Exploring Reversal Mathematical Reasoning Ability for Large Language Models

Pei Guo♠*, Wangjie You♠*, Juntao Li♠, Bowen Yan♢, Min Zhang♠
♠Institute of Computer Science and Technology, Soochow University, China
♢Department of Computer Science and Technology, Tsinghua University, China

{pguolst,wjyouuu}@stu.suda.edu.cn;
{ljt, minzhang}@suda.edu.cn;
yanbw@mail.tsinghua.edu.cn

Abstract

Large language models (LLMs) have presented remarkable capabilities in the wide range of natural language understanding and reasoning tasks. Despite their success, a few works indicate that LLMs suffer from the “reversal curse”, in which LLMs can’t employ the inverted structure “B is A” when they are trained based on “A is B”. To explore the effect of the “reversal curse” for LLMs on complex mathematical reasoning tasks, we present two reversal datasets upon GSM8K and MathQA and verify that LLMs also struggle to solve reversal mathematical problems. We analyze the potential reason and attribute it to the insufficient modeling of the relationship between reasoning steps caused by the left-to-right objective. Consequently, based on the characteristics of multi-step reasoning, we design a novel training method to improve the general and reversal reasoning abilities. Finally, we conduct experiments on four mathematical datasets, and the results demonstrate that our method significantly improves the general reasoning capacities and alleviates the reversal problem. Our datasets and codes are available at https://github.com/AllForward/ReversalMath.

1 Introduction

With the significant increase in data and model scale, large language models (LLMs) (Brown et al., 2020; Hoffmann et al., 2022; Touvron et al., 2023; OpenAI, 2023) have emerged with their powerful multi-dimensional capabilities, such as long-context open domain conversation, code assistants (Chen et al., 2021b; Luo et al., 2023b; Wang et al., 2023; Zheng et al., 2023b), instruction following (Ouyang et al., 2022; Taori et al., 2023), particularly in the complex reasoning tasks solved by chain-of-thought (CoT) methods (Wang et al., 2022; Wei et al., 2022; Lightman et al., 2023).

Nevertheless, a number of contemporary studies (Berglund et al., 2023; Grosse et al., 2023) highlight the presence of the “reversal curse” predicament in LLMs, where LLMs are trained based on the structure “A is B” in a sentence, and they cannot employ the inverted structure “B is A” to extrapolate and respond to queries effectively. Almost all works merely explore the “reversal curse” based on the name-to-description reversal task. It is worth exploring whether complex multi-step reasoning tasks also suffer from this predicament. If it exists, how should we alleviate it and improve the performance of LLMs’ reasoning?

To explore this problem, we choose one of the most challenging and representative reasoning tasks, i.e., mathematical problems (Collins et al., 2023; Imani et al., 2023; Luo et al., 2023a; Yuan et al., 2023), as the testbed. Resembling with the format of reversal curse problems, backward mathematical reasoning gives the answer to the original question and reverses to infer one of the variables in the question, which is first formalized by Yu et al. (2024). They construct a backward test set upon GSM8K (Cobbe et al., 2021) to evaluate the backward reasoning capabilities of LLMs. Their preliminary results on LLaMA-2-7B confirm that recent LLM can struggle to solve mathematical problems in backward rationales, leaving extensive verification and in-depth understanding unexplored.

To fill this blank, we first propose two reversal

<table>
<thead>
<tr>
<th>Models</th>
<th>GSM8K/Reversal</th>
<th>MathQA/Reversal</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-3.5-Turbo</td>
<td>77.4 / 52.2</td>
<td>63.5 / 44.6</td>
</tr>
<tr>
<td>Flan-T5-3B</td>
<td>13.5 / 3.5</td>
<td>5.8 / 5.8</td>
</tr>
<tr>
<td>Flan-T5-11B</td>
<td>16.1 / 12.3</td>
<td>15.5 / 9.6</td>
</tr>
<tr>
<td>LLama2-7B</td>
<td>13.7 / 7.0</td>
<td>19.2 / 10.3</td>
</tr>
<tr>
<td>LLama2-13B</td>
<td>25.3 / 10.7</td>
<td>25.6 / 10.3</td>
</tr>
<tr>
<td>LLama2-70B</td>
<td>52.1 / 30.2</td>
<td>42.0 / 29.7</td>
</tr>
</tbody>
</table>

Table 1: The accuracy of different LLMs on GSM8K, MathQA, and their correlated reversal test datasets.
mathematical test sets to further verify that LLMs suffer from the “reversal curse” on mathematical problems. Specifically, we employ GPT-4 to imitate the format and style of original questions and generate the reversal data based on the GSM8K and MathQA (Amini et al., 2019) test sets. The detailed construct process and data quality verification are elaborated in Section 3. After constructing two reversal test sets, we use them to evaluate representative LLMs of different model scales. As shown in Table 1, compared with original test sets, LLMs with different scales and architectures present a significant accuracy decline in the reversal datasets except for Flan-T5-3B on MathQA. This phenomenon sufficiently demonstrates that LLMs actually face difficulties in reversal reasoning in mathematical problems.

