Investigating Logical Reasoning Capabilities of LLMs

Arnav Agarwal Aayush Ranjan Harshvardhan Singh
Parthiv Dholaria Utsav Garg Pulkit Nargotra

September 2024

Abstract

This project aims to investigate and evaluate the logical reasoning capabilities of Large Language Models (LLMs). Logical reasoning, being a critical component of human cognition, is essential for tasks involving problem-solving, decision-making, and inferring conclusions. With the recent advancements in LLMs, there is a growing need to assess how these models handle logic-based tasks and whether their responses align with human-like reasoning. Through this study, we will design and implement a set of logical reasoning problems, encompassing tasks like natural language inference, deductive reasoning, and consistency checking. Using various LLM models such as LLAMA, CLAUDE, GEMINI, etc. we will evaluate their performance and analyze their strengths and limitations in reasoning. The findings are expected to contribute to a deeper understanding of the cognitive capabilities of LLMs and offer insights into improving model architectures for enhanced logical reasoning.

Objectives

1. To Assess Logical Reasoning Abilities of LLMs:

- Explore a comprehensive set of logical reasoning tasks that challenge different aspects of human-like reasoning, including natural language inference, deductive reasoning, and consistency checking.
- Evaluate the performance of various LLMs (e.g., LLAMA, CLAUDE, GEMINI) on these tasks to determine their capabilities and limitations in logical reasoning.

2. To Analyze LLMs' Reasoning Patterns and Limitations:

- Investigate the types of logical errors made by LLMs and the patterns underlying these errors.

- Identify which reasoning tasks are most challenging for the models and analyze the factors contributing to their difficulties, such as lack of contextual understanding or failure in inferential reasoning.

3. To Develop Enhanced Evaluation Frameworks:

- Try to propose new benchmarks and evaluation frameworks that better capture the reasoning abilities of LLMs, moving beyond simple task accuracy to assess models' deeper cognitive capabilities.
- Provide recommendations for refining model architectures and training methods to enhance logical reasoning performance.

Possible Applications

1. **Improvement in AI Decision-Making Systems:** Insights from this study could guide the development of LLMs with enhanced reasoning skills, making them more reliable in applications like legal reasoning, medical diagnosis, and strategic planning.
2. **Advancements in Human-AI Collaboration:** By enhancing LLMs' understanding of logical reasoning, these models could become more effective tools for collaboration, assisting humans in complex problem-solving and decision-making tasks.
3. **Development of More Reliable Conversational Agents:** Findings could contribute to creating chatbots and virtual assistants that provide more accurate and contextually relevant responses, reducing instances of overconfident or misleading outputs.
4. **Contributions to AI Safety and Ethics:** Improved logical reasoning in LLMs could reduce the risks associated with AI-generated misinformation, making AI systems more transparent and trustworthy.

Related Work

Easy Problems That LLMs Get Wrong

The paper "Easy Problems That LLMs Get Wrong" by Sean Williams and James Huckle [1] introduces a linguistic benchmark to evaluate the limitations of Large Language Models (LLMs) in areas such as logical reasoning, spatial intelligence, and linguistic abilities. The authors argue that despite advancements, LLMs struggle with seemingly simple problems requiring genuine reasoning, rather than relying on surface-level patterns in training data.

The paper critically reviews existing literature, emphasizing that LLMs often fail at tasks demanding a combination of skills like context understanding, inference-making, and logical reasoning. A key finding is that LLMs tend to generate overconfident and incorrect responses by confabulating answers based on statistical patterns in their training data, lacking awareness of their own knowledge limitations.

The authors advocate for more comprehensive evaluation frameworks that go beyond traditional benchmarks to truly assess the reasoning abilities of LLMs. By introducing diverse problems that challenge various linguistic and reasoning skills, the paper aims to provide a more accurate understanding of LLMs' current limitations and guide future research toward developing more reliable AI systems.

