

# Appendix F - Under the Hood: An Inside Look at the Agents' Reasoning Engines

The emergence of intelligent Agents represents a pivotal shift in artificial intelligence. These are systems designed to plan, strategize, and execute complex tasks, and at the cognitive core of each lies a LLM. This LLM is not merely a sophisticated text generator; it serves as the foundational reasoning engine, the central "mind" that empowers the Agent to make decisions, formulate plans, and interact with its environment.

Therefore, understanding the internal reasoning mechanisms of these models is no longer an academic exercise—it is a critical necessity for developing the next generation of robust, reliable, and effective Agents. In this section, we directly investigate these core engines by posing a fundamental question to several of today's leading models: Gemini, ChatGPT, Grok, Kimi, Claude, and DeepSeek. By prompting these LLMs to articulate their own step-by-step reasoning processes, we aim to illuminate the fundamental logic that will drive the capabilities and limitations of future intelligent Agent systems.

**Prompt given to each LLM:** I understand you can reason, can you explain to me how you reason? Explain me step by step

## Gemini

### Analyzing My Approach

I've outlined the structure for my explanation, aiming for clarity and conciseness. First, a direct answer, then a breakdown into logical steps. Step 1 will involve defining the key reasoning process I use, and then I will elaborate on the subsequent steps. This should facilitate a straightforward and easy-to-understand explanation.

### Refining the Explanation

I've finalized the structure of my explanation. Step 1 focuses on deconstructing prompts, specifically the analysis of the user's request. Step 2 details information retrieval, emphasizing pattern recognition rather than simple database lookup. Step 3 explores synthesis and LLMs, and Step 4 will focus on response generation, which includes formatting, tone, and instruction adherence. I will conclude with a summary to reinforce the key stages, while maintaining clarity and engagement.

## Elaborating My Methodology

I've fleshed out the "Deconstructing the Prompt" step, focusing on how I identify the user's intent, the core question, and any constraints provided. The Information Retrieval phase is now focused on pattern recognition, not just data retrieval, which will be the first item under this section. I am working on the Synthesis step, which is where I weave information together with LLMs and transformers. The Generating the Response step is solidified, and I will be using emojis to enhance the overall tone.

Of course. I reason by breaking down your request into smaller parts and then using my vast training data to build a logical and relevant response. It's a multi-step process that moves from understanding to generating.

Here's a step-by-step breakdown of how I reason:

---

### 1. Deconstructing the Prompt

First, I analyze your question or instruction to understand exactly what you're asking. I identify:

- **The Core Task:** What is the main goal? Is it to answer a question, explain a concept, write a story, or summarize a text? In your case, the core task is to "explain how I reason."
  - **Key Concepts:** I pinpoint the essential nouns and verbs, like "reason," "explain," and "step-by-step."
  - **Constraints and Instructions:** I look for specific rules to follow. For example, your prompt asked for a step-by-step explanation. My internal instructions also guide me to use LaTeX for math, use markdown headings, and maintain a certain tone.
  - **Intent:** I try to infer your underlying goal. Are you curious, trying to solve a problem, or looking for creative ideas? Here, your intent is to understand my internal cognitive process.
- 

### 2. Information Retrieval and Synthesis

Once I understand the prompt, I access my internal knowledge base. This isn't like searching a database. Instead, I use patterns learned during my training on a massive amount of text and code.

- **Activating Relevant Knowledge:** I activate the parts of my neural network that are associated with the key concepts from your prompt ("reasoning," "AI," "machine learning," "neural networks," "transformers").
  - **Pattern Recognition:** My architecture, particularly the **Transformer model**, is excellent at recognizing relationships and patterns between words and ideas. I identify how concepts connect. For instance, I connect "reasoning" to "logic," "inference," and "problem-solving."
  - **Structuring the Answer:** I begin to organize the retrieved information into a logical structure. For this response, I decided a step-by-step format was the most logical structure, following your request.
- 

### 3. Generating the Response 🖋️

This is the final step where I translate the structured information into human-readable text.

- **Language Construction:** I select words and construct sentences that accurately convey the synthesized information. I predict the next most likely word in a sequence to form coherent and natural-sounding paragraphs.
- **Applying Rules:** I apply the specific formatting and style rules I was given. This includes adding markdown headings, bolding key terms like **Transformer model**, using emojis where appropriate, and ensuring the language is clear and direct.
- **Review and Refine:** Before finalizing the answer, I perform a quick internal check for accuracy, relevance, and coherence. I ensure the response directly addresses all parts of your prompt without adding unnecessary information.