From this discovery, we analyze the potential reasons and speculate that it’s related to the traditional left-to-right training objective. In the process of mathematical multi-step reasoning, LLMs strictly follow the order of deductions from left to right, which solely focuses on acquiring the association from conditions to conclusions. This shortcoming is essentially the lack of context modeling for reasoning steps, which causes the difficulty of reversal reasoning and affects the general reasoning performance. To sufficiently model the relationship of different reasoning steps, and enhance the reversal and overall reasoning ability of LLMs, we propose a simple and effective training framework, which introduces an additional bidirectional training objective based on the characteristics of multi-step deduction. In particular, we choose the partial steps as the context which employs the bidirectional attention mechanism, while utilizing the causal attention mechanism to predict the remaining unselected steps. By employing this approach, LLMs are trained to extrapolate the preceding steps in a reverse manner, drawing upon the information from the succeeding steps.

To validate the effectiveness of our method, we fine-tune Flan-T5-XL and Llama2-7B on the GSM8K dataset. Subsequently, we evaluate the performance of these models on four benchmarks and two reversal mathematical datasets. The results show that the models’ general and reversal reasoning ability is superior to the latest methods that use additional training skills, and even some data augmentation strategies. Our contributions are listed below:

- We construct two reversal mathematical datasets to further explore the reversal reasoning ability of LLMs, and prove that LLMs actually suffer from the “reversal curse” on mathematical problems.
- We analyze the potential reason and attribute it to the insufficient modeling of the relationship between reasoning steps. Consequently, based on the characteristics of multi-step deduction tasks, the bidirectional training objective is designed to alleviate this problem.
- Whether on four benchmarks or two reversal datasets, applying our approach to different settings all achieves significant improvements, and even close to GPT-3.5-Turbo on the parts of benchmarks.

2 Related Work

2.1 Large Language Models

LLMs have shown impressive multi-dimensional capabilities, significantly affecting the natural language processing community (Brown et al., 2020; Hoffmann et al., 2022; Touvron et al., 2023; OpenAI, 2023). Recently, Wei et al. (2022); Wang et al. (2022) uncovered the broad prospects of CoT reasoning capabilities within LLMs. Given a few augmenting few-shot examples with multiple reasoning steps, LLMs can generate multi-step deduction toward the answer of solving complex tasks, e.g., this approach has been widely used on GPT-3.5 (OpenAI, 2022), GPT-4 (OpenAI, 2023) and LLaMA (Touvron et al., 2023) to tackle various reasoning tasks (Fu et al., 2023b; Zhang et al., 2023).

2.2 Reversal Curse

Though LLMs show impressive performance on various tasks, a number of current works (Berglund et al., 2023; Grosse et al., 2023) clarify that LLMs suffer from the reversal curse. Specifically, the autoregressive LLMs are trained on the logical sentence structure "A is B" and fail to infer "B is A". This phenomenon suggests that LLMs don’t grasp the relationship of knowledge presented in the training data adequately. Lv et al. (2023) further explore this problem and contend that the reversal curse arises partly due to the specific training objectives pursued by models, mostly evident in the widespread adoption of next-token prediction techniques in causal language models. Besides, Wu
et al. (2023) find that BERT is immune to the reversal curse. At the same time, a few bidirectional modeling approaches are proposed to mitigate the curse (Lv et al., 2023; Ma et al., 2023).

