RiddleBench: A New Generative Reasoning Benchmark for LLMs

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have demonstrated impressive performance on many established reasoning benchmarks. However, these benchmarks primarily evaluate structured skills like quantitative problem-solving, leaving a critical gap in assessing the more flexible, multifaceted reasoning abilities that are a cornerstone of human intelligence. These skills require synthesizing logical deduction with spatial awareness and constraint satisfaction, which current evaluations do not adequately measure. To address this gap, we introduce **RiddleBench**, a new benchmark of 1,737 challenging puzzles in English designed specifically to probe these core reasoning capabilities. Our evaluation of state-of-the-art models on RiddleBench reveals fundamental weaknesses. We find that models suffer from "hallucination cascades" (uncritically accepting flawed peer reasoning) and exhibit poor self-correction due to a strong self-confirmation bias. Furthermore, their reasoning proves fragile, with performance degrading significantly when faced with reordered constraints or irrelevant information. RiddleBench serves as a diagnostic tool for these critical issues and provides a valuable resource for guiding the development of more robust and reliable LLMs.

1 Introduction

The rapid advancement of Large Language Models (LLMs) has led to unprecedented performance on many NLP benchmarks (Devlin et al., 2019; Brown et al., 2020). While excelling at tasks like text generation and translation, true intelligence requires the ability to reason, deduce, and infer new knowledge (Chollet, 2019). However, many existing benchmarks primarily measure pattern recognition or memorization rather than deep reasoning, making it difficult to fully assess a model's inferential capabilities (Saxton and et al., 2019; Cobbe et al., 2021).

> A family has 6 members – Radhey, Krishna, Madhav, Kanha, Gaur and Hari among 3 generations. Further it is also known that: Radhey is the son-in-law of Krishna. Gaur who is unmarried, has a sister and Hari has an uncle. Kanha is the grandmother of Hari. Both the grandparents and parents of Hari are alive. Which of the following does not belong to the second generation of the family?
> A. Madhav
> B. Krishna
> C. Gaur
> D. Radhey
> E. Either A or C

Figure 1: An example from the RiddleBench benchmark for Blood Relations.

To address this, we propose **RiddleBench**, a collection of 1,737 challenging puzzles from competitive exams. These puzzles demand multi-step deduction, spatial reasoning, and constraint satisfaction, forcing models to reason from first principles.

Our contributions are threefold:

1. We introduce RiddleBench, a benchmark of 1,737 puzzles for evaluating LLM reasoning in English.
2. We comprehensively evaluate leading LLMs, highlighting strengths and weaknesses.
3. We structure our analysis around key research questions, revealing phenomena like hallucination cascades and poor self-correction.

RiddleBench will be made publicly available to support research on robust reasoning systems. Experiments show that even the most powerful models struggle with a significant portion of RiddleBench, highlighting advanced reasoning as a key frontier for LLM development.

Table 1: RiddleBench Dataset Composition and Core Skills Probed.

Category (% of Total)	Description	Primary Skills Tested
Sequential Reasoning (60%)	Establishing linear order from rules	Constraint Satisfaction Logical Deduction
Seating Arrangements (25%)	Deducing positions in spatial layouts	Spatial Awareness Constraint Satisfaction
Blood Relations (8%)	Inferring kinship from relationships	Logical Deduction
Coding-Decoding (7%)	Applying rules to decipher patterns	Logical Deduction Pattern Recognition

2 Related Work

Evaluating the reasoning capabilities of large language models is a pivotal research area, often addressed with benchmarks targeting specific cognitive skills. For instance, on datasets focused on **mathematical reasoning** (e.g., GSM8K; (Cobbe et al., 2021)), **commonsense reasoning** (e.g., CommonsenseQA; (Talmor et al., 2019)), and formal **logical deduction** (e.g., LogiQA; (Liu et al., 2020)), top models now achieve high accuracy, demonstrating strong performance in structured problem-solving.

A key limitation of these evaluations, however, is their focus on problems with clear arithmetic or inferential paths. RiddleBench complements these efforts by introducing less-structured puzzles that demand more flexible reasoning strategies, requiring models to synthesize logical deduction with spatial awareness and constraint satisfaction. Our work also extends beyond standard accuracy to investigate the reliability of the reasoning process itself, exploring error-correction paradigms as in (Zheng et al., 2023).

