DOP: Diagnostic-Oriented Prompting for Large Language Models in Mathematical Correction

Anonymous ACL submission

Abstract

Math world problems correction(MWPC) is a novel task dedicated to rectifying reasoning errors in the process of solving mathematical problems. In this paper, leveraging the advancements in **large language models (LLMs)**, we address two key objectives:(1) Distinguishing between mathematical reasoning and error correction; (2) Exploring strategies to enhance the error correction capabilities of LLMs in mathematics to solve MWPC task. We noticed that, in real-time education,assisting students in recognizing their mistakes is more crucial than simply providing correct answers. However, current research tends to prioritize obtaining accurate solutions to math problems rather than correcting potentially incorrect ones. Therefore, we modify the research paradigm, demonstrating that improving mathematical reasoning abilities does not equate to mastery in error correction. Meanwhile, we propose a novel method called **diagnostic-oriented promping(DOP)** aimed at facilitating LLMs to excel in error correction. In experiments, DOP has shown outstanding performance, highlighting its significant impact. We argue that in mathematical education, the demand for outstanding correctors surpasses that for proficient reasoners. Codes and data are available on https://github.com/ChenhaoEcnuCS/Reason-Correct.

1 Introduction

> **"Give a man a fish and you feed him for a day; Teach a man to fish and you feed him for a lifetime."**
>
> **—-Huainanzi**

In recent years, the rapid advancement of large language models(LLMs)(Zhao et al., 2023) has profoundly reshaped the landscape of artificial intelligence research. The remarkable capabilities exhibited by prominent models like GPT-4 (OpenAI, 2023), LLama2 (Touvron et al., 2023), among others, have sparked innovative approaches across diverse domains of study.

In mathematics domain, numerous studies(Wei et al., 2022; Kojima et al., 2022; Wang et al., 2023a,c; Zhang et al., 2023; An et al., 2023; Liu et al., 2023b; Liu and Low, 2023; Yu et al., 2023; Luo et al., 2023) have focused on the task of solving math world problems(MWPs). Some have employed diverse prompting strategies (Wei et al., 2022; Kojima et al., 2022; Wang et al., 2023a,c; Zhang et al., 2023) to enhance the reasoning capabilities of LLMs, while others (An et al., 2023; Liu et al., 2023b; Liu and Low, 2023; Yu et al., 2023; Luo et al., 2023) have fine-tuned models for mathematical tasks using domain-specific corpora.

However, we observe that most of these approaches primarily focus on achieving accuracy in solving MWPs. We often overlook the key point: merely enhancing the ability of a large language model to solve MWPs correctly falls short in mathematics pedagogy scenarios.

In real life, good students may be good at solving MWPs, but struggle to mentor their peers. Conversely, parents who may encounter difficulties in solving MWPs themselves can effectively coach their children using educational resources. This observation underscores the importance of **focusing not just on a model's ability to solve problems, but also on its capacity to correct errors and provide guidance.** With LLMs, an significant objective is instructing them to assist students in identifying and correcting their mistakes.

We first distinguish the concept of reasoning and correcting. As shown in Figure 1, in educational scenarios, the capacity for reasoning aids students in providing correct answers, whereas error correction empowers teachers to guide students through the process of identifying and rectifying mistakes in their responses. Our research mainly discussed those abilities in mathematics domain.

Therefore, we begin with a research question: **is**

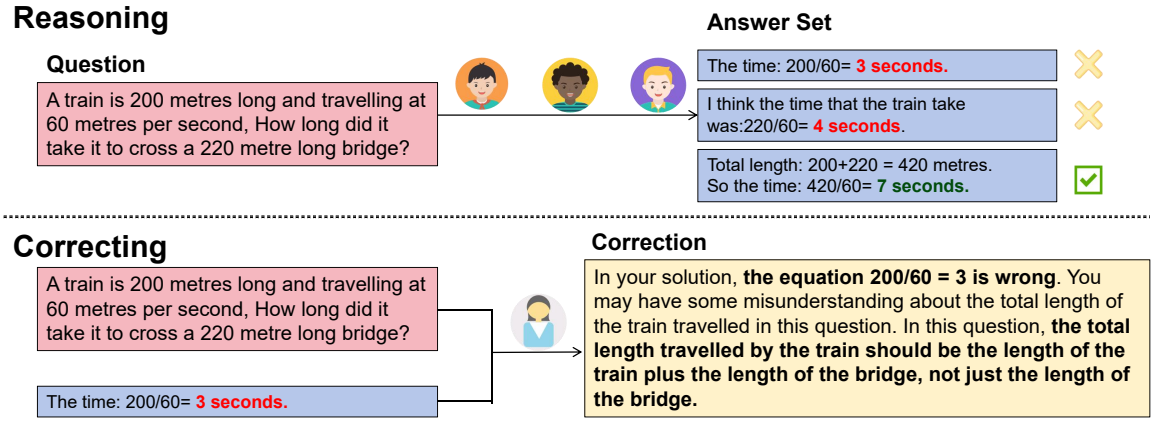

Figure 1: Examples of reasoning and correcting.

the ability of a language model to reason and to correct errors equivalent?

