

# LLM Foundations for Agents

Week 2: Prompting, Reasoning, and Context

PhD Course in Agentic Artificial Intelligence

## Bloom's Taxonomy Levels

- **Remember:** Define Chain-of-Thought, Tree-of-Thoughts, Self-Consistency
- **Understand:** Explain how reasoning emerges from prompting strategies
- **Apply:** Implement various prompting techniques for complex reasoning
- **Analyze:** Compare effectiveness of different prompting strategies
- **Evaluate:** Assess trade-offs between reasoning depth and cost
- **Create:** Design custom prompting strategies for specific domains

---

These prompting techniques are the foundation of agent reasoning.

## The Reasoning Gap

- LLMs can reason, but not always reliably
- Prompting is “programming in natural language”
- The right prompt can unlock latent capabilities

## Agent Applications

- **Planning:** Decompose complex tasks into steps
- **Decision-making:** Weigh alternatives systematically
- **Error recovery:** Self-diagnose and correct mistakes
- **Tool selection:** Choose appropriate actions

---

Effective prompting is essential for building reliable agents.

# Chain-of-Thought Prompting

## Key Insight (Wei et al., 2022)

- Elicit intermediate reasoning steps before final answer
- “Let’s think step by step” unlocks reasoning

## Example

### Without CoT

Q: Roger has 5 tennis balls. He buys 2 cans of 3 balls each. How many balls?

A: 11

### With CoT

Q: [same question] Let’s think step by step.

A: Roger starts with 5 balls. 2 cans of 3 = 6 balls. Total:  $5 + 6 = 11$ .

---

CoT significantly improves math and multi-step reasoning tasks.

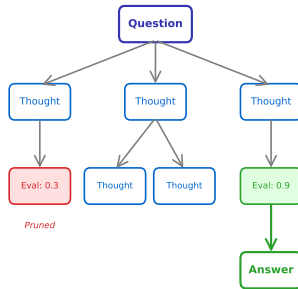
# Chain-of-Thought vs Tree-of-Thoughts

## Chain-of-Thought (CoT)



*Linear reasoning chain*  
One path to answer

## Tree-of-Thoughts (ToT)



*Explore multiple paths*  
Evaluate and prune

ToT (Yao et al., 2023) enables exploration and backtracking.

# Tree-of-Thoughts: Deliberate Problem Solving

## Key Innovation (Yao et al., 2023)

- Explore multiple reasoning paths simultaneously
- Evaluate intermediate states with value function
- Backtrack when paths look unpromising

## Algorithm

- ① Generate multiple candidate thoughts
- ② Evaluate each thought (LLM as evaluator)
- ③ Select best paths for expansion
- ④ Repeat until solution or budget exhausted

**Complexity:**  $O(b^d)$  where  $b$  = branching factor,  $d$  = depth

---

ToT excels at creative and planning tasks where exploration helps.

## Key Idea (Wang et al., 2023)

- Sample multiple reasoning paths at temperature  $> 0$
- Take majority vote on final answers
- Robust answers emerge from diverse reasoning

## Formula

$$\hat{a} = \arg \max_a \sum_{i=1}^k \mathbb{I}[a_i = a]$$

## When to Use

- Math problems, logical reasoning
- When single path may hallucinate
- Trade-off: More tokens, more cost, better accuracy

---

Self-consistency is simple but effective for improving reliability.

## **“Let’s think step by step” (Kojima et al., 2022)**

- No few-shot examples needed
- Simple prompt addition unlocks reasoning
- Works across many task types

## **Effective Phrases**

- “Let’s think step by step”
- “Let’s work this out in a step by step way”
- “First, ... Then, ... Finally, ...”
- “Let me break this down”