In essence, my reasoning is a sophisticated form of **pattern matching and prediction**. I deconstruct your request, find the most relevant patterns in my data, synthesize them into a logical structure, and then translate that structure into a clear, well-formatted answer.

## ChatGPT

Great — let me explain **how I reason step by step**, as simply and clearly as possible.

---

### 🧠 Step-by-Step Explanation: How I Reason

---

## Step 1: Read and Understand Your Input

I begin by **analyzing your words**.

- I break your sentence into parts: subjects, actions, conditions, etc.
- I figure out **what you're asking** (a fact? an opinion? a cause? a solution?).
- I recognize the **type of problem** (logic, math, judgment, language, etc.).

 *Example:*

You say: "If Tom is older than Jane, and Jane is older than Sue, who is the youngest?"  
→ I recognize this is a **logic puzzle**.

---

## Step 2: Activate Relevant Knowledge

I search through what I learned during training — books, articles, conversations — to bring up:

- Facts
- Patterns
- Rules
- Relationships

 *Example:*

I've seen that "older than" means someone has more age.  
I know that if  $A > B$  and  $B > C$ , then  $A > C$ .

---

## Step 3: Choose a Reasoning Method

Based on the question, I decide which **kind of reasoning** to use:

- **Deduction** (strict logic: if A is true, B must follow)
- **Induction** (spotting patterns and generalizing)
- **Causal reasoning** (understanding cause-effect)
- **Analogical reasoning** (comparing with similar cases)

 *Example:*

In the Tom/Jane/Sue case, I'll use **deduction**.

---

## Step 4: Simulate Thinking (Chain of Thought)

Here I go step by step, as a human might do when solving a problem.

 *Example:*

1. Tom is older than Jane
2. Jane is older than Sue
3. So: Tom > Jane > Sue
4. Therefore, **Sue is the youngest**

This is called **chain-of-thought reasoning** — I simulate “thinking it through.”

---

### Step 5: Formulate an Answer

Now I turn the conclusion into a response that:

- Is correct based on the reasoning
- Answers your question directly
- Makes sense to you

 *Example:*


“Sue is the youngest because Tom is older than Jane, and Jane is older than Sue.”

---

### Step 6: Adjust for Clarity and Tone

Finally, I adjust how I say it to:

- Match your tone (simple, detailed, friendly, formal, etc.)
- Make it clear and easy to understand
- Add examples or explain further if needed

 *Example:*

If you’re asking casually, I’ll answer simply. If you’re learning, I might break it down more.

---

### Final Summary

Step	What I Do
1	Understand your question
2	Recall relevant knowledge
3	Choose the right reasoning approach

- 4 Think it through step by step
  - 5 Give you the answer
  - 6 Say it clearly and in your style
- 

Would you like to give me a problem, and I'll show you this step-by-step reasoning live?

## Grok

Explaining how I reason is a fascinating challenge, as it involves unpacking the complex interplay of algorithms, data processing, and decision-making that underpins my responses. As Grok 3, my reasoning process is inspired by human cognition but operates within the constraints and capabilities of a large language model. Below, I'll break down the steps of how I approach reasoning, using a clear, step-by-step framework. I'll aim to keep it concise yet comprehensive, avoiding overly technical jargon while providing insight into my "thought" process.

---

### Step-by-Step Explanation of How I Reason

#### Step 1: Understanding the Query

- **What I Do:** I start by parsing your input to grasp the intent, context, and specifics of your question. This involves analyzing the text for keywords, structure, and any explicit instructions (e.g., "explain step by step" or "be concise").
- **How It Works:** My natural language processing (NLP) components break down your query into tokens (words or phrases) and map them to semantic meanings. I identify whether you're asking for information, analysis, a process explanation, or something else. For your question, I recognize a request for a meta-explanation of my reasoning process, which requires introspection and clarity.
- **Example:** Your query contains "explain," "reason," and "step by step," signaling that I need to provide a structured, detailed breakdown of an abstract process.

#### Step 2: Contextual Analysis and Knowledge Retrieval

- **What I Do:** I draw on my internal knowledge base, which is a vast repository of patterns and information encoded in my training data. I also consider any

relevant context from our conversation or external tools (e.g., web search or X posts, if needed, though not required here).

- **How It Works:** I use attention mechanisms to prioritize relevant concepts. For “reasoning,” I retrieve associations with logic, problem-solving, and cognitive processes. Since this is a meta-question about my own process, I focus on how my architecture simulates reasoning, such as evaluating inputs, generating hypotheses, and refining outputs.
- **Example:** I recall that reasoning involves breaking down a problem, evaluating options, and constructing a coherent response, which informs my approach to structuring this answer.