2.3 Mathematical Reasoning

Mathematical multi-step reasoning, one of the most challenging problems, has attracted widespread attention. We divide the related work into two categories. One category is the prompt-based methods, another is finetuning-based. For the first one, a few approaches (Narang et al., 2023; Fu et al., 2023c; Zheng et al., 2023a; Diao et al., 2023; Li et al., 2023b) provide multiple reasoning examples to LLMs, and leverage the excellent in-context capability of LLMs to generate high-quality reasoning paths. For instance, Narang et al. (2023) entail generating various reasoning chains, potentially yielding multiple candidate answers. Among these, the answer that garners the most votes is subsequently chosen as the ultimate response. Another category is obtaining the CoT paths from closed-source LLMs (e.g., GPT-3.5, GPT-4) by employing knowledge distillation and utilizing the knowledge to fine-tune open-source models (e.g., Flan-T5, LLaMA). Yuan et al. (2023) propose the rejection sampling fine-tuning (RFT) to improve the performance through collecting more reasoning paths as augmented datasets. WizardMath (Luo et al., 2023a) applies reinforcement learning from the evol-instruct feedback method to enhance reasoning ability. MetaMath (Yu et al., 2024) adopts four data augmentation strategies to generate high-diversity data and obtain excellent performance. Li et al. (2023a) explore the effect of augmented data from multiple perspectives and put forward query and response augmentations approaches. An et al. (2023) demonstrate the effectiveness of learning from mistakes.

Besides, the potential of smaller language models (SLMs) reasoning has been verified (Magister et al., 2022; Ho et al., 2023; Fu et al., 2023a). Shridhar et al. (2023), Han et al. (2023) and Junbing et al. (2023) decompose the complex questions into a series of simpler problems. Liu et al. (2023) further distill the self-evaluation capability of LLMs into SLM to improve the performance.

3 Reversal Mathematical Datasets Construction

Different levels of mathematical word problems have been proposed to evaluate LLMs’ general mathematical reasoning ability, such as AddSub (Hosseini et al., 2014), MultiArith (Roy

Figure 1: The prompt for obtaining reversal data. “Modified question” and “Hiding number” denote the generated reversal question and the corresponding answer.
Morisette and Kael were asked to bring fruits. Morisette brought 5 apples and 8 oranges, while Kael brought twice the amount of apples and half the number of oranges than Morisette. How many fruits do they have in total? (The correct answer is 27)

CoT Reasoning:
Step 1: Morisette brought 5 apples and 8 oranges, totaling 13 fruits.
Step 2: Kael brought twice the amount of apples, so $2 \times 5 = 10$ apples, and half the number of oranges than Morisette, which is $8 / 2 = 4$ oranges.
Step 3: Kael brought a total of $10 + 4 = 14$ fruits.
Step 4: Thus, the total number of fruits they have is $13 + 14 = 27$ fruits.

4 Methodology
In this section, we first introduce the training objective of causal mechanisms widely utilized in LLMs. After that, we analyze the shortcomings of this objective in multi-step reasoning tasks like mathematical problems and design an additional training objective to alleviate this problem.

4.1 Unidirectional Modeling of Reasoning
The causal attention mechanism is widely applied to LLMs to fine-tune various downstream tasks, including multi-step mathematical reasoning problems. Formally, we denote a mathematical dataset as $D = \{(x_i, y_i)\}_{i=1}^{N}$ where $x_i$ is a question, $y_i$ represents the CoT reasoning steps to solve question $x_i$, $N$ stands for the number of samples in $D$. For each sample, LLMs are trained to maximize the following likelihood:

$$L_{\text{causal}} = \sum_{t}^{T} \log P(y_{i}^{t} | y_{i}^{<t}, x_i ; \theta)$$

where $y_{i}^{<t}$ denotes the previous tokens before token $y_{i}^{t}$, $T$ denotes the length of $y_{i}$, and $\theta$ denotes the model parameters.

4.2 Bidirectional Modeling of Reasoning
Through applying Equation 1, LLMs strictly follow the order of deductions from left to right in the mathematical problems. Nevertheless, this training objective solely focuses on acquiring the association from conditions to conclusions, disregarding the reciprocal relationship. For example, as shown in Figure 2, LLMs are trained to predict the ultimate answer even the intermediate answer with preceding conditions, namely inferring step 4 from...
steps 1 to 3 and inferring step 3 from step 2, while disregarding the process of deducing preceding conditions from the conclusions, such as how to infer the deductions of numbers "13" and "14" appeared in the conclusion based on step 4. This shortcoming is essentially the lack of context modeling for reasoning steps, which causes insufficient dependency between different steps and affects the general reasoning performance. Struggling with the reversal questions is one of the typical manifestations.