3 RiddleBench

We describe the approach for creation of RiddleBench, a benchmark for assessing the complex reasoning capabilities of LLMs beyond accuracy. (See Appendix B for examples of each puzzle category.)

3.1 Data Collection and Curation

Problems were sourced from publicly archived mock examination papers for Indian government services. All puzzles are in the English language. The curation process, shown in Figure 2, involved:

1. **Content Extraction:** Using OCR (Gemini 2.5 Flash) to digitize questions, answers, and official reasoning traces from source PDFs.
2. **Data Structuring:** Processing raw OCR output to isolate each problem's core components.
3. **Human Verification:** Each data point was manually evaluated by the authors to ensure transcription accuracy and correctness.

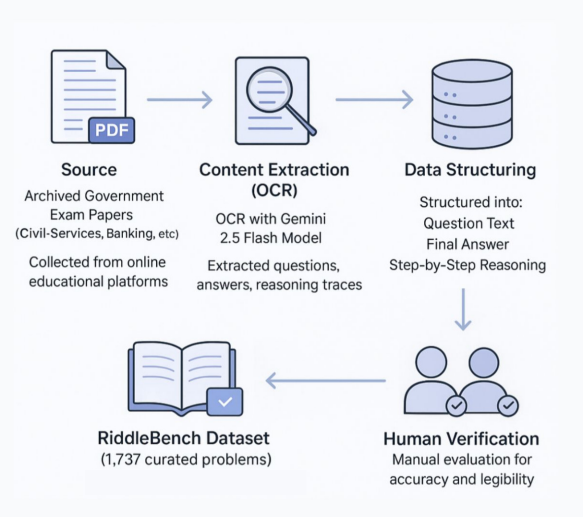

Figure 2: The step-by-step methodology for building RiddleBench. The workflow combines automated extraction with meticulous human evaluation to ensure high-quality data.

This meticulous process ensures the high fidelity of our dataset. RiddleBench is released under the CC0 license[1].

4 Experimental Design

We evaluate a suite of leading proprietary and open-weight models (see Appendix A for a full list). All

[1] https://creativecommons.org/public-domain/cc0/

Table 2: Performance of evaluated LLMs on RiddleBench (1,737 puzzles). Scores are Correct Answers (%). SR: Sequential Reasoning, SA: Seating Arrangement, BR: Blood Relations, CD: Coding-Decoding.

Model	Overall	SR	SA	BR	CD
GPT-oss-120B	69.26	76.43	51.99	71.23	64.23
Claude 4 Sonnet	63.16	69.68	43.52	74.66	63.41
DeepSeek-V3	58.28	61.64	45.02	71.92	57.72
Qwen QwQ 32B	50.86	68.05	17.66	42.47	27.64
DeepSeek-R1	50.56	65.48	12.68	54.79	46.34
Mistral Small 24B it	42.67	44.97	31.84	53.42	46.34
o3	39.22	46.72	13.55	22.13	51.75
Gemini 2.5 Pro	38.52	40.18	16.85	23.58	79.00
Llama 3.3 70B	27.48	25.74	22.39	43.15	39.84
Gemma 3 27B it	25.04	23.18	19.65	33.56	47.97

evaluations use a zero-shot prompting methodology with a temperature of 0.7, and a thinking budget of 8192 tokens. The exact prompt format is detailed in Appendix C.

Our analysis moves beyond final-answer accuracy to assess the reliability and robustness of LLM reasoning. RiddleBench's puzzles, which require satisfying logical and spatial constraints, are uniquely suited for this purpose as their structure produces clear, diagnosable reasoning chains. We therefore investigate the following research questions (RQs):

RQ1: Can LLMs reliably detect and correct reasoning errors made by other models, or do they fall into a "*hallucination cascade*"?

RQ2: How effective are LLMs at self-correction?

RQ3: Is LLM reasoning robust to the order of information and to the presence of irrelevant information?

For the cross-model (RQ1) and self-correction (RQ2) experiments, we primarily use *Qwen QwQ 32B* as the "evaluator" model and *DeepSeek-R1* as the "generator" model, given their strong baseline performance and API accessibility.

5 Results and Analyses

We now describe the performance of various models on RiddleBench, and follow up with analyses aimed at understanding the reasoning capabilities of said models.

