`Direct-Inverse Prompting`: Analyzing LLMs' Discriminative Capacity in Self-Improving Generation

Anonymous ACL submission

Abstract

Mainstream LLM research has primarily focused on enhancing their generative capabilities. However, even the most advanced LLMs experience uncertainty in their outputs, often producing varied results on different runs or when faced with minor changes in input, despite no substantial change in content. Given multiple responses from the same LLM to the same input, we advocate leveraging the LLMs' discriminative capability to reduce this generative uncertainty, aiding in identifying the correct answers. Specifically, we propose and analyze three discriminative prompts: `Direct Prompt`, `Inverse Prompt`, and `Combination`, to explore the potential of both closed-source and open-source LLMs in self-improving their generative performance on two benchmark datasets. Our insights reveal which discriminative prompt is most promising and when to use it. To our knowledge, this is the first work to systematically analyze LLMs' discriminative capacity to address generative uncertainty.

1 Introduction

Generative AI is revolutionizing various fields by utilizing large language models (LLMs) trained to generate human-like responses based on given instructions. Despite the increasing strength of existing LLMs in terms of generation capability, a widely recognized issue is their uncertainty in responses to inputs—the same model may produce significantly different responses on different runs or to equivalently varied inputs.

Previous studies have either relied on external human/tool supervision (Wang et al., 2023a; Paul et al., 2024; Gou et al., 2023; Chen et al., 2023b; Olausson et al., 2023; Gao et al., 2023) or have not successfully explored the inner capabilities of LLMs, such as their own discriminative capability, to reduce uncertainty (Jiang et al., 2024). We argue that LLMs should focus on both their generative and discriminative capabilities. In this work, we explore various discriminative capabilities of LLMs to reduce the uncertainty of their generations.

Specifically, we propose and analyze three types of discriminative prompts to identify the most promising answer from a group of generated responses: `Direct Prompt`: directly asking the LLM which responses are correct; `Inverse Prompt`: contrasting `Direct Prompt`by asking which responses are incorrect; `Combination`: combining `Direct Prompt`and `Inverse Prompt`, since intuitively they perform the same reasoning process from complementary perspectives.

We conduct analyses with two closed-source LLMs (GPT-4 (OpenAI, 2023) and GPT-4o (OpenAI, 2024)) and two open-source LLMs (Llama-3-8B-Instruct (Meta, 2024) and MetaMath-7B-V1.0 (Yu et al., 2023)) on two math-related datasets, MATH (Hendrycks et al., 2021) and MathQA (Amini et al., 2019). We observe: i)For closed-source LLMs, using discriminative capability, either `Direct Prompt` or `Inverse Prompt`, is highly effective for reducing generative uncertainty. ii) For open-source LLMs, if not instruction-tuned, using discriminative capability is not recommended. Even if instruction-tuned, only `Direct Prompt` is recommended due to likely issues with understanding negation in `Inverse Prompt`.

Our contributions are threefold:

- Proposing `Direct-Inverse Discriminative Prompting`, a multi-angle complementary method, to assess LLMs' discriminative capability;
- The first systematic analysis of the potential of LLMs' discriminative capability to reduce generative uncertainty;
- Providing insights and suggestions for future users on how to effectively utilize LLMs' discriminative capability in practice.

2 Related Work

LLM self-improves generation. Various methods are being devised to increase the certainty of LLM-generated answers. Chain-of-Thought (Wei et al., 2023) tries to add a detailed reasoning path from the input to the output answer so that the answer is more explainable and certain. Self-Consistency (Wang et al., 2023b) has the LLM solve the same problem multiple times to obtain several results. A majority vote is then conducted to choose the most consistent result as the final answer. This approach guarantees a higher success rate than Chain-of-Thought. Based on this, diverse variants of Self-Consistency exist; for example, Universal Self-Consistency (Chen et al., 2023a), which includes reasoning to select the most consistent value as the final answer, or Early Stop Self-Consistency (Li et al., 2024), which reduces the number of answer sets used in the majority vote to save cost and time. It is worth mentioning that the above approaches are fully unsupervised, namely no human or external signals are needed.

Exploring LLM discriminative capability to enhance generation. To assess the generative and discriminative capabilities of LLMs, Liu et al. (2023) and Arora and Kambhampati (2023) carried out experiments on summarization and planning problem, respectively. The most related work, (Jiang et al., 2024), concluded that LLMs struggle to enhance their generation performance through discriminative capability because their discriminative capability is not stronger than their generative capability. Our work differs from this study in two key ways: i) Jiang et al. (2024) only considered a simplified discriminative prompt similar to our `Direct Prompt`. They provided the discriminative prompt with all the generated final answers without the reasoning paths. In contrast, our `Direct Prompt` includes reasoning-path equipped answers, which we believe can help LLMs better determine the correct answer. ii) We further analyze another complementary discriminative capability expressed by `Inverse Prompt`. While `Inverse Prompt` should theoretically yield the same conclusions if applied to humans, the inconsistency between `Direct Prompt` and `Inverse Prompt` in LLMs allows us to better understand their discriminative potential in reducing generative uncertainty. iii) Our findings suggest a different conclusion: LLMs' discriminative capabilities can indeed enhance their generation if used skillfully.