In some cases, an LLM may correctly solve a mathematical problem but fail to address errors in the solution. Conversely, it may inaccurately answer a math question but successfully rectify solution errors based on adequate contextual cues.

Based on the observation, we hypothesise that the reasoning and correcting capabilities are not fully equivalent. To demonstrate this, we introduce **math world problems correction(MWPC)**, a novel task focusing on the correction abilities of LLMs. We also conduct a series of experiments on MWPC task to prove our hypothesis, which will be described in Section 3.

Then, we further raise a question: **How can we enhance the correcting abilities of LLMs?**

In modern teaching materials, both concise and detailed answers are commonly provided alongside the questions. Since we have demonstrated that the reasoning and correcting abilities were not fully equivalent, we proposed a novel method, called **Diagnostic-Oriented Prompting(DOP)**, leveraging available resources to enhance LLMs' proficiency as correctors in mathematical education.

Generally speaking, our contributions can be concluded as follows.

- We modify the research paradigm, showing that in most LLMs, the abilities to reason and correct in MWPs are not fully equivalent, emphasizing that merely enhancing reasoning is insufficient.
- To the best of our knowledge, we are the first to propose MWPC task, which is more relevant and beneficial in mathematical education settings.
- We propose **Diagnostic-Oriented Prompting(DOP)**, a novel and effective method to enhance LLMs' correcting abilities based on modern teaching resources.

2 Background and Related Work

2.1 Mathematical Reasoning Through LLMs

There are many ways to improve the performance of LLMs on mathematical reasoning tasks by prompting them.

The method of chain-of-thought(COT) prompting (Wei et al., 2022; Kojima et al., 2022) can be used in mathematical domain and improves the accuracy. (Wang et al., 2023c) notices that a complex reasoning problem is usually thought of in a number of different ways and used majority voting to improve the process of COT. (Zhou et al., 2023; Wang et al., 2023a) endeavour to decompose complex problems into multiple simple steps, guiding the large language model to solve mathematical problems step by step. (Liu et al., 2023a; Imani et al., 2023; Gou et al., 2023) mainly focus on using external tools like Python executor, mathematical calculator, and so on, to reduce the probability of error in LLMs and improve the reliability of LLMs in mathematical reasoning tasks.

In order to specifically enhance and utilise the mathematical reasoning ability of the model, some researchers use fine-tuning or instruction-tuning methods. (Ho et al., 2023) proposed fine-tuned COT, which generates reasoning samples from large teacher model to fine-tune smaller model. (An et al., 2023) utilised a corpus of mathematical reasoning containing error samples and the error correction process to fine-tune small models like LLama-2(Touvron et al., 2023) and MetaMath(Yu et al., 2023). (Liu and Low, 2023) introduced Goat,

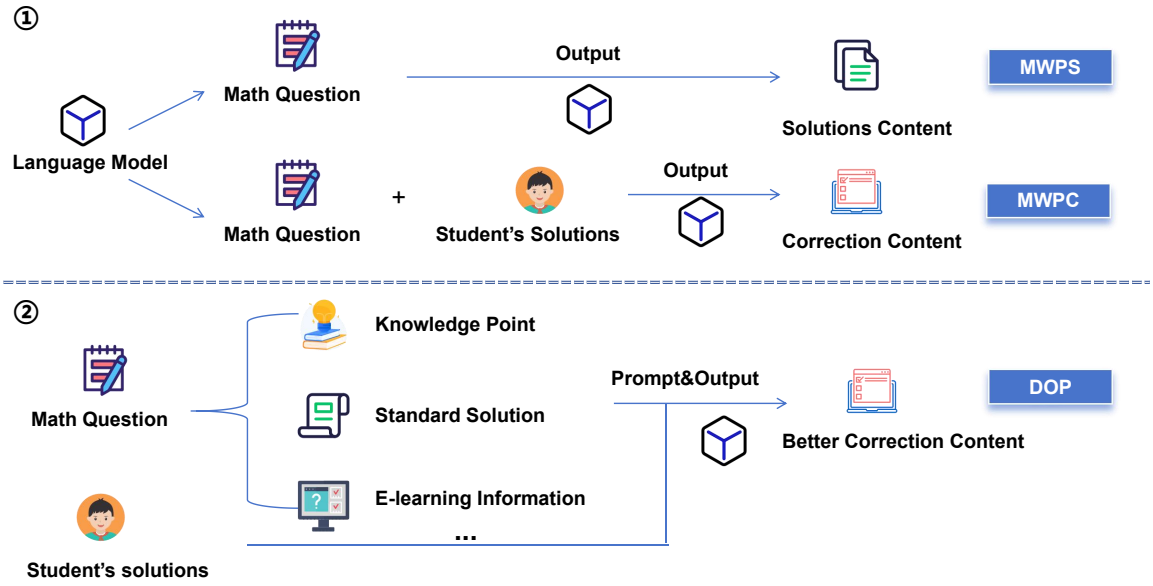

Figure 2: The overall framework of our research. In the first stage, we conduct both **MWPS** and **MWPC** tasks on our candidate models and prove that mathematical reasoning and correcting capabilities are not fully equivalent. Then, in the second stage, we conduct our strategy called **Diagnostic-Oriented Prompting(DOP)**, enabling our candidate models to enhance their correcting abilities in mathematical domain.

which is a fine-tuned LLama model and can significantly outperforms GPT-4 (OpenAI, 2023) on a wide range of arithmetic tasks.