LogicBench

The paper titled "LogicBench: Towards Systematic Evaluation of Logical Reasoning Ability of Large Language Models" [2] by Mihir Parmar et al. presents a new dataset called LogicBench, designed to evaluate the logical reasoning capabilities of Large Language Models (LLMs). The authors highlight a gap in existing research on logical reasoning in LLMs, which has been underexplored compared to other reasoning types like commonsense or numerical reasoning. LogicBench is a carefully crafted dataset that spans 25 distinct reasoning patterns across propositional, first-order, and non-monotonic logics, making it one of the most comprehensive datasets for this purpose.

The paper evaluates popular LLMs such as GPT-4, ChatGPT, and Llama-2 on two tasks: Binary Question Answering (BQA) and Multiple-Choice Question Answering (MCQA). The results reveal significant weaknesses in current LLMs, particularly in handling complex reasoning patterns and negations. The models, despite their size and capabilities, struggle with logical consistency and often overlook contextual information necessary for arriving at correct conclusions.

The authors also explore how fine-tuning models on LogicBench improves their performance, particularly in reasoning-based tasks. This study provides a critical benchmark for future research and proposes methods to enhance the logical reasoning abilities of LLMs, suggesting potential improvements in model architectures and training methods. The work is positioned as a key step toward enhancing AI's reasoning abilities in a more systematic and structured way.

Self-Correction of Reasoning

[3] investigates the self-correction abilities of large language models (LLMs) like GPT-3.5 and GPT-4, particularly in the domain of reasoning. While LLMs have shown exceptional performance across various tasks, their reasoning capabilities remain a concern, particularly when it comes to improving or correcting their initial responses without external feedback.

The study introduces the concept of intrinsic self-correction, where the LLM is tasked with correcting its reasoning without any external input, such as la-

bels or human guidance. The authors critically analyze whether LLMs can improve their reasoning-based answers through this self-correction process and explore multiple datasets such as GSM8K, CommonSenseQA, and HotpotQA for evaluation.

Their findings suggest that LLMs often fail at self-correction, and in some cases, performance even degrades after multiple rounds of self-correction. The paper also reveals that improvements observed in previous research were often the result of using oracle labels to guide the process, which is unrealistic in real-world applications.

The authors propose that the limitations arise from the LLMs' inability to judge the correctness of their reasoning. They also identify issues with prompt design, which can bias the results, and recommend focusing on more informative prompts for the initial task rather than relying on feedback during self-correction. The paper concludes by emphasizing the need for external feedback mechanisms to enhance the reasoning capabilities of LLMs.

LogicLM

The paper "Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning" [4] presents a framework that integrates Large Language Models (LLMs) with symbolic solvers to enhance logical reasoning abilities. The framework operates in two stages: translating natural language problems into symbolic formulations and performing inference via symbolic solvers. A self-refinement module is introduced to iteratively improve the symbolic formalizations based on error messages from the symbolic solver.

Logic-LM has been evaluated on five logical reasoning datasets, demonstrating significant performance improvements over standard prompting and chain-of-thought prompting methods. The model exhibits increasing effectiveness as the complexity of reasoning tasks increases, outperforming traditional methods by an average of 39.2% and 18.4%, respectively. However, the authors acknowledge limitations in capturing semantic intricacies, particularly when compared to smaller LLMs with different semantic comprehension capabilities.

The Logic-LM framework represents a promising advancement in logical reasoning within NLP, effectively combining LLMs with symbolic solvers to address the limitations of LLMs in handling complex logical tasks. The authors emphasize the need for further research to enhance semantic understanding and fully leverage the potential of this integrated approach in various reasoning contexts.

LoGiPT

The paper "Language Models can be Logical Solvers" [5] introduces LoGiPT, a novel language model designed to improve the logical reasoning abilities of LLMs by directly emulating the processes of traditional logical solvers. Jiazhan Feng and colleagues highlight the limitations of existing methods, which rely on converting natural language into symbolic representations before using external solvers, often leading to parsing errors and inefficient reasoning. LoGiPT

avoids these issues by internalizing the syntax and grammar of deductive solvers, allowing it to reason without external systems.

