Tree of Thoughts Prompting

Siam Ahamed
Student ID: 2105155
Al Shahriar Alif
Student ID: 2105158
Imdadul Hasan
Student ID: 2105160

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

Contents

List of Figures

List of Tables

Abstract

In the rapidly evolving field of artificial intelligence and natural language processing, prompting techniques play a pivotal role in eliciting accurate and coherent responses from language models. This report explores the novel Tree of Thoughts (ToT) prompting technique, which structures reasoning into a tree-like framework, enabling systematic exploration of solution paths. Using the Game of 24—a challenging arithmetic problem requiring reasoning and creativity—we demonstrate the advantages of ToT prompting over traditional methods such as Input-Output (IO) Prompting, Chain-of-Thought (CoT) prompting, and Self-Consistency with CoT prompting. Through a comparative analysis, we highlight how ToT effectively leverages hierarchical problem-solving, leading to higher accuracy and better interpretability. This study underscores the potential of ToT prompting in enhancing complex reasoning tasks, paving the way for its broader application across domains requiring structured thought processes.

1 Introduction

Prompting techniques play a pivotal role in unlocking the reasoning capabilities of large language models (LLMs), enabling them to tackle complex tasks that require logic, creativity, and structured problem-solving. While traditional methods such as IO Prompting, Chain-of-Thought (CoT) prompting [1], and Self-Consistency with CoT [2] have achieved significant success, they often fall short when addressing problems that require exploring multiple reasoning paths or navigating non-linear solution spaces.

To address these challenges, the Tree of Thoughts (ToT) prompting framework [3] was introduced. By structuring the problem-solving process as a tree, ToT enables systematic exploration of alternative paths, fostering creativity and improving the model's ability to reach accurate solutions.

This report evaluates the effectiveness of ToT prompting through the lens of the Game of 24, a mathematical puzzle that involves using four given numbers and basic arithmetic operations (addition, subtraction, multiplication, and division) to achieve a target value of 24. The problem demands both creativity and logical reasoning, making it an ideal benchmark for comparing prompting techniques. Specifically, we assess the performance of ToT prompting against IO Prompting, CoT prompting, and Self-Consistency with CoT prompting, providing insights into their strengths and limitations.

The remainder of this report is structured as follows: we introduce the Game of 24 as a representative problem, describe the prompting techniques under comparison, and later propose the ToT prompting framework. Finally, we present a detailed analysis of their performance.

2 The Game of 24

The Game of 24 is a mathematical reasoning challenge that tests logical thinking and creativity. The objective of the game is to combine four given numbers using basic arithmetic operations $(+, -, \times, \div)$ to reach the target value of 24. The game encourages players to explore multiple ways of problem-solving, often requiring innovative and multi-step reasoning.

2.1   Description of the Game

The Game of 24 is simple in premise but challenging in execution:

- **Goal:** Use four numbers and the four basic arithmetic operations to form an equation that equals 24.

- **Allowed Operations:** Addition $(+)$, subtraction $(-)$, multiplication $(\times)$, and division $(\div)$.

Example:

- **Input Numbers:** 4, 9, 10, 13

- **Solution:** $(10 - 4) \times (13 - 9) = 24$

This puzzle helps sharpen analytical thinking and demonstrates the importance of step-by-step reasoning when solving complex problems.

3   Prompting Techniques to Solve the Game

In this report, we explore different prompting techniques that can guide both humans and AI models in solving the Game of 24 effectively.

3.1   Input-Output (IO) Prompting

This technique establishes a direct relationship between the input (the numbers and operations) and the output (the solution). It works well for straightforward(just like the fig1) problems where minimal reasoning is needed.

Figure 1: Input-Output Prompting.

Example:

- **Input:** 4, 4, 6, 8

- **Output:** $(8 - 6) \times (4 + 4) = 24$

Merits:

- Simple and easy to use.

- Faster response time.

- Ideal for basic arithmetic problems.

2

Limitations:

- Limited to single-step reasoning.

- Fails in complex multi-step scenarios.

- Lacks contextual awareness.