3 `Direct-Inverse Discriminative Prompting`

Given multiple answer options by LLMs' generative process (here uses five for example), this section introduces our discriminative approach `Direct-Inverse Discriminative Prompting`, that asks LLMs with `Direct Prompt`, `Inverse Prompt`, and finally combines their lens to find the most certain answer.

`Direct Prompt`. Here, we directly ask LLMs which options are correct with the following prompt:

> This problem [*problem description*] has the following reasoning paths you generated: " A: [$path_1$]", "B: [$path_2$]", "C: [$path_3$]", "D: [$path_4$]", "E: [$path_5$]". Please output the correct ones.

`Inverse Prompt`. Here, we ask LLMs which options are incorrect with the following prompt:

> This problem [*problem description*] has the following reasoning paths you generated: " A: [$path_1$]", "B: [$path_2$]", "C: [$path_3$]", "D: [$path_4$]", "E: [$path_5$]". Please output the incorrect ones.

`Combination`. As humans, when asked using both `Direct Prompt` and `Inverse Prompt` prompts, their answers should be consistent. However, this is not the case with LLMs, as our analysis in Section 5.2 shows. For instance, using `Direct Prompt`, an LLM may believe "A and B" are correct, but when asked using `Inverse Prompt`, it might believe "B and C" are incorrect, implying that "A, D, and E" are correct. `Direct Prompt` and `Inverse Prompt` reflect LLMs' discriminative analysis of the problem from different perspectives, and we combine their results to improve accuracy. Specifically, we run `Direct Prompt` and `Inverse Prompt` separately multiple times and select the final answer by identifying the most consensus among the responses.

4 Experiments

Datasets. Two datasets. An example of each dataset is given in appendix A.

- `MATH` (Hendrycks et al., 2021): This dataset contains 7 types of open-ended math problems, including algebra and geometry, with average high school difficulty. For this project, we selected the entire test dataset of 5,000 problems. Each problem

	MATH				MathQA			
	GPT4	GPT-4o	Llama3	MetaMath	GPT4	GPT-4o	Llama3	MetaMath
Chain-of-Thought	47.58	50.67	21.55	10.83	72.57	82.73	39.03	11.96
Uni. Self-Consist.	55.14	54.72	26.72	12.04	79.50	85.33	42.58	11.79
`Direct Prompt`	54.18	**<u>57.44</u>**	**<u>27.54</u>**	0.18	<u>81.64</u>	**<u>86.73</u>**	**<u>46.40</u>**	0.00
`Inverse Prompt`	54.62	<u>55.48</u>	18.08	0.06	**<u>82.34</u>**	<u>86.40</u>	37.45	0.00
`Combination`	**<u>56.44</u>**	<u>56.82</u>	25.98	0.24	<u>82.04</u>	<u>86.63</u>	<u>42.98</u>	0.00

Table 1: Comparing discriminative prompts `Direct Prompt`, `Inverse Prompt`, and `Combination` on LLMs. Bold: top score. Underline: surpass the Universal Self-Consistency.

includes a "problem" label, representing the math word problem, and a "solution" label, which provides the explanation of how to solve the problem, including an answer formatted as `$\boxed{`A`}$`, where A is the answer. To maintain consistency, all models were instructed to return the final answer in the same format as the dataset.

- `MathQA` (Amini et al., 2019): This dataset includes 6 types of math problems, such as geometry and probability, with college-level difficulty. For this experiment, we selected all 2,985 problems from the test dataset. Each entry in MathQA contains a "problem," a "rationale" explaining how to solve it, "options" that list possible answers, and "correct," indicating the correct answer from the options. When models were asked to solve the MathQA problems, they were instructed to return the correct option's alphabet from the given choices.

LLMs. i) Two closed-source LLMs: GPT-4 (OpenAI, 2023) and GPT-4o (OpenAI, 2024). Both by OpenAI APIs. We do not consider more closed-source LLMs due to budget limits, and GPT-4 and GPT-4o are already widely recognized as the strongest LLMs. ii) Open-source LLMs: Llama-3 (Meta, 2024) and MetaMath (Yu et al., 2023)–a LLM specifically optimized for math problem solving. In our experiments, five A100 GPUs were used for running Llama-3 and MetaMath inference.