A key innovation is the fine-tuning of LoGiPT on a custom instruction-tuning dataset specifically crafted to refine the model's understanding of deductive logic. The model's performance was tested on two public deductive reasoning benchmarks, where it outperformed state-of-the-art models, including GPT-4 and other solver-augmented LLMs.

The paper's findings show that LoGiPT represents a significant advancement in integrating logical reasoning within language models, making it a more reliable tool for complex reasoning tasks. The authors propose that LoGiPT's architecture could lead to further breakthroughs in areas requiring logical problem-solving and decision-making systems, marking a promising direction for future research in LLMs.

Reasoning via Planning: World Model

The paper titled "Reasoning with Language Models is Planning with World Models" [6] introduces a novel framework, Reasoning via Planning (RAP), which augments Large Language Models (LLMs) with world models to improve their reasoning capabilities. Traditional LLMs, such as GPT-3 and GPT-4, excel in tasks involving basic reasoning or language comprehension but often fail in tasks requiring complex, multi-step reasoning or action planning. The paper addresses this gap by incorporating Monte Carlo Tree Search (MCTS) into the reasoning process, allowing LLMs to simulate future states and explore multiple reasoning paths.

The RAP framework treats the LLM as both a reasoning agent and a world model, enabling it to perform structured planning for tasks such as plan generation, mathematical reasoning, and logical inference. In this setup, the LLM builds a reasoning tree, guided by predicted outcomes and rewards from the world model, and refines its reasoning steps iteratively. This approach provides a balance between exploration (searching for new reasoning paths) and exploitation (optimizing the current best paths), improving the model's decision-making ability in complex scenarios.

The authors demonstrate RAP's effectiveness by applying it to three challenging reasoning tasks: plan generation in Blocksworld, mathematical reasoning on the GSM8K dataset, and logical inference in PrOntoQA. RAP consistently outperforms Chain-of-Thought (CoT) reasoning and even achieves a 33% improvement over GPT-4 in some cases. The results show that RAP holds significant potential for improving the logical reasoning abilities of LLMs across a variety of domains.

Approach

Methodologies and Techniques:

1. **Problem Set Exploration:** Choose a diverse set of tasks (e.g., natural language inference, deductive reasoning, consistency checking) across multiple domains to test LLMs' reasoning skills.

2. **Model Selection and Fine Tuning:** Evaluate a range of LLMs (e.g., LLAMA, CLAUDE, GEMINI) and compare against baselines; fine-tune models on custom datasets for logical reasoning.

3. **Evaluation Metrics:** Use accuracy, task-specific metrics, error analysis, and confidence scoring to assess performance; employ interpretability tools for deeper insights.

4. **Cross-Model Analysis:** Compare model performance across tasks, analyze feature attributions, and study the impact of model parameters on reasoning capabilities.

5. **Iterative Testing and Refinement:** Update benchmarks with new tasks, and involve human evaluators to provide qualitative feedback.

6. **Propose Model Enhancements:** Suggest architectural improvements and training methods (e.g., hybrid models, curriculum learning) to enhance logical reasoning abilities.

7. **Result Interpretation and Reporting:** Report findings on performance, error types, and propose future research directions to improve model reasoning.

Rough Timeline

- **September 5 - September 13: Literature Review and Project Proposal**
  - Perform a exhaustive review of all prior literature and prepare a project draft with rough appoach and timeline.
- **September 13 - September 20: Problem Set Design**
  - Develop a diverse set of logical reasoning tasks and ensure coverage across various domains.
- **September 20 - September 29: Baseline Model Selection and Preparation**
  - Select and prepare LLMs (e.g., LLAMA, CLAUDE, GEMINI); set up baseline models and fine-tune on custom datasets.