## **Agent Application**

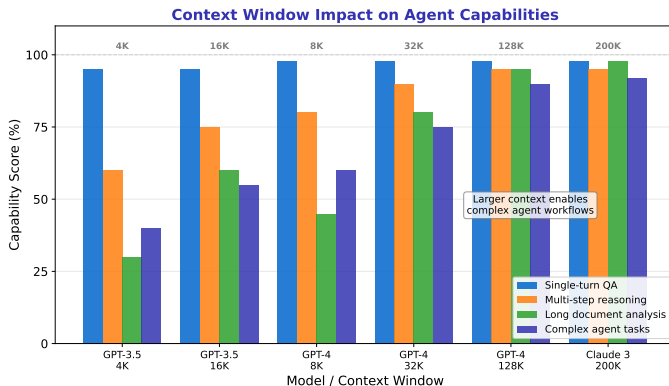
- Use in ReAct “Thought” steps
- Combine with tool-use for grounded reasoning

---

Zero-shot CoT is the simplest way to improve LLM reasoning.



# Context Window and Agent Performance



Context length determines how much history and context an agent can use.

## The Challenge

- Agents need: system prompt + history + tools + current task
- Context fills up quickly in multi-turn interactions

## Strategies

- **Sliding window:** Keep recent N turns
- **Summarization:** Compress old history
- **RAG:** Retrieve relevant context on demand
- **Hierarchical:** Use different models for different tasks

## Token Budget Planning

- Reserve space for: System (500-2K), Tools (1-5K), Response (1-4K)
- Remaining: Available for history and context

---

Context management is crucial for long-running agent tasks.

# Comparison of Prompting Strategies

Strategy	Tokens	Best For	Trade-off
Zero-shot	Low	Simple tasks	May miss nuance
Few-shot	Medium	Task demonstration	Example selection
Chain-of-Thought	Medium	Math, logic	Linear only
Tree-of-Thoughts	High	Creative, planning	Cost, latency
Self-Consistency	Very High	Reliability	Many samples

## For Agents

- Use Zero-shot CoT for simple reasoning
- ToT for planning and task decomposition
- Self-Consistency for critical decisions

---

Choose prompting strategy based on task complexity and cost constraints.

## Temperature Effects

- $T = 0$ : Deterministic, best for factual tasks
- $T = 0.3 - 0.7$ : Balanced creativity and coherence
- $T > 1.0$ : High creativity, risk of incoherence

## Agent Guidelines

- **Tool selection:**  $T = 0$  (deterministic)
- **Planning:**  $T = 0.2 - 0.5$  (some exploration)
- **Self-Consistency:**  $T = 0.7+$  (diversity)
- **Creative writing:**  $T = 0.8 - 1.0$

---

Temperature is a key hyperparameter for agent behavior.

## Prompt Structure for Agents

- 1 System context and role
- 2 Available tools and their descriptions
- 3 Output format specification
- 4 Few-shot examples (if applicable)
- 5 Current task and context

## Common Pitfalls

- Overloading system prompt (keep focused)
- Inconsistent formatting (LLM gets confused)
- Missing error handling instructions
- Vague tool descriptions

---

**Clear, consistent prompts lead to reliable agent behavior.**

## This Week

- Wei et al. (2022). “Chain-of-Thought Prompting Elicits Reasoning.” *NeurIPS 2022*. arXiv:2201.11903

## Supplementary

- Yao et al. (2023). “Tree of Thoughts.” arXiv:2305.10601
- Wang et al. (2023). “Self-Consistency.” arXiv:2203.11171
- Kojima et al. (2022). “Zero-Shot Reasoners.” arXiv:2205.11916

---

Chain-of-Thought is required; others are recommended.

# Summary and Key Takeaways

## Key Concepts

- **Chain-of-Thought:** Linear reasoning traces
- **Tree-of-Thoughts:** Branching exploration
- **Self-Consistency:** Majority vote over samples
- **Context Window:** Limits agent memory and complexity

## Key Equation

$$\hat{a} = \arg \max_a \sum_{i=1}^k \mathbb{I}[a_i = a] \quad (\text{Self-Consistency})$$

## Next Week

- Tool Use and Function Calling
- Model Context Protocol (MCP)

---

Prompting strategies form the reasoning backbone of LLM agents.