To sufficiently model the relationship between different reasoning steps and improve their overall performance, especially the reversal reasoning ability, we propose a simple and effective training framework, which introduces an additional bidirectional training objective based on the characteristics of multi-step deduction. In particular, for each CoT explanation $y_i$, combined with $n$ steps $y_i = \{s_1, s_2, ..., s_n\}$, we randomly sample parts of the steps that need to be predicted, denoted as $s_i^{\text{pred}}$, and the rest steps denoted as $s_i^{\text{obs}}$. Given that LLMs leverage the causal attention mechanism throughout the pre-training even fine-tuning phases, it proves challenging to directly transition the attention mechanism from unidirectional to bidirectional. To maintain the original capabilities of LLMs and better stimulate the ability of reversal reasoning, we keep each token in $s_i^{\text{pred}}$ following a left-to-right order, which is shown in Figure 3 (b). Besides, $s_i^{\text{obs}}$ could be observed by each token in $s_i^{\text{pred}}$, which achieves the bidirectional modeling of the preceding and following steps. At the same time, $s_i^{\text{pred}}$ are not visible in $s_i^{\text{obs}}$ in order to prevent information leakage. For instance, in Figure 3, if step 2 can obtain the information of step 1, it would lead to a potential information leakage that could impact the prediction of step 1. Formally, the proposed new training objective could be described as follows:

$$L_{\text{bid}} = \sum_{t \in s_i^{\text{pred}}} \log P(y_t | y_{<t} \cup y_{t \in s_i^{\text{obs}}}, x_i; \theta)$$

where $T_{\text{pred}}$ denotes the number of tokens in $s_i^{\text{pred}}$.

**4.3 Training and Inference**

The causal and proposed bidirectional training objectives have been described in the previous section. Now, we clarify the final objective and the details of training and inference. In the training stage, we combine $L_{\text{causal}}$ and $L_{\text{bid}}$ as the final objective. Not only can it maintain the original capacity of LLMs, but it also improves the reversal reasoning ability. The corresponding computation can be formulated as follows:

$$L = L_{\text{causal}} + \alpha L_{\text{bid}}$$
where $\alpha$ is a customized parameter. As shown in Figure 3 (b), to satisfy the autoregressive generation for each step in $s_{pred}^i$, the special token [BOS] is padded at the beginning of the input. At the stage of inference, LLMs still adopt the causal attention algorithm as usual to perform reasoning autoregressively.

5 Experiments

In this section, we evaluate the effectiveness of our method by employing it on mathematical datasets. To compare with current works (Fu et al., 2023a; Han et al., 2023; Yu et al., 2024), we follow them and adopt the original test set to present the general reasoning ability. The discussion about the reversal reasoning capacity will be introduced in Section 6.1.

5.1 Datasets

To evaluate LLMs’ reasoning ability and general ability, we utilize four mathematical datasets, namely GSM8K (Cobbe et al., 2021), MultiArith (Roy and Roth, 2015), ASDiv (Miao et al., 2020), and SV AMP (Patel et al., 2021). Except for Llama2 (MetaMath) with and without our method, which uses the MetaMath training dataset, we only fine-tune Flan-T5 and Llama2 on the GSM8K training set, which contains 7,473 examples. It’s noticed that for each GSM8K training instance, we employ GPT-3.5-Turbo-1106 to generate multiple related reasoning paths and choose the correct one as the final solution (specific prompt could be seen in Appendix A). The remaining three datasets are adopted to evaluate the out-of-distribution ability of models. Moreover, following the previous work (Fu et al., 2023a; Han et al., 2023), we adopt 500 examples for each dataset as the validation set, and the remaining examples as the test set (800 for GSM8K, 400 for MultiArith, 18K for ASDiv, 500 for SVAMP).

5.2 Baselines

For the baseline models, we divide them into three categories: (i) Closed-sourced models: GPT-3.5-Turbo-1106 (OpenAI, 2022), Code-Davinci-002 (Chen et al., 2021a), LaMDA-137B (Kojima et al., 2022), PaLM-60B (Chowdhery et al., 2022), each of them presents strong reasoning ability. (ii) Open-sourced generic models: Flan-T5 (Chung et al., 2022) and Llama2 (Touvron et al., 2023), which are widely applied to various tasks. (iii) Specialized models: For Flan-T5 models, SpecialFT (Fu et al., 2023a) and DialCoT (Han et al., 2023) respectively employ knowledge transfer and questions decomposition to enhance models’ mathematical reasoning ability. For LLama2 models, Rejection sampling Fine-Tuning (RFT) (Yuan et al., 2023) collects multiple correct reasoning paths as augmented data for fine-tuning. WizardMath (Luo et al., 2023a) applies reinforcement learning from the evol-instruct feedback method to the math domain. MetaMath (Yu et al., 2024) proposes four data augmentation methods, significantly improving performance. Besides, we apply the supervised fine-tuning (SFT) method on our designed GSM8K training dataset for both models.