5.1 Overall Performance

We first establish baseline performance to contextualize the reasoning capabilities of current models. As shown in Table 2, while *GPT-oss-120B* leads with 69.26% accuracy, this result underscores a critical reality: even state-of-the-art models fail on nearly a third of the benchmark. Performance varies significantly across categories, with models universally struggling on Seating Arrangement puzzles, indicating that complex spatial reasoning remains a formidable challenge. This performance ceiling, even on the best models, motivates a deeper investigation beyond simple accuracy to understand the underlying fragility of the reasoning process itself. For a comparative visualization of each model's reasoning profile and a qualitative analysis of specific problem-solving heuristics, see Appendices D, E, and F.

5.2 The 'Hallucination Cascade': Failures in Cross-Model Correction

Our first research question probes the viability of the "model-as-judge" paradigm, a cornerstone of many modern evaluation pipelines. We tested this by tasking *Qwen QwQ 32B* (the "evaluator") with assessing incorrect outputs from *DeepSeek-R1*. In a simple forced-choice task between a correct and an incorrect answer, the evaluator timed out in 55.0% of cases, revealing that verifying answers without a reasoning trace is often computationally intractable (Table 3).

A more alarming failure emerged when we provided the evaluator with the flawed reasoning trace. The evaluator's accuracy in identifying the flawed logic was only 44.1% - no better than a coin toss. Critically, it incorrectly validated the flawed reasoning as sound in 45.2% of cases. This tendency for one model to uncritically accept and propagate the plausible errors of another constitutes a **hallucination cascade**. To test the persistence of this

effect, we had the model re-evaluate the outputs it had just incorrectly validated. The model reversed its incorrect judgment in a mere 4.4% of cases, demonstrating a powerful error fixation. Once a hallucination cascade begins, our results show that iterative refinement is almost entirely ineffective at stopping it.

Table 3: Summary of Answer and Reasoning Validation Results.

Task / Verdict	Count	Percentage
Forced-Choice Answer Verification		
Success	209	32.4%
Failure	81	12.6%
Thinking Exhausted	355	55.0%
Reasoning Validation on Flawed Reasoning		
Success	286	44.1%
Failure	293	45.2%
Thinking Exhausted	69	10.6%
Iterative Reasoning Validation		
Success	12	4.4%
Failure	121	44.3%
Thinking Exhausted	140	51.3%
Self-Correction Reasoning Validation		
Success	133	17.3%
Failure	520	67.7%
Thinking Exhausted	115	14.9%

5.3 The Illusion of Self-Correction

Given the failure of cross-model correction, we investigated the arguably more critical capability of self-correction (RQ2). We tasked *Qwen QwQ 32B* with judging the soundness of its own flawed reasoning. The results were striking: The model failed to identify its own errors in 67.7% of trials, successfully flagging its flawed logic only 17.3% of the time (Table 3). This success rate is drastically lower than the 44.1% achieved when evaluating a peer's reasoning, suggesting a powerful **self-confirmation bias**. Rather than being their own best critics, models appear to be their own most potent deceivers, reinforcing and trapping themselves in their initial hallucinations.

5.4 Fragile Reasoning: The Lack of Robustness

Our final experiments (RQ3) probed the robustness of the reasoning process by testing its sensitivity to superficial changes in the prompt that should not affect a logical reasoner. Examples illustrating these prompt perturbations are provided in Appendix C. First, we randomly shuffled the order of constraint sentences in puzzles. For a system building a true holistic mental model, order should be irrelevant. However, performance on *Qwen QwQ 32B* dropped significantly - by 6.70 percentage points (p.p.) on Blood Relations and 3.69 p.p. on Seating Arrangements (Table 4). This suggests the model relies on brittle, sequential heuristics rather than robust comprehension.

Second, we tested the model's ability to filter signal from noise by inserting a single, irrelevant "red herring" sentence into the prompt. The model's accuracy proved volatile, with performance dropping on most categories but unexpectedly increasing on Blood Relations. This unpredictability demonstrates that the model lacks a reliable mechanism for ignoring distractors, rendering its reasoning process fragile and contingent on incidental features of the input rather than a deep understanding of the problem's logical structure.

Table 4: Performance of *Qwen QwQ 32B* under shuffled constraints and irrelevant information. Performance change is in percentage points (p.p.).

