Exploring Self-supervised Logic-enhanced Training for Large Language Models

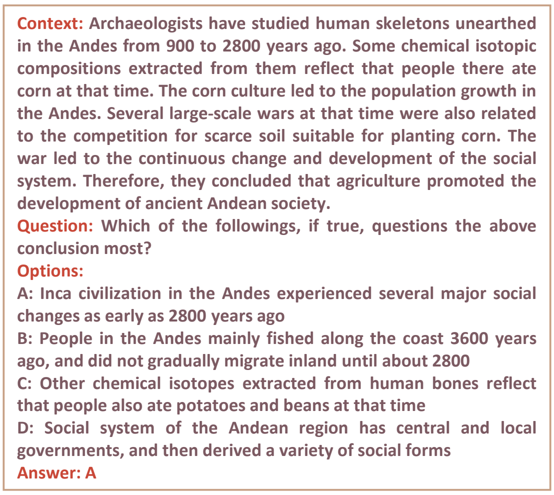
Fangkai Jiao[1,2] **Zhiyang Teng**[1] **Bosheng Ding**[1] **Zhengyuan Liu**[2]
Nancy F. Chen[2,†] **Shafiq Joty**[1,3,†]
[1]Nanyang Technological University, Singapore
[2]Institute for Infocomm Research (I^{2}R), A*STAR, Singapore [3]Salesforce Research
jiaofangkai@hotmail.com bosheng001@e.ntu.edu.sg zhiyang.teng@ntu.edu.sg
{nfychen, liu_zhengyuan}@i2r.a-star.edu.sg sjoty@salesforce.com

Abstract

Traditional attempts to enhance the logical reasoning abilities of language models often rely on supervised fine-tuning, limiting their generalization to new tasks or domains. Large Language Models (LLMs), with their capacity to condense vast knowledge, can effectively tackle many tasks. Yet, our experiments reveal a gap in their performance on logical reasoning benchmarks when compared to state-of-the-art fine-tuning based models. To bridge this gap, we present LogicLLM, a first-of-its-kind, fully self-supervised framework for integrating logical reasoning capabilities into LLMs, and activating them via in-context learning. We apply this to two LLM series, FLAN-T5 and LLaMA, with parameter sizes from 3 billion to 33 billion. LogicLLM demonstrates its effectiveness through successful improvements on two logical reasoning benchmarks (ReClor and LogiQA-v2). Additionally, LogicLLM based on FLAN-T5-11B attains comparable results to ChatGPT, and evaluations with LLaMA-based models on three language understanding benchmarks (RACE, MMLU and Big-Bench-Hard) confirm that the improvements come without compromising the model's general language understanding capabilities.[1]

1 Introduction

Logical reasoning serves as a bedrock for negotiation, debate and writing, underpinning our ability to engage with complex cognitive tasks (Yu et al., 2020). An example of logic reasoning in natural language is shown in Figure 1. As the complexity of relations and expressions presented in this task defy straightforward conversion into symbolic or formal languages, perfecting logical reasoning within language models has proven to be a significant challenge (Zhong et al., 2021).



Context: Archaeologists have studied human skeletons unearthed in the Andes from 900 to 2800 years ago. Some chemical isotopic compositions extracted from them reflect that people there ate corn at that time. The corn culture led to the population growth in the Andes. Several large-scale wars at that time were also related to the competition for scarce soil suitable for planting corn. The war led to the continuous change and development of the social system. Therefore, they concluded that agriculture promoted the development of ancient Andean society.
Question: Which of the followings, if true, questions the above conclusion most?
Options:
A: Inca civilization in the Andes experienced several major social changes as early as 2800 years ago
B: People in the Andes mainly fished along the coast 3600 years ago, and did not gradually migrate inland until about 2800
C: Other chemical isotopes extracted from human bones reflect that people also ate potatoes and beans at that time
D: Social system of the Andean region has central and local governments, and then derived a variety of social forms
Answer: A

Figure 1: An example logical reasoning task from LogiQA-v2 dataset (Liu et al., 2020). The relations between different constituents, e.g., *agriculture* and *development of Andean society*, include various predicates, and it is hard to be converted into logical form through either first-order logic or formal language.

Past attempts to incorporate logical reasoning into language models primarily focused on integrating knowledge about logic. For instance, Huang et al. (2021) employed graph neural networks to capture relational semantics, while Wang et al. (2022) used data augmentation to implement first-order logic. These techniques, however, are constrained by their need for extensive annotated training data, which hinders the model's ability to generalize across different tasks due to disparities in data distribution and optimization objectives.

Conversely, recent breakthroughs in Large Language Models (LLMs) like PaLM (Chowdhery et al., 2022), LLaMA (Touvron et al., 2023), ChatGPT[2], GPT-4 (OpenAI, 2023), and Bard[3] offer a promising alternative. These LLMs effectively encapsulate a vast array of knowledge and tackle diverse tasks with minimal specialization, guided by human instruction. Despite their potential, our

[†] Correspondence to: Nancy F. Chen and Shafiq Joty.

[1] The code and models are released at https://github.com/SparkJiao/LogicLLM.

[2] https://openai.com/blog/chatgpt

[3] https://bard.google.com/

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 926–941
June 16-21, 2024 ©2024 Association for Computational Linguistics

experiments on logical reasoning benchmarks revealed deficiencies in their logical reasoning capabilities as shown later in our experiments.

Contemporary efforts to fortify LLMs' specific capabilities fall broadly into two categories. The first employs external tools or APIs (Schick et al., 2023; Mialon et al., 2023; Cheng et al., 2022; Gao et al., 2022; Chen et al., 2022), aiding LLMs in argument parsing and semantic understanding. Yet, these tools' utility for logical reasoning remains limited due to the absence of a symbolic language for problem descriptions. The second category, instruction tuning, relies on data augmentation or enriched human feedback but struggles due to the scarcity of task-specific data and high annotation costs (Ouyang et al., 2022; Xu et al., 2023). In this work, we pivot away from these traditional methods and introduce LogicLLM, which performs self-supervised logic-enhanced meta-training for LLMs. It tackles two primary challenges: 1) synthesising logic-consistent data from raw texts ensuring fully self-supervised training, and 2) effectively incorporating logic prior into LLMs while preventing learning problems, such as memorization, forgetting and generalization.

To tackle the first challenge, LogicLLM emphasizes the necessity of understanding and exploiting **fuzzy logical consistency**. As mentioned previously, strict formal logic is often absent in natural language, we instead treat the **relational consistency** between different perspectives of relational expressions as an approximation to fuzzy logic consistency[4]. In fact, ensuring logical consistency in a discourse is a key requirement for text coherence and effective information conveyance (Jurafsky and Martin, 2009). We devise a method that inspects the implicit intra-sentence relation of entity pairs at the discourse level to extract logically consistent examples from Wikipedia articles (Figure 2). Specifically, we posit that direct and indirect relations of an anchor entity pair should be logically consistent, as they are derived from the "same" context. For the second challenge, LogicLLM adopts an auto-regressive objective optimizing on the logically consistent relation instances directly to make it seamlessly adapt to its pretraining objective. It tasks the model with generating the alternative perspective (indirect or direct) given a direct or indirect description of the anchor entity pair. We further employ counterfactual data augmentation through entity replacement to enforce relation-centric reasoning, which not only avoids the model's tendency to merely recall results from memory but also ensures the preservation of the logic-enhanced aspect of the learning process.

LogicLLM is task-agnostic and does not require any annotations, making it adaptable to various logical reasoning tasks. We have conducted experiments across two distinct LLM series, FLAN-T5 (Longpre et al., 2023) and LLaMA (Touvron et al., 2023), encompassing a variety of parameter sizes. These experiments are designed to investigate two main questions: (1) Can the logical reasoning capabilities be exclusively improved through self-supervised meta-training for LLMs, thereby circumventing the need for task-specific supervised fine-tuning? (2) How does the logic-enhanced meta training affect the LLM's language understanding capabilities, i.e., does it suffer from forgetting or generalization issues?

