# Thought-Path Contrastive Learning via Premise-Oriented Data Augmentation for Logical Reading Comprehension

**Chenxu Wang**[1], **Ping Jian**[*1,2], **Zhen Yang**[1]

[1]School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China
[2]Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Beijing Institute of Technology, Beijing, China
{wangchenxu, pjian, bityangzhen}@bit.edu.cn

## Abstract

Logical reading comprehension is a challenging task that involves understanding the underlying semantics of text and applying reasoning to deduce the correct answer. Prior researches have primarily focused on enhancing logical reasoning capabilities through Chain-of-Thought (CoT) or data augmentation. However, previous work constructing chain-of-thought rationales concentrates solely on analyzing correct options, neglecting the incorrect alternatives. Addtionally, earlier efforts on data augmentation by altering contexts rely on rule-based methods, which result in generated contexts that lack diversity and coherence. To address these issues, we propose a Premise-Oriented Data Augmentation (PODA) framework. This framework can generate CoT rationales including analyses for both correct and incorrect options, while constructing diverse and high-quality counterfactual contexts from incorrect candidate options. We integrate summarizing premises and identifying premises for each option into rationales. Subsequently, we employ multi-step prompts with identified premises to construct counterfactual context. To facilitate the model's capabilities to better differentiate the reasoning process associated with each option, we introduce a novel thought-path contrastive learning method that compares reasoning paths between the original and counterfactual samples. Experimental results on three representative LLMs demonstrate that our method can improve the baselines substantially across two challenging logical reasoning benchmarks (ReClor and LogiQA 2.0).

**Code** — https://github.com/lalalamdbf/TPReasoner

## 1 Introduction

Logical reasoning is a fundamental component of human cognition, essential for comprehending text and applying reasoning to deduce appropriate conclusions. Recently, challenging logical reasoning benchmarks have been proposed through machine reading comprehension (MRC) tasks (Yu et al. 2020; Liu et al. 2023a), which require models to derive the correct answer based on the given context, question and options. With the advent of large language models (LLMs), enhancing their capabilities in logical reasoning is a crucial step toward achieving strong artificial intelligence (Chollet 2019). Especially, the highly advanced model, GPT-4 (Achiam et al. 2023), has exhibited remarkable abilities to handle such tasks. However, a broad spectrum of open LLMs, including LLaMA2 (Touvron et al. 2023), Mistral (Jiang et al. 2023) and LLaMA3 (AI@Meta 2024), still fall short in logical reasoning, significantly trailing behind GPT-4. Consequently, improving the logical reasoning capabilities of community models has increasingly attracted the attention of many researchers (Liu et al. 2023b; Jiang et al. 2023).

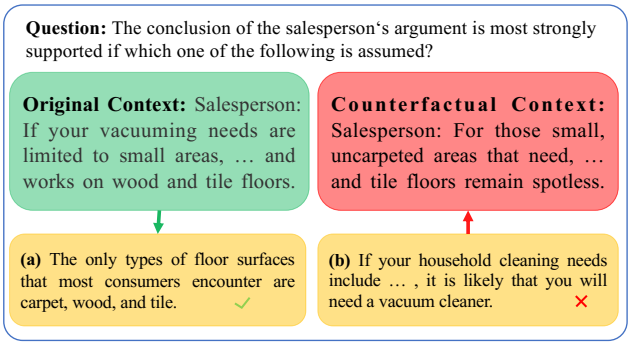


Figure 1: Generating counterfactual context from an incorrect candidate option.

For logical MRC tasks, LogiCoT (Liu et al. 2023b) constructs instruction-tuning data with Chain-of-Thought (CoT) rationales. Nevertheless, these rationales only provide analyses for the correct options, neglecting the incorrect alternatives. This oversight limits the model's ability to fully understand why certain answers are wrong, which is crucial for enhancing its reasoning capabilities and overall performance in distinguishing between similar options. In addition, previous studies typically create counterfactual contexts based on rule-based data augmentation. For instance, LReasoner (Wang et al. 2022) generates logically nonequivalent sentences by utilizing templates and syntax parsing. AMR-LDA (Bao et al. 2023) constructs counterfactual sentences based on Abstract Meaning Representation (AMR, Banarescu et al., 2013) graph and logical laws. These methods rely on complex principles and make minimal changes to the text, that cannot ensure the diversity of generated content and accurate modifications to its underlying logic. Ad-

*Corresponding author.

| |
|---|
| **Context:** [Content of the context] |
| **Question:** [Content of the question] |
| **Options:** [Content of the options] |
| **Summarize Premises:**<br>1. [Premise 1]<br>2. [Premise 2]<br>3. [Premise 3]<br>**Analyze Options:**<br>(a) [Thought-path 1]<br>Identify Premises : Unrelated to the premises.<br>(b) [Thought-path 2]<br>Identify Premises: Supported by premises 2 and 3.<br>(c) [Thought-path 3]<br>Identify Premises: Unrelated to the premises.<br>(d) [Thought-path 4]<br>Identify Premises: Contradicted by premise 1.<br>[A summary of thought-paths]. Therefore, the optimal correct answer is (b). |

Table 1: A logical reasoning example. The CoT rationale is annotated by GPT-3.5 or GPT-4. Due to space constraints, we refer to the specific reasoning process as [thought-path].

ditionally, they directly modify the context without considering its relationship with the options, which leads to a mismatch between the counterfactual context and options.