3.2 Chain of Thought (CoT) Prompting

Unlike IO prompting, CoT prompting involves breaking down the problem into intermediate steps(In Fig 2). It encourages multi-step reasoning and mimics how a human would think through the solution.
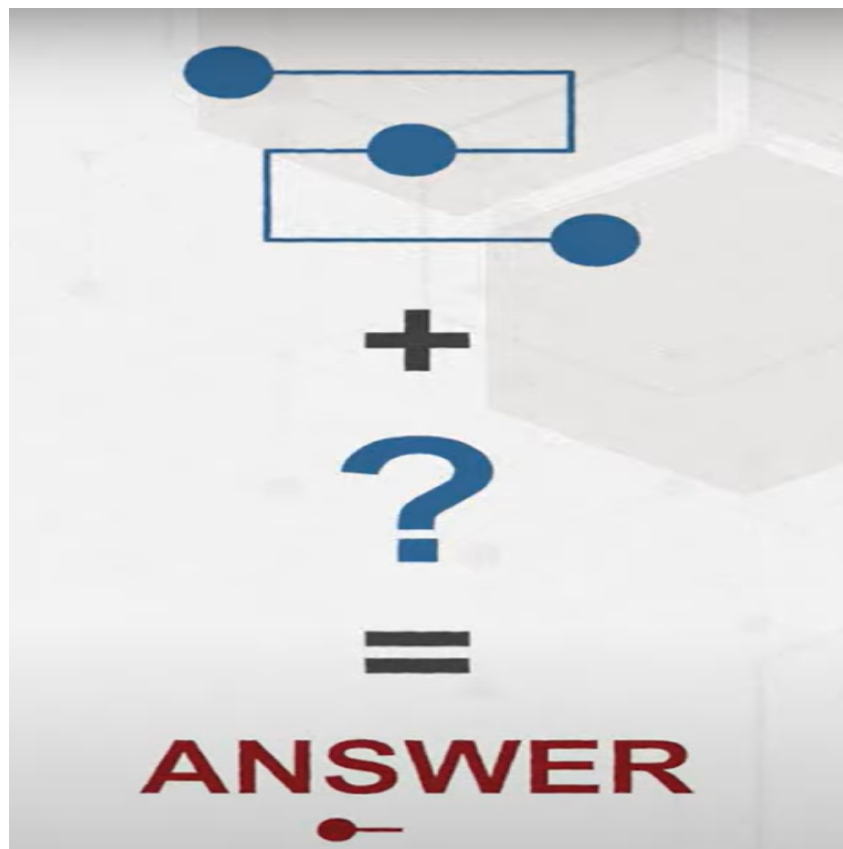

Figure 2: Chain of Thought Prompting.

Process:

1. Identify sub-problems.

2. Solve each step incrementally.

3. Combine results to reach the final answer.

Example Solution with CoT:

- **Input:** 4, 4, 6, 8

- **Step 1:** Subtract 6 from 8: $8 - 6 = 2$.

- **Step 2:** Add 4 and 4: $4 + 4 = 8$.

- **Step 3:** Multiply results from steps 1 and 2: $2 \times 8 = 24$.

Merits:

- Handles complex problems effectively.

- Encourages logical reasoning.

- Achieves better accuracy in multi-step scenarios.

