Evolving Reasoning in Large Language Models: From Linear Chain-of-Thought to Graph-Based Diagram-of-Thought Frameworks

Debabrata Pruseth
AI Architect & Applied AI Researcher
Singapore

Author Note

This article is a research-style companion version of the parent blog post, **"The Evolution of AI Thinking: From Chain of Thought to Diagram of Thought,"** and its related explanatory articles on Chain-of-Thought, Tree-of-Thought, Logic-of-Thought, Iteration-of-Thought, and Diagram-of-Thought prompting. The purpose of this paper is to provide a structured academic framing of reasoning-oriented prompting techniques in large language models, with emphasis on how prompting has evolved from linear reasoning to branching, logical, iterative, and graph-based reasoning frameworks.

Abstract

Large language models have shown strong capabilities in language generation, reasoning, coding, summarization, and decision support. However, their reasoning quality is highly sensitive to how problems are framed and how intermediate thinking is structured. This article presents a comparative research-style framework for understanding the evolution of reasoning-oriented prompting techniques: Chain-of-Thought, Tree-of-Thought, Logic-of-Thought, Iteration-of-Thought, and Diagram-of-Thought. Chain-of-Thought prompting encourages linear step-by-step reasoning; Tree-of-Thought expands reasoning into multiple possible paths; Logic-of-Thought emphasizes rules, facts, and deductive structure; Iteration-of-Thought introduces self-review and refinement; and Diagram-of-Thought organizes reasoning as an interconnected graph of ideas, critiques, dependencies, and synthesis. The paper argues that the movement from chains to trees to graphs reflects a broader shift in prompt engineering: from simple answer generation toward structured reasoning orchestration. It concludes that graph-based and hybrid prompting frameworks may provide more robust support for complex decision-making, strategy formulation, and multi-factor reasoning tasks.

Keywords: Large Language Models, Prompt Engineering, Chain-of-Thought, Tree-of-Thought, Logic-of-Thought, Iteration-of-Thought, Diagram-of-Thought, AI Reasoning, Generative AI, Graph-Based Reasoning, LLM Reasoning Frameworks

1. Introduction

Large language models are increasingly used for tasks that require more than fluent text generation. Users now expect these systems to solve problems, compare options, reason through ambiguity, explain decisions, generate strategies, critique assumptions, and support decision-making. However, the reasoning performance of large language models depends strongly on the structure of the prompt and the reasoning pathway encouraged during inference.

Early prompting approaches often asked models to provide direct answers. This worked for simple factual or generative tasks but often failed for multi-step reasoning problems. Chain-of-Thought prompting introduced a major shift by encouraging models to generate intermediate reasoning steps before producing a final answer. Wei et al. showed that providing chain-of-thought exemplars can improve performance on arithmetic, commonsense, and symbolic reasoning tasks in sufficiently large language models.

Since then, prompting research and practice have expanded beyond linear reasoning. Tree-of-Thought prompting introduced the idea that models can explore multiple reasoning paths, evaluate alternatives, and backtrack when necessary. Yao et al. argued that standard left-to-right inference can be limiting for tasks requiring exploration or strategic lookahead, and proposed Tree-of-Thought as a framework for deliberate problem solving.

This article extends that discussion by organizing five reasoning-oriented prompting techniques into an evolutionary framework:

1. **Chain-of-Thought:** linear step-by-step reasoning
2. **Tree-of-Thought:** branching exploration of alternatives
3. **Logic-of-Thought:** rule-based and fact-based reasoning
4. **Iteration-of-Thought:** self-review and refinement
5. **Diagram-of-Thought:** graph-based synthesis of connected ideas

The parent blog frames these methods as a progression from simple step-by-step thinking to exploration, logic, self-improvement, and connected reasoning systems. It also demonstrates how these methods can be combined in practical prompts for decision-making tasks such as buying versus renting a house or deciding whether to repay debt or invest.

---

2. Background: Reasoning and Prompt Engineering in Large Language Models

Prompt engineering is the practice of designing instructions, context, examples, constraints, and output structures to guide large language models toward better responses. In reasoning tasks, prompt engineering does more than change wording. It influences the model's inferred problem-solving procedure.

Large language models generate outputs token by token. Without explicit reasoning structure, they may produce plausible but shallow responses. They may skip assumptions, ignore alternatives, fail to check logic, or provide overconfident answers. Reasoning-oriented prompting attempts to reduce these weaknesses by encouraging the model to externalize, structure, test, and refine intermediate reasoning.

