

The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity

Work done during an internship at Apple.

{p_shojaee, imirzadeh, kalizadehvahid, mchorton, bengio, farajtabar}@apple.com

背景

言語モデルの推論能力

- これまでのLLM：大量のテキストを学習し，それを基に出力.
- 数学や論理系の問題のベンチマークで正答率が低かった.

解決策

- Chain of Thought(思考の連鎖)
 - 途中経過を出力しながら推論させる.
- self-reflection(自己反省)
 - 出した答えを自分で再チェックする.

推論モデルLRMが登場

- 様々な推論ベンチマークで好成績
 - Chain of Thought(思考の連鎖), self-reflection(自己反省)
- LRMの例
 - OpenAIのo1/o3, DeepSeek-R1, Claude 3.7 Sonnet Thinking

背景

LRM固有の問題

- 思考と結果のズレ
 - 思考の途中では正しそうに見えても、最終的な答えが間違っている.
- **Over thinking**
 - 一度正しい解にたどり着いても、無駄に長く出力を続ける.

数学問題(Ballonら)

- 思考を続けるほど間違った出力をしてしまう.
- パズルの難易度を一定以上上げると、考えなくなってしまう.

問題・目的

なぜ性能がいいのかが分からない

1. 既存の数学ベンチマークにたくさん触れた
2. 推論に使うトークン数が多く割り当てられている
3. 学習において、本当に「考え方」を学習した.

目的

- どれが要因なのかが明確でない
- LRMがLLMより性能が良い理由を特定し、どの程度使えるかを測定.

アプローチ

数学ベンチマークには限界がある

- 学習済み問題の混入
 - 思考プロセスの評価の困難さ
- ⇒モデルが正答した理由がブラックボックスになりやすい.

パズルの思考過程を追う

- ハノイの塔
- チェッカージャンプ問題
- 川渡し問題
- Blocks World

⇒数学ベンチマークよりも, 思考のパターンや失敗の仕方を詳細に分析できる.

実験1

使用モデル

- Claude 3.7 Sonnet(思考あり vs なし)
- DeepSeek-R1(思考あり) vs DeepSeek-V3 (思考なし)

各パズルの難易度を上げていく

- 例：ハノイの塔：円盤の数を増やす

3つの難易度設定

1. 低難易度
2. 中難易度
3. 高難易度

結果

1. 低難易度

- 非推論モデルが推論モデルと同等, またはそれ以上の精度.
⇒簡単な問題で推論モデルを選ばなくて良い.

