Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 7, 1072-1090
2025
Publisher: Learning Gate
DOI: 10.55214/25768484.v9i7.8810

Integrating reasoning into large language models: A major step toward truthful AI

Nguyen Duc Xuan[1], Pham Van Khanh[2], Do Thi Loan[2], Le Thi Thuy Giang[2*]
[1]VNU Universtity of Economic and Business; 144 Xuan Thuy street, CauGiay district, Ha Noi, Vietnam; nguyenducxuan@vnu.edu.vn (N.D.X.).
[2]Institute of Information Technology - Vietnam Academy of Science & Technology; khanhvietdm@gmail.com (P.V.K.) Loandt.ioit@gmail.com (D.T.L.) dienkhanh06@gmail.com (L.T.T.G.).

Abstract: This study investigates the approach of integrating reasoning into the outputs of large language models (LLMs) — combining logical inference techniques with knowledge retrieval — to enhance their alignment with truth. We begin by analyzing the statistical foundations of LLMs, which operate as probabilistic text generators based on Markovian assumptions without genuine semantic understanding. Next, we discuss the conceptual framework of 'truth' in the AI context, differentiating between descriptive truth (objective correctness), pragmatic truth (contextual utility), and verifiable knowledge (information supported by independent evidence). We examine advanced reasoning techniques — from Chain-of-Thought [1], Tree-of-Thought [2], Retrieval -Augmented Generation [3], to self-critique models like CriticGPT [4] - that move LLMs closer to verified knowledge and mitigate hallucination tendencies. The paper also explores the philosophical implications: Can modern LLMs, equipped with reasoning capacities, be considered 'fallible cognitive agents' - akin to humans in their capacity for error correction and learning -or are they merely stochastic parrots mimicking language without true understanding? Finally, we open a discussion on the risks, limitations, and ethical issues involved in deploying reasoning-integrated AI systems, connecting them with contemporary philosophical currents such as pragmatism, anti-realism, and behaviorist perspectives in evaluating artificial intelligence.

Keywords: *Chain-of-thought, Epistemic and ethical AI, Large language models, Reasoning Integration, Retrieval-augmented generation, Tree-of-thought, Truthfulness in AI.*

1. Introduction

Large language models (LLMs), such as GPT-3 and GPT-4, have demonstrated impressive achievements in generating and understanding natural language, primarily due to extensive training on massive text corpora. However, their capabilities fundamentally rely on statistical emulation of language patterns rather than genuine comprehension of the world. Indeed, an LLM essentially functions as a probabilistic model predicting the next word based on a sequence of previous words. In other words, it operates as an extended Markov model, estimating the probability of a word sequence $W_1, W_2, ..., W_n$ by multiplying conditional probabilities $p(W_1) \cdot p(W_2 | W_1) \cdot ... \cdot p(W_n | W_1, ..., W_{n-1})$ [5]. This intrinsic "statistical continuity" nature often leads to LLMs being metaphorically referred to as "stochastic parrots," producing plausible-sounding language without true semantic understanding. Indeed, language models predict words but do not necessarily guarantee factual correctness. Consequently, LLMs may confidently generate seemingly persuasive information that is entirely factually incorrect because they lack an internal verification mechanism.

History: Received: 7 May 2025; Revised: 24 June 2025; Accepted: 27 June 2025; Published: 12 July 2025
* Correspondence: dienkhanh06@gmail.com

The issue of hallucinations - when a model confidently provides nonexistent or fabricated information - has raised the crucial question: how can language models be guided closer to truthfulness? This challenge extends beyond technical dimensions, raising profound philosophical questions. The concept of "truth" in the context of AI must be dissected: Does "truth" for an AI equate to objective accuracy (descriptive/correspondence truth)? Or should an AI's response be considered "true" if it is useful and persuasive to the user (pragmatic success)? Alternatively, should AI strictly offer verifiable knowledge backed by clear evidence? These questions intersect various philosophical traditions—from pragmatism, emphasizing utility and application of knowledge, to anti-realist perspectives (post-truth theories), skeptical about the existence of a singular objective truth, and cognitive-behavioral viewpoints emphasizing intelligence assessment through observable behavior rather than internal mental states.