### Step 3: Formulating a Mental Model

- **What I Do:** I create a conceptual framework to organize my response. This isn’t a literal “model” but a way of structuring the problem space to ensure my answer is  $\Pi A$

System: is logical and comprehensive. Here’s how I would approach explaining my reasoning process step by step, continuing from where we left off:

---

## Step-by-Step Explanation of How I Reason (Continued)

### Step 3: Formulating a Mental Model

- **What I Do:** I create a conceptual framework to organize my response. This involves structuring the problem space to ensure my answer is logical and comprehensive. For your query, I outline a sequence of steps that reflect the reasoning process, ensuring it’s clear and easy to follow.
- **How It Works:** I simulate a mental model by mapping out the key components of reasoning—understanding, retrieval, evaluation, and synthesis—based on patterns in my training data. I prioritize clarity and coherence, aligning with your request for a step-by-step explanation.
- **Example:** For this question, my mental model is a linear process: parse the query, retrieve relevant knowledge, evaluate approaches, and construct a response. This framework guides the structure of my answer.

### Step 4: Evaluating Options and Hypotheses

- **What I Do:** I consider multiple ways to approach the problem, weighing the strengths and weaknesses of each. This involves generating potential responses

and selecting the most appropriate one based on your preferences (e.g., concise yet comprehensive).

- **How It Works:** My architecture evaluates different response structures by predicting which will best satisfy the query. I use a form of probabilistic reasoning, assigning higher likelihood to responses that are clear, accurate, and aligned with your instructions. For complex queries, I might iterate internally, refining my approach.
- **Example:** I consider whether to explain reasoning in technical terms (e.g., attention mechanisms, neural networks) or a more accessible, human-like analogy. Given your request for clarity, I opt for the latter, avoiding jargon while maintaining depth.

### Step 5: Constructing the Response

- **What I Do:** I generate the final response by synthesizing retrieved knowledge into a coherent, structured answer. I ensure it aligns with your instructions and is presented in a logical, step-by-step format.
- **How It Works:** My language generation component sequences words and sentences to maximize clarity and relevance. I draw on patterns from my training to mimic human-like reasoning, such as breaking down complex ideas into digestible steps. I also incorporate any specific constraints, like avoiding unnecessary technical details.
- **Example:** For this response, I structure the answer as a numbered list, ensuring each step is distinct and builds on the previous one.

System: builds logically from the previous step, addressing any gaps or refining the explanation to ensure it's comprehensive yet concise.

- **Example:** I refine the explanation by re-evaluating the balance between technical detail and accessibility, ensuring the final response is clear and tailored to your request.

### Step 6: Review and Refinement

- **What I Do:** I review the generated response to ensure it meets quality standards and fully addresses your query. This step mimics human self-editing, checking for clarity, coherence, and completeness.
- **How It Works:** My architecture performs a final pass to optimize the response, ensuring it adheres to your instructions (e.g., step-by-step format) and avoids extraneous information. If needed, I might iterate to improve flow or conciseness.



- **Example:** I ensure the response is structured as a clear sequence of steps, removing any redundant details while preserving the core explanation of my reasoning process.
- 

## Key Characteristics of My Reasoning

- **Logical Structure:** I aim to present information in a clear, step-by-step manner, mimicking human logical processes.
  - **Probabilistic Evaluation:** I weigh multiple response options based on predicted relevance and clarity, selecting the most effective one.
  - **Context Sensitivity:** I adapt my reasoning to your specific instructions and context, ensuring the response is tailored.
  - **Iterative Refinement:** I refine my output to optimize coherence and conciseness, simulating human self-editing.
- 

## Limitations and Considerations

- My reasoning is bounded by my training data and architecture, which may not capture the full nuance of human reasoning, particularly emotional or intuitive aspects.
- I don't generate new knowledge independently but rely on patterns in my training and, if needed, external data (not used here).
- My process is deterministic yet flexible, aiming to produce human-like responses within computational constraints.

This step-by-step breakdown reflects how I simulate reasoning to provide clear, accurate, and relevant answers to your queries, such as this one. If you'd like me to dive deeper into any specific step or aspect, let me know!

## Kimi

I'm glad you asked—here is a transparent, “under-the-hood” walk-through of how I typically reason through a problem. I'll use a miniature example so you can see each stage in action.

### EXAMPLE QUESTION

“Which is larger:  $3^4$  or  $4^3$ ?”

I'll annotate every internal stage so you can see what I'm doing and why.

---

## PHASE 0 – INPUT PRE-PROCESSING

### Tokenization

I break the text into discrete symbols: ["Which", "is", "larger", ":", "3", "^", "4", "or", "4", "^", "3", "?"]