In response to the first question, our findings suggest that LLMs trained with the LogicLLM objective demonstrate superior performance on logical reasoning benchmarks, eliminating the need for further fine-tuning. Our LogicLLM based on FLAN-T5-11B attain comparable results to ChatGPT on two logic reasoning benchmarks, ReClor (Yu et al., 2020) and LogiQA-v2 (Liu et al., 2022a), highlighting the feasibility of enhancing logical reasoning abilities through self-supervised training alone.

Regarding the second question, our evaluations with LLaMA-based models on three general language understanding benchmarks - RACE (Lai et al., 2017), MMLU (Hendrycks et al., 2021) and BIG-Bench-Hard (BBH) (Suzgun et al., 2022), confirm that the enhanced logical reasoning capabilities do not compromise the model's overall language understanding on MMLU and BBH. In fact, the learned logic ability appears to boost the model's performance in RACE.

2 Related Work

2.1 Large Language Models

In recent years, Large Language Models with in-context learning have emerged as a groundbreaking paradigm in the field of NLP. Unlike the traditional fine-tuning approach, in-context learning leverages natural language instructions or a small number

[4] In this paper, we will use the term **logical consistency** to represent **consistency in fuzzy logic** for simplification, which is further described by relational consistency. This means that the relationship between a **logically consistent** data pair has a higher degree of logical consistency but cannot be strictly proved considering the diverse expressions of relations.

of annotated examples as demonstrations to predict responses for new instances. This unique approach empowers LLMs to serve as a versatile tool for handling multiple tasks without requiring task-specific training. However, recent evaluations of LLMs (Qin et al., 2023; Bang et al., 2023; Jiao et al., 2023; Laskar et al., 2023; Wang et al., 2023a) have revealed a limitation in their ability to learn complex skills like logic and planning through language modeling alone. To address this, even the training of GPT-4 has incorporated labeled matching datasets to enhance its performance in solving math word problems (OpenAI, 2023). Nevertheless, due to the vast amount of data used in pre-training LLMs, annotated data for specific capabilities may be severely undersampled, and the cost of obtaining annotations should not be overlooked. Therefore, it remains crucial to develop various self-supervised or weakly-supervised training methods that do not rely on human annotation. These approaches are essential for constructing more robust and versatile LLMs that can perform a wider range of tasks with higher proficiency and lower resource.

2.2 Reasoning in Natural Language

Previous research aimed at natural language reasoning tasks can be broadly classified into three categories. The first category involves explicit prior knowledge, such as discourse structure or linguistic knowledge, to model implicit reasoning processes (Gao et al., 2020; Huang et al., 2021). The second category is neural-symbolic reasoning, where variables are first parsed, and then predefined programs are executed to obtain final results (Wang et al., 2022; Zhong et al., 2021). However, a significant challenge with these methods is the requirement of a robust semantic parser and a self-contained symbolic system for extracting variables or arguments, which is impractical for logic reasoning based on natural language. The third category encompasses methods that focus on general domain pre-training for reasoning via denoising auto-encoding (Jiao et al., 2021; Deng et al., 2021; Liu et al., 2022b). Nevertheless, restricted by the poor task generalization of discriminative models with few parameters, these methods are still in demand of task-specific fine-tuning to activate learned knowledge.

Our approach in this paper falls within the third category, which improves the efforts of MERIt (Jiao et al., 2022) by transforming it into auto-regressive framework to better align the nature of LLMs as generative model. We also drop the usage of knowledge graph enabling enhancing the logic of LLMs through purely self-supervised learning.

3 LogicLLM

Figure 2 shows the framework of LogicLLM. It involves three main steps: 1) Logic-consistent Data Construction (Section 3.1), which synthesises the logic-consistent data using relation discrimination between entity pairs; 2) Counterfactual Data Augmentation (Section 3.2), which augments the logic-consistent training data by entity sampling and replacement; 3) LLM Training (Section 3.3), which performs continual training of LLMs using the training data generated by the previous two steps.

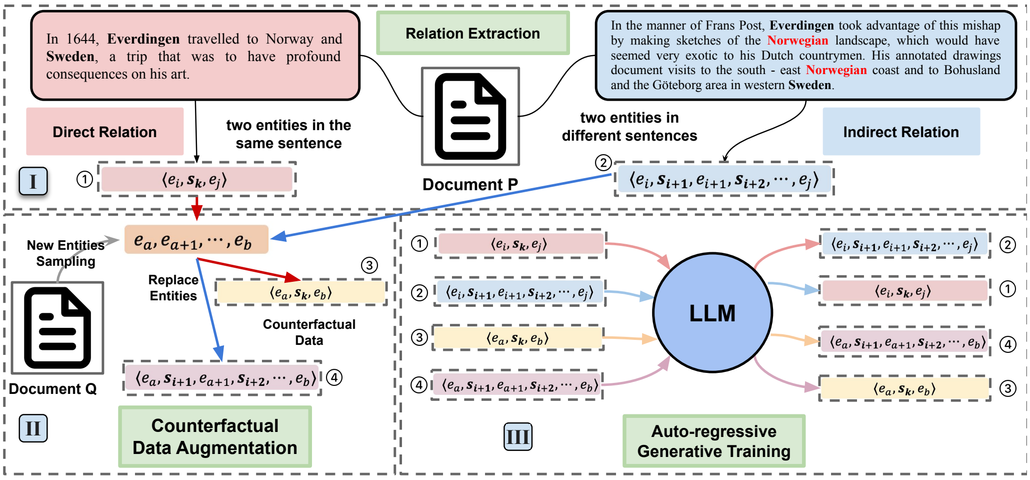
3.1 Logically consistent Data Construction

Ensuring logical consistency in discourse and pragmatics is a fundamental prerequisite for natural language to effectively convey information and maintain coherence. Consequently, logically consistent data is prevalent in text documents and various techniques can be applied to extract them. In this study, we implement this by inspecting intra-sentence relation of entity pairs at the discourse level to extract logically consistent examples from Wikipedia.

Direct relation Given an arbitrary paragraph and an anchor entity pair $\langle e_i, e_j \rangle$, **we assume there exists an *implicit* relation s_k between $\langle e_i, e_j \rangle$ if one sentence directly mentioning them can be found.** This comes from the distant supervision (Mintz et al., 2009) and has been employed and extended in self-supervised training by previous work (Deng et al., 2021). For example, the instance ① in Figure 2 is a direct relation. To this end, we simply treat $\langle e_i, s_k, e_j \rangle$ as the direct relation triplet for further data construction.

Indirect relation Entities e_i and e_j can be indirectly connected through multiple sentences within the input paragraph. In such situations, we identify a chain of triplets, such as $\langle e_i, s_{i+1}, e_{i+1}, \cdots, s_j, e_j \rangle$, which represents an indirect relation between the entity pair $\langle e_i, e_j \rangle$ through the relation composition of serial relation triplets $\langle e_i, s_{i+1}, e_{i+1} \rangle$, $\langle e_{i+1}, s_{i+2}, e_{i+2} \rangle$, $\cdots$, $\langle e_{j-1}, s_j, e_j \rangle$. For example, instance ② in Figure 2 demonstrates an indirect relation.[5]

[5] In practice, we find 87% indirect relations are composed of two relation triplets, 12% contain three triplets, and less than 1% have more than 4 triplets. This prevents the logical consistency be weakened by long context.



In 1644, **Everdingen** travelled to Norway and **Sweden**, a trip that was to have profound consequences on his art.

In the manner of Frans Post, **Everdingen** took advantage of this mishap by making sketches of the **Norwegian** landscape, which would have seemed very exotic to his Dutch countrymen. His annotated drawings document visits to the south - east **Norwegian** coast and to Bohusland and the Göteborg area in western **Sweden**.