3.3 Self-Consistency with Chain of Thought (CoT) Prompting

Instead of generating a single chain of thought, the model generates multiple reasoning paths (solutions) for the same problem(Figure 3). It then selects the final answer based on the most frequently reached conclusion across these diverse paths.
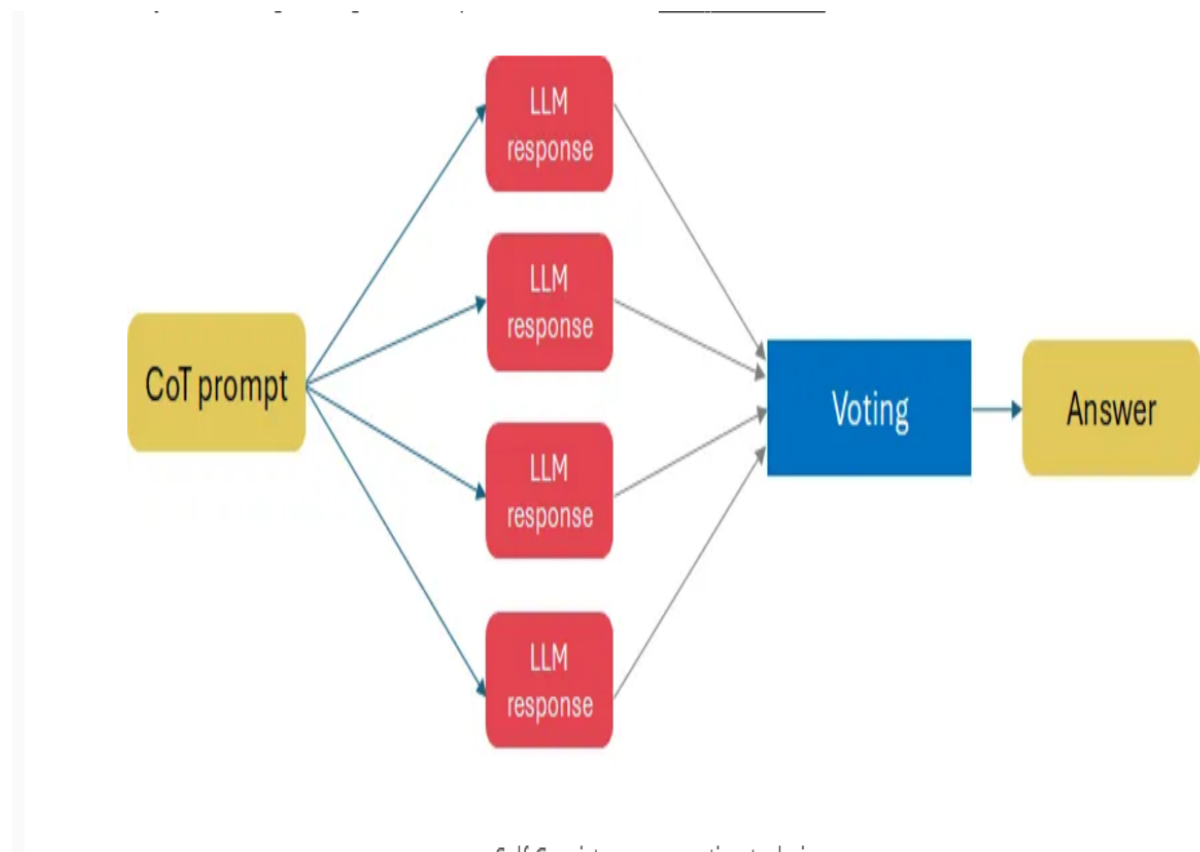

Figure 3: Self-Consistency with Chain of Thought Prompting.

Process:

1. **Initiate with CoT/Few-shot Prompting**

   - Use examples to demonstrate reasoning patterns.
   - Begin with a well-structured prompt that encourages step-by-step reasoning.

2. **Generate Multiple Outputs**

- Instead of providing a single response, run the prompt multiple times.
- Obtain a variety of plausible answers.

3. **Select the Most Consistent Answer**

   - Aggregate all outputs from Step 2.
   - Choose the most frequent or consistent response as the final answer.

Example Solution with SC-CoT Prompting:

s **Input:** $\{4, 4, 6, 8\}$

- **Step 1:** Subtract $6 - 4 = 2$ **Remaining Numbers:** $\{4, 8, 2\}$
- **Step 2:** Add $4 + 8 = 12$ **Remaining Numbers:** $\{12, 2\}$
- **Step 3:** Multiply $12 \times 2 = 24$ **Final Result:** 24

Merits:

- **Improved Accuracy and Reasoning:** Provides reliable solutions for complex problems.

- **Supports Complex Problem Solving:** Encourages breaking challenges into manageable steps.

- **Enhanced Explanations:** Solutions are clearer and more understandable.

Limitations:

- **Scalability Challenges:** Requires multiple iterations, increasing computational cost.

- **Vulnerability to Intermediate Errors:** Incorrect steps may lead to erroneous outputs.

- **Resource Intensive:** Demands additional processing time and resources.

Limitations:

- Slower than IO prompting.

- Requires structured instructions.

4 Comparison of Techniques

Technique	Merits	Limitations
IO Prompting	Quick and simple for single-step problems.	Struggles with complex or multi-step reasoning tasks.
Chain of Thought (CoT)	Enables step-by-step reasoning, improving logical accuracy.	Slower due to sequential steps and lacks robustness in branching decision-making.
Self-Consistency with CoT	Enhances accuracy by aggregating multiple reasoning paths.	Computationally expensive and limited in handling diverse solution spaces.

Table 1: Comparison of Prompting Techniques.