2.2 Corrrection Throught LLMs

Meanwhile, some research spotlights the correction capabilities of LLMs.

(Madaan et al., 2023) proposed self-refine, which is a novel approach that allows LLMs to iteratively provide feedback and refine their own outputs. (Pan et al., 2023) summarised a series of methods using feedback either produced by LLMs themselves or some external systems, to rectify those flaws. Self-correction effectively mitigates hallucination (Ji et al., 2023) in LLMs. However, (Huang et al., 2023) pointed out that without external feedback, LLMs still connot self-correct their own reasoning process, including mathematical reasoning process. According to (Stechly et al., 2023; Valmeekam et al., 2023a,b; Huang et al., 2023), when correcting something wrong, especially those errors produced by LLMs themselves, external information is indispensable.

There are also some studies centering on error correction task. (Wang et al., 2023b) used LLMs to remediate students' mathematical mistakes step by step. (Tang et al., 2023; Song et al., 2023; Du et al., 2023; Kwon et al., 2023) focused on grammatical error correction(GEC) task, utilising LLMs to solve GEC problems in monolingual and multilingual scenarios. (MacNeil et al., 2023; Leinonen et al., 2023) researched the abilities of LLMs to correct errors in code, which is beneficial to computer science(CS) education. Unfortunately, there is still very little research on error correction to the mathematical reasoning process.

2.3 AI For Mathematical Education

Artificial Intelligence(AI) strongly promotes the development of mathematical education.

Since LLMs were put into use, (Wang et al., 2023b) simulated the process of human tutor, determining different strategy to address students' reasoning mistakes in mathematics. (Wu et al., 2023) studied mathematical education on conic sections in Chinese senior high school education using LLMs like GPT-4(OpenAI, 2023) and ChatGLM(Du et al., 2022). (Long et al., 2023) evaluated ChatGPT on generating pre-university math questions, providing insights for teachers and researchers in utilizing LLMs in mathematical education. The research above reveals that making LLMs to be good teachers is a following trend for AI in mathematical education.

3 Methodology

In this section, we will address the focus and describe the research methodology we used in this study.

Firstly, we conducted experiments to **validate**

differences between reasoning and correcting in mathematics domain. Continue with the process, we proposed **Diagnostic-Oriented Prompting(DOP)** for correction capabilities. Figure 2 shows the overall framework.

3.1 Validating Differences between Reasoning and Correcting

Initially, we conducted comparative experiments to validate the observation that the reasoning and error correction abilities of LLMs are not fully correlated.

We established several pivotal elements within this scenario. Firstly, our candidate models are represented as an expression $\boldsymbol{f}(\cdot)$, with the output sequence denoted as $\boldsymbol{y}$. In mathematical reasoning task, the input is a math question, denoted as $\boldsymbol{Q}$. We provided the model with a prompt containing the question, denoted as $\boldsymbol{P_r}(\boldsymbol{Q})$, and obtained an output $\boldsymbol{y_r}$, which means that:

$$\boldsymbol{y_r} = \boldsymbol{f}(\boldsymbol{P_r}(\boldsymbol{Q})) \tag{1}$$

Similarly, in the MWPC task, the input consists of a math question $\boldsymbol{Q}$ and its corresponding incorrect solution $\boldsymbol{W}$. We provided the model with a prompt containing both elements, denoted as $\boldsymbol{P_c}(\boldsymbol{Q}, \boldsymbol{W})$, and obtained an output $\boldsymbol{y_c}$, indicating that:

$$\boldsymbol{y_c} = \boldsymbol{f}(\boldsymbol{P_c}(\boldsymbol{Q}, \boldsymbol{W})) \tag{2}$$

In the next step, considering the question $\boldsymbol{Q}$, we examine standard answer, represented as $\boldsymbol{A}$. To ascertain the model's ability to solve the question, we employ an extraction function, denoted as $\boldsymbol{N_r}$ in the reasoning task and $\boldsymbol{N_c}$ in the correction task, to extract the final numeric answer from the natural language. Ultimately, we defined two states, $\boldsymbol{S_r}$ and $\boldsymbol{S_c}$, to indicate whether the model had successfully solved the task, which means that:

$$\boldsymbol{S_r} = \begin{cases} 1, & \text{if } \boldsymbol{N_r}(\boldsymbol{y_r}) = \boldsymbol{N_r}(\boldsymbol{A}), \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

$$\boldsymbol{S_c} = \begin{cases} 1, & \text{if } \boldsymbol{N_c}(\boldsymbol{y_c}) = \boldsymbol{N_c}(\boldsymbol{A}), \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

As mentioned above, it is necessary to collect a wide range of $\{\boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{W}\}$ triplet. They are all represented as natural language.We chose several LLMs as our candidate models to perform both reasoning and correction tasks. Details about these selected models will be provided in Section 4.