The evolution of reasoning prompts can be understood as a movement across five levels:

Level	Prompting Pattern	Reasoning Structure	Primary Value
1	Chain-of-Thought	Linear sequence	Clarity
2	Tree-of-Thought	Branching alternatives	Exploration
3	Logic-of-Thought	Rules and facts	Consistency
4	Iteration-of-Thought	Review and refinement	Improvement
5	Diagram-of-Thought	Graph of connected ideas	Synthesis

This evolution reflects a shift from simple generation to structured reasoning orchestration. The model is not merely asked to answer; it is asked to reason through a designed cognitive workflow.

---

3. Chain-of-Thought Prompting: Linear Step-by-Step Reasoning

3.1 Concept

Chain-of-Thought prompting encourages a language model to break a problem into intermediate steps before giving a final answer. In its simplest form, the user asks the model to "think step by step" or provides examples showing how a problem can be decomposed into reasoning stages.

In the parent blog and child Chain-of-Thought article, Chain-of-Thought is described as making AI think step by step instead of jumping directly to an answer. The analogy used is solving a math problem line by line.

3.2 Research Basis

The original Chain-of-Thought paper by Wei et al. showed that generating intermediate reasoning steps can improve large language model performance on complex reasoning tasks. The paper reported improvements across arithmetic, commonsense, and symbolic reasoning tasks, especially for sufficiently large models.

The importance of Chain-of-Thought lies in its simplicity. It does not require model retraining. It changes the inference behavior through prompting. This made it one of the most widely adopted reasoning prompts in practical LLM usage.

3.3 Strengths

Chain-of-Thought is useful when a problem has a natural sequence of steps. Examples include:

- mathematical reasoning
- basic analytical reasoning
- explanation generation
- procedural tasks
- cause-and-effect analysis
- beginner-friendly teaching

Its main benefit is clarity. It helps users see how an answer was developed and makes the model less likely to skip essential intermediate steps.

3.4 Limitations

The main limitation is linearity. A chain has only one path. If an early assumption is wrong, the rest of the reasoning may inherit the error. Chain-of-Thought can also produce plausible but incorrect reasoning. In some cases, the explanation may sound coherent even when the conclusion is wrong.

Therefore, Chain-of-Thought is valuable but insufficient for problems requiring exploration, comparison, strategy, or uncertainty management.

---

4. Tree-of-Thought Prompting: Branching Reasoning and Exploration

4.1 Concept

Tree-of-Thought prompting extends Chain-of-Thought by allowing the model to explore multiple possible reasoning paths. Instead of producing one linear chain, the model generates several candidate approaches, evaluates them, and selects or synthesizes the best path.

In the parent blog, Tree-of-Thought is explained as "exploring different paths in a maze." The suggested prompt pattern asks the model to generate multiple possible solutions, evaluate each, and choose the best one.

4.2 Research Basis

Yao et al. introduced Tree-of-Thought as a framework for deliberate problem solving with large language models. Their paper argues that standard left-to-right token generation limits models on tasks that require exploration, strategic lookahead, or backtracking. Tree-of-Thought enables models to reason over coherent units of text called "thoughts," evaluate alternatives, and backtrack when necessary.

This approach is especially relevant for open-ended or complex problems where the first path may not be the best path.

4.3 Strengths

Tree-of-Thought is useful for:

- strategic planning
- creative problem solving
- decision-making
- product ideation
- scenario analysis
- trade-off evaluation
- complex planning

Its primary strength is exploration. It allows the model to avoid premature convergence on a single answer.

4.4 Limitations

Tree-of-Thought increases reasoning breadth but does not automatically guarantee logical correctness. The model may generate multiple weak alternatives or evaluate options using superficial criteria. It also requires more tokens and more careful prompt design.

For high-stakes problems, Tree-of-Thought should be combined with logical validation, evidence checks, or external tools.

---

5. Logic-of-Thought Prompting: Reasoning with Facts and Rules

5.1 Concept

Logic-of-Thought prompting emphasizes structured reasoning based on facts, conditions, constraints, and rules. Instead of asking the model only to reason step by step or explore alternatives, the prompt asks the model to identify relevant facts, formulate logical rules, and derive conclusions.

In the parent blog, Logic-of-Thought is described as making AI break problems into facts and logical rules before deriving conclusions. The analogy used is solving a puzzle using rules.

