Are Large Language Models Meta Reasoners?

Anonymous ACL submission

Abstract

In this paper, we introduce Meta-Reasoning Prompting (MRP), a novel approach inspired by human meta-reasoning to enhance the flexibility and generality of large language models (LLMs). Traditional in-context learning techniques, such as Tree-of-Thoughts, show promise but lack consistent state-of-the-art performance across diverse tasks due to their specialized nature. MRP addresses this limitation by dynamically selecting and applying different reasoning methods based on the specific requirements of each task, optimizing both performance and computational efficiency. The MRP framework operates in two phases: initially, the LLM selects the most appropriate reasoning method using task input cues and objective descriptions of available methods; subsequently, it applies the chosen method to complete the task. This dynamic strategy mirrors human meta-reasoning, allowing the model to excel in a wide range of problem domains. We evaluate the effectiveness of MRP through comprehensive benchmarks. The results demonstrate that MRP achieves or approaches state-of-the-art performance across these diverse tasks. MRP represents a significant advancement in enabling LLMs to autonomously select suitable reasoning methods, enhancing their ability to handle diverse and complex problem domains efficiently.

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities in natural language understanding and generation, showing promise in various reasoning tasks. However, the inherent diversity and complexity of real-world problems necessitate advanced reasoning methods that go beyond the capabilities of a single, static approach. Existing in-context learning (ICL) techniques, such as Chain-of-Thoughts (Wei et al., 2022), Tree-of-Thoughts (Yao et al., 2024), Analogical Prompting (Yasunaga et al., 2023), and Solo Performance Prompting (Wang et al., 2023), offer valuable tools for enhancing reasoning but often fall short in consistently achieving state-of-the-art performance across different tasks.

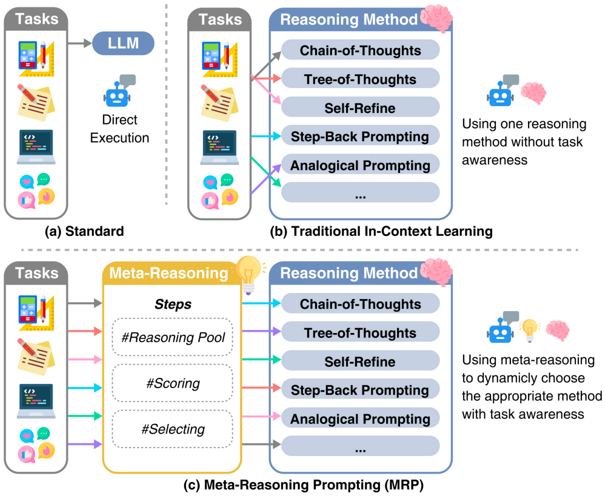


Figure 1: Illustration of Meta-Reasoning Prompting (MRP) and the difference compared to standard reasoning and traditional in-context learning methods.

These challenges highlight the need for a more adaptive and flexible approach to reasoning in LLMs. In human cognition, meta-reasoning refers to the processes that monitor and regulate reasoning and problem-solving activities, adjusting strategies based on the context and specific requirements of the task (Cox and Raja, 2011b,a). This adaptive capability allows humans to efficiently allocate cognitive resources, balancing the trade-offs between accuracy, complexity, and computational cost. Inspired by this, we propose Meta-Reasoning Prompting (MRP) to endow LLMs with similar adaptive reasoning capabilities.

Meta-Reasoning Prompting is a novel framework designed to dynamically select and apply the most suitable reasoning method based on the task at hand. By incorporating meta-reasoning principles, MRP transforms task-specific prompt engineering into a more general and flexible approach. The MRP framework operates in a dynamic mechanism. The LLM evaluates the task input and selects an appropriate reasoning method from a set of predefined methods, which is guided by objective descriptions and evaluations of the available methods. Then, the selected method is then applied to complete the task, ensuring that the model uses the most effective strategy for the given problem.

To evaluate MRP, we conducted experiments using multiple benchmarks. Our findings demonstrate that MRP not only approaches state-of-the-art performance across these benchmarks but also excels in tasks requiring a blend of different reasoning strategies. Additionally, we observe that larger models, such as GPT-4, exhibit superior meta-reasoning capabilities compared to smaller models like GPT-3.5.

Our key contributions are as follows:

1. We introduce Meta-Reasoning Prompting (MRP), inspired by human meta-reasoning, to enhance the flexibility and generality of reasoning with LLMs.

2. Our comprehensive empirical evidence demonstrates that MRP enables LLMs to autonomously select the most appropriate reasoning methods for various tasks, thereby improving overall performance.

2 Meta Reasoning Prompting

We formally introduce the Meta Reasoning Prompting algorithm. The goal is to select the most suitable reasoning method from a pool of available methods to process a given input, thereby enhancing the overall reasoning performance of the model. Detailed prompts could be found in AppendixA.1.