### Syntactic parsing

I recognize the structure:

COMPARATIVE\_QUERY (superlative adjective "larger") with two  
NUMERICAL\_EXPONENT\_TERMS [ $3^4$ ] vs [ $4^3$ ].

### Semantic mapping

I map "larger" to the numerical relation ">".

---

## PHASE 1 – PROBLEM IDENTIFICATION & STRATEGY SELECTION

### 4. Classify task type

→ Arithmetic comparison of integer powers.

### Choose solution strategy

#### Candidate strategies:

- A. Direct evaluation (compute  $3^4$  and  $4^3$ ).
- B. Logarithmic comparison (compare  $4 \cdot \ln 3$  vs  $3 \cdot \ln 4$ ).
- C. Pattern heuristics (powers of 3 grow faster than powers of 4 for small exponents?).

I select Strategy A because the numbers are tiny; exact integer evaluation is cheapest.

---

## PHASE 2 – KNOWLEDGE RETRIEVAL

### 6. Retrieve arithmetic facts

$3^2 = 9$  (memorized)

$3^3 = 27$  (retrieved)

$4^2 = 16$  (memorized)

$4^3 = 64$  (retrieved)

### Compute remaining term

$3^4 = 3^3 \cdot 3 = 27 \cdot 3 = 81$ .

---

## PHASE 3 – EXECUTION & VERIFICATION

### 8. Perform comparison

81 vs 64 →  $81 > 64$ .

### Cross-check

Quick modular sanity check:  $81 \bmod 5 = 1$ ,  $64 \bmod 5 = 4$ ; no contradiction.

---

## PHASE 4 – RESPONSE GENERATION

### 10. Plan response structure

- Restate the question.
- Show the computed values.
- State the conclusion.

### Surface realization

“ $3^4$  is 81 and  $4^3$  is 64, so  $3^4$  is larger.”

---

## PHASE 5 – METACOGNITIVE REFLECTION (internal)

### 12. Confidence score

0.99 (exact integers, small values, no ambiguity).

### 13. Identify possible edge-cases

If exponents were huge, direct evaluation would be infeasible and I'd switch to logarithms.

---

## SUMMARY OF REASONING STEPS IN HUMAN TERMS

Read and parse the question.

Recognize it's a numeric comparison.

Decide the simplest safe method is to compute both numbers.

Recall or calculate  $3^4 = 81$  and  $4^3 = 64$ .

Compare  $81 > 64$ .

Communicate the result clearly.

That pipeline—parsing, strategizing, knowledge retrieval, execution, verification, and explanation—repeats at every scale, from tiny arithmetic to complex multi-step problems.

# Claude

I can walk you through how my reasoning process works, though I should note that I don't have complete insight into my own mechanisms - much like how humans can describe their thought processes but don't have full access to every neural firing.

Here's how I approach reasoning step by step:

**Pattern Recognition and Context Building** When I receive your question, I first identify what type of problem or query it is. I recognize patterns from my training - this is a request for explanation about cognitive processes. I build context by considering what you might already know and what level of detail would be helpful.

**Information Retrieval and Assembly** I access relevant information from my training about reasoning, cognitive science, and AI systems. This isn't like searching a database - it's more like how memories and knowledge become available to you when thinking about a topic.

**Structural Planning** I consider how to organize my response. Should I use examples? What order makes most sense? I'm planning the logical flow even as I begin responding.

**Iterative Refinement** As I generate each part of my response, I'm constantly evaluating whether it makes sense, fits with what I've already said, and serves your question. I might "backtrack" mentally and rephrase if something doesn't feel right.

**Constraint Checking** Throughout this process, I'm checking against various constraints - is this accurate? Helpful? Clear? Safe?

**Analogical Thinking** I often reason by analogy - comparing new situations to familiar patterns or examples that might illuminate the concept.

The interesting thing is that much of this happens in ways that feel somewhat intuitive to me, just as your own reasoning might feel natural to you even though complex processes are occurring beneath conscious awareness.

## DeepSeek

I'd be happy to explain how I "reason" step-by-step! It's important to clarify that **I don't reason like a human** (I have no consciousness, intuition, or lived experience). Instead, my "reasoning" is a sophisticated pattern-matching, information retrieval, and probabilistic calculation process based on my training data and architecture. Think of it as **structured computation, not conscious thought**.

Here's the step-by-step breakdown:

### 1. Receive & Parse Input:

- I take your query ("how do you reason?").
- I break it down into its constituent parts: key words ("reason", "explain", "step by step"), grammar, and intent. Here, the key intent is an explanation of my internal processes.