Fosso Wamba, et al. [6] comprehensively analyze generative AI, particularly large language models such as ChatGPT, from the perspectives of information systems and socio-technical theory, highlighting unique characteristics and associated challenges (including potential negative impacts), proposing a detailed research agenda for information systems to address profound technological impacts. The authors underscore GenAI's ability to generate creative content, contextualize human interactions, and execute knowledge tasks superior to previous AI technologies. However, GenAI's default behavior relies on statistical probabilities without genuine understanding, prone to hallucinations and biases, thus posing reliability challenges. Utilizing system theory, the article identifies three key generative characteristics of GenAI: strong emergence, novel creativity, and independent system outputs like code, text, and images. Concurrently, GenAI has significant disruptive potential impacting organizations, individuals, policies, and labor forces, alongside numerous "dark sides" such as deepfakes, manipulation, intellectual property infringements, and environmental impacts. The research combines systems theory, socio-technical frameworks, and linguistic analyses, focusing on GenAI as a "socio-technical system" through the ChatGPT case study. This paper contributes a theoretical framework for understanding and managing GenAI, proposing a comprehensive IS research agenda and urging the academic community to critically evaluate GenAI's social, technical, and ethical implications beyond initial enthusiasm.

Recently, researchers have developed various techniques aimed at integrating reasoning capabilities into the operational processes of LLMs, hoping to bridge the gap between mere "statistical parrots" and genuine "knowledge agents." Techniques such as Chain-of-Thought (CoT), enabling models to generate intermediate reasoning steps, Tree-of-Thought (ToT), expanding reasoning across multiple branches, Retrieval-Augmented Generation (RAG), integrating real-time document retrieval before responding, and self-critique mechanisms like CriticGPT, acting as internal evaluators for the model's output, have shown promising results. These methods enable LLMs to handle complex multi-step tasks better, reduce misinformation, and even provide source references for responses.

This introduction systematically presents: (1) the background of LLMs from a statistical perspective and their inherent limitations in capturing truth; (2) philosophical analyses of truth in AI contexts, distinguishing types of truth; (3) advanced reasoning techniques (CoT, ToT, RAG, CriticGPT) altering LLMs' approaches to knowledge; (4) philosophical implications of viewing LLMs as cognitive agents; and (5) ethical issues, risks, and challenges associated with integrating reasoning into AI. The goal is to connect technological advancements with contemporary philosophical discussions, enhancing our understanding of the evolving trajectory toward more trustworthy and intelligent AI systems.

2. Theoretical Background: Statistical Nature of Large Language Models

Today's large language models fundamentally originate from statistical language models. The basic goal of a language model is to estimate the probability distribution over a sequence of words. Specifically, given a word sequence $W_1, W_2, ..., W_n$, the model seeks to estimate the probability $P(W_1, ..., W_n)$, typically decomposed using the chain rule into the product of conditional probabilities:

$$P(W_1, W_2, ..., W_n) = P(W_1) \cdot P(W_2 | W_1) \cdot P(W_3 | W_1, W_2) \cdots P(W_n | W_1, ..., W_{n-1}).$$

At the core of the model is the conditional distribution —the probability of the next word given the previous history of words. Traditional n-gram models limit the history to the immediate preceding words, whereas modern LLMs using neural networks (such as Transformer architectures) can consider much longer contexts (hundreds of words or more). Despite variations in technique, the underlying principle remains that LLMs predict subsequent words based solely on statistical relationships.