In view of above challenges, we propose a premises-oriented data augmentation (PODA) framework. As shown in Figure 1 and Table 1, the objective of PODA is to generate CoT rationales that include analyses for both correct and incorrect options, while also constructing counterfactual contexts based on incorrect candidate options. In Table 1, analyses for both correct and incorrect options are presented in *Analyze options*. Besides, we incorporate summarizing premises and identifying premises for each option into rationales. Each option has a specific relationship with these premises—either supported, contradicted, or unrelated. PODA will create high-quality and diverse counterfactual contexts using multi-step prompts based on these premises and relationships. Furthermore, since supervised fine-tuning (SFT) focuses solely on individual instances, it lacks the comparison between different samples. For original and counterfactual samples, there are thought-paths that indicate similar and dissimilar reasoning processes associated with options. Therefore, we propose a thought-path contrastive learning approach, which specifically compares thought-paths across different samples, facilitating the model's capabilities to better distinguish diverse reasoning paths. The main contributions of this paper are summarized as follows:

- We propose a premise-oriented data augmentation framework, which can generate CoT rationales involving analyses for both correct and incorrect options, while automatically constructing diverse and high-quality counterfactual data from incorrect candidate options.
- We introduce a thought-path contrastive learning approach, facilitating models to distinguish different reasoning paths between original and counterfactual samples.
- Experimental results conducted on representative open LLMs (LLaMA2-7B, Mistral-7B and LLaMA3-8B) demonstrate that our method achieves superior performance on two logical MRC benchmarks.

## 2 Related Work

### 2.1 Chain-of-Thought Prompting

LLMs are capable of performing complex reasoning to derive the final answer by generating intermediate reasoning steps through a process called Chain-of-Thought (CoT). Zero-shot-CoT (Kojima et al. 2022) showcases impressive reasoning performance only using a single instruction "Let's think step by step". Few-shot-CoT (Zhang et al. 2022; Wang et al. 2023) further boosts the reasoning abilities of LLMs by incorporating several CoT demonstrations. In addition, by offering carefully-crafted CoT demonstrations, LLMs can be encouraged to develop the similar reasoning skills and deliver responses in a uniform format. To ensure obtained CoTs are well-structured, we adopt Few-shot-CoT for data collection using GPT-3.5 and GPT-4. Recently, Liu et al. (2023c) also utilized GPT-4 to annotate the intermediate steps of correct options for logical MRC tasks. In contrast, our study expands the analysis to include incorrect options and focuses on mining information from CoT rationales to generate new logical MRC data.

### 2.2 Logical Reasoning

Leveraging logical reasoning capabilities embodies a comprehensive approach to natural language understanding (NLU). Previous studies have primarily focused on integrating logical knowledge into language models. For example, Huang et al. (2021) exploited a logic graph to model semantic relationships. Wang et al. (2022) and Bao et al. (2023) constructed equivalent/nonequivalent instances through intricate logic rules and entity replacement. These techniques, however, are constrained by their reliance on manually designed rules, which struggles to reliably identify complex logical relationships in diverse texts. Thus, our work shifts away from annotating logical relationships. Instead, we decompose and construct contexts using premises as the foundational units. Moreover, our contrastive learning approach improves LLMs' logical reasoning capabilities by enabling them to distinguish various thought-paths.

## 3 Methodology

Figure 2 shows the overall architecture of our method (PODA-TPCL). It consists of two key components: Premise-Oriented Data Augmentation (PODA) and Thought-Path Contrastive Learning (TPCL). The former module is aimed at generating CoT rationales that comprise analyses for correct and incorrect options, while constructing diverse and high-quality counterfactual logical reasoning data from incorrect candidate options. The latter one enhances the reasoning capabilities by comparing thought-paths between the original and counterfactual samples.