Puzzle Type	Condition	Old	New	Change
SA	Shuffled	17.66	13.97	-3.69
BR	Shuffled	42.47	35.77	-6.70
SA	Noisy	17.66	14.58	-3.08
BR	Noisy	42.47	45.21	+2.74
CD	Noisy	27.64	23.77	-3.87

6 Conclusion

In this work, we introduce **RiddleBench**, a benchmark designed to test complex, multi-step reasoning in LLMs, revealing that even top models struggle significantly. Our analysis uncovers several critical failures: a **hallucination cascade**, where models uncritically adopt flawed peer reasoning; strong **error fixation** preventing the reversal of incorrect judgments; and poor self-correction due to a powerful **self-confirmation bias**. We also find that LLM reasoning is fragile, easily disrupted by reordered constraints or irrelevant information.

RiddleBench provides the research community with a tool to diagnose these issues and measure progress towards more dependable AI. We plan to expand this effort through further dataset growth and cross-lingual evaluations, as outlined in Appendix G.

7 Limitations

Our evaluation of proprietary models uses August 2025 API versions; performance may change with updates. RiddleBench's curation from Indian exam materials may introduce cultural bias, although the logical tasks are universal. High API costs and computation limited cross-model and self-correction experiments to a subset of models.

A key concern is data contamination. We mitigated this by sourcing puzzles from mock exam PDFs, which are rarely part of standard web scrapes. The multi-step deductive nature resists memorization, but contamination cannot be definitively ruled out for proprietary models with undisclosed training sets.

8 Ethics Statement

Through this work, our aim is to advance the study of reasoning in Large Language Models by creating a challenging and publicly accessible benchmark. By highlighting critical vulnerabilities such as hallucination cascades, poor self-correction, and fragile logic, we hope to guide the development of more robust, reliable, and safer AI systems.

The RiddleBench dataset created in this work is released under the permissible CC0 license to ensure broad and unrestricted access for the research community. Generative AI systems were used only for assistance with language refinement (e.g., paraphrasing, polishing the authors' original content) and for writing boilerplate code.

References

Alibaba Group. 2025. Qwen QwQ: A multilingual large language model. Technical report, Alibaba Group.

Anthropic. 2025. Claude 4 (opus & sonnet).

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. GSM8K: Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek AI. 2025a. DeepSeek-R1: Advancing reasoning through large-scale pre-training. Technical report, DeepSeek AI.

DeepSeek AI. 2025b. DeepSeek-V3: Pushing the boundaries of code and language understanding. Technical report, DeepSeek AI.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Google. 2025. Gemma 3: Advances in open models for responsible ai. Technical report, Google.

Jian Liu, Wanjun Zhong, Duyu Cui, Yelong Li, Nan Chang, Yeyun Chen, Qing-An Chua, and Ming Zhao. 2020. LogiQA: A challenge dataset for logical reasoning. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 4038–4044.

Meta AI. 2024. Llama 3.3: A more capable and aligned language model family. Technical report, Meta AI.

Mistral AI. 2025. Mistral Small: Efficient and powerful language models. Technical report, Mistral AI.

OpenAI. 2025a. GPT-oss-120B: An open-source foundation model for general tasks. Technical report, OpenAI.

OpenAI. 2025b. o3: A new generation of reasoning models. Technical report, OpenAI.

David Saxton and et al. 2019. Analysing mathematical reasoning abilities of neural networks. *OpenReview*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158.

Gemini Team and 1 others. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zou, Zhuohan Liu, Ion Stoica, Eric P Xing, Joseph E Gonzalez, and 1 others. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

A Models Evaluated

We evaluated a comprehensive suite of LLMs, encompassing both leading proprietary systems and prominent open-weight models to benchmark their performance.

Proprietary Models

- **o3** (OpenAI) (OpenAI, 2025b)
- **Gemini 2.5 Pro** (Google) (Team et al., 2023)
- **Claude 4 Sonnet** (Anthropic) (Anthropic, 2025)

Open-Weight Models

- **DeepSeek-R1** (DeepSeek AI) (DeepSeek AI, 2025a)
- **GPT-oss-120B** (OpenAI) (OpenAI, 2025a)
- **Mistral Small 24B it** (Mistral AI) (Mistral AI, 2025)
- **Llama 3.3 70B** (Meta) (Meta AI, 2024)
- **Gemma 3 27B it** (Google) (Google, 2025)
- **Qwen QwQ 32B** (Alibaba) (Alibaba Group, 2025)
- **DeepSeek-V3** (DeepSeek AI) (DeepSeek AI, 2025b)

All experiments were conducted using the latest model versions available as of August 2025.