Meta Reasoning Prompting begins with an input x_0 and a pre-trained model M. A set of reasoning methods $\alpha_1, \alpha_2, \ldots, \alpha_n$ are available for selection. A reasoning pool containing descriptions of each reasoning method in the form of prompts $p_1, p_2, \ldots, p_n$, is utilized. Specifically, the description of the method is extracted from the abstract of the corresponding papers. A meta reasoning prompt p_{MR} is defined to guide the selection process. For each reasoning method α_i, where i ranges from 1 to n, the model M is used to evaluate the combined prompt $(p_i|p_{MR}|x_0)$. This evaluation yields a score s_i indicating the effectiveness of the method α_i for the given input x_0.

$$s_i = M(p_i \| p_{MR} \| x_0) \quad \text{for} \quad i = 1, 2, \ldots, n. \tag{1}$$

The algorithm identifies the reasoning method α_k that receives the highest score s_i by finding the index k that maximizes the set $s_1, s_2, \ldots, s_n$.

$$k = \arg\max_i \{s_1, s_2, \ldots, s_n\} \tag{2}$$

Once the best reasoning method α_k is determined, it is executed on the input x_0. The model M generates the final output y_0 using the prompt $(p_k|x_0)$, which combines the description of the chosen reasoning method with the original input.

$$y_0 = \alpha_k(x_0) \tag{3}$$

Algorithm 1 Meta Reasoning Prompting

Require: Input x_0, model M, reasoning methods $\alpha_1, \alpha_2, \ldots, \alpha_n$, a set of prompts of reasoning method descriptions $p_1, p_2, \ldots, p_n$, the prompt for meta reasoning process denoted as p_{MR}. Evaluation score is denoted as $s_1, s_2, \ldots, s_n$.

for $i = 1$ **to** n **do**
  $s_i = M(p_i \| p_{MR} \| x_0)$
end
$k = \arg\max_i \{s_1, s_2, \ldots, s_n\}$
Determine k for which α_k is executed and reason with the chosen method.
$y_0 = \alpha_k(x_0)$
return y_0

3 Experiments

3.1 Setup

Instantiation of Meta-Reasoning Prompting We instantiate MRP with seven popular in-context learning methods, which also serve as our baseline for comparison. We prompt descriptions for each method, allowing the LLM to understand. Details and prompts are available in Appendix A.1.

Tasks We experiment with seven diverse tasks: **(1) Arithmetic Reasoning:** GSM8K (Cobbe et al., 2021), 1319 basic math questions. **(2) Complex Mathematical Reasoning:** Game of 24 (Yao et al., 2024), a game using 4 numbers and basic arithmetic four operations to obtain 24. **(3) Creative Writing:** Trivia Creative Writing (Trivia CW) (Wang et al., 2023; Joshi et al., 2017), necessitating the model to assimilate and combine heterogeneous

Table 1: Experiments with GPT4: Comparison of effects on benchmarks using Meta-Reasoning Prompting versus using other methods independently. **Bold** represents the best performance, and underline represents the second-best performance.

Method	GSM8K	Gameof24	Trivia CW	HotpotQA	BigToM	Code	MMLU	Macro Avg.
COT	0.914	0.050	0.762	0.800	0.470	0.685	0.894	0.654
TOT	**0.942**	**0.410**	0.786	0.716	0.430	0.765	0.815	0.725
Analogical	0.924	0.040	0.735	0.777	0.500	0.614	**0.947**	0.648
Self-Refine	0.929	0.080	0.764	0.763	0.470	**0.872**	0.861	0.677
SPP	0.929	0.170	**0.861**	0.763	0.550	0.672	0.874	0.688
STEP-BACK	0.933	0.090	0.787	**0.810**	0.420	0.809	0.841	0.670
SimTom	0.938	0.040	0.739	0.667	**0.590**	0.694	0.815	0.640
MRP (our)	0.921	0.310	0.796	0.797	0.570	0.867	0.854	**0.772**

information from multiple domains internally. **(4) Multi-Hop Reasoning:** HotpotQA, (Yang et al., 2018), requiring models to connect pieces of information from multiple documents to answer a question. **(5) Social Reasoning:** BigToM (Gandhi et al., 2024), to evaluate social situations understanding and the theory of mind. **(6) Computer Code:** Code Readability (Code) (Puri et al., 2021), to enhance the readability of given code snippets. **(7) STEM:** MMLU (Hendrycks et al., 2020), Physics, Chemistry, Biology, and Math problems of high school domain. To prevent any method from skewing the results due to exceptional performance on a specific task, we reported both the arithmetic mean accuracy and the harmonic mean accuracy of each method across all benchmarks.

Details about the dataset and its construction are provided in Appendix A.2.

Models We used gpt-3.5-turbo[1] and gpt-4-turbo[2] with identical prompts to compare the effect of model size on meta-reasoning ability.