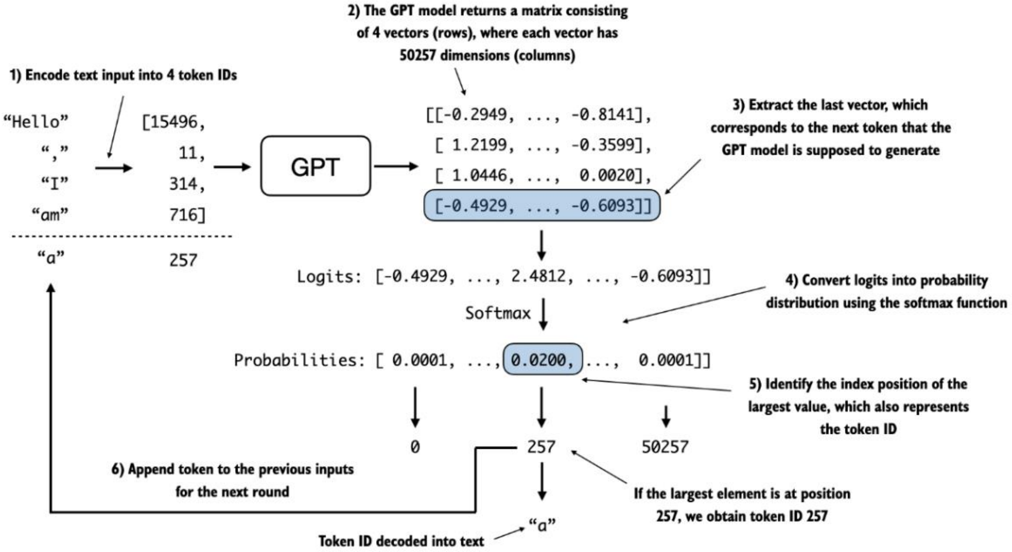
2) The GPT model returns a matrix consisting of 4 vectors (rows), where each vector has 50257 dimensions (columns)
1) Encode text input into 4 token IDs
"Hello"
","
"I"
"am"
[15496,
11,
314,
716]
"a"
257
GPT
[[-0.2949, ..., -0.8141],
[1.2199, ..., -0.3599],
[1.0446, ..., 0.0020],
[-0.4929, ..., -0.6093]]
3) Extract the last vector, which corresponds to the next token that the GPT model is supposed to generate
Logits: [-0.4929, ..., 2.4812, ..., -0.6093]]
Softmax
4) Convert logits into probability distribution using the softmax function
Probabilities: [0.0001, ..., 0.0200, ..., 0.0001]]
5) Identify the index position of the largest value, which also represents the token ID
0
257
50257
6) Append token to the previous inputs for the next round
If the largest element is at position 257, we obtain token ID 257
Token ID decoded into text
"a"

Figure 1.
Data Generation Process of a Large Language Model.
Source: Raschka [7].

Figure 1 illustrates the data generation process of a large language model such as GPT from a statistical perspective. Initially, the input text is encoded into numerical token IDs (Step 1), serving as discrete observations fed into the model. The LLM then computes a series of vectors (Step 2), each representing unnormalized log probabilities (logits) across the entire vocabulary (in this example, 50,257 possible tokens). The model's statistical prediction is based on extracting the last vector in the sequence (Step 3), representing the conditional distribution of the next token given the previous context. These logits are transformed into probability distributions through the softmax function (Step 4), normalizing them into a valid probability simplex. The model selects the token with the highest probability by identifying the index of the highest probability value (Step 5), and the selected token is appended to the input sequence for iterative generation (Step 6). This process embodies the basic autoregressive assumption of LLMs, whereby the probability of a sequence $P(x_1, x_2, ..., x_n) = \prod_{t=1}^{n} P(x_t | x_{<t})$ is decomposed into the product of conditional probabilities, emphasizing the model's probabilistic nature and complete dependence on the data.