So we see existing techniques fall short when faced with problems that require systematic exploration of diverse pathways and iterative decision-making, as seen in human problem-solving. Tree of Thought (ToT) prompting addresses these limitations by:

- Encouraging exploration of multiple branches of reasoning simultaneously.

- Adopting a hierarchical structure to evaluate and prune solutions, similar to how humans reason through complex scenarios.

- Providing a more comprehensive and adaptable approach to solving problems like the Game of 24, where intermediate decisions significantly impact the final outcome.

ToT prompting not only mirrors human thinking but also introduces a scalable framework for tackling problems involving intricate reasoning and decision trees, making it the ideal technique for our next exploration.

5 Understanding Human Problem-Solving and Structured Thinking

Human problem-solving methods can be broadly classified into two categories:

1. **Random Thinking:** A spontaneous and unstructured approach often adopted by common individuals.

2. **Structured Thinking:** A logical and well-organized approach characteristic of mathematicians and systematic thinkers.

This part of the report analyzes these approaches, demonstrating their strengths and weaknesses through examples and applying them to AI modeling.

5.1 Methods

5.1.1 Random Thinking and Its Limitations

Definition: Random thinking involves trial-and-error methods without a coherent strategy.

Example: Solving the "Game of 24" with numbers :

1. Choose 14 and 3 and multiply them to get 42.

2. Divide 42 by 7, resulting in 6 .

3. Multiply 6 with 3 to get 18, which is not the target 24.

Limitation: Random approaches often lead to failure as they lack a systematic pathway to the solution.

5.1.2 Structured Thinking by Mathematicians

Definition: Structured thinking involves logical steps to decompose and solve a problem.
Example: Solving the "Game of 24" systematically:

1. **Sorting Inputs:** Arrange numbers .

2. **Initial Evaluation:**

   - Check if any number exceeds : None.
   - Multiply all numbers to confirm a solution is possible: .

3. **Logical Exploration:**

   - Identify promising operations (e.g., $5 * 7 = 35, 14 - 3 = 11$).
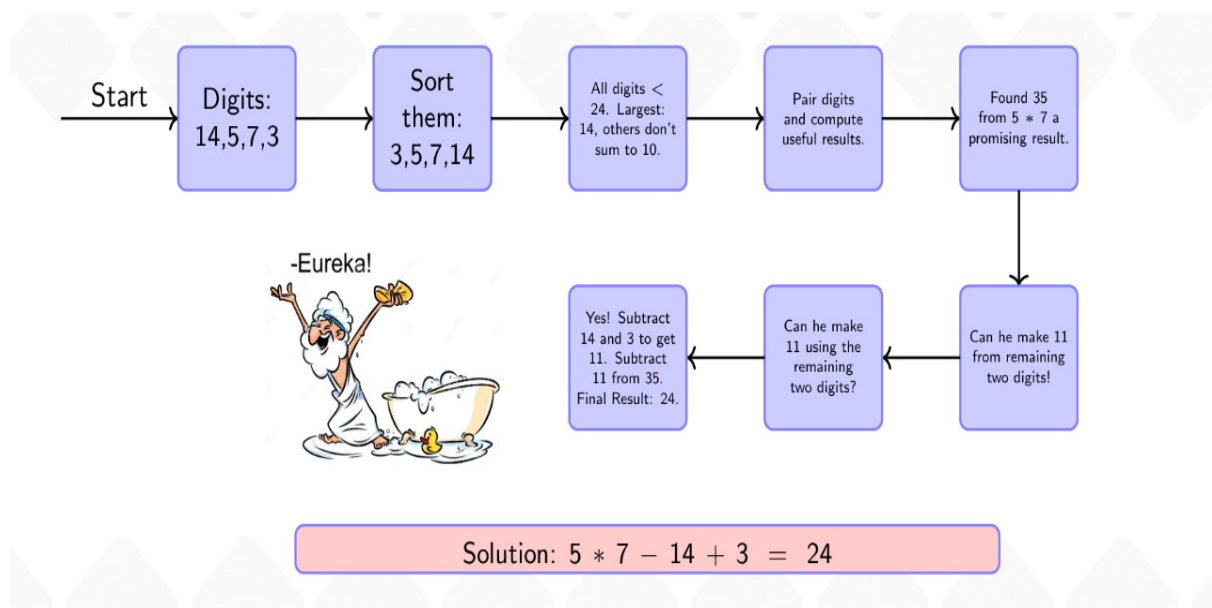   - Subtract 11 from 35 to reach 24.