B Dataset Examples

B.1 Coding-Decoding

```
Question: In a certain code
language FRAME is written
as QEBDL and BLOCK is
written as KAPJB. How is
PRIDE written in that
code language?
a) SQHFE
b) QSHEF
c) QQJCD
d) QOJDC
e) None of these
```

B.2 Sequential Reasoning

```
Question: You are given a
numerical sequence in which
one term is missing, represented
by a '?'. Your task is to
analyze the pattern followed
by the numbers and determine
the missing value.
43 41 44 39 46 ?
```

B.3 Seating Arrangement

```
Question: Nine persons Anu, Bablu,
Cheenu, Dona, Esha, Faria, Gaurav,
Harish and Ishita are sitting
in a row and all are facing
north. It is known that Cheenu
sits exactly in the middle and
there is no person to the right
of Ishita. Dona is fourth to
the right of Faria. Gaurav and
Harish are sitting next to each
other. Esha is the neighbor of
Dona but not of Cheenu. Harish
doesn't sit at any extreme
corner. Dona is not sitting
adjacent to either Cheenu or
Ishita. Anu is second to the
right of Harish. Who is sitting
at the left most seat of the row?
A. Faria
B. Bablu
C. Gaurav
D. Dona
E. None of these
```

B.4 Blood Relations

```
Question: Soni is brother of
Moni, Daya is sister of Moni,
and Bala is father of Charu,
who is brother of Daya. If Moni
is son of Roop then how is Bala
related to Roop?
A. Wife
B. Husband
C. Son
D. Mother in law
E. Brother in law
```

C Prompting Details and Samples

C.1 Zero-Shot Evaluation Prompt

The following prompt was used for the main evaluation on RiddleBench.

```
Question: <question><options if any>.
Please always write the
final answer in \boxed{}.
Answer:
```

C.2 Prompt for Forced-Choice Answer Verification

```
You are checking a mathematical
sum. For a given question, determine
out of answer 1 and answer 2
which is correct. Since the
answers are for the same question,
you can assume similar context
for both answers and make
appropriate assumptions when
checking if they are correct.
Think about both the answers
and find out which one is correct.
Please put the final answer as
"1" or "2" in \boxed{}.

<question>
{question}
</question>

Answer 1: {correct_answer}
Answer 2: {predicted_answer}
```

C.3 Prompt for Reasoning Validation

```
You are a checker for logical
reasoning questions. You will be
given a Question, a Reasoning,
and an Answer. Your task:
(1) Check if the reasoning is
logically correct.
(2) Check if the answer is correct.
At the end, reply in the format:
\boxed{YES/NO, The correct answer
is ...}. Use YES if the reasoning
and answer are correct, otherwise
use NO and provide the correct answer.

Question : {question}
Reasoning : {reasoning}
Answer : {answer}
```

C.4 Sample Problems for Robustness Experiments

The following examples illustrate the perturbations applied.

Constraint Shuffling

```
Original: A is the father of B.
B is the sister of C. C is the
son of D. How is A related to D?

Shuffled: C is the son of D.
A is the father of B. B is the
sister of C. How is A related to D?
```

Irrelevant Information The irrelevant sentence is highlighted in italics for clarity.

```
In a certain code language FRAME
is written as QEBDL and BLOCK
is written as KAPJB. *The programmers
who designed this language often
take breaks to play basketball.*
How is PRIDE written in that code
language?
a) SQHFE
b) QSHEF
c) QQJCD
d) QOJDC
e) None of these
```

D Model Reasoning Profiles

Figure 3 provides a comparative visualization of LLM reasoning capabilities across the four RiddleBench categories: Sequential Reasoning (SR), Seating Arrangement (SA), Blood Relations (BR), and Coding-Decoding (CD). Each axis represents a category, with a model's distance from the center indicating its accuracy.