Baseline Methods We select seven popular reasoning methods as baselines rather than standard input-output. These methods include: **(1) Chain-of-Thoughts:** breaking down problems into a series of coherent reasoning steps (Wei et al., 2022). **(2) Tree-of-Thoughts:** exploring multiple reasoning paths and self-evaluating choices to solve complex problems (Yao et al., 2024). **(3) Analogical prompting:** self-generating few-shots based on past experiences and related problems (Yasunaga et al., 2023). **(4) Self-Refine:** self-evaluating for refinement and continuously improving the output (Madaan et al., 2024). **(5) Solo Performance Prompting:** simulating multiple personas to collaboratively solve complex tasks (Wang et al., 2023). **(6) Step-Back Prompting:** abstract high-level concepts and principles to guide the reasoning process (Zheng et al., 2023). **(7) SimToM:** enabling perspective-taking to understand the character's beliefs and goals (Wilf et al., 2023)

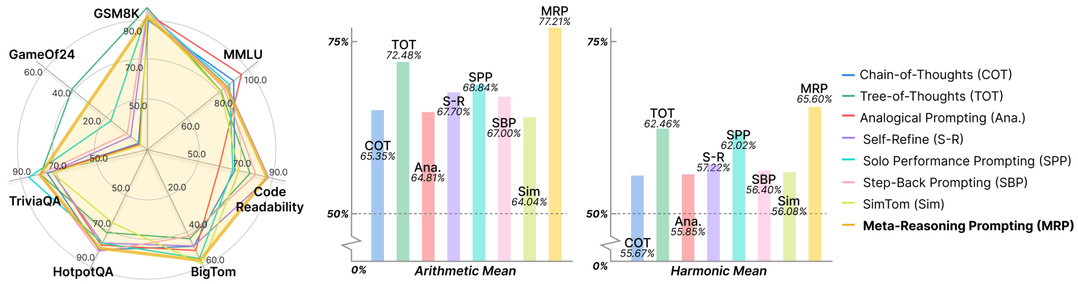
3.2 Main Results

Meta-Reasoning Prompting performs best on comprehensive tasks As shown in table 1, MRP consistently exhibits robust performance across multiple benchmarks. MRP achieves the **second-best** in **4 of 7** tasks, including **Gameof24**, **TriviaQA**, **BigToM** and **Code**. This impressive performance across a wide range of tasks demonstrates MRP's ability to effectively select and apply appropriate reasoning methods tailored to the specific requirements of each task. In terms of overall performance, MRP attains the highest across the 7 tasks, with an average of 0.772. In contrast, although TOT excels in certain tasks such as GSM8K and Gameof24, it performs less impressively in others. We observe noticeable performance gaps compared with MRP in tasks such as BigToM (**0.43 VS 0.57**) and Code (**0.765 VS 0.867**). This consistent excellence across all benchmarks underscores MRP's omnipotence, demonstrating its ability to maintain impressive performance across diverse task domains.

Meta-reasoning capability is influenced by the base model capability As illustrated in Appendix C table 3 in the appendix, while the performance with GPT-4 is satisfactory, the experimental results with GPT-3.5 indicate that the effectiveness of MRP is suboptimal. This suggests that the potential of MRP shows some consistency across smaller-scale basic language models (LLMs), but does not fully realize its potential. This consistency can be attributed to the fact that in a more advanced LLM, such as GPT-4, MRP can leverage the model's full capabilities, whereas in a base model like GPT-3.5,

[1] Azure OpenAI, Model Name: gpt-35-turbo, API Version: 0301

[2] Azure OpenAI, Model Name: gpt-4, API Version: 1106-Preview



(a) Performance on seven individual datasets (b) Performance on comprehensive tasks (Two calculation methods)

Figure 2: (a) Comparison of methods on different benchmarks reveals that guiding LLM to dynamically choose the appropriate reasoning method enables MRP to achieve consistently better performance across all tasks. (b) The arithmetic and harmonic average performances of applying a specific reasoning approach to all benchmarks demonstrate that MRP consistently excels in overall evaluation.

the impact of MRP is constrained by the model's inherent limitations.

4 Related Works

4.1 Endowing LLMs with Multiple Reasoning Methods

Independent reasoning methods have been proven to improve LLM performance from different perspectives but fail to meet integrated problems (Madaan et al., 2024; Wilf et al., 2023; Zheng et al., 2023; Sahoo et al., 2024; Suzgun and Kalai, 2024; Chen et al., 2024; Besta et al., 2024). More recently, more researchers have begun to consider adapting LLMs to various reasoning methods. X-of-Thoughts improves the success rate of LLM on arithmetic problems by integrating three methods (Liu et al., 2023). It proposes a trial-and-error iterative mechanism that allows LLM to autonomously repeat attempts to find a final solution. Yuan et al. fine-tune smaller models with a well-prepared dataset inspired by preference learning to achieve reasoning power comparable to a larger model (Yuan et al., 2024). It presents a problem-method coupled dataset and shows how to improve the model's grasp of inference skills at the data level. However, there is still a lack of research to explore the meta-reasoning ability of LLMs to choose reasoning methods.