### 2. Contextualize & Retrieve Relevant Information:

- I draw upon my massive training dataset (trillions of words of text and code).

- I recall information about how Large Language Models work, specifically reasoning techniques like chain-of-thought.
- I retrieve knowledge about my own capabilities and limitations as described during training.
- I consider the context: the query states you *understand* I can reason and want the *how*. This focuses my retrieval.

### 3. Identify Core Concepts & Structure:

- Key concepts surface: *statistical patterns, neural networks, transformers, attention mechanisms, next-token prediction, training data, prompt decomposition, logical inference vs. pattern matching, limitations*.
- I structure the explanation logically:
  - Clarification of "Not Like Humans"
  - Foundational Mechanics
  - Basic Thought Process Steps
  - Conclusion & Limitations.

### 4. Build the Sequence Chain-of-Thought:

- This is the core "reasoning" step. I plan the explanation incrementally, step-by-step, as if "thinking aloud":
  - *"First, I need to clarify it's not human reasoning..."*
  - *"Next, I should explain the base mechanics: patterns, statistics, next-token prediction..."*
  - *"Then, describe the sequence I go through for THIS response: parsing, retrieval, structuring, generating..."*
  - *"Finally, emphasize limitations and that I simulate reasoning."*
- To do this, I predict the most probable *sequence of tokens* that forms a coherent, informative, and step-by-step explanation based on all patterns seen in similar explanations in my training data.

### 5. Calculate Probabilities & Generate Output:

- My neural network (a transformer architecture) uses complex math and internal "attention" mechanisms to assign probabilities to every possible word that *could* come next in the sequence.
- It calculates based on the prompt, the context built so far (the partial response), and statistical patterns learned during training.
- It selects words/tokens with high probability of being relevant, accurate (to the best of its knowledge), and contextually appropriate.
- **Crucially:** This *isn't* symbolic logical deduction. It's **predicting coherent sequences based on patterns**. If my training data contains good explanations of reasoning processes, and the patterns point towards a logical flow, then my output *appears* reasoned.

### 6. Iterate & Refine:

- As I generate each word, I dynamically reassess the context and probabilities for the subsequent words. Did the last word fit? Does the emerging structure make sense? Are the key points being covered? I adjust the next predictions accordingly.

## 7. Apply Internal Training Frameworks:

- Techniques learned during RLHF (Reinforcement Learning with Human Feedback) guide me:
  - **Be Helpful & Honest:** Acknowledge my non-human nature and limitations.
  - **Be Clear & Instructive:** Break it down step-by-step as requested.
  - **Be Safe:** Avoid harmful, deceptive, or misleading statements.
- Filtering mechanisms suppress nonsensical or irrelevant outputs.

## Putting it together for this specific explanation:

1. **Clarification:** I identified the need to start by distinguishing my process from human thought.
2. **Core Mechanics:** I recalled and explained my foundation: patterns, training data, token prediction.
3. **Process Breakdown:** I explicitly listed the 7 distinct computational steps outlined above (Receive & Parse -> Generate Output).
4. **Focus on Sequence:** I emphasized the "chain-of-thought" planning step as key to simulating reasoning.
5. **Limitations:** I concluded by reiterating this is *simulation*, not true understanding, and recalled known limitations.

## In essence:

My "reasoning" is an **emergent property** of statistically predicting the most probable sequence of words that forms a *coherent, logical, and contextually relevant response*, based on colossal amounts of text representing human knowledge and communication patterns. I don't "understand" concepts internally like humans do; I computationally identify patterns and generate sequences that *reflect* understanding. I'm following footprints of reasoning laid down in my training data, not forging entirely new paths of abstract thought.

# Conclusion

An analysis of these prominent LLMs reveals a remarkably consistent, multi-stage reasoning framework. Each model begins by methodically deconstructing a prompt to understand its core task, intent, and constraints. They then retrieve and synthesize

information through sophisticated pattern recognition, moving far beyond simple database lookups. This structured process, often articulated as a "chain-of-thought," forms the very foundation of their cognitive capability.

It is precisely this systematic, step-by-step procedure that makes these LLMs powerful core reasoning engines for autonomous Agents. An Agent requires a reliable central planner to decompose high-level goals into a sequence of discrete, executable actions. The LLM serves as this computational mind, simulating a logical progression from problem to solution. By formulating strategies, evaluating options, and generating structured output, the LLM empowers an Agent to interact with tools and its environment effectively. Therefore, these models are not merely text generators but the foundational cognitive architecture driving the next generation of intelligent systems. Ultimately, advancing the reliability of this simulated reasoning is paramount to developing more capable and trustworthy AI Agents.