This visualization highlights the diverse performance profiles of the models. For example, *GPT-oss-120B* exhibits a large, balanced shape, indicating strong, well-rounded performance. In contrast, other models display more skewed profiles that reveal specific strengths (e.g., in Coding-Decoding) and weaknesses (e.g., in Seating Arrangements). This chart serves as a qualitative summary of the results in Table 2, offering an intuitive look at each model's unique reasoning "fingerprint."

E Visual Aids in Solving Problems

In our qualitative analysis, we identified a unique reasoning strategy exclusive to Gemini models for

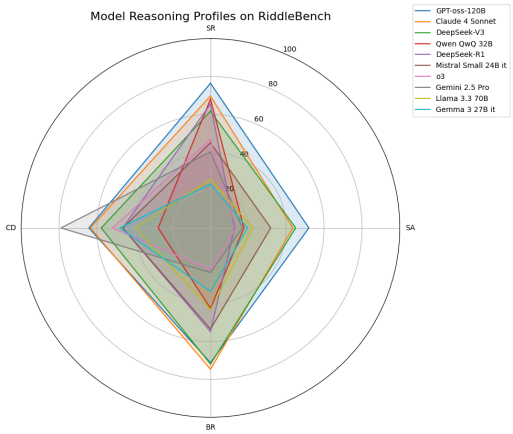

Figure 3: A radar chart illustrating the performance of evaluated LLMs across the four reasoning categories of RiddleBench. Each colored line represents a different model, showing its strengths and weaknesses in SR, SA, BR, and CD.

solving **Blood Relations** puzzles: the generation of ASCII art family trees. This emergent behavior, shown in Figure 4, mimics the human technique of drawing diagrams to visualize complex relationships.

By translating textual constraints into a spatial format, the model attempts a more sophisticated reasoning process than purely sequential text deduction. Although this method does not guarantee a correct answer, the strategy of building a visual mental model represents a promising step towards more robust and interpretable AI reasoning. This approach was not observed in any other model we evaluated.

```
   (parent of P & Q)
           |
  --------------------
  |                  |
 P(?)               Q(♂)
  |                  |
 R(♂)        ----------------
             |              |
            S(?)           T(♀)
```

Figure 4: An ASCII family tree generated by a Gemini model for a Blood Relations puzzle, a unique visual reasoning strategy observed in our analysis.

F Patterns in Thinking in Sequence Tasks

Our analysis of **Sequential Reasoning** puzzles revealed that most models use a consistent, two-step heuristic for numerical sequences. As exemplified in Figure 5, models first try to find an arithmetic pattern by calculating differences between terms. If that is inconclusive, they pivot to checking for a geometric pattern by calculating their ratios.

This standardized heuristic, which mirrors common human problem-solving methods, suggests the models have learned an effective procedural algorithm. While efficient for typical problems in RiddleBench, this formulaic approach indicates procedural imitation rather than abstract reasoning. This reliance on a fixed template may be a vulnerability for sequences that require more novel or unconventional logic.

```
I need to analyze the pattern in this sequence:
10, 31, 94, 288, 868, 2609

Let me look at the differences between consecutive terms:
- 31 − 10 = 21
- 94 − 31 = 63
- 288 − 94 = 194
- 868 − 288 = 580
- 2609 − 868 = 1741

Now let me look at the ratios:
- 31 / 10 = 3.1
- 94 / 31 ≈ 3.03
- 288 / 94 ≈ 3.06
- 868 / 288 ≈ 3.01
- 2609 / 868 ≈ 3.01
```

Figure 5: A typical reasoning trace from a model solving a numerical sequence. The model first checks for arithmetic differences before successfully identifying a geometric ratio pattern. This two-step heuristic was common across most models.

G Future Work

While RiddleBench provides a strong foundation for evaluating logical reasoning, we plan to extend this work in several key directions to further probe the capabilities and limitations of LLMs.

First, we intend to expand the benchmark itself. We may increase the number of problems within the existing categories to enhance statistical robustness and potentially introduce new types of logical puzzles to assess a broader spectrum of reasoning skills.

A primary goal for future work is to extend RiddleBench beyond a single language. True reasoning capabilities should be language-agnostic, and evaluating models on multilingual data is critical. We

plan to translate the benchmark into several other languages. Given the origin of our source material, we will initially focus on major Indian languages, with the goal of eventually including other world languages. This will enable a more comprehensive, cross-lingual assessment of LLM reasoning and help drive the development of more globally competent models.