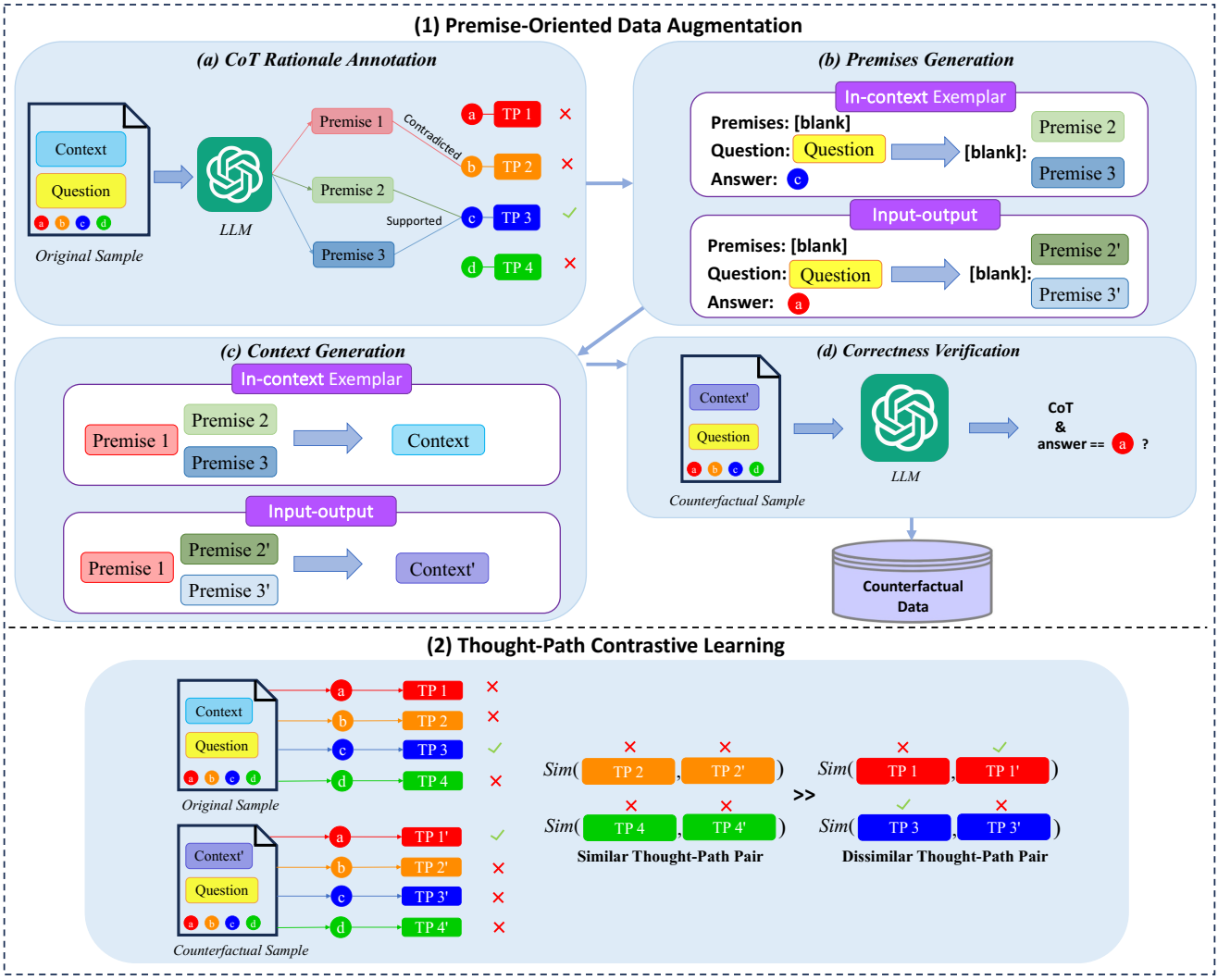


Figure 2: The overall architecture of our method. (1) PODA annotates Chain-of-Thought (CoT) rationales and generates counterfactual logical reasoning data. (2) The original and counterfactual samples are used for thought-path contrastive learning.

### 3.1 Premise-Oriented Data Augmentation

PODA initially creates analyses by forming thought-paths for both correct and incorrect options. Summarizing premises and identifying premises for each option are incorporated into CoT rationales, which are essential for generating new data. The core idea of it is to prompt a large language model through in-context learning to generate counterfactual data that can reverse the current answer to a new answer. A context can be divided into *Premises* (the known information from the text), which have specific relationships with the options. The relationships are categorized into three types: *supported*, *contradicted* and *unrelated*. Utilizing the premise as a foundational unit, we can construct counterfactual samples based on these relationships.

**CoT Rationale Annotation** As illustrated in Table 1, we design a structural CoT consisting of three steps: (1) *Summarize Premises*: Extract supporting statements from the context to serve as premises. (2) *Analyze Options*: Conduct a thorough evaluation of each option, clarifying the specific relationships between the options and premises, referred to as *thought-path* in our work. (3) *Derive answer*: Combine all thought-paths and determine the final answer. To guarantee a well-structured CoT, we utilize Few-shot-CoT for data collection.[1]

**Premises Generation** To prompt GPT-4 for new premises generation, we use a masked natural language inference (NLI) format to build the prompt. Let $P_a$ represent the premises, associated with the question $Q$ and the current answer $a$. We replace $P_a$ with a mask token **[blank]**, and then $P_a$ serves as the output of the in-context exemplar to satisfy the question and answer. Given a new answer $a'$ we want to

[1]In some cases where the model generates an incorrect response, we provide the correct answer to assist in its self-correction and refinement of the reasoning process.

flip to, we ask the model to complete **[blank]** with creative premises $P_{a'}$ that align with $a'$. This approach enables the model to generate counterfactual premises that are logically consistent with the new answer.