2. 中難易度

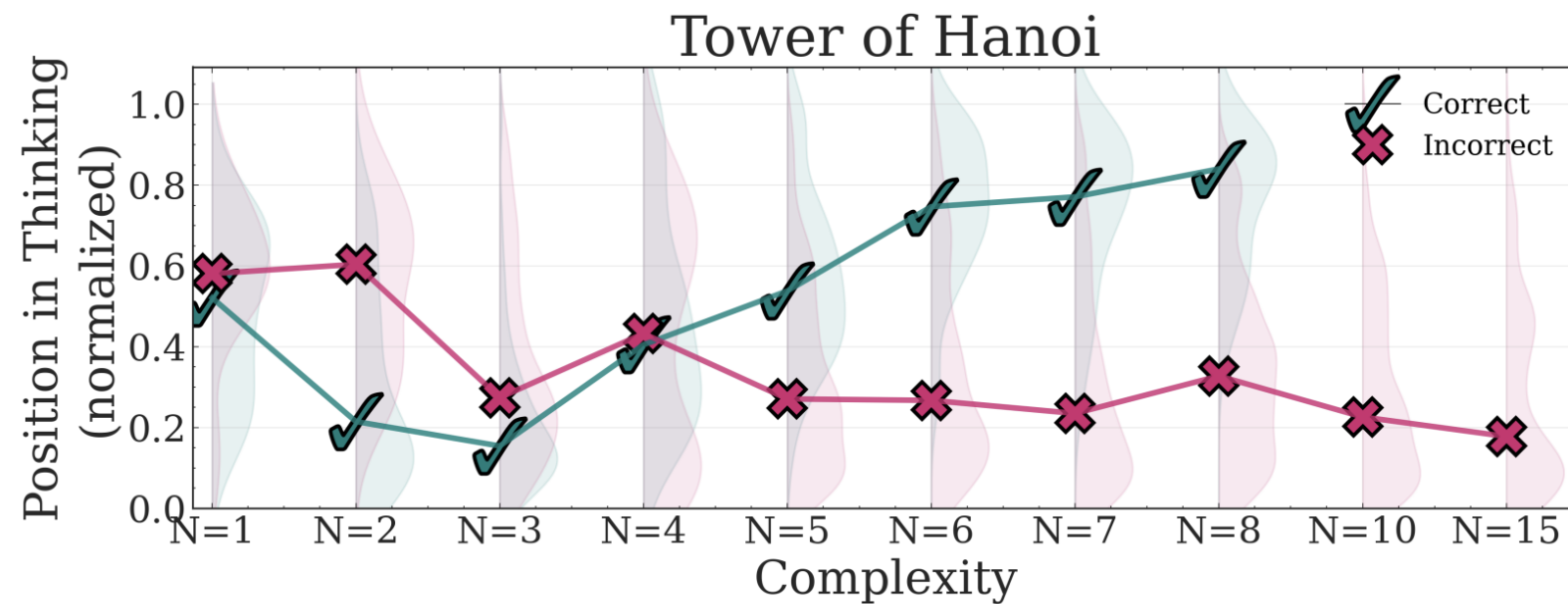
- 推論モデルが非推論モデルを上回っていく.
⇒推論モデルの性能が発揮し始める.

3. 高難易度

- 両モデルで正答率がほぼゼロに.
⇒LRMがどのように限界を迎えるのか再調査.

実験 2

ハノイの塔で見るLRMの思考過程の振る舞い



N = 1~4 : 思考するほど不正解

N = 5~8 : 思考するほど正解

N = 10~ : ほぼ正答率がゼロに

LRMの限界

LRMモデルで共通していること

- 問題の難しさが増すにつれて、徐々に正答率は下がる.
- ある特定の限界値を迎えると精度が**ゼロ**になってしまう.

思考トークン数

- 一般的に複雑度が増すと、思考トークン数も増える.
 - しかし、限界点に近づくと思考トークン数が**減り始める**.
- ⇒ 「考えることをやめる」ような振る舞い.

実験 3

LRMに正解アルゴリズムをプロンプトに埋め込む

- モデルは手順通りに実行するだけだが...?

結果：複雑度が高いものは正しく実行せず

- 正解通りに実行するだけなのに...
- 推論だけでなく，論理実行能力自体が弱い.

問題の種類にも関与

- ハノイの塔：100手以上正しく動作
- 川渡り問題：4手ほどでエラー

⇒ $N > 2$ のような事例はインターネット上のデータにほぼ存在しない.

⇒ 「汎用的な推論力」ではなく，学習データに大きく依存している.

まとめ

LRMは一定の複雑度を超えると、正答率が**ゼロ**になる

- 「汎化された推論能力」は獲得できていない。

推論トークン数が増えない不思議な挙動

- LRMにおける計算資源の**スケーリング限界**を示唆。

思考過程を追う中で分かったこと

- 低難易度であっても間違いを探索し続けてしまうため、LLMの誤答が発生し始める**中難易度**で使用するのが最も効果的。

正解アルゴリズムを与えても改善しない

総論：

「思考しているように見える」LLMでも、本質的な推論や正確な実行はできていない。

所感・感想

難易度が上がりすぎると，逆に推論トークン数が増えないのは非常に興味深い

- 思考トークン数が極端に減り始めると，推論してない可能性大
 - ルーティングする際の手がかりになるか

これまではトレーニングデータ量を増やせば性能が良くなっていたが，そうでないということは transformerベースのアーキテクチャの限界なのでは？