- **September 29 - October 6: Initial Evaluation and Testing**
  - Evaluate models on the designed tasks using initial metrics; conduct preliminary error analysis.
- **October 6 - October 15: Cross-Model Analysis**
  - Perform comparative analysis across different models, focusing on performance patterns and parameter impact.
- **October 15 - October 23: Iterative Refinement**
  - Update benchmarks with additional complex tasks; incorporate human feedback for qualitative assessment.
- **November 23 - November 3: Propose and Implement Enhancements**
  - Identify model enhancements and implement changes; experiment with proposed improvements (e.g., curriculum learning).
- **November 3 - November 10: Final Evaluation**
  - Conduct a comprehensive evaluation of all models; finalize error analysis and confidence scoring.
- **November 10 - November 14: Result Interpretation and Reporting**
  - Prepare detailed reports on findings, conclusions, and recommendations for future research; finalize documentation and presentation.

Evaluation Metrics

Traditional Metrics

1. Accuracy

Accuracy measures the proportion of correct predictions over the total number of questions or tasks. In logical reasoning, accuracy reflects how often the LLM provides the correct answer based on logical principles and rules. While accuracy is a good starting point for evaluating logical reasoning, it does not account for partial correctness or distinguish between easy and difficult reasoning tasks, limiting its effectiveness in more nuanced logical evaluation. Despite shortcomings all LogicLM,LoGiPt and LogicBench have used accuracy to evaluate their performance.

2. Exact Match Score

Exact Match (EM) determines whether the model's predicted answer exactly matches the reference answer. Since logical reasoning is often modelled as Q/A task, it is only natural to use EM score as a metric. In logical reasoning tasks, this metric is especially strict and suitable for situations where precision is paramount, such as binary logic questions or structured reasoning tasks. However, it may be too rigid in cases where logical correctness can be expressed in various ways, making it less flexible for tasks involving natural language responses or diverse expressions of logical reasoning.

3. Semantic Answer Similarity

Semantic answer similarity evaluates the meaning of the model's answer compared to the reference answer, even if the words or phrasing differ. This metric is particularly useful for logical reasoning tasks where multiple phrasings of the same logical conclusion can still be correct. Using techniques like word embeddings or cosine similarity, it ensures that logically equivalent but differently worded answers are recognized as correct, making it more forgiving than exact match metrics.

Specially Designed Metrics

We also discuss certain metrics presented in research papers which are designed with the motive of evalutaing logical reasoning capabilities in LLMs. These metrics largely test the correctness and informativeness of the reasoning generated by the models and whether reasoning is consistent and contributes to the final generated answer.

ROSCOE: A Suite of Metrics for Evaluating Logical Reasoning in LLMs

ROSCOE [7] provides a fine-grained evaluation framework for step-by-step reasoning produced by Large Language Models (LLMs). It goes beyond traditional metrics by focusing on evaluating the quality of reasoning at a granular level, which is essential when analyzing the logical reasoning capabilities of LLMs. ROSCOE operates across four key metrics:

Semantic Alignment: This metric evaluates whether the reasoning steps align with the source context. For logical reasoning, it checks if each reasoning step is grounded in the provided information, ensuring that models do not stray from the original data while making inferences.

Logical Inference: ROSCOE's logical inference metric identifies contradictions or inconsistencies within reasoning steps. It assesses whether the conclusions drawn by LLMs follow logically from the premises. This is crucial for logical reasoning tasks where the internal consistency of the argumentation matters.

Semantic Similarity: This metric focuses on the degree of similarity between reasoning steps or between the generated reasoning and reference answers. For logical reasoning, it helps detect redundancy, hallucinations, or unnecessary steps, ensuring the logical flow remains intact and efficient.

Language Coherence: This measures the fluency and coherence of the generated text, ensuring that the reasoning is not only logically sound but also articulated clearly and cohesively.