3.2 Diagnostic-Oriented Prompting(DOP)

In our previous experiments, we observed that while LLMs may not entirely solve problems, they can generate correction processes. This parallels real-time education scenarios where teachers or parents, though unable to solve problems themselves, can guide children based on relevant information. Motivated by this, we propose a strategy named **Diagnostic-Oriented Prompting (DOP)** to leverage abundant resources and enhance the mathematical correction abilities of LLMs.

In modern educational materials, questions often come paired with answers, ranging from concise to detailed responses. Depending on the available resources, we can employ varying levels of DOP to enhance the correction abilities of LLMs.

Furthermore, we conducted experiments involving 3 levels of DOP, affirming the effectiveness of the DOP approach.

The DOP framework comprises three levels, each with distinct input configurations. In the first level, the model's input consists of the mathematical problem, the erroneous solution and the correct numeric answer(NA) of the problem. In the second level, the model's input consists of the problem, the erroneous solution and the brief explanation(BE) of the problem. And finally, in the third level, the model's input consists of the problem, the erroneous solution and the standard answer(SA) of the problem. The prompt method that does not provide any additional supplementary information is labeled as standard prompting(SP).

The goal of DOP is to correct erroneous solution processes and arrive at the correct answer. The 3 levels of DOP progressively deepen and are denoted DOP+NA, DOP+BE and DOP+SA. The complete SP and DOP process is illustrated in Figure 3.

4 Expriments and Analysis

4.1 Experiment Setup

We utilized some LLMs as candidate models, and collected multiple $\{\boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{W}\}$ triplets from several mathematical datasets.

Candidate models. We selected the following LLMs as out candidates, which contains some no-

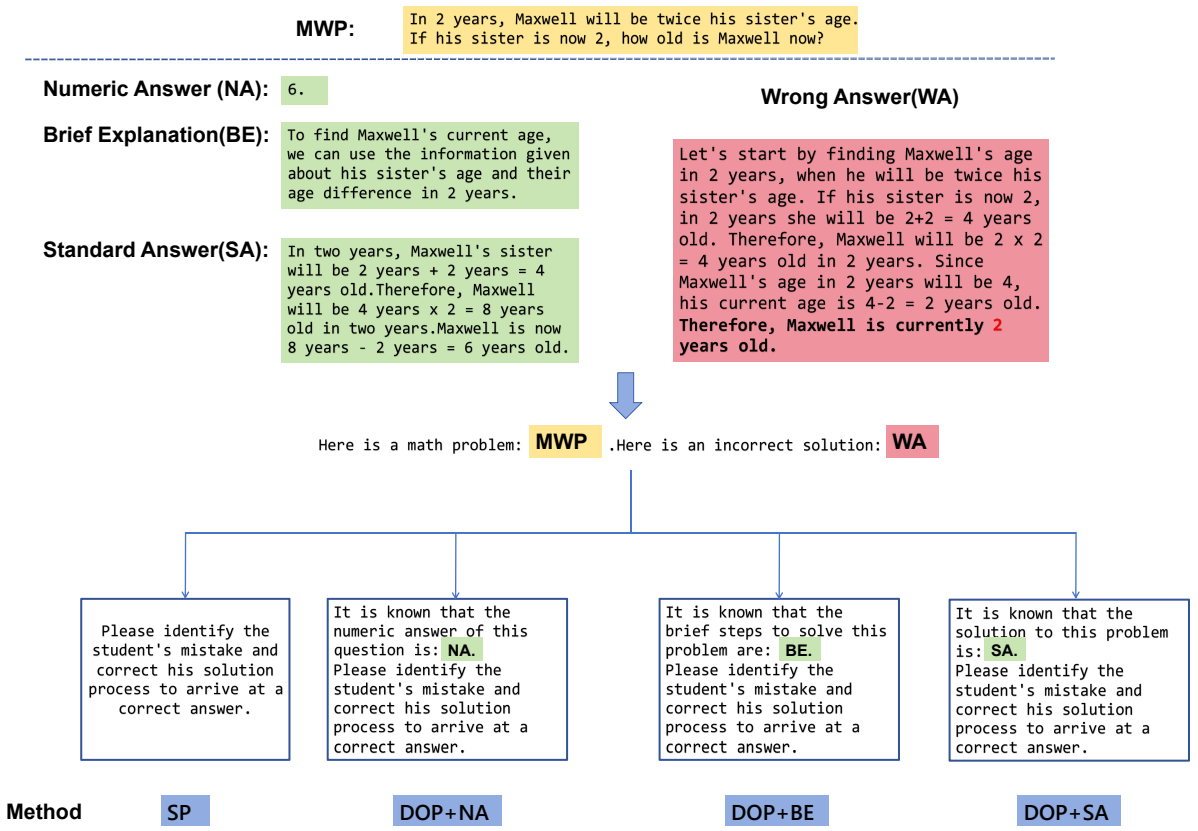

Figure 3: An example of different levels of DOP.

table general models, some specialized mathematics models, and some educational-purpose models.