Relation Extraction
Direct Relation
two entities in the same sentence
Document P
two entities in different sentences
Indirect Relation
I
① $\langle e_i, s_k, e_j \rangle$
② $\langle e_i, s_{i+1}, e_{i+1}, s_{i+2}, \cdots, e_j \rangle$
New Entities Sampling
$e_a, e_{a+1}, \cdots, e_b$
Replace Entities
③ $\langle e_a, s_k, e_b \rangle$
Counterfactual Data
④ $\langle e_a, s_{i+1}, e_{a+1}, s_{i+2}, \cdots, e_b \rangle$
Document Q
II
Counterfactual Data Augmentation
① $\langle e_i, s_k, e_j \rangle$
② $\langle e_i, s_{i+1}, e_{i+1}, s_{i+2}, \cdots, e_j \rangle$
③ $\langle e_a, s_k, e_b \rangle$
④ $\langle e_a, s_{i+1}, e_{a+1}, s_{i+2}, \cdots, e_b \rangle$
LLM
$\langle e_i, s_{i+1}, e_{i+1}, s_{i+2}, \cdots, e_j \rangle$ ②
$\langle e_i, s_k, e_j \rangle$ ①
$\langle e_a, s_{i+1}, e_{a+1}, s_{i+2}, \cdots, e_b \rangle$ ④
$\langle e_a, s_k, e_b \rangle$ ③
III
Auto-regressive Generative Training

Figure 2: The LogicLLM framework. P and Q are two arbitrary paragraphs from Wikipedia. In Step 1, we extract intra-sentence relations ①: $\langle\, e_i, s_k, e_j \,\rangle$, and the compositions of them ②: $\langle e_i, s_{i+1}, e_{i+1}, \cdots, s_j, e_j \rangle$ from P for an entity pair $\langle\, e_i,\ e_j \,\rangle$; ① and ② are direct and indirection relations, respectively. Here s_k is a relation, represented by the sentence that mentions $\langle\, e_i,\ e_j \,\rangle$. ① and ② are viewed as logically consistent since both of them describe the "same" relation between $\langle\, e_i,\ e_j \,\rangle$ from different view. In Part I of the figure, e_i refers to *Everdigen* and e_j represents *Sweden*. The intermediate entity is *Norwegian* here. The direct relation on the left says that *Everdigen* has traveled to *Sweden*, and the indirect relation implies the fact that *Everdigen* has probably visited *Sweden* as well as its nearby area, otherwise he could not complete the sketches of *Norwegian*, demonstrating the fuzzy logic consistency with high probability. Step 2 is the process of counterfactual data augmentation, where counterfactual relation composition is generated by random entity replacement. ③ and ④ are the counterfactual augmentations of ① and ②, respectively. Finally, in Step 3, the LLM is optimized to generate direct/indirect relations with their logically consistent indirect/direct counterparts as inputs. Here, ①→ ②, ②→ ①, ③→ ④, and ④→ ③ are considered.

Logical consistency Intuitively, the direct and indirect relations between $\langle\, e_i,\ e_j \,\rangle$ should be logically consistent since they are derived from same context and describing the same entity pairs. Instances ① and ② in Figure 2 exemplify logically consistent relations. By establishing implicit connections between single-step and multi-hop reasoning, LLMs gain the ability to understand relation composition process between s_k and $\langle s_{i+1}, s_{i+2}, \cdots, s_{j-1} \rangle$. This capability consequently enhances the LLMs' logical reasoning abilities.

To retrieve logically consistent relation pairs, we follow a two-step process. First, we recognize all entities within each paragraph via distant annotation from WikiData (Wang et al., 2021). And secondly, we enumerate every possible entity pair and search for a series of sentences and check if both direct and indirect relations can be extracted.

3.2 Counterfactual Data Augmentation

The work we have described in Section 3.1 produces logically consistent data that correlates entities and relations within reasoning paths. To enhance entity-irrelevant reasoning and ensure LLM focuses more on the process of relational composition rather than the entities themselves, we have additionally introduced counterfactual data augmentation. This approach, similar to the method suggested by Jiao et al. (2022), includes the random replacement of entities.

To create counterfactual examples of $\langle\, e_i,\ e_j \,\rangle$ within paragraph P, we initially select a random paragraph, denoted as Q, from a separate document. Subsequently, we sample a new set of entities, such as $e_a, e_{a+1}, \cdots, e_b$ from Q. The head and tail entities in the original relation instances of $\langle\, e_i,\ e_j \,\rangle$ are then substituted by these randomly sampled entities, maintaining the relationships unchanged. For instance, after substituting e_i and e_j with e_a and e_b, ③ and ④ become the counterfactual augmentations of ① and ②, respectively. In our research, we postulate that the logic-consistency between s_k and $s_{i+1}, e_{i+1}, s_{i+2}, \cdots, s_{j-1}$ remains undisturbed in the counterfactual examples. This assertion is based on the idea that logical relationships within a paragraph's context are primarily driven by shared entities and their interconnections rather than the specific entities themselves.

3.3 Training Objective

During the training phase, we apply continual training to LLMs using logic-consistent data. Drawing inspiration from the success of in-context learning, we treat one relation from a logic-consistent relation pair as the in-context example and task the LLM with generating the other relation. As depicted in Figure 2, using the logic-consistent pair ⟨①, ②⟩ as an example, when ① is given as the conditional input, the LLM is expected to produce ② as the output, and vice versa. This process intuitively forces the LLM to reason the logic-consistent connections between the input and output relations since they are from the same context and the entity pairs of ① and ② are both e_i and e_j.

Formally, we denote the data extracted from Section 3.1 and Section 3.2 as $D = \{\langle R_i^1, R_i^2\rangle\}_{i=1}^{N}$, where N represents the number of training examples, and $\langle R_i^1, R_i^2\rangle$ is the i-th logic-consistent record. Here, R_i^1 refers to the direct relation-related instance, while R_i^2 represents the instance with an indirect relation. The goal of LLM training is to minimize the negative log-likelihood function as follows:

$$\begin{aligned}\mathcal{L}_{\text{logic}} &= -\sum_{i=1}^{N}[\log P(R_i^1|R_i^2) + \log P(R_i^2|R_i^1)] \\ &= -\sum_{i=1}^{N}[\sum_{j=1}^{|R_i^1|}\log P(R_{i,j}^1|R_{i,<j}^1, R_i^2) \\ &\quad + \sum_{j=1}^{|R_i^2|}\log P(R_{i,j}^2|R_{i,<j}^2, R_i^1)],\end{aligned} \tag{1}$$

where $R_{i,j}^1$, $R_{i,j}^2$ denotes the j-th token of R_i^1 and R_i^2, respectively.

Furthermore, we incorporate the another causal language modeling loss $\mathcal{L}_{\text{lm}}$ to mitigate the catastrophic forgetting problem. Both $\mathcal{L}_{\text{lm}}$ and $\mathcal{L}_{\text{logic}}$ are implemented as auto-regressive decoding. The only difference is that they sample from different data source. $\mathcal{L}_{\text{lm}}$ continuously samples data from the subset of training corpus used during the last-stage pre-training, i.e., Wikipedia paragraphs for LLaMA series models, and FLAN-collection-v2 for FLAN-T5 series models. Therefore, the overall training objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{logic}} + \mathcal{L}_{\text{lm}}. \tag{2}$$

During training, for each forward-backward, we randomly sample two mini-batches with the same size from the datasets for logic-enhanced training and language modeling, respectively, and merge them into a single one.