5.2 Role in the Reasoning Evolution

Logic-of-Thought can be viewed as a corrective mechanism for Chain-of-Thought and Tree-of-Thought. Chain-of-Thought provides sequence, and Tree-of-Thought provides alternatives, but both can still be weak if the reasoning is not grounded in explicit logic.

Logic-of-Thought introduces a more formal structure:

1. Extract relevant facts
2. Identify constraints
3. Define rules
4. Apply rules to facts
5. Derive conclusions
6. Check consistency

This makes it useful for domains where correctness depends on rules and conditions.

5.3 Strengths

Logic-of-Thought is useful for:

- legal-style reasoning
- compliance interpretation
- structured decision rules
- policy evaluation
- diagnostic reasoning
- analytical problem solving
- business rule analysis

Its main strength is consistency. It encourages the model to make assumptions and rules explicit.

5.4 Limitations

Logic-of-Thought may become rigid. It can be less effective for creative or ambiguous tasks where rules are incomplete or evolving. It also depends on the correctness of the extracted facts and rules. If the model extracts the wrong rule, the final conclusion may be wrong even if the reasoning structure appears valid.

Thus, Logic-of-Thought is most effective when paired with fact-checking and domain validation.

---

6. Iteration-of-Thought Prompting: Self-Review and Refinement

6.1 Concept

Iteration-of-Thought prompting asks the model to generate an initial answer, review it, identify weaknesses, and improve it through one or more refinement cycles. The parent blog describes this as “writing and editing a draft.”

This pattern reflects an important insight: the first model response is not always the best response. By explicitly asking the model to critique and refine its output, the user can often improve clarity, completeness, and quality.

6.2 Reasoning Role

Iteration-of-Thought introduces a feedback loop. The reasoning process is no longer a one-pass generation. Instead, it includes evaluation and correction.

A typical structure is:

1. Generate initial answer
2. Review assumptions
3. Identify missing factors
4. Check weaknesses
5. Improve the answer
6. Produce final version

This is especially useful for writing, strategy, planning, and decision support.

6.3 Strengths

Iteration-of-Thought is useful for:

- writing improvement
- executive summaries
- strategy refinement
- risk review
- proposal development
- policy drafting
- analysis improvement
- decision support

Its primary strength is refinement. It encourages the model to detect gaps and improve the output.

6.4 Limitations

Iteration-of-Thought can become circular if not bounded. The model may continue revising without meaningful improvement. It may also reinforce its own incorrect assumptions if the review step is not guided by clear criteria.

Effective Iteration-of-Thought prompts should specify:

- number of iterations

- review criteria
- expected output format
- constraints
- stopping point

For example, a prompt may ask the model to perform one review pass focused only on missing risks, then produce a final answer.

---

7. Diagram-of-Thought Prompting: Graph-Based Connected Reasoning

7.1 Concept

Diagram-of-Thought prompting extends reasoning beyond chains and trees into a network of connected ideas. In this approach, the model proposes ideas, connects them, critiques them, refines them, and synthesizes a final answer.

The parent blog describes Diagram-of-Thought as making AI build a network of ideas, critiques, refinements, and combinations. The analogy used is a mind map with feedback loops.

7.2 Why Graph-Based Reasoning Matters

Many real-world problems are not linear. They involve interacting variables, dependencies, feedback loops, trade-offs, and competing perspectives. For example, deciding whether to buy a house involves financial cost, lifestyle preference, interest rates, liquidity, family plans, tax treatment, property-market risk, and psychological comfort. These factors do not form a simple chain. They form an interconnected decision graph.

Diagram-of-Thought prompting attempts to represent this complexity. It encourages the model to reason across relationships rather than simply through a sequence.

7.3 Relationship to Chain and Tree Reasoning

Diagram-of-Thought can be understood as a generalization of prior prompting structures:

- Chain-of-Thought creates a linear path.
- Tree-of-Thought creates multiple branches.
- Logic-of-Thought creates rule-based structure.
- Iteration-of-Thought creates feedback loops.
- Diagram-of-Thought creates a connected reasoning graph.

This makes Diagram-of-Thought particularly useful for complex decision-making, strategy, systems thinking, and multi-factor synthesis.

7.4 Strengths

Diagram-of-Thought is useful for:

- strategic decision-making
- systems thinking
- enterprise architecture analysis
- multi-stakeholder decisions
- policy analysis
- risk assessment
- business transformation planning
- complex personal finance decisions
- research synthesis

Its main strength is connected depth. It allows the model to consider how different factors interact.