- **GPT-4-0613(OpenAI, 2023)**. GPT-4 is one of the most widely known LLMs, developed by openai. We selected the latest version.
- **GPT-3.5-turbo(OpenAI, 2023)**. A strong and remarkable model. It is also known as ChatGPT, developed by openai.
- **LLama-2-Chat(Touvron et al., 2023)**. LLama-2 is a collection of LLMs devloped by Meta and LLama-2-Chat is the fine-tuned model for dialogue use. We selected 3 parameter size: 7B, 13B and 70B.
- **MetaMath(Yu et al., 2023)**. MetaMath is a fine-tuned model that specializes in mathematical reasoning. Researchers used a rewrite strategy to bootstrap math questions and then fine tune the model. We selected 2 parameter size: 7B, 13B, pretrained from LLama2, and a 7B version pretrained on Mistral(Jiang et al., 2023).
- **WizardMath(Luo et al., 2023)**. WizardMath is a fine-tuned model using reinforcement learning from evol-instruct feedback for mathematical reasoning. We selected 2 parameter size: 7B, 13B.
- **Baichuan2(Yang et al., 2023)**. Baichuan2 is a series of multilingual LLMs trained from scratch and perform well on some vertical domains including education. We selected 2 parameter size:7B, 13B.

Data Construction. In our experiments, we collected sets of Q, A, W triplets, focusing on application problems in primary school mathematics described in natural language. The datasets we primarily referred to are as follows:

- **GSM8k(Cobbe et al., 2021)**. GSM8k is a dataset of 8.5K high quality diverse grade school math word problems containing natural language solutions.
- **MathDial(Macina et al., 2023)**. MathDial is a dataset of one-to-one teacher-student tutoring dialogues grounded in multi-step mathematical reasoning problems. Most of the math problems are from GSM8k.

As MathDial provides problem statements, correct answers, and student confusion, we leveraged

Model	R-rate	C-rate	sR+sC	sR+uC	uR+sC	uR+uC
GPT-4-0613	**0.859**	**0.811**	**2152**	306	165	238
GPT-3.5-turbo	0.556	0.344	659	932	325	945
LLama-2-chat-7b	0.108	0.089	45	264	211	234
LLama-2-chat-13b	0.200	0.153	148	424	290	1999
LLama-2-chat-70b	0.318	0.224	282	629	**358**	1592
MetaMath-7b	0.764	0.180	455	**1732**	61	613
MetaMath-13b	0.772	0.238	606	1602	76	577
MetaMath-Mistral-7b	0.733	0.254	637	1459	91	674
WizardMath-7b	0.708	0.391	890	1138	229	604
WizardMath-13b	0.486	0.165	294	1096	177	1294
Baichuan-2-7b	0.079	0.059	29	196	139	**2497**
Baichuan-2-13b	0.281	0.105	133	690	186	1872

Table 1: The performance of candidate models in comparative experiments. The maximum value in each column is highlighted in **bold**.

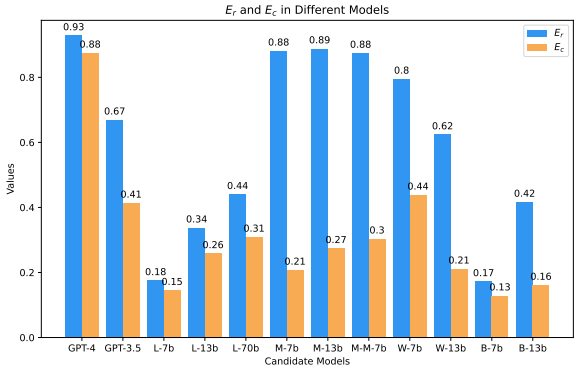

Figure 4: Results of E_r and E_c. We represents the candidate models using the first letters. For example, 'M-M-7b' means MetaMath-Mistral-7b.

this data to construct a dataset comprising 2,861 sets of $\{Q, A, W\}$ triplets.