Model / Dataset	ReClor Dev Acc.	 Test Acc.	LogiQA-v2 Dev Acc.	 Test Acc.
ChatGPT	56.6	61.2	54.5	52.7
LLaMA-7B	30.2	30.3	27.4	28.1
w/ LogicLLM	**32.4**	**31.0**	**27.7**	**28.6**
LLaMA-13B	30.4	33.5	33.0	32.1
w/ LogicLLM	**37.4**	**36.3**	**34.1**	**34.0**
LLaMA-33B	45.2	50.3	41.2	41.6
w/ LogicLLM†	**50.2**	**54.4**	**45.9**	**42.6**
Falcon-40B	38.4	37.1	35.9	36.1
w/ LogicLLM†	**41.4**	**43.0**	**38.6**	**37.2**
FLAN-T5-3B	54.6	52.5	48.7	48.7
w/ LogicLLM & FLAN	**55.8**	**54.1**	**50.8**	**50.1**
FLAN-T5-11B	57.4	59.9	55.3	53.1
w/ LogicLLM & FLAN	**61.2**	**61.1**	**56.0**	**54.0**

Table 1: The results on logical reasoning benchmarks. Better results are annotated in bold. † refers that the corresponding model is trained through QLoRA (Dettmers et al., 2023).

4 Experiment

We integrate our pre-training approach into two prominent LLMs: LLaMA (Touvron et al., 2023) and FLAN-T5 (Wei et al., 2022a). These models boast parameter sizes ranging from 3 billion to 30 billion. To thoroughly evaluate the capability of LLMs from various angles, we have carefully selected five datasets representing three distinct categories. ReClor (Yu et al., 2020) and LogiQA-V2 (Liu et al., 2020) are two logical reasoning benchmarks sourced respectively from standardized graduate admission examinations and logical examination papers intended for reading comprehension. RACE (Lai et al., 2017) is a reading comprehension task that assesses general reasoning abilities. MMLU (Hendrycks et al., 2021) is used for measuring the learned knowledge and massive multitask language understanding, and BIG-Bench-Hard (BBH) (Suzgun et al., 2022) is a collection of multiple challenging tasks where LLMs fall behind human being. By employing MMLU and BBH, we aim to verify whether the logic-oriented meta-training negatively impacts the models' ability to generalize across a wide range of tasks. Due to space limitation, more implementation details can be found in Appendix A.

5 Results and Analysis

5.1 Logical Reasoning

Table 1 shows the results on ReClor and LogiQA-v2 under zero-shot setting. From the table we can find that the performance of LLaMA-based models is notably lower compared to ChatGPT. By training LLaMA models with LogicLLM, we observe significant enhancement in their zero-shot logical

Model / Dataset	RACE Dev Acc.	RACE Test Acc.	MMLU 0-shot Acc.	MMLU 5-shot Acc.
LLaMA-7B	31.3	32.3	33.3	36.2
w/ LogicLLM	**37.3**	**37.9**	**34.6**	**36.6**
LLaMA-13B	55.8	54.5	41.1	46.7
w/ LogicLLM	**57.7**	**55.6**	**43.3**	**47.3**
LLaMA-33B	68.4	**68.1**	54.3	**58.3**
w/ LogicLLM†	**68.8**	**68.1**	**54.4**	**58.3**

Table 2: The results of LLaMA models on RACE and MMLU. † means training through QLoRA.

Model / Dataset	ReClor Dev Acc.	ReClor Test Acc.	LogiQA-v2 Dev Acc.	LogiQA-v2 Test Acc.
LLaMA-13B	30.4	33.5	33.0	32.1
w/ LogicLLM (*ctr*)	33.4	33.3	33.1	32.7
w/ LogicLLM (*ar*)	**37.4**	**36.3**	**34.1**	**34.0**
LLaMA-33B	45.2	50.3	41.2	41.6
w/ LogicLLM† (*no aug.*)	49.4	53.0	44.2	40.8
w/ LogicLLM† (*1 aug.*)	50.8	52.7	45.6	41.5
w/ LogicLLM†	**50.2**	**54.4**	**45.9**	**42.6**

Table 3: The effect of different training objectives. *Ctr* refers contrastive learning and *ar* means the auto-regressive variant. *no aug.* means the counterfactual data augmentation is removed from the LogicLLM framework. † means that the model is trained with QLoRA.

reasoning capabilities. For instance, on LLaMA-13B and LLaMA-33B, the average improvements across the four dataset splits are 3.2 and 3.7 points, respectively. The benefits are more substantial than those observed in the 7B models (0.9 points), which aligns with the findings on emergent abilities (Wei et al., 2022b). This could be attributed to the fact that larger models possess stronger generalization abilities and better apply their learned capabilities to different tasks. We also conducted experiments on Falcon-40B (Penedo et al., 2023), and found that LogicLLM brings an average improvement of 3.2 points.

Consistent with LLaMA-based models, we can draw similar conclusions for those based on FLAN-T5, where logic-oriented meta-training also yields improvements for both FLAN-T5-3B and FLAN-T5-11B. For FLAN-T5-11B, our model achieves accuracies of 61.2 and 61.1 on the development and test sets of ReClor, respectively. On the development and test sets of LogiQA-v2, our logic-oriented FLAN-T5-11B model achieves accuracies of 56.0 and 54.0, respectively. Notably, on the development set of ReClor, our logic-oriented FLAN-T5-11B model outperforms ChatGPT by a significant margin of 4.8 accuracy points. Similarly, on the development and test sets of LogiQA-v2, our logic-oriented FLAN-T5-11B model surpasses ChatGPT by 1.5 and 1.3 accuracy points, respectively. These overall results indicate that instruction tuning on multiple supervised datasets, such as the FLAN collection, can still be improved for learning logic. We hypothesize that this may be attributed to the sparsity of reasoning-relevant data in the entire collection and the conflicts between different tasks.

5.2 Hybrid Reasoning and Application

In addition to logical reasoning in text, we are also curious about whether logic-enhanced training contributes to general language understanding (RACE), and maintain the general capabilities on massive knowledge based tasks (MMLU). To investigate this, we evaluate the performance of the enhanced LLaMA models on these two datasets.

As shown in Table 2, from 7B to 33B, LogicLLM can consistently improve the performance on RACE, except the one of LLaMA-33B w/ LogicLLMon the test set. Specifically, LLaMA-7B w/ LogicLLM obtain around 4.2 absolute improvements, and LLaMA-13B w/ LogicLLM achieves 1.5 improvements, which has verified that the logic-enhanced training is also beneficial to general reasoning and reading comprehension. Additionally, we find that LogicLLM can also benefits the massive multitask language understanding (MMLU) on LLaMA-7B and 13B. We find that the improvements of both RACE and MMLU on LLaMA-33B are marginal, probably because low-rank adaptation have restricted the generalization.

5.3 Pre-training Strategy

LogicLLM draws inspiration from the contrastive learning framework for logical reasoning, i.e., MERIt, which has demonstrated its efficacy in fine-tuning based approaches. As mentioned earlier, we hypothesize that contrastive learning may be inadequate for LLM with in-context learning. To validate this assumption, we examine the effects of contrastive learning (*ctr*) and auto-regressive generation (*ar*). In the case of contrastive learning, we adopt the methodology of MERIt to construct logically inconsistent instances and optimize the model by maximizing the distance between logically consistent instances and the inconsistent counterparts. Referring to the table, it can be observed that LogicLLM (*ctr*) fails to yield significant improvements compared to LLaMA-13B, except for the dev set of ReClor. Conversely, the auto-regressive models consistently outperform both the baseline models and the contrastive methods by considerable margins across all dataset splits. We propose two primary reasons to explain the superiority of auto-regressive models over the contrastive approach.

First, the heuristic construction process for negative candidates used in contrastive learning fails to identify true contradictory relations, resulting in randomly chosen negative samples that lack logically opposite relationships with the positive instances. To this end, the contrastive learning process can degrade into a positive-only optimization process, which is similar to auto-regressive learning but receives less token-level supervision.

Second, the divergence between the training objectives of contrastive learning and auto-regressive generation undermines the model's ability to effectively do in-context reasoning. Contrastive learning primarily focuses on discriminating positive pairs from negative pairs based on a *global* semantic perspective. Auto-regressive models, on the other hand, accumulate their ability through *local* token prediction. During inference, LLMs are expected to understand instruction, and jointly consider the logical relations between different hypothesises within single input. By placing emphasis on fine-grained relations, the auto-regressive objective can better support in-context learning, enabling the model to grasp the nuanced connections and reasoning processes required for logical understanding.