Figure 4: Solving Game of 24 using structured thinking

Result: Solution achieved efficiently through logical steps.

5.2 Results

5.2.1 Key Observations

1. **Random Thinking:**

   - Inefficient and prone to errors.
   - May succeed occasionally but lacks reliability.

2. **Structured Thinking:**

   - Eliminates unnecessary calculations.
   - Guarantees a solution when possible through logical exploration.

Visualization of Process

Step	Action	Result
1	Sort inputs	
2	Evaluate Possibilities	Solution Possible
3	Logical Operations	
4	Final Calculation	

Table 2: Structured Thinking.

6 Application to AI: Modeling Human Problem-Solving

6.1 Tree of Thought (ToT) Prompting

ToT is an AI methodology inspired by structured human thinking. It includes the following steps:

1. **Problem Decomposition:** Break down the problem into smaller components.

2. **Thought Generation:** Explore possible solutions using logical operations.

3. **State Evaluation:** Assess intermediate results for progress toward the goal.

4. **Decision Making:** Select the most promising path to reach the solution.

6.2 Example: AI Solving "Game of 24"

1. **Problem Decomposition:**

   - Numbers:3,5,7,14.
   - Target: 24.
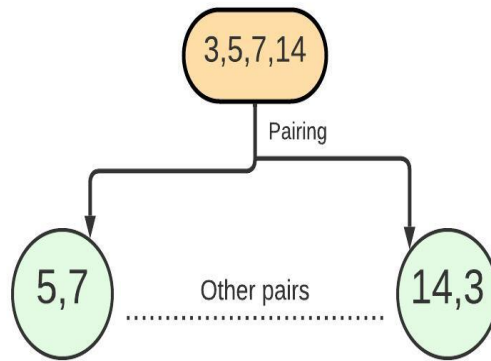
- Pair the inputs.

Figure 5: Pairing Inputs

2. **Thought Generation:**

   - Systematically compute results from the pairs generated in the decomposition step and apply operations to approach the target 24.
   - From the decomposition step , pairs such as (5,7) and (14,3) are identified.
   - Different arithmetic operations are performed on these pairs to generate intermediate values.
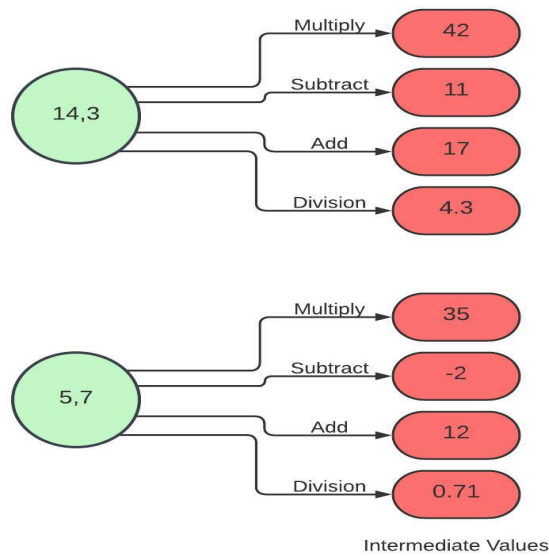   - For example:

Figure 6: Performing Arithmetic Operations on Paired Values to Generate Intermediate Values.

3. **State Evaluation:**

   - Evaluate as promising.

- Check if can be formed from remaining numbers.

4. **Decision Making:**

   - Combine results to achieve .

7 Advantages of Structured Thinking

1. **Efficiency:** Reduces unnecessary calculations and random guesses.

2. **Reliability:** Ensures a logical path to the solution.

3. **Applicability:** Can be extended beyond puzzles to real-world problems.