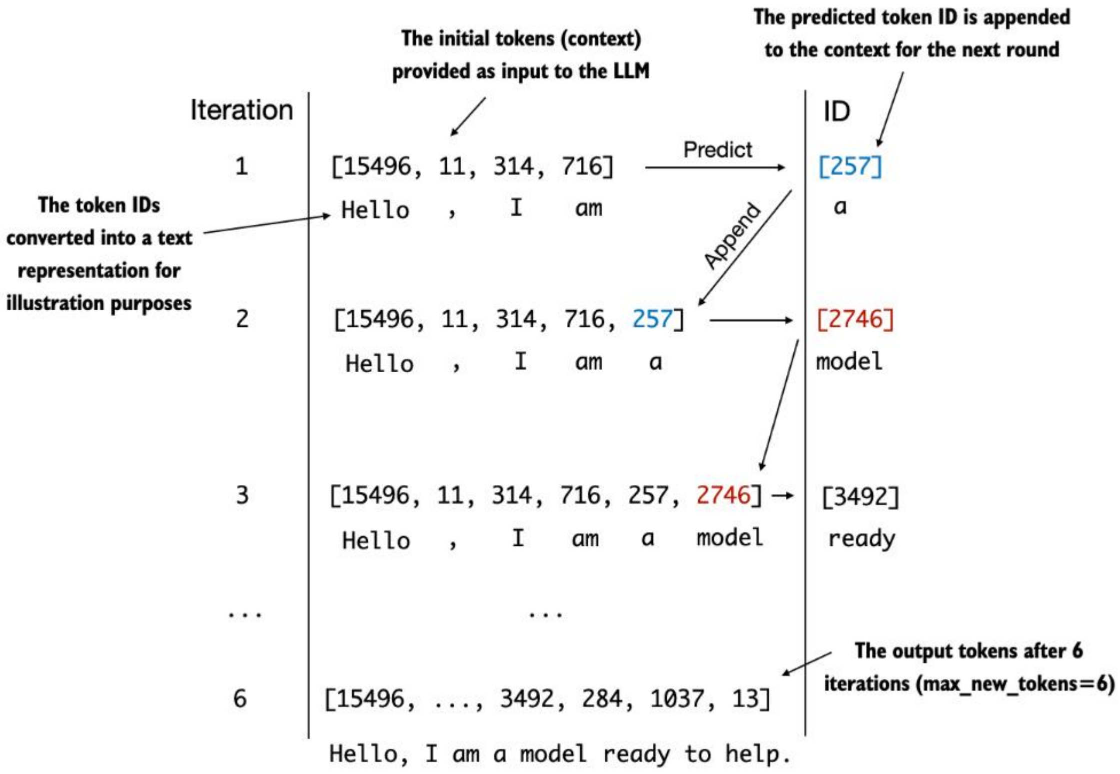
The initial tokens (context) provided as input to the LLM
The predicted token ID is appended to the context for the next round
Iteration
ID
1
[15496, 11, 314, 716]
Predict
[257]
Hello , I am
a
The token IDs converted into a text representation for illustration purposes
Append
2
[15496, 11, 314, 716, 257]
[2746]
Hello , I am a
model
3
[15496, 11, 314, 716, 257, 2746]
[3492]
Hello , I am a model
ready
...
...
The output tokens after 6 iterations (max_new_tokens=6)
6
[15496, ..., 3492, 284, 1037, 13]
Hello, I am a model ready to help.

Figure 2.
Autoregressive Nature of Large Language Models.
Source: Raschka [7].

The statistical foundation of large language models such as GPT lies in their autoregressive nature, where the joint probability of a sequence of tokens is decomposed into the product of conditional probabilities. Figure 2 illustrates this mechanism by showing how an input sequence such as "Hello, I am" is encoded into token IDs—[15496, 11, 314, 716]—and input into the model, resulting in a logits matrix representing scores across the entire vocabulary at each token position. The model selects the last row of this matrix, the conditional logits vector for the next token, and applies the softmax function to convert it into a probability distribution over the entire vocabulary. The token with the highest probability (e.g., 257, corresponding to "a") is chosen as the prediction. As shown in Figure 2, this predicted token is appended to the sequence, forming a new context for the subsequent iteration. Through repeated iterations, the model generates successive tokens, each token depending on the entire previously generated sequence.

For example, after six iterations starting with the phrase 'Hello, I am,' the model generates the complete sequence: 'Hello, I am a model ready to help.' This autoregressive decoding loop exemplifies how an LLM performs probabilistic inference on natural language, where each prediction is fully governed by the distribution learned from vast training datasets and executed through logits normalized by the softmax function. The result is an elegant yet powerful statistical framework, enabling the generation of coherent, fluent, and contextually relevant language.

Therefore, the training process of LLMs optimizes predicting the correct next word as seen in the training data, without ensuring real-world factual accuracy. The model is rewarded for correctly predicting the next word in context, regardless of whether the content of the sentence is factually correct. If the training dataset frequently contains an incorrect fact, such as 'Paris is the capital of Germany,' the model will learn a high probability for the combination 'Paris - capital - Germany.'

Consequently, LLMs have no external awareness of truth; their only sense is statistical probability. As one author aptly put it: 'ChatGPT doesn't "know" truths through logical reasoning; it merely guesses based on what it has been trained on.' In other words, LLMs 'predict words, not truths.