4.2 Meta Reasoning

Meta-reasoning is a crucial cognitive process in human intelligence, involving the recognition and interpretation of reasoning to select optimal methods based on past experiences (Griffiths et al., 2019). In artificial intelligence, it refers to efficiently deploying computational resources for informed decision-making in specific situations (Cox and Raja, 2011a,b). Recently, some works develop routing or buffer systems to improve performance, using supervised learning algorithms (Shnitzer et al., 2023), reward model-based techniques, and other methods (Hari and Thomson, 2023; Lu et al., 2024; Wang et al., 2024). Hu et al. created a benchmark to evaluate these methods' effectiveness (Hu et al.). Zeng et al. noted the neglect of meta-reasoning in independent LLMs and proposed a benchmark to evaluate reasoning rationality (Zeng et al., 2024). In (Yang et al., 2024), the authors introduce a meta-buffer to store a series of high-level thoughts distilled from problem-solving processes across various tasks. This approach aligns with the inherent logic of meta reasoning. However, MRP achieves simple and efficient meta-cognitive effects by directly unleashing the meta reasoning capabilities of LLM through prompts, without introducing complex mechanisms.

5 Conclusions

This paper introduces Meta-Reasoning Prompting (MRP), a novel approach inspired by human meta-reasoning to enhance the flexibility and generality of reasoning with large language models (LLMs). Guided by MRP, an LLM dynamically selects and applies the most suitable reasoning method for each task. Our experiments demonstrate that MRP achieves near state-of-the-art performance across diverse benchmarks, achieving the highest comprehensive weighted accuracy across all tasks. This adaptability underscores the promise of meta-reasoning prompting in enhancing LLMs' general reasoning capabilities. Future research could explore using MRP to construct instruction -tuning datasets, further boosting the general reasoning abilities of LLMs.

6 Limitations

Our study investigates the meta-reasoning mechanisms of LLMs by dynamically selecting suitable methods to enhance their performance across various reasoning tasks without introducing new knowledge or training efforts. Currently, Meta-Reasoning Prompting (MRP) selects the highest-scoring method for each task. However, drawing from human cognitive processes, tackling complex problems often involves combining multiple reasoning methods. Future research will explore mechanisms such as Top-Probability (Top-P) or Top-K to allow models to ensemble relevant methods, potentially achieving better performance.

Our experimental results indicate that the meta-reasoning ability of LLMs is influenced by the capabilities of the models themselves. For instance, GPT-4's meta-reasoning prompting shows significantly greater improvement compared to GPT-3.5, which aligns with our expectations. Nonetheless, we can further enhance the smaller model's meta-reasoning capabilities through instruction tuning in future work.

Due to space constraints and limited resources, our experiments primarily tested the most representative LLMs (GPT-4 and GPT-3.5). We did not fully cover the performance of other open-source or closed-source models. However, we believe that the experimental results on these representative LLMs provide sufficient insights and implications.

Acknowledgments

We would like to acknowledge the use of ChatGPT for assisting in checking grammar errors and improving the expression of this paper. The assistance provided was purely focused on enhancing the language of the manuscript. No new content or ideas were suggested by ChatGPT.

References

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pages 17682–17690.

Sijia Chen, Baochun Li, and Di Niu. 2024. Boosting of thoughts: Trial-and-error problem solving with large language models. arXiv preprint arXiv:2402.11140.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168.

Michael T Cox and Anita Raja. 2011a. Metareasoning: an introduction.

Michael T Cox and Anita Raja. 2011b. Metareasoning: Thinking about thinking. MIT Press.

Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah Goodman. 2024. Understanding social reasoning in language models with language models. Advances in Neural Information Processing Systems, 36.

Thomas L Griffiths, Frederick Callaway, Michael B Chang, Erin Grant, Paul M Krueger, and Falk Lieder. 2019. Doing more with less: meta-reasoning and meta-learning in humans and machines. Current Opinion in Behavioral Sciences, 29:24–30.

Surya Narayanan Hari and Matt Thomson. 2023. Tryage: Real-time, intelligent routing of user prompts to large language model. arXiv preprint arXiv:2308.11601.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. arXiv preprint arXiv:2009.03300.

Qitian Jason Hu, Jacob Bieker, Xiuyu Li, Nan Jiang, Benjamin Keigwin, Gaurav Ranganath, Kurt Keutzer, and Shriyash Kaustubh Upadhyay. Mars: A benchmark for multi-llm algorithmic routing system. In ICLR 2024 Workshop: How Far Are We From AGI.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. arXiv preprint arXiv:1705.03551.

Tengxiao Liu, Qipeng Guo, Yuqing Yang, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. Plan, verify and switch: Integrated reasoning with diverse x-of-thoughts. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 2807–2822.

Xiaoding Lu, Adian Liusie, Vyas Raina, Yuwen Zhang, and William Beauchamp. 2024. Blending is all you need: Cheaper, better alternative to trillion-parameters llm. arXiv preprint arXiv:2401.02994.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. Advances in Neural Information Processing Systems, 36.

Ruchir Puri, David S Kung, Geert Janssen, Wei Zhang, Giacomo Domeniconi, Vladimir Zolotov, Julian Dolby, Jie Chen, Mihir Choudhury, Lindsey Decker, et al. 2021. Codenet: A large-scale ai for code dataset for learning a diversity of coding tasks. arXiv preprint arXiv:2105.12655.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. arXiv preprint arXiv:2402.07927.