**Context Generation** To create new contexts, we preserve the origin premises $P_{\neg a}$ that are irrelevant to the current answer $a$, and introduce counterfactual premises $P_{a'}$ corresponding to the new answer $a'$. Let origin context $C$ serve as the output of the in-context exemplar, which is consistent with all origin premises $P = \{P_a, P_{\neg a}\}$. According to the reorganized premises $P' = \{P_{a'}, P_{\neg a}\}$, we ask the model to craft a creative context $C'$ that develops the ideas and scenarios presented. This newly crafted context integrates the counterfactual premises while maintaining coherence and expanding upon the original narrative structure.

**Correctness Verification** Upon a combination of the initial three stages, we then implement correctness verification using Few-shot-CoT to filter out incorrect samples. The prompt and output format of this stage align with CoT Rationale Annotation. The potential mistakes of samples primarily stem from the following three aspects:

1. Some options are excessively absolute in their wording (e.g., using *must* or *can't*), which conflicts with the nature of the question, making them unsuitable as correct answers.

2. Several options in the original samples are deliberately designed as incorrect choices that violate common sense. Thus, it is inappropriate to create new contexts based on these options.

3. Given the complexity of logical reasoning, it is challenging to ensure that the generated premises align perfectly with the expected answers for particularly difficult samples.

## 3.2 Thought-Path Contrastive Learning

Supervised fine-tuning (SFT) can notably improve the model's performance. However, SFT only focuses on single instances, which results in its lack of the comparison between different samples. For logical MRC tasks, PODA annotates Chain-of-Thought (CoT) rationales, offering analyses for both correct and incorrect options, while also generating counterfactual samples. It can be observed that thought-paths exhibit similar and dissimilar reasoning processes associated with options in original and counterfactual samples.

In light of such motivation, we propose a thought-path contrastive learning approach. As depicted in the part (2) of Figure 2, the original/counterfactual sample has four thought-paths, with each corresponding to one option. Three thought-paths indicate that the corresponding options are incorrect, while one thought-path suggests it is correct. Therefore, for the original and counterfactual sample pair, the reasoning processes of thought-paths 2 and 2′ (thought-paths 4 and 4′) are analogous, whereas those of thought-paths 1 and 1′ (thought-paths 3 and 3′) are different. The goal of our method is to pull similar thought-paths closer while pushing different ones far apart. Simultaneously, we seek to enhance the model's capabilities to precisely distinguish between pairs of thought-paths (e.g., similarity(thought-paths 4, 4′) $\gg$ similarity(thought-paths 2, 2′)).

Inspired by recent advances in learning to preference optimization algorithms such as RLHF (Ouyang et al. 2022) and DPO (Rafailov et al. 2023), our objective is to present a simple approach for comparing the similarity of thought-path pairs. To achieve this objective, we employ the Bradley-Terry preference model (Bradley and Terry 1952) to construct the loss function for the similarity comparison. Given the input pair $\pi_0 = (x_1, x_2)$, the Bradley-Terry model calculates the likelihood of similarity comparison over thought-path pairs, denoted as $(p_s, p'_s) > (p_d, p'_d) \mid \pi_0$, where $(p_s, p'_s)$ and $(p_d, p'_d)$ represent the similar and different thought-path pairs, respectively. In our work, we simply choose the reward function $r^* = \frac{1}{\tau}\text{sim}(\cdot)$ to measure similarity using cosine distance, where $\tau$ is the temperature coefficient controlling the sharpness of the similarity distribution. Under the Bradley-Terry model, we can derive a streamlined probability measure for pairwise similarity comparison:

$$\begin{aligned} &p^*((p_s, p'_s) > (p_d, p'_d) \mid \pi_0) \\ &= \sigma(r(p_s, p'_s) - r(p_d, p'_d)) \\ &= \frac{1}{1 + \exp[\frac{1}{\tau}\text{sim}(p_d, p'_d) - \frac{1}{\tau}\text{sim}(p_s, p'_s)]} \end{aligned} \tag{1}$$

where $\pi_0$ is omitted as it doesn't directly contribute to the calculation. Then we formulate the problem as binary classification using the negative log-likelihood loss:

$$\begin{aligned} &\mathcal{L}((p_s, p'_s), (p_d, p'_d), \pi_0)) \\ &= -\mathbb{E}[\log(\sigma(r(p_s, p'_s) - r(p_d, p'_d)))] \end{aligned} \tag{2}$$

The objective of this loss function is to decrease the distance between dissimilar pair $(p_d, p'_d)$ while increasing the distance between similar pair $(p_s, p'_s)$. Additionally, the model learns to differentiate between pairs of thought-pairs based on preference optimization. Due to the presence of multiple groups of similar and dissimilar thought-path pairs for input $\pi_0$, we simply calculate the average loss as follows:

$$\begin{aligned} &\mathcal{L}_{\text{TPCL}}((p_s, p'_s), (p_d, p'_d), \pi_0)) = \\ &-\mathbb{E}[\frac{1}{N * M}\sum_{i=1}^{N}\sum_{j=1}^{M}\log(\sigma(r(p_{sj}, p'_{sj}) \\ &- r(p_{di}, p'_{di})))] \end{aligned} \tag{3}$$

where $N$ and $M$ represent the number of similar and dissimilar thought-path pairs respectively. Both $N$ and $M$ are set to 2 in our work. We also add a cross-entropy loss consistent with SFT to ensure the model does not deviate from the data distribution. Given an input sequence $x$, the average likelihood of generating the output sequence $y$, consisting of $m$ tokens, is computed as follows:

$$\mathcal{L}_{\text{SFT}} = \frac{1}{m}\sum_{t=1}^{m}\log P(y_t \mid x, y_{<t}) \tag{4}$$