7.5 Limitations

Diagram-of-Thought is more complex to prompt. It may require clearer structure, such as asking the model to identify nodes, relationships, tensions, feedback loops, and final synthesis. It may also be harder for beginners to use because the output can become too broad if not constrained.

A good Diagram-of-Thought prompt should specify:

- key factors or nodes
- relationships among factors
- conflicts or trade-offs
- refinement steps
- synthesis format
- final recommendation criteria

---

8. Comparative Framework

The five prompting techniques can be compared as follows:

Technique	Reasoning Shape	Best Use	Main Strength	Main Limitation
Chain-of-Thought	Line	Step-by-step reasoning	Clarity	Error propagation
Tree-of-Thought	Tree	Exploring alternatives	Breadth	May lack logic
Logic-of-Thought	Rule structure	Analytical reasoning	Consistency	Can be rigid

Technique	Reasoning Shape	Best Use	Main Strength	Main Limitation
Iteration-of-Thought	Loop	Refinement	Improvement	Can loop endlessly
Diagram-of-Thought	Graph	Complex synthesis	Connected depth	Requires careful prompting

This framework suggests that reasoning prompts should be selected based on task type.

For simple explanation tasks, Chain-of-Thought may be enough. For strategy tasks, Tree-of-Thought may be better. For compliance or rule-based tasks, Logic-of-Thought is useful. For writing and refinement, Iteration-of-Thought improves quality. For complex multi-factor decisions, Diagram-of-Thought provides a more holistic structure.

---

9. Integrated Reasoning Workflow

The parent blog proposes a combined prompt structure that uses all five techniques together. The sequence is:

1. Chain-of-Thought: break the problem step by step
2. Tree-of-Thought: generate multiple approaches
3. Logic-of-Thought: extract facts and apply logic
4. Iteration-of-Thought: review and improve the answer
5. Diagram-of-Thought: combine insights into a structured solution

This can be interpreted as an integrated reasoning workflow:

Stage	Function	Output
CoT	Decomposition	Stepwise understanding
ToT	Exploration	Multiple options
LoT	Validation	Rule-based evaluation
IoT	Refinement	Improved answer
DoT	Synthesis	Integrated recommendation

This workflow is useful because each stage compensates for limitations in the previous stage. Chain-of-Thought provides clarity but may be narrow. Tree-of-Thought broadens the search but may lack rigor. Logic-of-Thought introduces structure. Iteration-of-Thought improves quality. Diagram-of-Thought connects the final insights.

10. Practical Use Case: Decision-Making Under Multiple Constraints

The parent blog demonstrates the combined framework using practical financial decision prompts, including whether to buy a house or continue renting, and whether to repay debt or invest. These examples are useful because they involve multiple interacting variables rather than simple factual recall.

For example, the buy-versus-rent decision includes:

- upfront cost
- mortgage cost
- maintenance cost
- rent cost
- time horizon
- interest rates
- property appreciation
- liquidity
- flexibility
- lifestyle preferences
- job stability
- risk tolerance

A purely linear answer may oversimplify the decision. A Tree-of-Thought approach can generate multiple scenarios: buy now, rent long-term, or rent now and buy later. A Logic-of-Thought layer can apply rules such as: if debt cost is high, buying becomes less attractive; if flexibility is important, renting becomes more attractive. An Iteration-of-Thought step can review missing variables, such as tax, liquidity, or job uncertainty. Finally, Diagram-of-Thought can synthesize financial, lifestyle, and risk factors into a balanced recommendation.

This example shows why graph-based reasoning is valuable. Real decisions are rarely single-path problems. They are networks of trade-offs.

---

11. Implications for Prompt Engineering Practice

The evolution from Chain-of-Thought to Diagram-of-Thought has several implications for practical prompt engineering.

First, prompt engineering should be task-sensitive. Not every task requires an advanced reasoning framework. Simple factual queries may not need Chain-of-Thought or Diagram-of-Thought. However, complex decisions benefit from structured reasoning.

Second, reasoning frameworks should be modular. Users can combine techniques depending on the problem. A legal analysis may combine Logic-of-Thought and Iteration-of-Thought. A strategy problem may combine Tree-of-Thought and Diagram-of-Thought. A technical explanation may use Chain-of-Thought followed by Iteration-of-Thought.

Third, visible reasoning should be used carefully. In practical applications, users may not need to see every internal reasoning step. Instead, prompts can ask for concise reasoning summaries, assumptions, alternatives, and final recommendations.

Fourth, reasoning prompts should include evaluation criteria. A model asked to "think better" may not know what better means. A stronger prompt defines criteria such as accuracy, feasibility, cost, risk, compliance, or stakeholder impact.