8 Solving Game of 24 Using Tree of Thought (ToT)

8.1 Conceptual Stages or Processes of Tree of Thought (ToT) Prompting

Tree of Thought (ToT) prompting involves breaking down problem-solving into two key processes: **Propose Prompt** and **Value Prompt**. These stages enable systematic exploration of potential solutions and evaluation of their validity, guiding a language model (LM) through reasoning like a human would while solving complex problems.
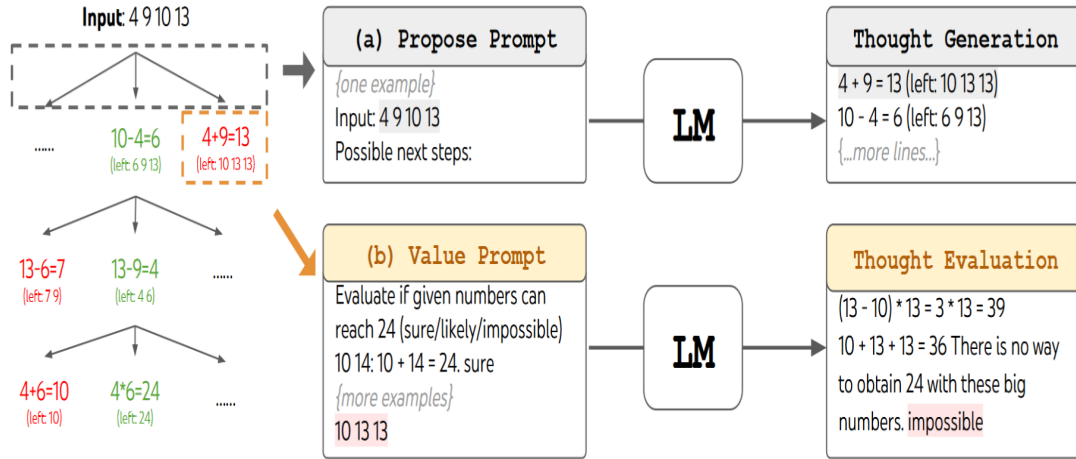

Figure 7: ToT in a game of 24. The LM is prompted for (a) thought generation and (b) valuation

1. **Propose Prompt (Thought Generation)**
   Objective: To generate possible next steps or ideas that lead closer to the solution of a problem.

   Key Characteristics:

- **Exploration:** The LM is prompted to think of multiple possible actions or operations based on the current input or state.
- **Tree Structure:** Each proposed step creates a "branch" in a tree, representing a new state in the reasoning process.
- **Open-ended Generation:** Encourages exploration of all viable options, expanding the solution space.

Example: In the *Game of 24*, given the input numbers 4, 9, 10, 13, the Propose Prompt might ask:
"What operations can be performed on these numbers to move closer to the target value of 24?"

The LM might generate:

- $4 + 9 = 13$ (new state: $\{13, 10, 13\}$)
- $10 - 4 = 6$ (new state: $\{6, 9, 13\}$)

This step focuses on what could be done next, creating a roadmap of possibilities.

2. **Value Prompt (Thought Evaluation)**
   Objective: To evaluate the feasibility or desirability of a particular branch or idea in the solution tree.

   Key Characteristics:

   - **Judgment:** The LM assesses whether the current state is likely, unlikely, or certain to lead to the solution.
   - **Pruning the Tree:** Irrelevant or unproductive paths are discarded, allowing focus on promising branches.
   - **Qualitative Assessment:** Results are categorized into labels such as:
     - **Sure:** The solution can be achieved from this state.
     - **Likely:** The solution may be possible, but further steps are needed.
     - **Impossible:** The solution cannot be achieved from this state.

   Example: For the *Game of 24*, given the state $\{10, 13, 13\}$, the Value Prompt might ask:
   "Can these numbers be combined to reach 24?"

   The LM evaluates:

   - $10 + 13 + 13 = 39 \rightarrow$ **Impossible**, as there is no way to reduce this to 24.

   This step focuses on whether the current idea works or is worth pursuing further.

The fig 8 demonstrates a "Tree of Thought" prompting technique to solve the arithmetic expression (5 * 7) - 14 + 3 using the input numbers 3, 5, 7, 14. It shows a computation tree where each node represents an intermediate step, applying arithmetic operations such as addition, subtraction, multiplication, or division to pairs of numbers. Valid operations producing integer results are expanded further, while invalid paths, such as non-integer division results, are discarded. The tree systematically explores all possible computation