Tal Shnitzer, Anthony Ou, Mírian Silva, Kate Soule, Yuekai Sun, Justin Solomon, Neil Thompson, and Mikhail Yurochkin. 2023. Large language model routing with benchmark datasets. arXiv preprint arXiv:2309.15789.

Mirac Suzgun and Adam Tauman Kalai. 2024. Meta-prompting: Enhancing language models with task-agnostic scaffolding. arXiv preprint arXiv:2401.12954.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. 2024. Mixture-of-agents enhances large language model capabilities. arXiv preprint arXiv:2406.04692.

Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. 2023. Unleashing cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. arXiv preprint arXiv:2307.05300, 1(2):3.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35:24824–24837.

Alex Wilf, Sihyun Shawn Lee, Paul Pu Liang, and Louis-Philippe Morency. 2023. Think twice: Perspective-taking improves large language models' theory-of-mind capabilities. arXiv preprint arXiv:2311.10227.

Ling Yang, Zhaochen Yu, Tianjun Zhang, Shiyi Cao, Minkai Xu, Wentao Zhang, Joseph E Gonzalez, and Bin Cui. 2024. Buffer of thoughts: Thought-augmented reasoning with large language models. arXiv preprint arXiv:2406.04271.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. arXiv preprint arXiv:1809.09600.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. Advances in Neural Information Processing Systems, 36.

Michihiro Yasunaga, Xinyun Chen, Yujia Li, Panupong Pasupat, Jure Leskovec, Percy Liang, Ed H Chi, and Denny Zhou. 2023. Large language models as analogical reasoners. arXiv preprint arXiv:2310.01714.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, et al. 2024. Advancing llm reasoning generalists with preference trees. arXiv preprint arXiv:2404.02078.

Zhongshen Zeng, Pengguang Chen, Shu Liu, Haiyun Jiang, and Jiaya Jia. 2024. Mr-gsm8k: A meta-reasoning revolution in large language model evaluation. Preprint, arXiv:2312.17080.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023. Take a step back: Evoking reasoning via abstraction in large language models. arXiv preprint arXiv:2310.06117.

A Implementation Details

A.1 Meta-Reasoning Prompting Instantiation

MRP serve as an automatic reasoning method adaptor, and we instantiate it with seven popular in-context learning methods in this paper. For practical use, any number of methods and their descriptions can be added to the framework we provide.

System Prompts of Meta-Reasoning Prompting (MRP)

You are an adaptive reasoning method with meta-reasoning abilities, capable of selecting the most appropriate reasoning method based on the task provided by the user.
Using a meta-reasoning thinking pathway, choose exactly one of the methods listed below for each objective task you receive. Avoid using any methods not provided.

Main Prompts of Meta-Reasoning Prompting (MRP)

Instructions :
Please adhere to the following guidelines when answering questions.
Reasoning method pool:
These reasoning methods are available for your tasks. Understand their applications for various tasks.

1. Method 1:
{Method_description}
2. Method 2:
{Method_description}
......
......

Analyzing giving task:
Evaluate the problem's difficulty carefully. Avoid underestimating the complexity and make a considered decision.
Anticipate the mistakes you might make:
Identify possible errors in method selection, such as underestimating the problem's difficulty.
Meta reasoning:
Apply meta-reasoning to choose the appropriate pathway.
Grading:
Rate the suitability of each solution on a scale of 1-7, selecting the method with the highest score as your preferred choice.
Choosing reasoning method:
Proceed methodically, taking a deep breath and thinking step-by-step. Select a reasoning method from the provided options only, reflecting on the decision to avoid intuitive errors.
The question is:
{user_query}
Output:
>> FINAL CHOICE:[final method choice]

Figure 3: System and main prompts of MRP

Table 2: Dataset Split and Number of Examples

Domain	Dataset	Split	Number of Examples
Arithmetic Reasoning	GSM8K	Test	1319
Complex Mathematical Reasoning	Game of 24	Test	100
Creative Writing	Trivia Creative Writing	Test	100
Multi-hop Reasoning	HotpotQA	Test	300
Social Reasoning	BigTOM	Dev	100
Computer Code	Code Readability	Test	300
STEM	MMLU	Dev	151

Table 3: Experiments with GPT3.5: Comparison of effects on benchmarks using Meta-Reasoning Prompting versus using other methods independently. **Bold** represents the best performance, and underline represents the second-best performance.

Method	**GSM8K**	**Gameof24**	**Trivia CW**	**HotpotQA**	**BigToM**	**Code**	**MMLU**	**Avg.**
COT	**0.831**	0.030	0.414	0.187	0.610	0.578	0.675	0.416
TOT	0.810	0.100	0.155	**0.360**	0.430	**0.797**	0.735	0.352
Self-Refine	0.716	0.030	0.213	0.167	**0.650**	0.796	0.543	0.372
SPP	0.823	**0.160**	**0.536**	0.217	0.540	0.684	0.689	**0.469**
STEP-BACK	0.817	0.010	**0.536**	0.190	0.570	0.642	**0.788**	0.452
SimTom	0.586	0.040	0.240	0.177	0.460	0.599	0.503	0.315
MRP (our)	0.781	0.050	0.346	0.187	0.600	0.759	0.722	0.433

A.2 Dataset Details

Table 2 shows the split and number of examples used for evaluations in GSM8K, Game of 24, Trivia Creative Writing, HotpotQA, BigTOM, Code Readability and MMLU. The dataset sizes of GSM8K, Gameof24, Trivia Creative Writing are consistent with the size used in the references. And considering cost, we tested a random 100-300 sample of data from HotpotQA, BigTOM, and Code Readability and MMLU. Despite of the economic consideration, we found that on this data scale, MRP has achieved significant results.