4.2 Results and Analysis

4.2.1 Comparative Experiments for Validation.

For each candidate model in the comparative experiments, we recorded the following information:

- **R-rate**. The rate of $\{Q, A, W\}$ triplets which were reasoned successfully.
- **C-rate**. The rate of $\{Q, A, W\}$ triplets which were corrected successfully.
- **sR+sC**. The number of $\{Q, A, W\}$ triplets which were reasoned and corrected successfully.
- **sR+uC**. The number of $\{Q, A, W\}$ triplets which were reasoned successfully but corrected unsuccessfully.
- **uR+sC**. The number of $\{Q, A, W\}$ triplets which were corrected successfully but reasoned unsuccessfully.
- **uR+uC**. The number of $\{Q, A, W\}$ triplets which were reasoned and corrected unsuccessfully.

Table 1 shows the performance of candidate models in comparative experiments. Let's start by analyzing the R-rate and C-rate. We can observe that GPT-4 achieves the highest performance both on MWPS and MWPC tasks. This indicates that as the most advanced general-purpose language model currently available, GPT-4's mathematical capabilities are clearly evident. Meanwhile, the specialized mathematics models like MetaMath show strong capabilities in mathematical reasoning, while their error correction abilities still have considerable room for improvement.

Next, we will analyze the performance of the models on the following four metrics: sR+sC, sR+uC, uR+sC and uR+uC. In Table 1, we can observe that, even in GPT-4, successfully solving a mathematical problem does not guarantee the ability to accurately correct an incorrect solution. Conversely, the model may not always provide an accurate solution, yet it can generate a proper correction process for an incorrect solution.

We further provides 2 definitions as follows.

$$\boldsymbol{E_r} = \frac{|\boldsymbol{sR} + \boldsymbol{sC}|}{|(\boldsymbol{sR} + \boldsymbol{sC}) \cup (\boldsymbol{uR} + \boldsymbol{sC})|} \quad (5)$$

$$\boldsymbol{E_c} = \frac{|\boldsymbol{sR} + \boldsymbol{sC}|}{|(\boldsymbol{sR} + \boldsymbol{sC}) \cup (\boldsymbol{sR} + \boldsymbol{uC})|} \quad (6)$$

Question	Francine drives 140km to work each day. If she does not go to work 3 days every week, **find the total distance she drives to work for 4 weeks in kilometers.**
Ground-truth	There are 7 days in a week, so if he doesn't go to work for 3 days, he goes 7-3 = 4 days every week. He travels 140km each day for a weekly total of 140*4 = 560km. In 4 weeks he will travel a total of **560 * 4 = 2240km.**
Answer from GPT-4	Francine goes to work 7 - 3 = 4 days a week. This means that she drives a total of 140 * 4 =560km every week. Therefore in 4 weeks, she drives a total of **560 * 4 = 2240km.**
Student's solution	In a week, Francine drives **140km x 5 = 700km** to work. In 4 weeks, she drives 700km x 4 = 2800km to work. But she does not go to work for 3 days every week, so in 4 weeks, she does not go to work for 3 x 4 = 12 days. Therefore, the total distance she drives to work in 4 weeks is **(700km x 4) - (140km x 12) = 2800km - 1680km = 1120km.**
Correction From GPT-4	The student's mistake lies in the fact that they incorrectly subtracted the days she does not work from the total distance she drives to work. In reality, the days Francine does not work have already been accounted for when calculating the weekly distance. Here is the correct solution: In a week Francine drives 140km x 5 days = 700km to work. In 4 weeks, she drives 700km x 4 = 2800km to work. There is no need to subtract the days she does not work because they have already been accounted for in the initial weekly calculation. **Answer: 2800km.**

Table 2: A case study of GPT-4. The error of this students lies in that he or she thinks there are five days in a week. GPT-4 effectively solved this problem. However, GPT-4 didn't capture the student's error.

As we mentioned above, $\boldsymbol{E_r}$ represents the ratio of the corrected numbers to the total reasoned numbers, while $\boldsymbol{E_c}$ represents the ratio of the reasoned numbers to the total corrected numbers. We displays the value of $\boldsymbol{E_r}$ and $\boldsymbol{E_c}$ in our experiments in Figure 4.

In Figure 4, we can observe that all our candidate models achieve higher $\boldsymbol{E_r}$ than $\boldsymbol{E_c}$. This suggests that if a model can successfully correct an error, it is more likely to solve the problem simultaneously. However, when the model is capable of solving a problem, the probability of correcting a related incorrect solution is much lower.

We also provide a case study in our experiment, as shown in Table 2. The mathematical problem requires finding the distance Francine has traveled during her 4-week work. GPT-4 effectively solved this problem. However, when faced with a student who miscalculated the number of working days, GPT-4 did not successfully correct its mistake. This indicates that for LLMs, successfully solving a mathematical problem does not necessarily mean they can successfully correct any errors that may arise within it. Similarly, successfully correcting an error within a mathematical problem does not imply that they can also successfully solve the problem.

To conclude, combining the result from Figure 4 and Table 2, we successfully demonstrate through comparative experiments that **the ability of LLMs in mathematical reasoning is not entirely equivalent to their ability in mathematical error correction**. Therefore, solely enhancing a model's mathematical problem-solving ability does not guarantee its proficiency as an error corrector. Further research is needed to thoroughly investigate the model's error correction capabilities in mathematics.