Moreover, the auto-regressive objective significantly reduces computation costs during training by eliminating the need for negative candidates encoding. The streamlining of training process leads to more efficient and resource-friendly training without sacrificing performance. We also add another experiment by adjusting the ratio between counterfactual data and the normal ones as 1:1, and the comparison reveal that mixing more counterfactual data can also benefit the performance, which could be especially useful for low-resource domain, like finance and multi-lingual LLMs.

In summary, considering the advantages in both performance and training cost, the auto-regressive variant proves to be a superior choice for incorporating logic reasoning into LLMs.

	ReClor		**LogiQA-v2**	
Model / Dataset	Dev	Test	Dev	Test
FLAN-T5-3B				
w/ FLAN	53.6	53.8	49.5	49.5
w/ LogicLLM & FLAN	**55.8**	**54.1**	**50.8**	**50.1**
FLAN-T5-11B				
w/ FLAN	58.0	60.5	**56.9**	53.6
w/ LogicLLM & FLAN	**61.2**	**61.1**	56.0	**54.0**
LLaMA-13B				
w/ GPT4ALL	37.4	36.1	37.2	34.3
w/ LogicLLM & GPT4All	**39.2**	**37.7**	37.2	**35.1**

Table 4: Ablation study to explore if LogicLLM can be combined with instruction tuning. For FLAN-T5 , we use the subset of FLAN collection. For LLaMA, we introduce GPT4All (Anand et al., 2023).

5.4 Factors Relevant to Logic Prior

In Table 3, we also present the ablation results on LLaMA-33B when the counterfactual data augmentation strategy is omitted. Without the inclusion of counterfactual data, LogicLLM degrades into a conditional generative task that can be solved through memorization, as each sample has its own prototypes within Wikipedia.

As indicated in the table, even without the augmentation (*no aug.*), LogicLLM still contributes to the enhancement of logical reasoning abilities, albeit with more limited improvements. However, the introduction of counterfactual data augmentation to eliminate memorization effects can further amplify the benefits. The overall experimental results point out that relation construction serves as effective supervision signal for introducing logic prior. We leave the work about developing novel techniques to prevent memorization but less involve factual noise as future work.

5.5 Compatibility with Instruction Tuning

Instruction tuning has served as a critical step to make LLMs better in following human instruction, and/or generating with less toxic. In this section, we hope to study if LogicLLM can be well integrated with supervised instruction tuning so that LogicLLM has the potential to serve as a basic approach to train logic-enhanced foundation model before building applications. For FLAN-T5, we directly use the same subset of FLAN collection with our approach as the instruction tuning data. For LLaMA models, we introduce GPT4All (Anand et al., 2023) data for extra supervision. During training, we simply sum the loss of instruction tuning and LogicLLM in multitask training manner to keep the same data ratio.

Model	Normal	Normal (Anony.)	C.F.	C.F. (Anony.)
ChatGPT	94%	77.4% (-16.6%)	49.2%	65.0% (+14.8%)
GPT-4	99.8%	99.2% (-0.6%)	71.4%	94.2% (+22.8%)

Table 5: The ratio of consistent data deemed by ChatGPT and GPT-4. *Anony.* refers to *anonymization* and *C.F.* is the simplification of *Counterfactual*.

As shown in Table 4, on most dataset splits, LogicLLM can achieve additional improvements compared with the instruction tuning-only baselines. Specifically, we find that the improvements are more significant on ReClor that those on LogiQA-v2. One possible reason is that the language style in LogiQA-v2 is more close to formal language, leaving a gap with the natural user questions.

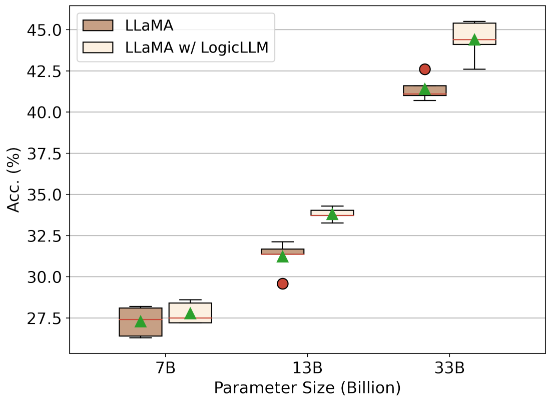


Figure 3: Results of 5 experiments with different option input orders across different model sizes on the test set of LogiQA-v2. **Brown circular marker**: outlier, **green triangle**: arithmetic mean value.

5.6 Data Assumption Auto-Verification

In order to verify the rationality of our assumption that the *direct* and *indirect* relations are logically consistent, we employ ChatGPT and GPT-4 for automatic evaluations. Specifically, we randomly sample 1,000 examples from the development set for our pre-training with the ratio of normal data and counterfactual ones as 1:1. For each data pair, we ask ChatGPT/GPT-4 to determine if the relation between the target entities are logically consistent. The prompt we used is shown in Appendix E. We have involved four different settings. Beside the normal data and the counterfactual ones, we have also applied anonymization (Qiu et al., 2020) to them to decouple the background knowledge from entity. Specifically, the target entities are replaced with [X] and [Y], and for counterfactual data, the other replaced entities during data augmentation are not further anonymized. Some cases can also be found in Appendix E for clearer understanding.

Our results are shown in Tabel 5, from which we can observe that: (1) for normal data, ChatGPT and GPT-4 deem that the logically consistent data occupie high ratios, which has initially verified the rationality of our data construction assumption. (2) For counterfactual data, the ratios significantly decrease. Yet, in the view of GPT-4, there is still more than 70% of logically consistent data in the whole corpus. (3) When combined with entity anonymization, the ratios become much higher for counterfactual data, i.e., nearly 15% absolute improvements for ChatGPT and 23% for GPT-4. Besides, the ratio of normal data decreases significantly for ChatGPT, but is less perturbed for GPT-4. The observation further demonstrates that most counterfactual data should also hold the assumption since the anonymization only remove the backgrounds of entities, yet leaving the context as original. And the great variation brought by counterfactual data augmentation also reveals the potential weakness of current LLMs on identifying the true causal relations.

5.7 Robustness

By training LLMs on logic-consistent data and counterfactual augmentations, they are exposed to a wide range of input variations. This exposure helps them become less sensitive to minor perturbations such as shuffling of input options. To determine the robustness of LogicLLM , we conducted experiments on LogiQA-v2 using models of varying sizes. We shuffled the input order of different options and reperformed the inference process.

Figure 3 illustrates the findings of our experiments. We observed that LLaMA exhibited higher variance across different input option orders, as indicated by the greater spread in results. The circular outlier values that indicate specific input orders causing significant variations, leading to substantially higher or lower performance results. Our observation is consistent with the recent findings of Wang et al. (2023b), suggesting that the normal LLMs heavily suffer from position bias. In contrast, when LLaMA is enhanced with LogicLLM, it achieves more stable performance across different parameter sizes. Moreover, the averaged performance of LLaMA w/ LogicLLM is significantly superior to that of LLaMA alone. These results show that LogicLLM produces consistent and improved results compared to traditional LLMs, demonstrat-

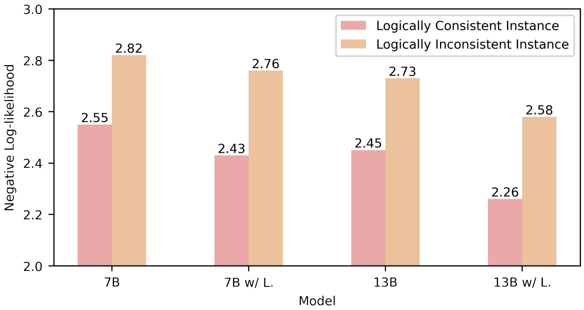


Figure 4: The averaged log-likelihood value of different models on the self-constructed logically consistent and inconsistent instances, respectively. w/ L. refers to the models augmented with LogicLLM.

ing the value of incorporating logic-enhanced training techniques into LLMs.