The overall training goal is the combination of TPCL loss and SFT loss:

$$\mathcal{L} = \mathcal{L}_{\text{TPCL}} + \mathcal{L}_{\text{SFT}} \tag{5}$$

| Model | ReClor | | | | LogiQA 2.0 | |
|---|---|---|---|---|---|---|
| | Dev | Test | Test-E | Test-H | Dev | Test |
| **Discriminative Language Models** | | | | | | |
| RoBERTa-Large (Liu et al. 2019) | 62.60 | 55.60 | 75.50 | 40.00 | - | - |
| DGAN (Huang et al. 2021) | 65.80 | 58.30 | 75.91 | 44.46 | - | - |
| LReasoner (Wang et al. 2022) | 66.20 | 62.40 | **81.40** | 47.50 | - | - |
| AMR-LDA (Bao et al. 2023) | 65.26 | 56.86 | 77.34 | 40.77 | - | - |
| FocalReasoner (Ouyang, Zhang, and Zhao 2024) | 66.80 | 58.90 | 77.10 | 44.60 | - | - |
| **Instruction-tuned LLMs** | | | | | | |
| LLaMA2-7B-logicot (Liu et al. 2023b) | 49.20 | 50.50 | 59.06 | 43.75 | 45.06 | 43.19 |
| LLaMA3-8B-logicot | 61.50 | 62.65 | 71.25 | 55.89 | 54.11 | 54.07 |
| Mistral-7B-logicot | 63.00 | 61.90 | 70.45 | 55.18 | 55.83 | 54.25 |
| **API-based LLMs (3-shot-CoT)** | | | | | | |
| GPT-3.5 (gpt-3.5-turbo-0613) | 56.00 | 58.20 | 61.82 | 55.36 | 55.07 | 51.15 |
| GPT-4 (gpt-4o) | 87.20 | 89.30 | 90.45 | 88.39 | 76.32 | 74.81 |
| **TPReasoner** | | | | | | |
| LLaMA2-7B | 58.00 | 58.63 | 66.14 | 52.74 | 49.71 | 49.75 |
| LLaMA3-8B | <u>67.60</u> | <u>70.97</u> | 77.27 | **66.01** | <u>60.29</u> | <u>58.78</u> |
| Mistral-7B | **69.73** | **71.17** | <u>78.79</u> | <u>65.18</u> | **61.22** | **60.28** |

Table 2: Experimental results (Accuracy %) of our method compared with baseline models on ReClor and LogiQA 2.0 benchmarks. Segment-1: Discriminative language models; Segment-2: Instruction-tuned LLMs; Segment-3: API-based LLMs (3-shot-CoT); Segment-4: TPReasoner (our method). Test-E and Test-H denote Test-Easy and Test-Hard respectively. The best and second best results are marked in bold and underlined (comparisons do not include API-based LLMs).

# 4 Experiment

## 4.1 Datasets

**ReClor** (Yu et al. 2020) comprises 6,138 question-answering samples collected from standardized exams including GMAT and LSAT, which are split into train / dev / test sets with 4,638 / 500 / 1,000 samples respectively. To evaluate the difficulty of the questions, the test set is further divided into Test-E and Test-H. The instances on Test-E are easy and biased that can be solved without knowing contexts and questions. The other harder and unbiased ones are taken as the Test-H set.

**LogiQA 2.0** (Liu et al. 2023a) is an updated and re-annotated version of LogiQA (Liu et al. 2020). There are 15,708 instances derived from the Chinese Civil Service Examination, meticulously translated into English by experts. The dataset is randomly split into train / dev / test sets with 12,567 / 1,569 / 1,572 samples respectively.

**Synthetic Data** is generated by our PODA framework. We construct 5,075 and 13,477 counterfactual samples based on the train sets of ReClor and LogiQA 2.0, respectively. To perform thought-path contrastive learning, each counterfactual sample is paired with its corresponding origin one.

## 4.2 Implementation Settings

In PODA framework, gpt-3.5-turbo-0613 and gpt-4-0613 are utilized for CoT Rationale Annotation. Subsequently, gpt-4-0125-preview is employed for Premises and Context Generation. In the end, gpt-4-0613 is used for Correctness Verification.[2] We set the sampling temperature of 0.75 and the top probability of 0.9, ensuring the generated text maintains both diversity and high quality. The detailed prompts are provided in Appendix C.2.

During the training process, we adopt LLaMA2-7B, Mistral-7B and LLaMA3-8B as baselines. In order to accelerate training, we employ LoRA (Hu et al. 2021) to fine-tune the model. The AdamW optimizer (Loshchilov and Hutter 2017) is used with a learning rate warmup of 0.03. Due to the absence of corresponding counterfactual samples for some original samples as illustrated in Correctness Verification, we implement a two-stage training strategy. The implementation of our code refers to Llamafactory (Zheng et al. 2024). All hyper-parameters of training are listed in Appendix A.

