., Wu, Y., Zettlemoyer, L., and Sun, H. Towards understanding chain-of-thought prompting: An empirical study of what matters. *arXiv preprint arXiv:2212.10001*, 2022.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022a.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., and Zhou, D. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022b.

Wu, Y., Jiang, A. Q., Li, W., Rabe, M. N., Staats, C., Jamnik, M., and Szegedy, C. Autoformalization with large language models. *arXiv preprint arXiv:2205.12615*, 2022.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*, 2023.