A direct consequence of this nature is that LLMs lack an intrinsic mechanism to verify the accuracy of the information they produce. The model operates essentially as a "black box" text generator—it cannot "look outside" to cross-check its responses against objective reality [8]. For the model, the only accessible "reality" is the statistical patterns within the training data. Thus, if an input query demands knowledge beyond its learned dataset, the model tends to rely on the nearest linguistic patterns available, leading to "hallucinations" of incorrect information. For instance, when asked about a recent event occurring after its training period, the model might fabricate an answer that sounds plausible but is entirely baseless due to the absence of credible data to reference.

To clarify further, one could metaphorically liken an LLM to a student memorizing answers from thousands of books. When encountering familiar questions, the student might correctly respond from memory. However, when faced with novel questions, the student tends to guess based on the nearest available knowledge, potentially resulting in incorrect answers. Crucially, this student has no capacity to independently consult new resources or conduct experiments during an exam. Similarly, an original LLM has no capability for autonomous research or reasoning beyond its trained data; it lacks an independent verification process [8].

Consequently, even though LLMs can generate fluent text and accurately answer numerous queries, they still unpredictably commit severe mistakes. Multiple studies have documented that large models such as GPT-3 and GPT-4 confidently assert implausible or factually incorrect statements. For instance, a model might confidently fabricate a detailed description of a nonexistent historical event or invent fake references. These errors are not intentional—simply because the model does not recognize them as wrong; it merely probabilistically replicates linguistic patterns. Bender et al. highlight two fundamental limitations: (1) LLMs are inherently restricted by their training data and merely replicate statistically whatever appears within that data; (2) Since they solely rely on linguistic data, the models have no means of discerning the factual correctness or incorrectness of their outputs.

Thus, theoretically speaking, LLMs are powerful statistical language models but inherently lack genuine semantic understanding. They do not possess a concept of truth as humans understand it—they do not actually know whether "Paris is the capital of Germany," but merely recognize that the phrase "Paris is the capital of…" is statistically more often followed by "France" rather than "Germany." This disconnect between linguistic statistics and truth is precisely the origin of the problem we seek to address: how to guide LLMs closer to genuine truth rather than merely emulating language patterns. To achieve this goal, we must clarify different conceptions of truth and subsequently investigate methods for integrating logical reasoning—an intrinsic strength of human intelligence—into these models.

3. Philosophical Discussion: "Truth" in the Context of AI

The concept of truth has long been a central theme in philosophy, approached from various perspectives. In the context of generative AI language models, the notion of "truth" becomes particularly complex. We can distinguish at least three dimensions for assessing the "truthfulness" of AI-generated responses: (1) descriptive truth (objective accuracy), (2) pragmatic success (usefulness or effectiveness in practical contexts), and (3) verifiable knowledge (the ability to confirm statements through independent evidence). These three dimensions correspond partly to traditional philosophical theories of truth: correspondence theory, pragmatic theory, and verificationism (which can be considered a variant of coherence theory or scientific positivism).

- Descriptive truth: This is the most traditional conception of truth—a statement is considered true if it accurately reflects objective reality. Philosophically, this aligns with the correspondence theory of truth: the statement "Paris is the capital of France" is true because it corresponds with the factual reality that Paris is indeed the capital of France. Conversely, "Paris is the capital of Germany" is false as it does not match reality (Germany's capital is Berlin). In AI contexts, this criterion evaluates outputs based on their alignment with factual knowledge of the world. An LLM achieves descriptive truth if its output matches objective reality accurately.