5.8 Training Quality Analysis

In order to analyze the quality of our meta-training, we have constructed a test set using the framework of MERIt (Jiao et al., 2022), which contains both logically consistent and inconsistent data. We have measured the log-likelihood on each sample as illustrated by Equation 1, and report the averaged results in Figure 4.

As shown in the figure, for logically consistent data, LogicLLM significantly reduced the negative log-likelihood. Moreover, the 7B-based model with LogicLLM surpasses the performance of LLaMA-13B. Notably, the disparity between the negative log-likelihood of logically consistent and inconsistent instances is further amplified, highlighting the effectiveness of LogicLLM in logical relation reconstruction. Furthermore, our experiments suggest a decrease in the negative log-likelihood for logically inconsistent data. This observation exposes a weakness in the contrastive learning-based method, i.e., MERIt, wherein the heuristic process for generating negative candidates introduces considerable noise. Consequently, some negative instances may not genuinely present contradictory logical relations.

6 Conclusion

In this paper, we have explored the feasibility and effectiveness of enhancing logical reasoning of LLMs via purely self-supervised training. We evaluate the performance based on two LLM series, i.e., FLAN-T5 and LLaMA. The experimental results on two logical reasoning benchmarks, LogiQA-v2 and ReClor, demonstrate the effectiveness of our method. And the performance on RACE, MMLU and Big-Bench-Hard have also verified that the framework do not hurt the generalization of LLMs. Finally, we have analyzed the factors relevant to logic during training, and the compability with supervised instruction tuning. We hope the analysis could bring new insights to future research.

Acknowledgements

This research is supported by the Ministry of Education, Singapore, under its Science of Learning Grant (award ID: MOE-MOESOL2021-0006). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore.

Besides, we sincerely appreciate the valuable comments from all the reviewers to help us make the paper polished. We also greatly thank to Chengwei Qin and Professor Aixin Sun for their kind suggestions.

Limitations

In this paper, we have explored the feasibility to introduce logical reasoning capability into LLMs via purely self-supervised meta-training. Though the results have demonstrated significant improvements on logical reasoning benchmarks, there are also some limitations:

Randomness from Diverse Prompt/Instruction. In our experiments, we find that the performance of LLMs, especially those never optimized by instruction tuning, is varying to different prompts. We try to reduce the variance by (1) using simpler prompt (as shown in Section D or (2) using the released prompt by commonly accepted benchmark or leaderboard, e.g., MMLU, Big-Bench-Hard and Chain-of-Thought Hub (Fu et al., 2023). Nevertheless, this still cannot entirely keep the certainty of the experimental results.

Non-uniform Evaluation Strategy. Currently, there is no *de facto* technical standard for LLMs evaluation. Some work just let language models generate the response and match the content. However, this can be unfair for non-instruction-tuned models since they often cannot generate meaningful and complete sentences, especially those under 13 billion parameters.

Scaling. Due to the resource limitation, we can only scale the method into models with 40 billion parameters under the help of low-rank adaptation.

References

Yuvanesh Anand, Zach Nussbaum, Brandon Duderstadt, Benjamin Schmidt, and Andriy Mulyar. 2023. Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *CoRR*, abs/2302.04023.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *CoRR*, abs/2211.12588.

Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2022. Binding language models in symbolic languages. *CoRR*, abs/2210.02875.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. *CoRR*, abs/2204.02311.

Xiang Deng, Yu Su, Alyssa Lees, You Wu, Cong Yu, and Huan Sun. 2021. Reasonbert: Pre-trained to reason with distant supervision. In *EMNLP*, pages 6112–6127. ACL.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *CoRR*, abs/2305.14314.

Yao Fu, Litu Ou, Mingyu Chen, Yuhao Wan, Hao Peng, and Tushar Khot. 2023. Chain-of-thought hub: A continuous effort to measure large language models' reasoning performance. *CoRR*, abs/2305.17306.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2022. PAL: program-aided language models. *CoRR*, abs/2211.10435.

Yifan Gao, Chien-Sheng Wu, Jingjing Li, Shafiq R. Joty, Steven C. H. Hoi, Caiming Xiong, Irwin King, and Michael R. Lyu. 2020. Discern: Discourse-aware entailment reasoning network for conversational machine reading. In *EMNLP*, pages 2439–2449. ACL.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *ICLR*. OpenReview.

Yinya Huang, Meng Fang, Yu Cao, Liwei Wang, and Xiaodan Liang. 2021. DAGN: discourse-aware graph network for logical reasoning. In *NAACL-HLT*, pages 5848–5855. ACL.

Fangkai Jiao, Yangyang Guo, Yilin Niu, Feng Ji, Feng-Lin Li, and Liqiang Nie. 2021. REPT: bridging language models and machine reading comprehension via retrieval-based pre-training. In *Findings of ACL/IJCNLP*, pages 150–163. ACL.

Fangkai Jiao, Yangyang Guo, Xuemeng Song, and Liqiang Nie. 2022. Merit: Meta-path guided contrastive learning for logical reasoning. In *Findings of ACL*, pages 3496–3509. ACL.

Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Xing Wang, and Zhaopeng Tu. 2023. Is chatgpt A good translator? A preliminary study. *CoRR*, abs/2301.08745.

Daniel Jurafsky and James H. Martin. 2009. *Speech and language processing*, 2. ed., [pearson international edition] edition. Prentice Hall series in artificial intelligence. Prentice Hall, Pearson Education International.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. RACE: large-scale reading comprehension dataset from examinations. In *EMNLP*, pages 785–794. ACL.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Xiangji Huang. 2023. A systematic study and comprehensive evaluation of chatgpt on benchmark datasets. In *ACL*. ACL.

Hanmeng Liu, Jian Liu, Leyang Cui, Nan Duan, Ming Zhou, and Yue Zhang. 2022a. Logiqa2.0 dataset - logical reasoning in mrc and nli tasks. *TASLP*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In *IJCAI*, pages 3622–3628.

Linlin Liu, Xin Li, Ruidan He, Lidong Bing, Shafiq R. Joty, and Luo Si. 2022b. Knowledge based multilingual language model. In *EMNLP*, pages 1–13. ACL.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning. *CoRR*, abs/2301.13688.

Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ramakanth Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, Edouard Grave, Yann LeCun, and Thomas Scialom. 2023. Augmented language models: a survey. *CoRR*, abs/2302.07842.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011. ACL.

OpenAI. 2023. Gpt-4 technical report. *Preprint*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *CoRR*, abs/2203.02155.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon LLM: outperforming curated corpora with web data, and web data only. *CoRR*, abs/2306.01116.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *CoRR*, abs/2302.06476.

Yujia Qin, Yankai Lin, Ryuichi Takanobu, Zhiyuan Liu, Peng Li, Heng Ji, Minlie Huang, Maosong Sun, and Jie Zhou. 2021. ERICA: Improving entity and relation understanding for pre-trained language models via contrastive learning. In *ACL/IJCNLP*, pages 3350–3363. ACL.

Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. 2020. GCC: graph contrastive coding for graph neural network pre-training. In *KDD*, pages 1150–1160. ACM.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *CoRR*, abs/2302.04761.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *CoRR*, abs/2210.09261.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Bin Wang, Zhengyuan Liu, Xin Huang, Fangkai Jiao, Yang Ding, Ai Ti Aw, and Nancy F. Chen. 2023a. Seaeval for multilingual foundation models: From cross-lingual alignment to cultural reasoning. *CoRR*, abs/2309.04766.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023b. Large language models are not fair evaluators. *CoRR*, abs/2305.17926.

Siyuan Wang, Wanjun Zhong, Duyu Tang, Zhongyu Wei, Zhihao Fan, Daxin Jiang, Ming Zhou, and Nan Duan. 2022. Logic-driven context extension and data augmentation for logical reasoning of text. In *ACL*, pages 1619–1629. ACL.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *TACL*, 9:176–194.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. In *ICLR*. OpenReview.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022b. Emergent abilities of large language models. *CoRR*, abs/2206.07682.