By integrating these metrics, ROSCOE allows for a more detailed and interpretable assessment of LLMs' reasoning capabilities, helping to identify specific weaknesses such as missing steps, contradictions, or irrelevant information in complex reasoning task

RECEVAL: Evaluation of Reasoning Chains

RECEVAL [8] (Reasoning Chain Evaluation) is a framework proposed for evaluating the logical reasoning abilities of language models by focusing on two key properties: correctness and informativeness.

Correctness: Each step in a reasoning chain must derive a valid conclusion from the information available within that step and any prior steps or input context. RECEVAL evaluates both intra-step and inter-step correctness, ensuring logical consistency both within individual steps and across the entire reasoning chain.

Informativeness: This property assesses how much new and useful information each step in the chain contributes toward deriving the final answer. RECEVAL uses information-theoretic techniques (like V-Information) to measure the gain in information at each step, ensuring that every part of the reasoning process moves closer to the correct conclusion.

By applying these metrics, RECEVAL can effectively identify reasoning errors, such as hallucinations, redundancy, or logical inconsistencies, making it a comprehensive tool for evaluating the quality of multi-step reasoning generated by LLMs. This method provides a detailed, step-by-step breakdown of reasoning, which goes beyond just evaluating final answers, offering a deeper insight into the reasoning processes of models.

Human Evaluation

Human evaluation involves annotators assessing the outputs of LLMs for logical correctness, consistency, and reasoning quality on the basis of pre-decided guidelines. An inter-annotator agreement score is calculated to ensure all evaluators are on the same page with respect to the guidelines. It is particularly effective for evaluating complex logical reasoning tasks where the context or subtle nuances may be beyond the capacity of automated metrics. In logical reasoning evaluation, human feedback is invaluable for identifying issues like logical fallacies, incorrect inferences, or reasoning gaps, helping to pinpoint deeper limitations in the model's capabilities.

While accuracy would remain a relevant metric to compare performance across baselines and different models, we could leverage a host of specially designed metrics like ROSCOE and RECEVAL to supplement the results and explain them. Human Evaluation could also be useful to detect any anomalies and findings these metrics are unable to capture.

Team Details

1. Arnav Agarwal
2. Aayush Ranjan
3. Harshvardhan Singh
4. Pulkit Nargotra
5. Parthiv Dholaria
6. Utsav Garg

References

[1] S. Williams and J. Huckle, "Easy problems that llms get wrong," *arXiv preprint arXiv:2405.19616v2*, 2024.

[2] M. Parmar, N. Patel, N. Varshney, M. Nakamura, M. Luo, S. Mashetty, A. Mitra, and C. Baral, "Logicbench: Towards systematic evaluation of logical reasoning ability of large language models," *arXiv preprint arXiv:2404.15522v2*, 2024.

[3] J. Huang, X. Chen, S. Mishra, H. S. Zheng, A. W. Yu, X. Song, and D. Zhou, "Large language models cannot self-correct reasoning yet," *arXiv preprint arXiv:2212.07919v2*, 2023.

[4] X. W. W. Y. W. Liangming Pan, Alon Albalak, "Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning," *arXiv preprint arXiv:2305.12295v2*, 2023.

[5] J. Feng, R. Xu, J. Hao, H. Sharma, Y. Shen, D. Zhao, and W. Chen, "Language models can be logical solvers," *arXiv preprint arXiv:2311.06158v1*, 2023.

[6] S. Hao, Y. Gu, H. Ma, J. J. Hong, Z. Wang, D. Z. Wang, and Z. Hu, "Reasoning with language model is planning with world model," *arXiv preprint arXiv:2305.14992v2*, 2023.

[7] O. Golovneva, M. Chen, S. Poff, M. Corredor, L. Zettlemoyer, M. Fazel-Zarandi, and A. Celikyilmaz, "Roscoe: A suite of metrics for scoring step-by-step reasoning," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[8] A. Prasad, S. Saha, X. Zhou, and M. Bansal, "Receval: Evaluating reasoning chains via correctness and informativeness," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 10066–10086, 2023.