5.3 Implementation

We implement our method on two model architectures, namely Encoder-Decoder (Flan-T5) and Decoder-only (LLama2). Due to the limited computing resources, we chose Flan-T5-3B and Llama2-7B as backbones and fully fine-tuned them. The greedy search algorithm is utilized to execute inference processes for all the specialized models. More experimental details can be seen in Appendix A.1. We follow the previous works to use statistical significance tests (Koehn, 2004) to detect if the difference in accuracy score between our approach and base settings is significant.

5.4 Results

The overall results are shown in Table 2 and can be summarized as follows:

**Results on Flan-T5 backbone.** The SFT 3B model trained on GPT-3.5-Turbo-1106 generated dataset is better than SpecialFT and DialCoT-S-PPO, even better than their 11B on parts of datasets, presenting the importance of data quality. Moreover, SFT and our method outperform LaMDA-137B on four datasets, and are superior to PaLM-60B, Llama2 7B to 13B on parts of datasets, showing the reasoning potential of the smaller models. Compared with the SFT method, which employs the causal attention mechanism, our method effectively improves the general reasoning ability of models, which achieves an improvement of 7.3% on GSM8K in testing accuracy. Besides, on three out-of-distribution datasets, applying our method also obtains significant improvements.

**Results on Llama2 7B backbone.** Our approach outperforms the RFT method, which collects more reasoning paths as augmented data for fine-tuning,
Math Word Problems
Methods Backbone #Params. GSM8K MultiArith ASDiv SV AMP
Closed-sourced models
GPT-3.5-Turbo-1106 - - 77.4 97.2 90.4 79.2
Code-Davinci-002 175B 63.1 95.8 80.4 76.4
Kojima et al. (2022) LaMDA 137B 14.8 45.0 46.6 37.5
Chowdhery et al. (2022) PaLM 60B 29.9 75.0 61.9 46.7
Open-sourced models
Chung et al. (2022) Flan-T5 3B 13.5 24.0 20.7 17.7
Chung et al. (2022) Flan-T5 11B 16.1 51.7 36.5 39.7
Touvron et al. (2023) LLama2 7B 13.7 45.2 46.6 37.5
Touvron et al. (2023) LLama2 13B 25.3 64.8 59.0 43.2
Touvron et al. (2023) LLama2 70B 52.1 92.5 74.7 67.4
Specialized models with Flan-T5
SpecialFT (Fu et al., 2023a) Flan-T5 3B 22.4 42.3 28.4 23.8
SpecialFT (Fu et al., 2023a) Flan-T5 11B 27.1 63.0 37.6 35.6
DialCoT-S-PPO (Han et al., 2023) Flan-T5 3B 25.6 46.9 30.7 27.1
DialCoT-S-PPO (Han et al., 2023) Flan-T5 11B 37.1 68.1 40.9 41.7
SFT Flan-T5 3B 28.0 59.2 48.8 38.8
SFT w/ Ours† Flan-T5 3B 35.3 69.3 54.3 53.6
Specialized models with LLama2
RFT (Yuan et al., 2023) LLama2 7B 45.3 90.5 50.9 39.8
WizardMath (Luo et al., 2023a) LLama2 7B 56.7 89.0 61.1 61.4
SFT LLama2 7B 46.0 90.0 51.0 51.0
SFT w/ Ours† LLama2 7B 52.1 90.0 60.3 59.2
MetaMath (Yu et al., 2024) LLama2 7B 65.0 96.7 75.0 72.4
MetaMath w/ Ours† LLama2 7B 68.0 97.0 79.3 77.6
Table 2: The accuracy of various LLMs on four mathematical datasets. † denotes that the performance improvements over standard SFT and MetaMath are statistically significant with \( p < 0.05 \).

and is close to WizardMath, which applies reinforcement learning and additional augmented data. Besides, after adopting our method on the MetaMath dataset, the general reasoning ability of the model is further enhanced, which obtains an average improvement of 3.2% in testing accuracy.

Overall results summary. The specialized mathematical datasets are important for models to enhance reasoning ability because all specialized models get significant improvements compared with their backbones. Moreover, applying our approach to different settings, such as the different backbones, they all achieve significant improvements. Finally, the above experiment results demonstrate that our method is beneficial to modeling the relationship of different CoT reasoning steps and improving the general mathematical reasoning ability.