Lionel Wong, Gabriel Grand, Alexander K. Lew, Noah D. Goodman, Vikash K. Mansinghka, Jacob Andreas, and Joshua B. Tenenbaum. 2023. From word models to world models: Translating from natural language to the probabilistic language of thought. *CoRR*, abs/2306.12672.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *CoRR*, abs/2304.12244.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021. Discriminative reasoning for document-level relation extraction. In *Findings of ACL*, pages 1653–1663. ACL.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *ICLR*. OpenReview.

Shuang Zeng, Yuting Wu, and Baobao Chang. 2021. SIRE: separate intra- and inter-sentential reasoning for document-level relation extraction. In *Findings of ACL*, pages 524–534. ACL.

Wanjun Zhong, Siyuan Wang, Duyu Tang, Zenan Xu, Daya Guo, Jiahai Wang, Jian Yin, Ming Zhou, and Nan Duan. 2021. AR-LSAT: investigating analytical reasoning of text. *CoRR*, abs/2104.06598.

	ReClor		**LogiQA-v2**	
Model / Dataset	Dev Acc.	Test Acc.	Dev Acc.	Test Acc.
RoBERTa-L.	62.6	55.6	59.8	57.0
MERIt (RoBERTa-L)	69.4	61.6	**62.6**	**59.3**
MERIt (DeBERTa-XXL)	**80.6**	**78.1**	—	—
LLaMA-7B	28.8	28.3	24.4	23.7
LLaMA-13B	31.6	34.4	31.6	31.1
LLaMA-33B	45.2	50.3	41.2	41.6
GPT-3.5-turbo	56.6	61.2	54.5	52.7
w/ CoT	58.8	57.7	—	53.1

Table 6: The overall accuracy of LLMs, i.e., ChatGPT (GPT-3.5-turbo) and LLaMA, and existing state-of-the-art methods (Jiao et al., 2022) on logical reasoning benchmarks. The evaluation of LLMs follows zero-shot in-context learning setting, where the models are expected to decode the answer based on the given instruction, context, and question.

A Implementation Details

A.1 LLM Prompting

In order to evaluate the generalization capabilities of LLMs across different tasks after post-training, we adopt a prompting-based approach. Here, the input to the LLMs is structured as `Instruction [Exemplars] Task input`. The instruction is tailored to the specific task at hand, while exemplars are utilized only in a few-shot setting. Each exemplar comprises both the task input and its corresponding output. For tasks such as multiple-choice question answering, the task input is a concatenation of the context, the question, and all potential options. The correct option index is used as the output. Besides, in a Chain-of-Thought (CoT) setting, we include a reasoning process formulated in natural language between the task input and output.

A.2 Data

We have constructed our self-supervised logic-enhanced training data from Wikipedia, where we directly used the paragraph corpus pre-processed by Qin et al. (2021). We have constructed around 200K logically consistent sample pairs. After that, we further performed counterfactual data augmentation with the ratio of 1:3, and finally induced 800K training sample pairs in total. The data construction process mainly follows the original setting of Jiao et al. (2022) except two differences. First, we remove the usage of knowledge graph for relation annotation to enable fully self-supervision and simplify the construction workflow. Secondly, we have dropped the negative candidates since we employed auto-regressive training.

	High	Middle	Weighted
LLaMA-7B	46.9	61.1	51.0
LLaMA-7B (Ours)	—	—	32.3
LLaMA-13B	47.2	61.6	51.4
LLaMA-13B (Ours)	—	—	**54.5**
LLaMA-33B	48.3	64.1	52.9
LLaMA-33B (Ours)	—	—	**68.1**

Table 7: The comparison on RACE dataset between our reproduced results and those reported by the opriginal paper of LLaMA.

For language modeling, we employed different dataset with respect to the data used in their last stage training. For FLAN-T5 series models, we used the subset of FLAN-collection-v2 (Longpre et al., 2023); while for LLaMA series models, we used the same Wikipedia paragraphs from the corpus of Qin et al. (2021).

A.3 Hyper-parameters of Training

During the pre-training process, we set the batch size to 4,096, which is implemented using gradient accumulation. The maximum sequence length is truncated at 1,024 for the FLAN collection and 512 for the MERIt corpus. For the FLAN-T5 series models, we conduct training steps for 200 iterations, while for the LLaMA series models, we perform training steps for 500 iterations. The learning rates are set as follows: 1e-4 for FLAN-T5-3B, 5e-5 for FLAN-T5-11B, 1e-5 for LLaMA-7B, and 5e-6 for LLaMA-13B. To carry out the training process, we utilize 8 NVIDIA A100 80G GPUs. However, due to hardware limitations, models larger than 13B are trained using QLoRA (Dettmers et al., 2023), a low-rank adaptation approach specifically designed for quantized LLMs. We follow the setting used in QLoRA with α as 16 and r as 64. All linear layers are used for adaptation and the LoRA dropout is 0.05. The learning rate for LLaMA-33B and Falcon-40B is set as 5e-4.

A.4 Evaluation

To ensure a fair comparison, we maintain consistency across different models for each dataset. This involves using identical instructions and few-shot samples. We use accuracy as the evaluation metric across all experiments. The prompts for different dataset can be found in Appendix D.

Model / Dataset	Zero-shot	Direct	CoT
LLaMA-7B	24.9	30.4	**27.0**
w/ LogicLLM	**25.2**	**30.8**	25.9
LLaMA-13B	25.0	34.7	32.3
w/ LogicLLM	**26.3**	**35.0**	**33.9**
FLAN-T5-3B	38.0	40.2	35.1
w/ LogicLLM & FLAN	**40.5**	**41.2**	**36.7**
FLAN-T5-11B	43.0	**42.6**	**40.9**
w/ LogicLLM & FLAN	**44.1**	36.2	40.2

Table 8: The accuracy of LLaMA and FLAN-T5 based models on BIG-Bench-Hard. *Direct* refer to few-shot setting through direct prompting, where only the final answer is given. Instead, in *CoT* setting, the reasoning process is also concatenated. The exemplars used for direct few-shot prompting and CoT prompting are consistent in each task, which are officially provided.

B Interpretation for Different Results on RACE

In this section, we will discuss the different results on RACE between ours and those reported by the original paper of LLaMA. Specifically, Touvron et al. (2023) do not report the weighted results, so we convert them by ourselves. The results are shown in Table 7. From the table we can find that only LLaMA-7B cannot match the performance reported by the authors. On LLaMA-13B and LLaMA-33B, our reproduced accuracies are much higher than the reported ones, which can help address the concern of unfair comparison, and demonstrate the effectiveness of our proposed LogicLLM.

C Logic-enhanced Meta-training for Complex Task Understanding

We evaluated the performance of logic-enhanced pre-trained models on BIG-Bench-Hard, a benchmark comprising challenging tasks where human performance surpasses that of LLMs. Table 8 presents the results achieved by the LLaMA and FLAN-T5 models under three evaluation settings: zero-shot, direct few-shot, and CoT.

In the zero-shot setting, our logic-enhanced meta-training significantly improves all four investigated models. For instance, the zero-shot accuracies of LLaMA-13B and FLAN-T5-T5-11B are 25.0% and 38.0%, respectively. When combined with the LogicLLM model, the accuracy scores of LLaMA-13B and FLAN-T5-11B improve to 26.3% and 44.1%, respectively. Some tasks included in BBH require free-form answers thus we cannot evaluate the models by selecting the candidate with lowest perplexity or log likelihood. Instead, we need to follow the evaluation of API-based models, which employs regularization expression to capture the answer from the response. However, smaller language models, especially those without being instruction tuned, fail to accept diverse instruction, and generate structured response. As a result, the absolute performance under zero-setting setting of LLaMA-based models are relatively limited.