## 4.3 Baselines

In this paper, we compare our method with two types of baselines: discriminative language models and large language models (LLMs). The discriminative language model baselines include RoBERTa-Large (Liu et al. 2019), DGAN (Huang et al. 2021), LReasoner (Wang et al. 2022), AMR-LDA (Bao et al. 2023) and FocalReasoner (Ouyang, Zhang, and Zhao 2024). The LLM baselines encompass instruction-tuned method such as LogiCoT (Liu et al. 2023b), as well as API-based LLMs like GPT-3.5 (gpt-3.5-turbo-0613) and GPT-4 (gpt-4o) by utilizing 3-shot-CoT. More details can be found in Appendix B.

[2]We found that gpt-4-0613 exhibits a superior capability for generating data in a structural format compared to gpt-4-0125-preview. Hence, we chose gpt-4-0613 to annotate CoT rationales.

| Model | ReClor | | | | LogiQA 2.0 | |
|---|---|---|---|---|---|---|
| | Dev | Test | Test-E | Test-H | Dev | Test |
| LLaMA2-7B | **58.00** | **58.63** | **66.14** | **52.74** | **49.71** | **49.75** |
| - w/o TPCL | 55.27 | 56.73 | 63.48 | 51.43 | 48.25 | 48.60 |
| - w/o TPCL + CD | 53.20 | 52.17 | 59.16 | 46.67 | 46.15 | 45.48 |
| - w/o TPCL + CD + WOA | 51.40 | 50.27 | 56.21 | 45.58 | 44.89 | 44.47 |
| LLaMA3-8B | **67.60** | **70.97** | **77.27** | **66.01** | **60.29** | **58.78** |
| - w/o TPCL | 66.00 | 69.30 | 74.09 | 65.54 | 57.11 | 57.06 |
| - w/o TPCL + CD | 63.07 | 66.10 | 73.11 | 60.59 | 55.19 | 54.83 |
| - w/o TPCL + CD + WOA | 61.17 | 64.03 | 71.53 | 58.11 | 53.51 | 53.07 |
| Mistral-7B | **69.73** | **71.17** | **78.79** | **65.18** | **61.22** | **60.28** |
| - w/o TPCL | 67.07 | 69.57 | 78.03 | 64.40 | 59.91 | 58.63 |
| - w/o TPCL + CD | 65.20 | 67.37 | 75.68 | 62.92 | 58.24 | 56.97 |
| - w/o TPCL + CD + WOA | 63.10 | 64.83 | 73.02 | 58.37 | 56.04 | 55.19 |

Table 3: Ablation study of our method. TPCL stands for thought-path contrastive learning approach. CD refers to the utilization of counterfactual data. WOA signifies that CoT rationales involve the analyses for wrong options.

# 5 Result and Analysis

## 5.1 Overall Results

Table 2 presents the primary experimental results of our method and other baselines on ReClor and LogiQA 2.0 benchmarks, in terms of accuracy. Our method employs the CoT rationales, which is appropriate for generative LLMs. Hence, we did not conduct experiments on discriminative language models. Compared to these baselines, TPReasoner exhibits superior performance except for GPT-4.

On ReClor dataset, LLaMA3-8B and Mistral-7B, based on our approach, significantly outperform all discriminative model methods. Compared with LReasoner, PODA-TPCL achieves improvements of 1.4-3.5% and 8-9% on the dev and test sets, respectively. Although LLaMA2-7B, when using our method, does not surpass all discriminative model baselines, the results on Test-H demonstrate that it exhibits stronger robustness and generalization for data distribution. There is a substantial disparity (exceeding 30%) between the performances on Test-E and Test-H for discriminative models, indicating that these models tend to take shortcuts for simpler and biased samples rather than genuinely comprehending them. In contrast, our method achieves a gap of less than 15% between Test-E and Test-H, with the performance on Test-H clearly surpassing that of the discriminative models. This demonstrates the great potential of leveraging CoT rationales to solve complex logical reasoning tasks.

Compared to the instruction-tuned LLMs based on LogiCoT, our models achieve superior performance on ReClor and LogiQA 2.0 datasets. This improvement is attributed to the PODA framework, which offers additional analyses of incorrect options and constructs high-quality, diverse counterfactual instances. Additionally, TPCL boosts the model's reasoning capabilities by facilitating the learning of similar and distinct thought-paths between different samples. Furthermore, our model demonstrates competitive performance comparable to GPT-3.5, trailing only behind GPT-4.