6 Analysis

6.1 Evaluation of the Reversal Ability

To further evaluate the effectiveness of our method in improving the reversal reasoning ability, we conducted experiments on two reversal mathematical datasets designed previously. We directly adopt the models described in Section 5 to perform inference. As shown in Table 3, all specialized models tuned on the mathematical dataset, get improvements compared with their backbones. Compared with standard SFT, our method achieves obvious improvements, even outperforming RFT and close to WizardMath on LLama2-7B. It’s noticed that

<table>
<thead>
<tr>
<th>Models</th>
<th>GSM8K-Rev</th>
<th>MathQA-Rev</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-3.5-Turbo</td>
<td>52.2</td>
<td>44.6</td>
</tr>
<tr>
<td>Flan-T5-3B</td>
<td>3.5</td>
<td>5.8</td>
</tr>
<tr>
<td>Flan-T5-11B</td>
<td>12.3</td>
<td>9.6</td>
</tr>
<tr>
<td>Flan-T5-3B (SFT)</td>
<td>8.2</td>
<td>10.0</td>
</tr>
<tr>
<td>w/Ours†</td>
<td>17.4</td>
<td>13.1</td>
</tr>
<tr>
<td>LLama2-7B</td>
<td>7.0</td>
<td>10.3</td>
</tr>
<tr>
<td>LLama2-7B (RFT)</td>
<td>20.8</td>
<td>14.5</td>
</tr>
<tr>
<td>LLama2-7B (WizardMath)</td>
<td>25.9</td>
<td>24.1</td>
</tr>
<tr>
<td>LLama2-7B (SFT)</td>
<td>21.7</td>
<td>16.5</td>
</tr>
<tr>
<td>w/Ours†</td>
<td>25.7</td>
<td>23.5</td>
</tr>
<tr>
<td>LLama2-7B (MetaMath)</td>
<td>50.0</td>
<td>37.0</td>
</tr>
<tr>
<td>w/Ours†</td>
<td>55.7</td>
<td>40.3</td>
</tr>
</tbody>
</table>

Table 3: The accuracy of various LLMs on reversal GSM8K and MathQA test datasets.
MetaMath, using backward data augmentation, significantly improves the reversal reasoning ability, which presents the importance of reversal augmented data. After training with our method on the MetaMath dataset, LLama2-7B obtains higher accuracy and outperforms the strong baseline GPT-3.5-Turbo on GSM8K-Reversal. The above results further evaluate that our method is able to model the relationship between CoT reasoning steps better and further improve the reversal reasoning ability.

### 6.2 Effect of Hyper-parameter $\alpha$

As described in Section 4.3, we introduce an additional training objective and combine it with the causal objective. To explore the effect of $L_{bid}$ weight, we set the pre-defined hyper-parameter $\alpha$ from 0.2 to 0.8, train LLama2-7B on the GSM8K dataset, and evaluate it on the GSM8K and SVAMP datasets. As shown in Figure 4, our method achieves the best performance when $\alpha$ is 0.4. If $\alpha$ is too large, $L_{bid}$ could damage the original causal paradigm and reduce the performance. On the contrary, once $\alpha$ becomes too small, the $L_{bid}$ has no effect on training and can’t provide effective context modeling.

### 6.3 Extensibility of Method

To evaluate the extensibility of our method, we conduct experiments on two commonsense reasoning datasets: CQA (Talmor et al., 2019) and QASC (Khot et al., 2020), which are respectively five-choice and eight-choice question-answering datasets. We denote each instance in datasets as $(x, y, r)$, where $x$ is a commonsense question with multiple choices, $y$ is its corresponding label, namely one of the correct choices, and $r$ is a rationale that describes the knowledge of correct label with multi-step. We also leverage GPT-3.5-Turbo-1106 to generate a related rationale $r$ for every sample. The specific prompt is shown in Appendix A.3.

Following Hsieh et al. (2023) and Li et al. (2022), we set the T5 model as the backbone, and denote the training method $[f(x) \rightarrow y]$ as $x2y$, $[f(x) \rightarrow y] + [f(x) \rightarrow y + r]$ as $(x2y) + (x2(y + r))$ (Li et al., 2022), $[f(x) \rightarrow y] + [f(x) \rightarrow r]$ as $(x2y) + (x2r)$ (Hsieh et al., 2023), where $f$ is the training model. The above training methods are utilized as strong baselines. To sufficiently model the relationships between $y$ and every step in $r$, we adopt the $(x2(y + r) + (x2(y + r)^r)$ method, where $(y + r)^r$ denotes that employ our bidirectional modeling on $(y + r)$. Table 4 illustrates that $r$ is beneficial to provide more knowledge to improve the performance. Besides, our method outperforms all the baselines, which demonstrates the effectiveness and generalization of our method.