paths, breaking down the problem into smaller subproblems. The successful path leading to the final result is highlighted with red arrows, showcasing the reasoning process behind solving the problem.
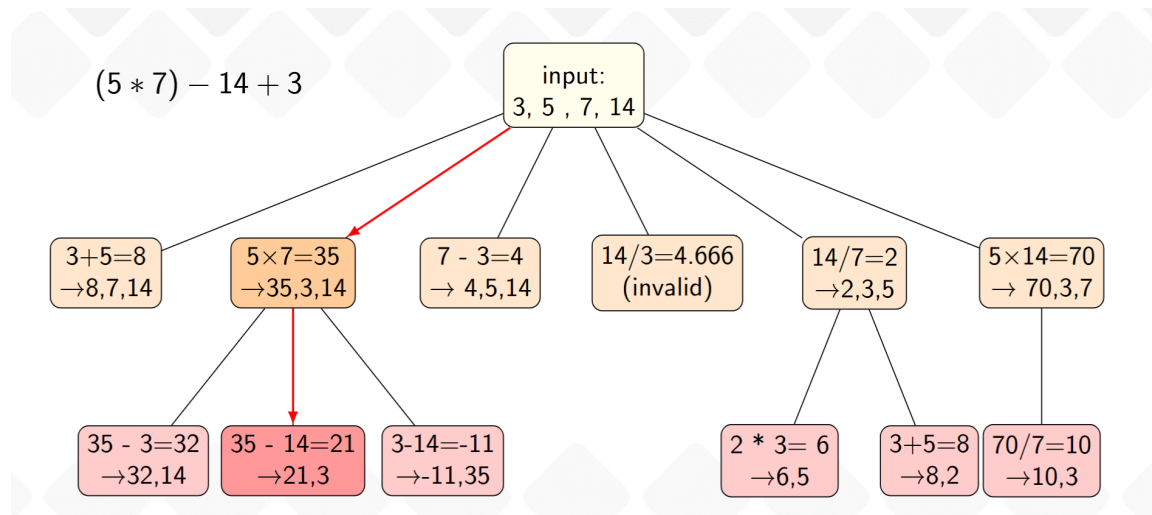

Figure 8: Visualization of game of 24

9   Conclusion

The experiments conducted on Tree of Thought (ToT) prompting in solving Game of 24 problems demonstrate its potential to systematically explore and evaluate multiple reasoning paths, mirroring human problem-solving processes. By leveraging intermediate steps and decision-making at each node, ToT enhances problem-solving accuracy, particularly in tasks involving logical reasoning, planning, and decision-making.

9.1   Capabilities

Tree of Thought prompting exhibits several strengths:

- **Enhanced Accuracy:** Compared to traditional methods like Chain of Thought (CoT) reasoning, ToT achieves a significantly higher success rate, as evidenced by its 74% success in solving Game of 24 problems.

- **Structured Exploration:** By systematically branching and evaluating multiple solution pathways, ToT enables deliberate exploration of complex problems, leading to more reliable outcomes.

- **Broad Applicability:** While particularly effective in structured reasoning tasks, this approach holds promise across a variety of domains requiring systematic problem-solving.

9.2   Limitations

Despite its advantages, ToT prompting has notable limitations:

- **Resource Intensity:** As shown in the cost analysis (Table 3), ToT requires a high number of generated and prompt tokens, resulting in greater computational costs compared to other methods.

Method	Generate/Prompt Tokens	Cost per Case	Success Rate
IO (best of 100)	1.8k / 1.0k	$0.13	33%
CoT (best of 100)	6.7k / 2.2k	$0.47	49%
ToT	5.5k / 1.4k	$0.74	74%

Table 3: Cost analysis on Game of 24.

- **Task Specificity:** The approach is best suited for tasks requiring structured and logical problem-solving, limiting its applicability to less structured or highly creative tasks.

Relevance to AI Advancements

The structured problem-solving paradigm exemplified by Tree of Thought prompting represents a significant step forward for AI systems. By improving reasoning capabilities and enabling efficient solution pathways, this approach contributes to advancing AI's ability to tackle complex, multi-step problems. While resource demands and limited task generality present challenges, ongoing optimization efforts could broaden its usability and cost-effectiveness.

References

[1] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou. *"Chain-of-Thought Prompting Elicits Reasoning in Large Language Models."* Submitted on 28 Jan 2022

[2] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, Denny Zhou. *Self-Consistency Improves Chain of Thought Reasoning in Language Models.* Published at ICLR 2023. V2: added PaLM results; V3: added UL2 results; V4: camera ready version at ICLR 2023.

[3] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, Karthik Narasimhan. *Tree of Thoughts: Deliberate Problem Solving with Large Language Models.* NeurIPS 2023 camera ready version..