Finally, advanced prompting should not replace human judgment. These frameworks can improve structure, but they do not guarantee truth. External validation, domain expertise, and evidence checking remain necessary.

---

12. Governance and Safety Considerations

Reasoning-oriented prompting can improve output quality, but it also introduces governance questions.

A model may generate persuasive reasoning that is wrong. This is particularly risky in domains such as finance, law, healthcare, compliance, and enterprise decision-making. Therefore, reasoning outputs should be treated as decision support, not as authoritative conclusions.

There is also a transparency challenge. Chain-of-Thought outputs may appear to reveal reasoning, but generated reasoning is not necessarily a faithful representation of the model's internal process. Recent public discussions around reasoning models and chain-of-thought monitoring highlight both the usefulness and limitations of visible reasoning traces for safety and alignment.

For enterprise use, reasoning prompts should include:

- source grounding
- assumption disclosure
- uncertainty indicators
- risk flags
- human review
- domain expert validation
- audit-friendly summaries
- controlled output formats

This is especially important when prompt frameworks are used in business strategy, regulatory analysis, risk management, or customer-facing advisory systems.

---

13. Limitations

This article is a conceptual and framework-oriented research-style paper. It does not present a controlled benchmark comparing the five prompting techniques across datasets. The framework is based on published prompting research, practical prompt design patterns, and the author's explanatory blog series.

Several limitations should be noted.

First, the terms Logic-of-Thought, Iteration-of-Thought, and Diagram-of-Thought are used here as practical prompting frameworks rather than universally standardized academic taxonomies. Second, the effectiveness of these techniques depends on model capability, prompt quality, task type, and evaluation method. Third, more complex prompting may increase token cost and latency. Fourth, highly structured prompts can still produce incorrect answers if the model lacks knowledge or uses flawed assumptions.

Future work could evaluate these prompting techniques across standardized tasks, compare output quality using human and automated metrics, and test whether Diagram-of-Thought improves performance on multi-factor decision problems.

---

14. Conclusion

Reasoning-oriented prompting has evolved from simple step-by-step instructions toward richer structures for exploration, logic, refinement, and synthesis. Chain-of-Thought introduced the idea that models can improve reasoning by producing intermediate steps. Tree-of-Thought expanded this into branching search across alternatives. Logic-of-Thought emphasizes facts, rules, and structured deduction. Iteration-of-Thought adds self-review and improvement. Diagram-of-Thought moves toward graph-based reasoning, where ideas, constraints, critiques, and trade-offs are connected into a coherent synthesis.

This evolution reflects a broader movement in AI usage: large language models are no longer being used only as text generators. They are increasingly being used as reasoning assistants, decision-support tools, and structured thinking partners. However, better prompting does not eliminate the need for validation, evidence, and human judgment.

The central contribution of this paper is a comparative framework showing how different reasoning prompts can be selected and combined. For simple tasks, linear reasoning may be sufficient. For complex decisions, graph-based Diagram-of-Thought frameworks may provide a more powerful structure by integrating multiple perspectives, trade-offs, and feedback loops.

The future of prompt engineering is likely to be hybrid: combining chains, trees, logic, iteration, diagrams, tools, retrieval, and human oversight into structured reasoning workflows.

References

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. arXiv:2201.11903.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*. arXiv:2305.10601.

Pruseth, D. (2026). *The Evolution of AI Thinking: From Chain of Thought to Diagram of Thought.* Debabrata Pruseth AI Blog.

Pruseth, D. (2026). *How to Make Your AI Think Out Loud: Chain-of-Thought Prompting*. Debabrata Pruseth AI Blog.

Pruseth, D. (2026). *Tree of Thought Prompting Explained: Make AI Think Smarter*. Debabrata Pruseth AI Blog.

Pruseth, D. (2026). *Logic of Thought Prompting: Making AI Reason Like a Logician*. Debabrata Pruseth AI Blog.

Pruseth, D. (2026). *Iteration of Thought Prompting: Making AI Improve Its Own Thinking*. Debabrata Pruseth AI Blog.

Pruseth, D. (2026). *Diagram of Thought Prompting: Making AI Think Like a System*. Debabrata Pruseth AI Blog.

Suggested Citation

Pruseth, D. (2026). *Evolving Reasoning in Large Language Models: From Linear Chain-of-Thought to Graph-Based Diagram-of-Thought Frameworks*. Debabrata Pruseth AI Blog.