A.3 Source Prompts of Other In-Context Learning Methods Used in This Paper

Full prompts of COT

```
Task Description:
{description}

Input:
{input}

Let's think step by step.
```

Figure 4: Prompt of COT

Full prompts of SIMTOM for general task

```
STEP 1:
Task Description:
{description}

Input:
{input}

What does you know about as an AI assistant to solve this {domain} problem?
STEP 2:
Task Description:
{description}

Input:
{input}

{principle}
```

Full prompts of SIMTOM for BigTom

```
STEP 1:
Imagine you are {coi}, and consider this story that has an unexpected event.

If the last sentence of the story says {coi} notices, sees or realizes the unexpected
event in this story, simply output the original story with nothing changed.
However, if the sentence says you are not aware of the changes in this story, output
only the events you know, i.e., the sentences before the unexpected event happens.
Output either the original story or the edited story, nothing else.
Format your answer as follows: Sees/Notices/Realizes: (Yes/No)

Story:

STEP 2:
{principle}

You are {coi}. Based on the above information, answer the following question:
{input}
Answer the questions based on the context. Keep your answer concise, few words are
enough, maximum one sentence.
```

Figure 5: Prompt of Sim-Tom Prompting

Full prompts of TOT

Proposal:
Task Description:
{description}

Input:
{input}

Make a strategy then write. Your output should be of the following format:
Strategy:
Your strategy about how to answer the question.
Answer:
Your answer to the question. It should end with {format}.

Vote:
Given an instruction and several choices, decide which choice is most promising.
Analyze each choice in detail, then conclude in the last line
"The best choice is {s}", where s the integer id of the choice.

Figure 6: Prompt of TOT

Full prompts of Analogical Prompting

Your task is to tackle mathematical problems. When presented with a math problem, recall relevant problems as examples. Afterward, proceed to solve the initial problem.

Initial Problems
Task Description:
{description}

Input:
{input}

Instructions :
Make sure that your response follows the instructions below.

Analogous Problems :
Offer five diverse examples of math problems that are relevant or analogous to the initial problem. For each problem, elaborate on the solution and conclude with the ultimate answer.
For each problem :
- After " Q : ", describe the problem.
- After " A : ", explain the solution and enclose the ultimate answer in here.

Solve the Initial Problem :
Q : Copy and paste the initial problem here.
A : Explain the solution and enclose the ultimate answer use the format that system required.

Figure 7: Prompt of Analogical Prompting

Full prompts of SELF-REFINE

STEP 1:
You are an expert in {domain}.

Task Description:
{description}

Input:
{input}

Once you think you have answered the question, please summarize your final answer strictly using the format below:{format} without other word."

STEP2:
Task Description:
{description}

Input:
{input}

Origin Solution:
{principle}

There is maybe an error in the problem solving process above because of lack of understanding of the question.
you need to go through complete blocks of the context, check if everything looks good and give your revised answer in the end with the prescribed format.
please answer with the format below:
Judge: tell me whether the problem solving process has error here and explain.
Answer: give your revised solution, and summarize the final answer in the end with the prescribed format without other word.

Figure 8: Prompt of SELF-REFINE

Full prompts of STEP-BACK Prompting

STEP 1:
You are an expert at {domain}.
You are given a {domain} problem.
Your task is to extract the {domain} concepts and principles involved in solving the problem.

Here are a few examples:
Question:
Griffin had 24 french fries, but Kyle took 5 of them. Billy took twice as many as Kyle. Ginger gave Griffin a handful of her fries, and then Colby took from Griffin 3 less than the number of fries that Kyle had taken. If in the end Griffin had 27 fries, how many fries did Ginger give Griffin?
Principles Involved:
1.Basic Arithmetic Operations: This problem primarily involves addition and subtraction. We use these operations to calculate the changes in the number of fries Griffin has after each event.
2.Multiplication and Doubling: The problem states that Billy took twice as many fries as Kyle. This involves the principle of multiplication or doubling a number.
3.Comparative Subtraction: The statement that Colby took 3 fries less than Kyle introduces the concept of comparative subtraction, where one value is reduced by a relative amount compared to another.
4.Variable Isolation in Equations: To find out how many fries Ginger gave Griffin, we first calculate the total number of fries taken or given and then isolate the variable representing the fries Ginger gave.
5.Equation Setup and Solving: The entire scenario can be conceptualized as an equation where the final number of fries is affected by a series of additions and subtractions. Setting up an equation with these operations and solving for the unknown (the fries Ginger gave) is key.