On the other hand, the direct few-shot results outperform the zero-shot results in three out of four models, with the exception of FLAN-T5-11B. Similarly, logic-enhanced meta-training boosts the performance of models, except for FLAN-T5-11B. In the CoT setting, our method further enhances the performances of LLaMA-13B and FLAN-T5-3B. However, the best direct few-shot and CoT results (42.6% and 40.9%, respectively) are both inferior to the best zero-shot result (44.1%). Notably, the CoT results on FLAN-T5-3B are significantly worse than the zero-shot and direct few-shot results. These observations suggest the potential drawback that learning CoT from annotated training data, i.e., FLAN collection, has difficulty in generalizing to different task categories, for example, learning CoT from math word problem solving and solving logical puzzles. We provide further discussion on these findings in Appendix G.

D Prompt Template

D.1 ReClor

Answer the following question with the given context through logical reasoning:

Context: #Context

Question: #Question

Options:
A: #Option A.
B: #Option B.
C: #Option C.
D: #Option D.

The answer is

D.2 LogiQA-v2 & RACE

Answer the following question with the given context:

Context: #Context

Question: #Question

Options:
A: #Option A.
B: #Option B.
C: #Option C.
D: #Option D.

The answer is

D.3 MMLU

The following are multiple choice questions (with answers) about #Subject.

#Question

A: #Option A.
B: #Option B.
C: #Option C.
D: #Option D.

Answer:

E Auto-Verification Cases for Logical Consistency

E.1 Prompt Template

[User]:

Determine whether the relation between "[Entity A]" and "[Entity B]" in the given two sentences are logically consistent.

Directly give the answer from either Yes or No.

Sentence 1:

[Sentence(s) 1]

Sentence 2:

[Sentence(s) 2]

[ChatGPT/GPT-4]:

Yes/No.

E.2 Normal Version

[User]:

Determine whether the relation between "**Everdingen**" and "**Sweden**" in the given two sentences are logically consistent.

Sentence 1:

In the manner of Frans Post, **Everdingen** took advantage of this mishap by making sketches of the Norwegian landscape, which would have seemed very exotic to his Dutch countrymen. His annotated drawings document visits to the south - east Norwegian coast and to Bohusland and the Göteborg area in western **Sweden**.

Sentence 2:

In 1644 **Everdingen** travelled to Norway and **Sweden**, a trip that was to have profound consequences on his art.

The output should either be Yes or No.

[ChatGPT]:

Yes.

E.3 Counterfactual Version

[User]:

Determine whether the relation between "**Nicholas Roerich**" and "**Master**" in the given two sentences are logically consistent.

Sentence 1:

In the manner of Frans Post, **Nicholas Roerich** took advantage of this mishap by making sketches of the Canal del Dique landscape, which would have seemed very exotic to his Dutch countrymen. His annotated drawings document visits to the south - east Canal del Dique coast and to Bohusland and the Göteborg area in western **Master**.

Sentence 2:

In 1644 Nicholas Roerich travelled to Norway and Master , a trip that was to have profound consequences on his art .

The output should either be Yes or No.

[ChatGPT]:

No.

Entity replacement:

- Everdingen → Nicholas Roerich;
- Sweden → Master;
- Norwegian (connecting entity) → Canal del Dique;

E.4 Anonymized Version

[User]:

Determine whether the relation between "**[X]**" and "**[Y]**" in the given two sentences are logically consistent.

Sentence 1:

In the manner of Frans Post, **[X]** took advantage of this mishap by making sketches of the Canal del Dique landscape , which would have seemed very exotic to his Dutch countrymen. His annotated drawings document visits to the south - east Canal del Dique coast and to Bohusland and the Göteborg area in western **[Y]**.

Sentence 2:

In 1644 **[X]** travelled to Norway and **[Y]**, a trip that was to have profound consequences on his art .

The output should either be Yes or No.

[ChatGPT]:

Yes.

Model / Dataset	**ReClor** Dev Acc.	 Test Acc.	**LogiQA-v2** Dev Acc.	 Test Acc.
zero-shot				
ChatGPT	56.6	61.2	54.5	52.7
w/ CoT	58.8	57.7	54.5	53.1
5-shot				
ChatGPT	61.0	**63.0**	55.1	54.5
w/ CoT	**62.0**	62.5	47.6	**55.6**
w/ CoT + Cate.	N/A	N/A	**55.8**	55.0

Table 9: The results on logical reasoning benchmarks with enhanced Chain-of-Thought prompting.

F Discussion about Different Perspectives of Logical Reasoning

In our opinion, logic can be reflected through multiple aspects. Here, we use a simple logic rule to discuss the different perspectives:

$$(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \gamma) \leftrightarrow \alpha \rightarrow \gamma. \tag{3}$$

The above equation shows the simplest case of first-order logic reasoning, where α, β and γ are different variables, and $\wedge$ is *logical and*. We can also introduce the necessary logical connectives in natural language to make it easier for understanding:

$$\text{IF } \alpha \rightarrow \beta \text{ AND } \beta \rightarrow \gamma, \text{ THEN } \alpha \rightarrow \gamma. \tag{4}$$

It should be noted that, in symbolic logic, we often ignore the actual meaning of relations. However, we can always find a path, i.e., a series of relation triplets from knowledge graph to transform the above symbolic form into natural language based logical reasoning process:

$$\text{IF } \alpha \xrightarrow{r_1} \beta \text{ AND } \beta \xrightarrow{r_2} \gamma, \text{ THEN } \alpha \xrightarrow{r_3} \gamma. \tag{5}$$

One example here can be: r_1 refers to *is the father of*, r_2 refers to *is the mother of*, and r_3 refers to *is the grandpa of*.

From the above discussion, we can conclude that (1) logical connectives focus on discourse-level connections, (2) symbolic logic can be viewed as the simplified version of logical reasoning in natural language, where we focus more on the formal rules of atomic logic operations, and (3) relational reasoning concentrates on the actual logic operations built on world knowledge. Both of what we have discussed in the paper and the reviewers have mentioned in comments, i.e., logical connectives, are indeed different perspectives of logical reasoning. They do not contradict to each other, and discussing them separately is beneficial to make the problem easier. Besides, there are also several studies also discuss logical reasoning from the relational reasoning perspective (Wong et al., 2023; Xu et al., 2021; Zeng et al., 2021; Wang et al., 2022). And Figure 1 also shows the case emphasizing relational reasoning.

G Weakness of LLMs on Logical Reasoning

Table 9 showcases the evaluation results of LLMs' performance in both few-shot and CoT settings. The intermediate reasoning process is automatically generated by ChatGPT using the prompt *"Let's think step by step."* In the case of zero-shot CoT, we include the suffix prompt *"So the answer is"* to guide the models in summarizing and concluding the answer. For few-shot CoT, the reasoning process is initially generated for each sample in the training set. Subsequently, we retain the samples where the final prediction is correct, following the steps outlined in zero-shot CoT. During testing, we randomly select samples from the retained candidates, as well as the automatically generated CoT, to serve as exemplars.

However, our observations indicate that both few-shot learning and the use of CoT do not significantly improve the models' performance. For example, ChatGPT w/ CoT performs much worse than that without CoT on the development set of LogiQA-v2. One potential reason for this is that the selected samples differ substantially from the target example. To investigate further, we incorporate reasoning category information during exemplar selection. In LogiQA-V2, each question is annotated with a reasoning category, such as categorical reasoning, sufficient conditional reasoning, or necessary conditional reasoning. For few-shot CoT prompting, we only consider candidates that share at least two common reasoning categories. This particular variant is denoted as *"ChatGPT w/ CoT + Cate."* in the table.

Despite these efforts, we find that carefully selecting prompting exemplars only provides limited improvement. The results indicate that LLMs struggle to comprehend the reasoning structure from a limited number of observed examples. Consequently, they face challenges in effectively learning the mapping between input-label and input-rationale-label. Additionally, as shown in Table 1, we observe that LogicLLM also contributes minimally to addressing this issue. We recognize the need for further investigation in this area and leave it as a potential avenue for future research.