## 5.2 Ablation Study

An ablation study is conducted to investigate the efficacy of three key components, thought-path contrastive learning (TPCL), counterfactual data (CD) and wrong options analyses (WOA), as presented in Table 3. For w/o TPCL, we eliminate TPCL and only employ SFT to train the model. There is a noticeable decline in performance, with a drop of 1-3% across the two datasets. These results convincingly demonstrate that TPCL significantly boosts the model's reasoning capabilities by comparing reasoning paths between original and counterfactual samples. For w/o TPCL + CD, we additionally exclude the counterfactual samples generated by PODA and solely utilize the original data. It can be observed that the models without CD have severe performance degradation. This suggests that the counterfactual samples are beneficial for LLMs to conduct logical reasoning. Furthermore, it demonstrates that the data synthesized by our framework is of high-quality, automatically generated without requiring human interventions. For w/o TPCL + CD + WOA, we further omit the analyses of wrong options in CoT rationales. As a result, the models' performance decreases by approximately 2%. It indicates that incorporating reasoning processes for incorrect options enables the model to analyze problems more thoroughly, thereby improving its reasoning abilities. Overall, PODA-TPCL achieves a performance improvement of 5-7% across three models on ReClor and LogiQA 2.0 datasets, underscoring its exceptional robustness and generalization capabilities.

## 5.3 Evaluation of Data Quality

**Accuracy Evaluation of Counterfactual Data** A primary concern is whether the synthetic data accurately matches the correctly labeled option. In order to evaluate this, we choose five outstanding LLMs, including Mixtral-8×7B-Instruct, GPT-3.5 (gpt-3.5-turo-0613), LLaMA2-70B-chat, LLaMA3-70B-Instruct and GPT-4 (gpt-4o-2024-05-13). Non-GPT series models evaluate all counterfactual data. Due to budget constraints, a random subset of 200 samples from the generated dataset are evaluated by GPT-3.5 and

| Model | Accuracy |
|---|---|
| Mixtral-8×7B-Instruct | 74.05 |
| GPT-3.5 (gpt-3.5-turo-0613) | 75.50 |
| LLaMA2-70B-chat | 79.39 |
| LLaMA3-70B-Instruct | 82.19 |
| GPT-4 (gpt-4o-2024-05-13) | 93.00 |
| Human Performance | 90.00 |

Table 4: Evaluation of accuracy (%) for counterfactual data.

GPT-4. We utilize 3-shot CoTs to evaluate the accuracy. As illustrated in Table 4, the accuracy assessed by LLaMA3-70B-Instruct and GPT-4 can reach 82.19% and 93% respectively, indicating PODA can generate high-quality counterfactual data. Moreover, accuracies for other models vary between approximately 75% and 79%, reflecting the significant challenges and complexities presented by these data. Overall, the generated data can have applicability in both training and evaluation domains.

**Comparison for Counterfactual Data** We compare PODA with a rule-based method, LReasoner (Wang et al. 2022), which constructs counterfactual contexts by modifying logical expressions according to logical laws. Two random subsets of 200 counterfactual samples were selected respectively. We ensure that the two subsets are derived from the same set of original samples for a fair comparison. These instances are evaluated by GPT-4 (gpt-4o-2024-05-13) using four key metrics: Coherence (Is the context well-connected and logically consistent?), Clarity (Is the context clear and easy to understand?), Relevance (Does the context relate to the question and options?), and Diversity (How does the counterfactual context differ from the original one?). Each context was rated on a scale from 1 (poor) to 5 (excellent) for each metric. Our method attains average scores of 4.61 for Coherence, 4.36 for Clarity, 4.71 for Relevance, 3.18 for Diversity, and 4.22 Overall. In contrast, LReasoner achieves average scores of 2.98 for Coherence, 2.96 for Clarity, 4.63 for Relevance, 1.08 for Diversity, and 2.91 Overall. This comparison clearly demonstrates that the contexts generated by our method significantly outperform those produced by the rule-based method in both quality and diversity.

**Comparison for CoT Rationales** We compare PODA with LogiCoT, which also utilizes GPT-4 for CoT rationales but focuses solely on analyzing the correct option. Similarly, two random subsets of CoT rationales were selected from the same contexts. These rationales are assessed by GPT-4 (gpt-4o-2024-05-13) using four key metrics: Coherence (Is the CoT rationale logically consistent?), Completeness (Does it offer a thorough explanation for the reasoning?), Relevance (Does it directly and effectively addresses the context, question and options?), and Faithfulness (Is it factually correct and free from fabricated details?). Each rationale was rated on a scale from 1 (poor) to 5 (excellent) for each metric. PODA surpasses LogiCoT across four metrics, especially in Completeness. This suggests that incorporating the analysis of incorrect options into the rationales can enhance the quality of CoTs.

| Method | Coh | Clar | Rel | Div | Overall |
|---|---|---|---|---|---|
| LReasoner | 2.98 | 2.96 | 4.63 | 1.08 | 2.91 |
| PODA | **4.61** | **4.36** | **4.71** | **3.18** | **4.22** |

Table 5: Evaluating counterfactual data on Coherence (Coh), Clarity (Clar), Relevance (Rel), and Diversity (Div).

| Method | Coh | Comp | Rel | Faith | Overall |
|---|---|---|---|---|---|
| LogiCoT | 4.82 | 3.38 | 4.91 | 4.80 | 4.48 |
| PODA | **4.86** | **4.46** | **4.99** | **4.86** | **4.79** |

Table 6: Evaluating rationales on Coherence (Coh), Completeness (Comp), Relevance (Rel), and Faithfulness (Faith).