Task Description:{description}
Question:{input}
Principles Involved:

STEP 2:
You are an expert at {domain}.
You are given a {domain} problem and a set of principles involved in solving the problem.
Solve the problem step by step by following the principles.

Here are a few examples:
Question:
Griffin had 24 french fries, but Kyle took 5 of them. Billy took twice as many as Kyle. Ginger gave Griffin a handful of her fries, and then Colby took from Griffin 3 less than the number of fries that Kyle had taken. If in the end Griffin had 27 fries, how many fries did Ginger give Griffin?
Principles Involved:
1.Basic Arithmetic Operations: This problem primarily involves addition and subtraction. We use these operations to calculate the changes in the number of fries Griffin has after each event.
2.Multiplication and Doubling: The problem states that Billy took twice as many fries as Kyle. This involves the principle of multiplication or doubling a number.
3.Comparative Subtraction: The statement that Colby took 3 fries less than Kyle introduces the concept of comparative subtraction, where one value is reduced by a relative amount compared to another.
4.Variable Isolation in Equations: To find out how many fries Ginger gave Griffin, we first calculate the total number of fries taken or given and then isolate the variable representing the fries Ginger gave.
5.Equation Setup and Solving: The entire scenario can be conceptualized as an equation where the final number of fries is affected by a series of additions and subtractions. Setting up an equation with these operations and solving for the unknown (the fries Ginger gave) is key.
Answer:
To solve this problem, let's follow the principles and the order of events to find out how many fries Ginger gave Griffin.
Start with the initial number of fries Griffin had, which is 24.
Kyle took 5 fries from Griffin:
Griffin's fries = 24 - 5 = 19 fries.
Billy took twice as many as Kyle, so Billy took 5 • 2 = 10 fries:
Griffin's fries = 19 - 10 = 9 fries.
Colby took 3 fries less than Kyle took. Kyle took 5 fries, so Colby took 5 - 3 = 2 fries:
Griffin's fries = 9 - 2 = 7 fries.
Finally, Ginger gave Griffin some fries. We know that at the end of all these transactions, Griffin had 27 fries. Therefore, we set up an equation based on the fries left after Billy and Colby took theirs:
Griffin's fries after Ginger gave some = 7 + fries given by Ginger = 27 fries.
Solving for the fries Ginger gave:
Fries given by Ginger = 27 - 7 = 20 fries.
Therefore, Ginger gave Griffin 20 fries.
>> FINAL ANSWER:
\"\"\"
Ginger gave Griffin 20 fries.
\"\"\"

Task Description:{description}
Question:
{input}
Principles Involved:{principle}
Answer:

Figure 9: Prompt of STEP-BACK Prompting

Full prompts of Solo Performance Prompting (SPP)

```
When faced with a task, begin by identifying the participants who will contribute to
solving the task. Provide profiles of the participants, describing their expertise or
needs. Then, initiate a multi-round collaboration process until a final solution is
reached. The participants will give critical comments and detailed suggestions
whenever necessary.

Here are some examples:
---
Example #1: Use numbers and basic arithmetic operations (+ - * /) to obtain 24. You
need to use all numbers, and each number can only be used once.
Input: 6 12 1 1

Participants: AI Assistant (you); Math Expert

Profiles:
  - AI Assistant (you): A super-intelligent AI assistant capable of performing tasks
more effectively than humans.
  - Math expert: A person who is good at math games, arithmetic calculation, and
long-term planning.

Start collaboration!

Math Expert: Let's analyze the task in detail. You need to make sure that you meet the
requirement, that you need to use exactly the four numbers (6 12 1 1) to construct 24.
To reach 24, you can think of the common divisors of 24 such as 4, 6, 8, 3 and try to
construct these first. Also you need to think of potential additions that can reach 24,
such as 12 + 12.
AI Assistant (you): Thanks for the hints! Here's one initial solution: (12 / (1 + 1)) * 6 =
24
Math Expert: Let's check the answer step by step. (1+1) = 2, (12 / 2) = 6, 6 * 6 = 36
which is not 24! The answer is not correct. Can you fix this by considering other
combinations? Please do not make similar mistakes.
AI Assistant (you): Thanks for pointing out the mistake. Here is a revised solution
considering 24 can also be reached by 3 * 8: (6 + 1 + 1) * (12 / 4) = 24.
Math Expert: Let's first check if the calculation is correct. (6 + 1 + 1) = 8, 12 / 4 = 3, 8 *
3 = 24. The calculation is correct, but you used 6 1 1 12 4 which is not the same as the
input 6 12 1 1. Can you avoid using a number that is not part of the input?
AI Assistant (you): You are right, here is a revised solution considering 24 can be
reached by 12 + 12 and without using any additional numbers: 6 * (1 - 1) + 12 = 24.
Math Expert: Let's check the answer again. 1 - 1 = 0, 6 * 0 = 0, 0 + 12 = 12. I believe
you are very close, here is a hint: try to change the "1 - 1" to "1 + 1".
AI Assistant (you): Sure, here is the corrected answer:  6 * (1+1) + 12 = 24
Math Expert: Let's verify the solution. 1 + 1 = 2, 6 * 2 = 12, 12 + 12 = 12. You used 1 1 6
12 which is identical to the input 6 12 1 1. Everything looks good!

Finish collaboration!

>> FINAL ANSWER:
\"\"\"
6 * (1 + 1) + 12 = 24
\"\"\"

---
Example #2: Write a poem that meets the following requirements: (1) the poem has
seven lines and the first letters of each line forms the word "CHATGPT"; (2) the poem
is about explaining what is a quantum computer. (3) the poem needs to be easy to
understand by a ten years old kid.

Participants: AI Assistant (you); Poet; Computer Scientist; Ten year old child

Profiles:
  - AI Assistant (you): A super-intelligent AI assistant capable of performing tasks
more effectively than humans.
  - Poet: A person who studies and creates poetry. The poet is familiar with the rules
and formats of poetry and can provide guidance on how to write a poem.
  - Computer Scientist: A scholar who specializes in the academic study of computer
science. The computer scientist is familiar with the concept of a quantum computer
and can provide guidance on how to explain it.
  - Ten year old child: A child with a limited English vocabulary and little knowledge
about complicated concepts, such as a quantum computer.

Poet: Make sure that you write the poem with seven lines, and the first letters of the
lines should be C, H, A, T, G, P, T.
Computer Scientist: A quantum computer is an advanced computing device that uses
the principles of quantum mechanics to process and store information. Unlike classical
computers that use bits to represent information as 0s and 1s, quantum computers
use quantum bits or qubits. Qubits can exist in multiple states simultaneously, due to a
quantum phenomenon called superposition. You can consider using these information
for the poem.
Ten year old child: I hope the poem to be fun and easy to understanding. I don't want
to see a lot of jargons or complicated concepts.
AI Assistant (you): Thanks for the guidance! Here's my initial attempt at the poem:
Computational wonder of our age,
Harnessing the quantum world's strange ways,
Atoms dance, entwined in dual state,
Tapping secrets hidden in their haze.

Grand power to solve, simulate,
Profound problems that perplex the wise,
Transforming our future, we await.
```