Baselines. i) Chain-of-Thought (Wei et al., 2022). We run it three times and report the average performance. ii) Universal Self-Consistency (Chen et al., 2023a), the state-of-the-art approach that runs Chain-of-Thought reasoning process five times, and finally choosing the answer with majority voting.

Setting. To prevent the LLMs' responses to options like "A, B, C, etc." from being biased due to their pretraining, we will shuffle these options and re-index them for each run. The final performance will be the average of three runs.

5 Results

5.1 Main Results

Table 1 presents the main results comparing different discriminative prompts (`Direct Prompt`, `Inverse Prompt`, and `Combination`) of LLMs on the MATH and MathQA datasets. Here are some key observations:

- Discriminative prompts (`Direct Prompt`, `Inverse Prompt`, and `Combination`) do not work for MetaMath. This is because MetaMath was specifically optimized for solving math problems rather than following instructions. In our experiments, MetaMath responded to our discriminative prompts with noise and unstructured outputs, making answer parsing impossible.

- Excluding MetaMath, `Inverse Prompt` outperforms `Direct Prompt` in 2 out of 6 cases, performs equally in one case (GPT-4o on MathQA), and underperforms in the remaining three cases. This is expected because negation is often more challenging for AI models to understand.

- In most cases (except for MetaMath), both `Direct Prompt` and `Combination` outperform Universal Self-Consistency (and even `Inverse Prompt` generally surpasses it on closed-source LLMs), indicating the effectiveness of using LLMs' discriminative capabilities to find the most certain answer.

5.2 Analysis

$\mathcal{Q}_1$: How frequently do LLMs experience uncertainty in their decisions, indicated by conflicts

	MATH	MathQA
GPT-4	36.88	23.75
GPT-4o	46.00	23.85
Llama-3	97.34	97.96
MetaMath	100.00	100.00

Table 2: Conflicting percentage per dataset for each LLM.

	MATH	MathQA
GPT-4	71.86 / 25.49	89.02 / 58.81
GPT-4o	77.93 / 30.30	93.36 / 62.64
Llama-3	76.69 / 23.07	73.77 / 42.23
MetaMath	0.00 / 0.12	0.00 / 0.00

Table 3: Fine-grained `Combination` performance on `Direct Prompt` and `Inverse Prompt` agreed/disagreed subsets.

between `Direct Prompt`and `Inverse Prompt`? When `Inverse Prompt` outputs, for instance, "B, C" as incorrect answers, we consider the remaining options, i.e., "A, D, E" as the correct answer inferred by `Inverse Prompt`. Conflicts arise when `Direct Prompt` and `Inverse Prompt` reach different conclusions. The conflict degree is calculated as the number of conflicts divided by the total number of problems for each dataset.

Table 2 provides a summary of the severity of self-conflict within each LLM. GPT-4 demonstrates the highest consistency and self-confidence, with the lowest conflict percentages across both datasets. GPT-4o shows moderate consistency, performing better on the MathQA dataset than on MATH. Llama-3 exhibits the weakest performance in terms of consistency on the MathQA dataset, with the second-highest conflict rates in the MATH dataset, indicating its unreliability in this analysis. Lastly, MetaMath shows the highest conflict rates in both datasets having 100% of conflict rates. These results underscore the enhanced reliability of advanced models like GPT-4. They also emphasize the interestingness of our work, which leverages the inconsistency in discriminative capability to enhance the certainty in generative decision-making.

$\mathcal{Q}_2$: **How are LLMs performing when their choice is agreed or disagreed by `Direct Prompt` and `Inverse Prompt`?** To answer this question, we check the fine-grained `Combination` performance for the agreed and disagreed subsets between `Direct Prompt` and `Inverse Prompt`.

Table 3 presents the performance of LLMs when they are certain (both `Direct Prompt` and `Inverse Prompt` agree) or uncertain (they conflict). It is clear that when `Direct Prompt` and `Inverse Prompt` agree, the answers are more likely to be correct, demonstrating significantly higher performance than both their disagreed subset and the overall dataset in Table 1. This further suggests that combining `Direct Prompt` and `Inverse Prompt` is an effective method for reducing uncertainty. If `Direct Prompt` and `Inverse Prompt` disagree, a comparison between Table 1 and Table 3 indicates that `Direct Prompt` is the preferred approach. These conclusions generally apply to most LLMs, except for MetaMath, which is non-functional due to its pretraining limitations.