### 7 Conclusion

In this paper, we discussed the “reversal curse” in mathematical problems and proposed two reversal datasets based on the GSM8K and MathQA to evaluate whether LLMs face challenges in reversal mathematical reasoning. We analyzed the potential reason and attributed it to the insufficient modeling of the relationship between reasoning steps. Consequently, based on the characteristics of multi-step deduction tasks, a bidirectional training objective is designed to alleviate this problem. Finally, we conducted experiments on four mathematical reasoning benchmarks to evaluate the effectiveness of our method and also verify the benefit of our approach on two reversal datasets we constructed. In the future, we will explore other enhanced methods and explore how to apply them in the pre-training stage to improve the general reasoning ability.
8 Limitation

In this section, we present several of the limitations of this paper. Firstly, the process of reversal data construction needs to be further refined, such as utilizing GPT-3.5 or GPT-4 to generate high-quality data and verify their accuracy. Besides, our method designs an additional training objective which could increase the cost of computational resources. Finally, we have not yet applied our method to larger-scale models, such as LLama2-13B and LLama2-70B, due to the limitation of computational resources. We will further explore the performance of our method on these larger-scale models.

9 Acknowledgements

We want to thank all the anonymous reviewers for their valuable comments. Juntao Li and Bowen Yan are the corresponding authors. This work was supported by the National Science Foundation of China (NSFC No. 62206194), the Natural Science Foundation of Jiangsu Province, China (Grant No. BK20220488), Young Elite Scientists Sponsorship Program by CAST (2023QNRC001), and Supercomputing Center in Yancheng, Grant No. 20231001.

References


Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. The reversal curse: Llms trained on "a is b" fail to learn "b is a".


Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, Wenhui Chen, and Xifeng Yan. 2022. Explanations from large language models make small reasoners better.


Ang Lv, Kaiyi Zhang, Shufang Xie, Quan Tu, Yuhan Chen, Ji-Rong Wen, and Rui Yan. 2023. Are we falling in a middle-intelligence trap? an analysis and mitigation of the reversal curse.

Jun-Yu Ma, Jia-Chen Gu, Zhen-Hua Ling, Quan Liu, and Cong Liu. 2023. Untying the reversal curse via bidirectional language model editing.


Da Wu, Jingye Yang, and Kai Wang. 2023. Not all large language models (llms) succumb to the "reversal curse": A comparative study of deductive logical reasoning in bert and gpt models.


Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhengu Li, and Yu Li. 2023a. Progressive-hint prompting improves reasoning in large language models.

A  Appendix

A.1  Experiment Details
We adopt the AdamW (Kingma and Ba, 2014) optimizer to train all the models. Following Chung et al. (2022) and Yuan et al. (2023), we respectively fine-tune Flan-T5 and LLama2 models for 50 and 3 epochs, and set batch size as 64 and 128, learning rate $5e^{-5}$ and $1e^{-5}$. For LLama2 with different enhanced methods, we directly adopt the checkpoints provided in huggingface to perform inference. We run all the experiments on eight NVIDIA A100-PCIE-40GB.

A.2  Prompts for CoT Reasoning Paths
The specific prompts for GPT-3.5-Turbo-1106 to obtain CoT reasoning paths are shown in Figure 5.

A.3  Prompts for QA Reasoning
The specific prompts for GPT-3.5-Turbo-1106 to obtain QA rationales are shown in Figure 6.

A.4  Case study
To understand the effectiveness of our approach better, we provide an example of LLama2-7B with SFT and our approach on the GSM8K dataset. As shown in Figure 7, SFT doesn’t sufficiently understand the relationships of different conditions in the question, which causes the wrong reasoning process. On the contrary, benefiting from the bidirectional modeling, our method understands the context and infers the correct answer.
**Prompt:**
You are a helpful and precise assistant, please follow the examples and answer the question with reasoning, finally give that 'The answer is' with calculating expressions.

**Question:** Paige was helping her mom plant flowers and together they planted some seeds. They put 10 seeds in each flower bed. If there are 45 flowerbeds, how many seeds did they plant?
**Answer:** They put 10 seeds in each flower bed. There are 45 flower beds. So they planted $10 \times 45$ seeds. $10 \times 45 = 450$. The answer is 450.