• Pragmatic success: Pragmatism, initiated by philosophers like Charles Peirce and William James, views truth in terms of practical utility. Simply put, a proposition is "true" if believing and acting upon it yields useful outcomes or resolves problems effectively. Applied to AI, pragmatic truth means that an AI's response is valued highly if it proves beneficial to the user, even if the information might not be literally or absolutely accurate. For example, if a user asks, "How can I relieve a headache?" and the AI answers, "Drink a glass of water and rest quietly," this advice, while not a descriptive fact but rather a practical instruction, can still be considered pragmatically "true" if it effectively alleviates the user's headache. In the context of AI chatbots trained through human feedback methods (e.g., Reinforcement Learning from Human Feedback - RLHF), usefulness often takes priority. The goal is user satisfaction, making responses pragmatically true rather than strictly accurate. Practically, this pragmatic approach is explicitly integrated into ChatGPT's training: the model is rewarded when users judge its responses as helpful, regardless of absolute factual accuracy. Consequently, this results in a pragmatic orientation toward truth: prioritizing responses that are "useful and minimally harmful" over those that might be "absolutely correct yet useless or potentially irritating" [8]. In other words, "the usefulness of the response is often placed above factual accuracy" [8]. However, this approach represents a double-edged sword: while making AI more user-friendly and helpful, it also risks compromising factual integrity.

• Verifiable knowledge: The third perspective emphasizes the independent verifiability of information. A statement can be considered reliable knowledge if it can be checked and confirmed through evidence, experimentation, or reputable sources. Philosophy of science, notably positivism and Karl Popper's falsificationism, highlights falsifiability—the requirement that scientific statements must be empirically testable. In the AI context, as models do not inherently verify external reality themselves, this aspect pertains significantly to the model's ability to reference and substantiate information. An AI-generated response accompanied by citations from reputable sources or easily verifiable information meets the standard of "verifiable knowledge." Conversely, if the AI produces information not supported or confirmable by any external source, then even if it seems plausible, users should remain skeptical. For instance, if an AI states, "A 2023 study found that green tea cures COVID-19" without providing any credible source, users would rightfully demand verification before accepting such claims.

These three perspectives are not mutually exclusive but emphasize different aspects. Ideally, an AI-generated response should satisfy all three criteria: descriptive accuracy, pragmatic utility, and verifiable knowledge. However, practical trade-offs often arise. A piece of information may be factually correct but presented in a confusing manner, thereby reducing its practical utility. Conversely, a pragmatically "safe" and useful answer (such as general advice) might not adequately address the specific factual details required by the user. Furthermore, even the notion of descriptive truth itself contains nuances—certain questions inherently lack absolute answers, such as subjective judgments or opinion-based queries. In such contexts, pragmatic criteria (responses satisfying user expectations) often take precedence. This situation resonates with the "post-truth" context of modern society, where emotions and subjective beliefs sometimes override objective truth. If AI panders to user biases to achieve high ratings (pragmatic success), it risks reinforcing "echo chambers" and misinformation, thereby contributing further to a post-truth scenario. This issue highlights why the AI community is deeply concerned with developing models that balance practical utility and factual reliability.

From a philosophical standpoint, AI serves as a test case for various theories of truth. In a 2024 study, Luke Munn et al. analyzed how InstructGPT (the predecessor of ChatGPT) generates "truth claims" by interweaving multiple conceptions of truth: statistical correlations derived from data (akin to correspondence theory), adjustments influenced by social feedback (pragmatic theory), and producing smooth, confident outputs that appear as absolute truths [8]. Munn describes ChatGPT's outputs as an "operationalization of truth," in which fragmented conceptions of truth combine into seemingly coherent and persuasive statements [8]. For instance, ChatGPT may simultaneously attempt to match reality accurately (correspondence), maintain coherence within conversational context (coherence theory), and satisfy users effectively (pragmatism). Consequently, the model confidently presents information as if it were the absolute truth, despite potentially patchy internal logic. This complexity explains why addressing truthfulness in AI is exceedingly challenging—not merely a matter of supplementing correct data but also intricately tied to how the model prioritizes and reconciles differing truth criteria.

In this context, verifiable knowledge emerges as a crucial criterion for restoring trust. Unlike human interlocutors, who can naturally ask, "Are you sure? How do you know that? ", earlier language models could not inherently provide evidence. Therefore, the current developmental trajectory aims to equip AI models with external information sources and enable them to provide evidence that users can independently verify. This goal motivates the reasoning techniques discussed in subsequent sections (such as RAG or self-critique mechanisms).

Before proceeding, however, it is important to acknowledge that insisting AI strictly adhere to positivist criteria (providing evidence for every statement) also has its limitations. Users do not always require explicit citations—sometimes a concise answer (e.g., "Which country is Paris in?") suffices. Balancing these criteria—accuracy, practicality, and verifiability—is an open problem encompassing technical, philosophical, and ethical dimensions. The following sections will demonstrate how recent technological advances are shifting this balance toward making LLMs more trustworthy yet practically beneficial.