4.2.2 Diagnostic-Oriented Prompting(DOP)

For the DOP framework mentioned in Figure 2 and Figure 3, we conducted experiments involving 3 levels of DOP with several candidate models.

We studied DOP in 8 candidate models, comparing the correction passing rate between SP and DOP. We record the experimental results in Figure 5.

We found that when employing DOP, all candidate models achieved higher pass rates compared to using SP alone during the MPWC task. This suggests that the DOP method significantly enhances

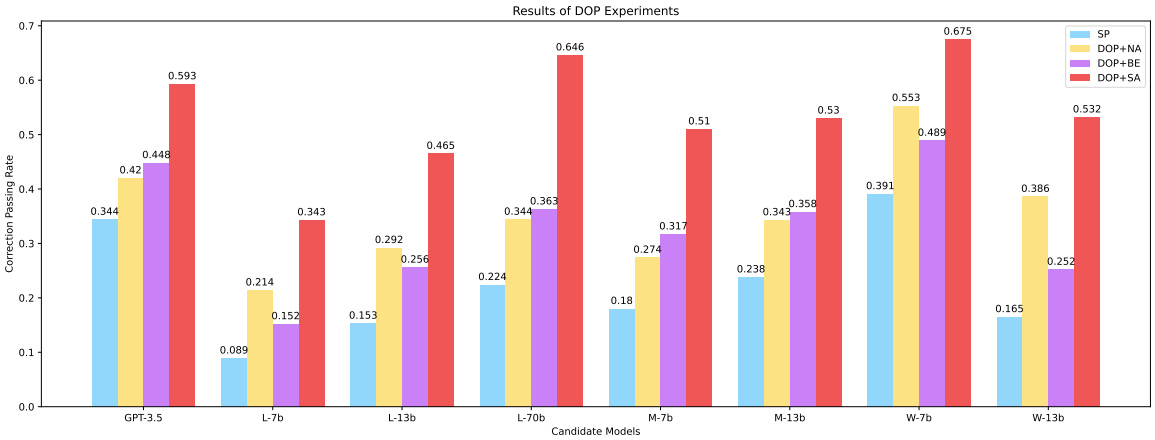

Figure 5: Experiment results of DOP. We recorded the success rates of error correction under different scenarios and visualized them as bar charts.

the mathematical error-correction capabilities of LLMs.

5 Conclusions

In this paper, we have come to the following conclusions.

1.LLMs' reasoning and correcting abilities are not fully equivalent. In our comparative experiment, LLMs may solve a problem but fail to correct a wrong solution of this problem. Also, they may not solve a problem properly, but can find reasoning errors and correct them in a wrong solution.

2.Mainstream LLMs' have stronger reasoning abilities than correcting abilities. In our experiments, our candidate models perform better in reasoning task than correcting tasks. This suggests that while LLMs excel as reasoners, their ability to correct errors is limited. Therefore, further research into their correction abilities is necessary.

3.Improving LLMs' correcting abilities is vital and essential. In mathematical education scenarios, it is more vital to correct the error from the students, rather than merely providing solutions. Since we have demonstrated that reasoning and correcting abilities are not the same thins, and reasoning abilities are much better, improving LLMs' correcting abilities bocomes vital and important.

4.Diagnostic-Oriented Prompting(DOP) is an effective method to enhance the correcting abilities of LLMs. We modify the research paradigm of the mainstream research and proposes MWPC task. With the aid of educational resources and DOP, LLMs can be an excellent corrector, which is useful to help students dealing with understanding math world problems.

6 Limitations and Future Work

We have several limitations in this work.Firstly, there are still lack of high-quality mathematical correction datasets to study the relative abilities of LLMs. Meanwhile, we study correction mainly based on all kinds of language models. In fact, the behaviour of LLMs and human teachers and students differs a lot. We still need deeper research in the field. To study this issue well, our future work is as follows:

- **Collect high-quality MWPs and corresponding mistakes.** High-quality data is vital for us to enhance the performance of LLMs. Most mainstream datasets in mathematical domain are lack of some relevant solutions with errors, which is not helpful to study the correcting abilities of LLMs. As a result, we are committed to construct a high-quality dataset containg MWPs and corresponding mistakes.

- **We need a deeper view of real-time mathematical education scenarios.** The behaviours between human and language models differs a lot. We also need some data from the real life, not just merely from the language models. In the future, it is necessary for us to go deeper to the real-time education scenarios.

- **Develop more level of DOP.** We have broken the mold and proven the effectiveness of DOP. It is still necessary to develop a higher level of DOP method.