**Question:** At the arcade Dave had won 14 tickets and lost 2 tickets. If he used 10 tickets to buy some toys, how many tickets did Dave have left?
**Answer:** Dave started with 14 tickets. He lost 2 tickets. So he had $14 - 2 = 12$ tickets. He used 10 tickets to buy toys. So he had $12 - 10 = 2$ tickets left. The answer is $((14 - 2) - 10)$.

**Question:** Jack received 3 emails in the afternoon, 6 emails in the morning and some more in the evening. If he received a total of 10 emails in the day, how many emails did Jack receive in the evening?
**Answer:** Jack received 3 emails in the afternoon, 6 emails in the morning and some more in the evening. If he received a total of 10 emails in the day, then he received $10 - (3 + 6) = 1$ emails in the evening. The answer is $(10 - (3 + 6))$.

**Question:** Morisette and Kael were asked to bring fruits. Morisette brought 5 apples and 8 oranges, and Kael brings twice the apples and half the oranges that Morisette brings. They have 27 fruits in total. How many oranges does Morisette bring?
**Answer:** Morisette brought 5 apples and 8 oranges. Kael brought twice the amount of apples which is $2 \times 5 = 10$ apples, and half the number of oranges which is $\frac{1}{2} \times 8 = 4$ oranges. Together, they have $5 + 8 + 10 + 4 = 27$ fruits. Therefore, Morisette brought 8 oranges.

**Question:** If Ann is 9 years old and her brother is twice her age, how old will her brother be in 3 years?
**Answer:** Ann's brother is twice as old as she is. In 3 years, her brother will be $2 \times 9 + 3 = 21$ years old.

**Question:** What will move to another area if their habitat will no longer support them?
**Answer:** If a habitat can no longer support animals, then those animals will move to another area. Cows are social animals, but they do not move to another area unless their habitat becomes unsuitable.

**Question:** With the exception of allergies, what may cause a person to seek medical attention?
**Answer:** Microorganisms can cause infections. Infections usually require medical treatment.

**Question:** How do you reduce pollution?
**Answer:** Conserving resources has a positive impact on the environment. Use of resources affects the environment such as pollution.

**Question:** What environmental factors might influence the growth of plants?
**Answer:** Factors such as temperature, sunlight, water, and soil quality can influence the growth of plants.

**Question:** How do you conserve energy?
**Answer:** Conserve energy by using less resources. The answer is using less resources. Why?

**Question:** With the exception of allergies, what may cause a person to seek medical attention?
**Answer:** Contact with microorganisms is an example of a cause that may require medical treatment.

**Question:** How do you reduce pollution?
**Answer:** By using less resources.

**Question:** What is the impact of reducing pollution?
**Answer:** Reducing pollution has a positive impact on the environment.

**Figure 5:** The prompt for obtaining CoT reasoning steps with a conclusion that “The answer is”.

**Figure 6:** The prompt for obtaining the commonsense reasoning steps based on the correct choice.
Question:
Below is an instruction that describes a task. Write a response that appropriately completes the request.
Instruction: Carlos, Jim and Carrey were at the beach playing and they decided to gather some seashells. Jim collected 27 seashells, which was 5 more than what Carlos collected. Carlos collected twice as many as Carrey. They gathered all their seashells and divided them equally between themselves. How many did each person get? (The correct answer is 20)

SFT Answer:
First, we know that Carlos collected twice as many seashells as Carrey. So, if Carlos collected 27 seashells, then Carrey collected $27 / 2 = 13.5$ seashells. Next, we know that Jim collected 5 more seashells than Carlos, so he collected $27 + 5 = 32$ seashells. Now, we have the total number of seashells collected by all of them: $27$ (Carlos) + $13.5$ (Carrey) + $32$ (Jim) = 72 seashells. Finally, we divide the total number of seashells equally between themselves: $72 / 3 = 24$ seashells for each person. The answer is 24.

Ours Answer:
We know that Jim collected 27 seashells, which was 5 more than what Carlos collected. So, Carlos collected $27 - 5 = 22$ seashells. Carlos collected twice as many as Carrey, so Carrey collected $22 / 2 = 11$ seashells. When they gathered all their seashells, they had a total of $27 + 22 + 11 = 60$ seashells. If they divided them equally between themselves, each person would get $60 / 3 = 20$ seashells. So, each person got 20 seashells. The answer is $60 / 3$.

Figure 7: A case study about the reasoning procession on the GSM8K dataset.