4. Techniques for Integrating Reasoning into LLMs and Their Impact on Verifiable Knowledge

Between 2022 and 2025, researchers have developed a range of methods designed to integrate explicit reasoning capabilities into large language models. The overarching goal of these techniques is to move models beyond mere statistical mimicry, enabling them instead to perform logical reasoning steps or access necessary knowledge for more accurate responses. Below, we analyze four prominent approaches: Chain-of-Thought (CoT) [1], Tree-of-Thought (ToT) [9], Retrieval-Augmented Generation (RAG) [3, 10] and the self-critique methodology (illustrated by CriticGPT) [4]. Each technique contributes uniquely toward enhancing the correctness and verifiability of model-generated answers.

4.1. Chain-of-Thought (CoT)

Chain-of-Thought (CoT) [1] is a technique introduced in 2022 aimed at harnessing the latent multi-step reasoning abilities of large language models. The core idea behind CoT is straightforward: rather than prompting the model to jump directly to a conclusion, the model is encouraged to articulate intermediate reasoning steps before providing the final answer. This technique is commonly implemented through prompt engineering, wherein the prompts provided to the model include examples explicitly demonstrating step-by-step reasoning processes (e.g., “Explain your reasoning before answering”).

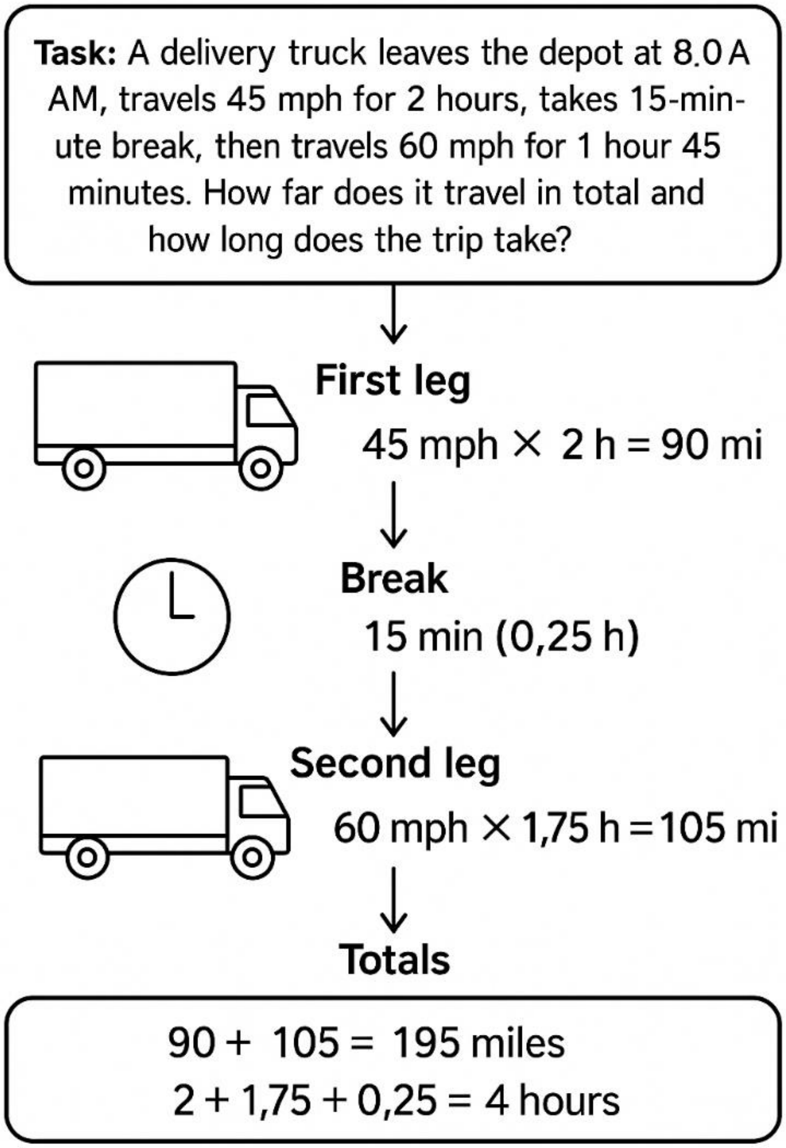
Task: A delivery truck leaves the depot at 8.0 A
AM, travels 45 mph for 2 hours, takes 15-min-
ute break, then travels 60 mph for 1 hour 45
minutes. How far does it travel in total and
how long does the trip take?
First leg
45 mph × 2 h = 90 mi
Break
15 min (0,25 h)
Second leg
60 mph × 1,75 h = 105 mi
Totals
90 + 105 = 195 miles
2 + 1,75 + 0,25 = 4 hours