$\mathcal{Q}_3$: **When to suggest using `Direct Prompt` and `Inverse Prompt` to self-improve generation?** Based on Table 1, we can summarize two criteria: i) For top-performing closed-source LLMs like GPT-4 and GPT-4o, using either `Direct Prompt` or `Inverse Prompt`, or their combination `Combination`, shows promise. These top LLMs perform similarly when `Direct Prompt` and `Inverse Prompt` are used separately. Combining them can result in robust performance, but the additional time and budget required for `Combination` may not be appealing. Therefore, the concise conclusion for the top-performing closed-source models is that either `Direct Prompt` or `Inverse Prompt` is sufficient. ii) For open-source LLMs, the decision to try discriminative prompts depends on two factors: a) If the LLMs are not optimized to follow instructions, such as MetaMath, neither `Direct Prompt` nor `Inverse Prompt` is recommended. b) Even if the model is instruction-tuned, open-source LLMs are more likely to struggle with understanding negation, so only `Direct Prompt` is strongly and exclusively recommended.

6 Conclusion

This study analyzed the development of LLM's discriminative capability to enhance generation performance. Specifically, we introduce `Direct-Inverse Discriminative Prompting`, a multi-faceted complementary approach to evaluating LLMs' discriminative potential. Our findings indicate that both `Direct Prompt` and `Inverse Prompt` are effective for closed-source LLMs, while for open-source LLMs, using `Direct Prompt` is highly and solely recommended.

Limitations

Our study is limited by the fact that experiments were conducted using only two datasets. In addition, if budget permits, exploring more closed-source LLMs is preferred.

Ethics Statement

This study uses publicly and automatically accessed datasets, and no ethical issues are present.

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of NAACL-HLT*, pages 2357–2367.

Daman Arora and Subbarao Kambhampati. 2023. Learning and leveraging verifiers to improve planning capabilities of pre-trained language models. *CoRR*, abs/2305.17077.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. 2023a. Universal self-consistency for large language model generation. *Preprint*, arXiv:2311.17311.

Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. 2023b. Teaching large language models to self-debug. *CoRR*, abs/2304.05128.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 16477–16508. Association for Computational Linguistics.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2023. CRITIC: large language models can self-correct with tool-interactive critiquing. *CoRR*, abs/2305.11738.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of NeurIPS*.

Dongwei Jiang, Jingyu Zhang, Orion Weller, Nathaniel Weir, Benjamin Van Durme, and Daniel Khashabi. 2024. Self-[in]correct: Llms struggle with refining self-generated responses. *Preprint*, arXiv:2404.04298.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. 2024. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. *Preprint*, arXiv:2401.10480.

Yixin Liu, Alexander R. Fabbri, Jiawen Chen, Yilun Zhao, Simeng Han, Shafiq Joty, Pengfei Liu, Dragomir Radev, Chien-Sheng Wu, and Arman Cohan. 2023. Benchmarking generation and evaluation capabilities of large language models for instruction controllable summarization. *CoRR*, abs/2311.09184.

Meta. 2024. Build the future of ai with meta llama 3. Accessed: 2024-06-07.

Theo X. Olausson, Jeevana Priya Inala, Chenglong Wang, Jianfeng Gao, and Armando Solar-Lezama. 2023. Demystifying GPT self-repair for code generation. *CoRR*, abs/2306.09896.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2024. Hello gpt-4o. Accessed: 2024-06-07.

Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. REFINER: reasoning feedback on intermediate representations. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, pages 1100–1126. Association for Computational Linguistics.

Tianlu Wang, Ping Yu, Xiaoqing Ellen Tan, Sean O'Brien, Ramakanth Pasunuru, Jane Dwivedi-Yu, Olga Golovneva, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023a. Shepherd: A critic for language model generation. *CoRR*, abs/2308.04592.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

A Example Appendix

A.1 MATH

$\mathcal{Q}$: What is the 100th term of the arithmetic sequence 6, 10, 14, 18, ...?
$\mathcal{R}$: The common difference is \$$10-6=4$\$, so the 100th term is \$$6+99\cdot 4 = boxed\{402\}$\$.

where "$\mathcal{Q}$" denotes questions and "$\mathcal{R}$" for rationale. "$\mathcal{R}$" includes the answer in a specific format which is boxed$\{A\}$, where A is the answer for the problem.

A.2 MathQA

$\mathcal{Q}$: what will be the difference between simple and compound interest at 14 % per annum on a sum of rs . 1000 after 4 years ?
$\mathcal{R}$: s . i . = (1000 * 14 * 4) / 100 = rs . 560 c . i . = [1000 * (1 + 14 / 100) 4 - 1000] = rs . 689 difference = (689 - 560) = rs . 129 answer : a
$\mathcal{O}$: a) 129 , b) 130 , c) 124 , d) 133 , e) 145
$\mathcal{A}$: a

where "$\mathcal{Q}$" denotes questions, "$\mathcal{R}$" for rationale, "$\mathcal{O}$" for options, and "$\mathcal{A}$" for answers.