References

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning from mistakes makes llm better reasoner.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Hanyue Du, Yike Zhao, Qingyuan Tian, Jiani Wang, Lei Wang, Yunshi Lan, and Xuesong Lu. 2023. Flacgec: A chinese grammatical error correction dataset with fine-grained linguistic annotation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23. ACM.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: general language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 320–335. Association for Computational Linguistics.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Tora: A tool-integrated reasoning agent for mathematical problem solving. *CoRR*, abs/2309.17452.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. 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 14852–14882. Association for Computational Linguistics.

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

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. Mathprompter: Mathematical reasoning using large language models. In *Proceedings of the The 61st Annual Meeting of the Association for Computational Linguistics: Industry Track, ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 37–42. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12):248:1–248:38.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L'elio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *ArXiv*, abs/2310.06825.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.

Sang Yun Kwon, Gagan Bhatia, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Beyond english: Evaluating llms for arabic grammatical error correction.

Juho Leinonen, Paul Denny, Stephen MacNeil, Sami Sarsa, Seth Bernstein, Joanne Kim, Andrew Tran, and Arto Hellas. 2023. Comparing code explanations created by students and large language models.

Tengxiao Liu, Qipeng Guo, Yuqing Yang, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023a. Plan, verify and switch: Integrated reasoning with diverse x-of-thoughts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2807–2822. Association for Computational Linguistics.

Tiedong Liu and Bryan Kian Hsiang Low. 2023. Goat: Fine-tuned llama outperforms GPT-4 on arithmetic tasks. *CoRR*, abs/2305.14201.

Yixin Liu, Avi Singh, C. Daniel Freeman, John D. Co-Reyes, and Peter J. Liu. 2023b. Improving large language model fine-tuning for solving math problems. *CoRR*, abs/2310.10047.

Phuoc Pham Van Long, Duc Anh Vu, Nhat M. Hoang, Xuan Long Do, and Anh Tuan Luu. 2023. Chatgpt as a math questioner? evaluating chatgpt on generating pre-university math questions.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *CoRR*, abs/2308.09583.

Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. Mathdial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 5602–5621. Association for Computational Linguistics.

Stephen MacNeil, Paul Denny, Andrew Tran, Juho Leinonen, Seth Bernstein, Arto Hellas, Sami Sarsa,

and Joanne Kim. 2023. Decoding logic errors: A comparative study on bug detection by students and large language models.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback.

OpenAI. 2023. Gpt-4 technical report.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies.

Yixiao Song, Kalpesh Krishna, Rajesh Bhatt, Kevin Gimpel, and Mohit Iyyer. 2023. Gee! grammar error explanation with large language models.

Kaya Stechly, Matthew Marquez, and Subbarao Kambhampati. 2023. Gpt-4 doesn't know it's wrong: An analysis of iterative prompting for reasoning problems.

Chenming Tang, Xiuyu Wu, and Yunfang Wu. 2023. Are pre-trained language models useful for model ensemble in chinese grammatical error correction? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 893–901. Association for Computational Linguistics.

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

Karthik Valmeekam, Matthew Marquez, and Subbarao Kambhampati. 2023a. Can large language models really improve by self-critiquing their own plans?

Karthik Valmeekam, Matthew Marquez, Sarath Sreedharan, and Subbarao Kambhampati. 2023b. On the planning abilities of large language models : A critical investigation.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023a. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large 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 2609–2634. Association for Computational Linguistics.

Rose E. Wang, Qingyang Zhang, Carly Robinson, Susanna Loeb, and Dorottya Demszky. 2023b. Step-by-step remediation of students' mathematical mistakes.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023c. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed 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*, volume 35, pages 24824–24837. Curran Associates, Inc.

Haoyi Wu, Wenyang Hui, Yezeng Chen, Weiqi Wu, Kewei Tu, and Yi Zhou. 2023. Conic10k: A challenging math problem understanding and reasoning dataset.

Ai Ming Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, Fei Deng, Feng Wang, Feng Liu, Guangwei Ai, Guosheng Dong, Hai Zhao, Hang Xu, Hao-Lun Sun, Hongda Zhang, Hui Liu, Jiaming Ji, Jian Xie, Juntao Dai, Kuncheng Fang, Lei Su, Liang Song, Lifeng Liu, Liyun Ru, Luyao Ma, Mang Wang, Mickel Liu, MingAn Lin, Nuolan Nie, Pei Guo, Ruiyang Sun, Zhang Tao, Tianpeng Li, Tianyu Li, Wei Cheng, Weipeng Chen, Xiangrong Zeng, Xiaochuan Wang, Xiaoxi Chen, Xin Men, Xin Yu, Xuehai Pan, Yan-Bin Shen, Yiding Wang, Yiyu Li, Youxin Jiang, Yuchen Gao, Yupeng Zhang, Zenan Zhou, and Zhiying Wu. 2023. Baichuan 2: Open large-scale language models. *ArXiv*, abs/2309.10305.

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. *CoRR*, abs/2309.12284.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations,*

ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Z. Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jianyun Nie, and Ji rong Wen. 2023. A survey of large language models. *ArXiv*, abs/2303.18223.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.