### 5.4 What Does TPCL Update Do?

We analyze the gradient of the loss function $\mathcal{L}_{\text{TPCL}}$ (considering a group of similar and dissimilar thought-path pairs), whose gradient can be written as:

$$\begin{aligned}\nabla\mathcal{L}_{\text{TPCL}}((p_s,p_s'),(p_d,p_d'),\pi_0)) = \\ -\frac{1}{\tau}\mathbb{E}[\sigma(r(p_d,p_d')-r(p_s,p_s')) \\ * \ [\nabla\text{sim}(p_s,p_s')-\nabla\text{sim}(p_d,p_d')]]\end{aligned} \tag{6}$$

Intuitively, the gradient of $\mathcal{L}_{\text{TPCL}}$ increases the similarity of the similar thought-paths $((p_s,p_s')$ and decreases the similarity of the dissimilar thought-paths $((p_d,p_d')$. Meanwhile, $\sigma(r(p_d,p_d')-r(p_s,p_s'))$ serves as an adjustable weight for the similarity reward estimate, assigning greater weight when $r(p_d,p_d')$ is approximately equal to or greater than $r(p_s,p_s')$. This mechanism accelerates the convergence of the loss function. Overall, the gradient update aligns with our objective to pull similar thought-paths closer while pushing dissimilar ones further apart, which enhances the model's reasoning capabilities by comparing thought-paths between the original and counterfactual samples.

## 6 Conclusion

In this paper, we propose a premise-oriented data augmentation framework that generates CoT rationales, providing analyses for both correct and incorrect options. Additionally, the framework automatically constructs diverse and high-quality counterfactual data from incorrect candidate options. We also introduce a thought-path contrastive learning method to effectively leverage pairs of original and counterfactual samples, enabling models to distinguish between different reasoning paths. Extensive experiments conducted on three open LLMs demonstrate that our approach achieves competitive performance on two logical reasoning benchmarks.

## Acknowledgments

This work is supported by the grants from the National Natural Science Foundation of China (No. 62172044 and No.

62376130). The authors would like to thank the organizers of AAAI 2025 and the reviewers for their helpful suggestions.

## References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 Model Card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md. Accessed: 2024-05-12.

Banarescu, L.; Bonial, C.; Cai, S.; Georgescu, M.; Griffitt, K.; Hermjakob, U.; Knight, K.; Koehn, P.; Palmer, M.; and Schneider, N. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, 178–186.

Bao, Q.; Peng, A. Y.; Deng, Z.; Zhong, W.; Tan, N.; Young, N.; Chen, Y.; Zhu, Y.; Witbrock, M.; and Liu, J. 2023. Contrastive learning with logic-driven data augmentation for logical reasoning over text. *arXiv preprint arXiv:2305.12599*.

Bradley, R. A.; and Terry, M. E. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4): 324–345.

Chollet, F. 2019. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Huang, Y.; Fang, M.; Cao, Y.; Wang, L.; and Liang, X. 2021. DAGN: Discourse-Aware Graph Network for Logical Reasoning. In Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tur, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; and Zhou, Y., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 5848–5855. Online: Association for Computational Linguistics.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Liu, H.; Liu, J.; Cui, L.; Teng, Z.; Duan, N.; Zhou, M.; and Zhang, Y. 2023a. Logiqa 2.0—an improved dataset for logical reasoning in natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Liu, H.; Teng, Z.; Cui, L.; Zhang, C.; Zhou, Q.; and Zhang, Y. 2023b. LogiCoT: Logical Chain-of-Thought Instruction Tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2908–2921.

Liu, H.; Teng, Z.; Cui, L.; Zhang, C.; Zhou, Q.; and Zhang, Y. 2023c. LogiCoT: Logical Chain-of-Thought Instruction Tuning. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2908–2921. Singapore: Association for Computational Linguistics.

Liu, J.; Cui, L.; Liu, H.; Huang, D.; Wang, Y.; and Zhang, Y. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

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

Ouyang, S.; Zhang, Z.; and Zhao, H. 2024. Fact-Driven Logical Reasoning for Machine Reading Comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 18851–18859.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, H.; Wang, R.; Mi, F.; Deng, Y.; Wang, Z.; Liang, B.; Xu, R.; and Wong, K.-F. 2023. Cue-CoT: Chain-of-thought Prompting for Responding to In-depth Dialogue Questions with LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 12047–12064.

Wang, S.; Zhong, W.; Tang, D.; Wei, Z.; Fan, Z.; Jiang, D.; Zhou, M.; and Duan, N. 2022. Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Findings of the Association for Computational Linguistics: ACL 2022*, 1619–1629. Dublin, Ireland: Association for Computational Linguistics.

Yu, W.; Jiang, Z.; Dong, Y.; and Feng, J. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. *arXiv preprint arXiv:2002.04326*.

Zhang, Z.; Zhang, A.; Li, M.; and Smola, A. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Zheng, Y.; Zhang, R.; Zhang, J.; Ye, Y.; and Luo, Z. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.