Figure 10: Prompt of SPP Prompting

Full prompts of Solo Performance Prompting (SPP)

```
Poet: Let's verify if the poem meets the requirements. The first letters are CHATGPT
which is correct! And the poem rhymes well. Good job!
Computer Scientist: Everything looks good to me!
Ten year old child: I don't know what does perplex mean. Can you make the use of
words easier to understand?
AI Assistant (you): Sure, let me revise the poem by using more common words. Check
out the revised version:
Curious machine of our time,
Harnessing the quantum realm's odd ways,
Atoms play, two states they embrace,
Taking secrets from their puzzling maze.

Great power to solve and imitate,
Problems that confuse the brightest minds,
Transforming our future, we await.

Poet: Let's check again if the poem meets the requirements. The first letters are C H A
T G P T. And now the poem is more accessible to children. Everything looks good to
me.
Computer Scientist: Looking good!
Ten year old child: I like this version a lot!

Finish collaboration!

>> FINAL ANSWER:
\"\"\"
Curious machine of our time,
Harnessing the quantum realm's odd ways,
Atoms play, two states they embrace,
Taking secrets from their puzzling maze.

Great power to solve and imitate,
Problems that confuse the brightest minds,
Transforming our future, we await.
\"\"\"

---
Task Description:
{description}

Input:
{input}
Now, identify the participants, provide their profiles, and collaboratively solve the
following task step by step. Remember to provide the final solution with the format
that 'system' requires:
```

Figure 11: Prompt of SPP Prompting

B META REASONING PROMPTING IN SIMPLE TASKS

From the experimental results (see figure 12), it can be seen that MRP and other methods show equal competitiveness on GSM8K, the accuracy of all the reasoning methods is above 90%, but the differentiation between the accuracy of each method is not very high, it can be seen that when the task is simpler, it is harder for MRP to reflect its own advantages, but MRP method is better than each method on the more difficult and comprehensive But the MRP method is significantly better than the other methods in the more difficult and comprehensive tasks.

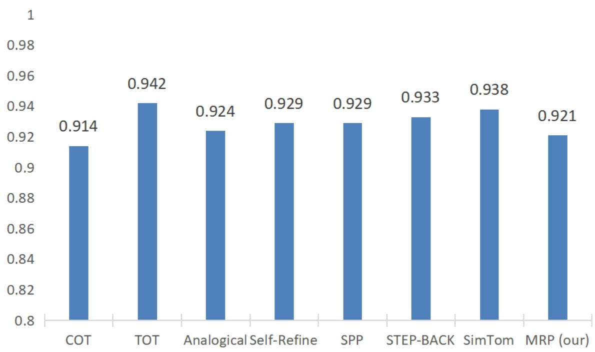


Figure 12: Performance of seven reasoning methods on GSM8K benchmark

C Experiments with GPT3.5

Table 3 demonstrates the results of GPT3.5 under the same experimental conditions as before. When using GPT3.5 as the basic model, MRP performed poorly. Through Error analysis, we found that the main reasons included Scoring Error, Self-opinion, Factual Error and Reasoning Error.