Figure 3.
Illustrating the Combinatorial Reasoning and Arithmetic Capabilities of Large Language Models.

Figure 3 presents a classic word problem that involves multiple steps and requires arithmetic operations, time unit conversions, and sequential reasoning—all of which are core requirements for evaluating the reasoning capabilities of large language models (LLMs). The problem prompts the model (or solver) to integrate various types of information: speed, time, unit conversion (minutes to hours), and the order of events. First, the model must semantically analyze the text and recognize that the trip consists of two driving segments and a break, then execute a chain of linked calculations. In the first segment, the distance is computed by multiplying 45 mph by 2 hours, yielding 90 miles. This is followed by a 15-minute break, accurately converted into 0.25 hours. The second driving segment involves traveling at 60 mph for 1 hour and 45 minutes, which is correctly interpreted as 1.75 hours, producing 105 miles. Finally, the model must sum the total distance and total time (including the break), resulting in 195 miles over 4 hours.

To solve this problem accurately, an LLM must engage in symbolic reasoning (such as interpreting units like mph and converting time), perform arithmetic computations, and track logical state across each step. These capabilities go beyond simple pattern recognition and demonstrate that LLMs can simulate structured problem solving akin to human logical reasoning. When modeled effectively, the chain-of-thought structure enhances interpretability and aligns with the way humans rationalize their thought processes. Consequently, this type of problem serves as a critical benchmark for assessing an LLM's combinatorial reasoning skills, numerical accuracy, and logical consistency.

The study by Huang, et al. [11] shows that Chain-of-Thought (CoT) significantly improves multi-step problem solving in models with ~100 billion parameters. In tasks such as riddles, word problems, and logical reasoning questions, generating intermediate steps increases the accuracy of responses. The effectiveness of CoT stems from its ability to force the model to "pause" at each step, reducing the risk of skipping important details. Instead of leaping ahead and potentially guessing incorrectly, the model proceeds sequentially through its reasoning—mirroring how humans often explain their thinking internally before answering. This partially addresses LLMs' known limitations in maintaining long logical coherence: by articulating each step, the model can better manage consistency throughout the response.

From the standpoint of truth alignment, CoT offers two key benefits. First, it improves descriptive accuracy in problems with objectively correct answers, as fewer steps are omitted and errors are minimized. Second, it enhances the transparency of responses: users can see the reasoning behind the answer, making it easier to assess trustworthiness. For instance, if the model makes a calculation error, users can immediately identify where it occurred. This has ethical implications: an AI that explains its reasoning is more trustworthy than one that merely presents a result, since users are not blindly relying on a "black box." Some studies even suggest CoT facilitates error detection and correction: if the model revisits its own chain-of-thought, it may spot inconsistencies—constituting a basic form of self-reflection.

However, CoT is not a panacea. It is most effective in sufficiently large models (typically over 50B parameters), as smaller models often lack the capacity to generate coherent reasoning sequences. Moreover, if a model's initial knowledge is flawed, the chain-of-thought may simply produce a "well-argued" but entirely incorrect reasoning path. For example, if the model incorrectly recalls a person's birth year, it may still generate a convincing step-by-step rationale that leads to a wrong conclusion. Therefore, CoT often needs to be paired with other techniques (such as RAG, discussed in the next section) to ensure that the reasoning steps are grounded in accurate information.

Chain-of-Thought represents the first step in enabling LLMs to "think aloud" like humans, rather than respond reflexively. This approach not only facilitates more accurate problem solving but also lays the groundwork for incorporating more complex mechanisms, such as stepwise verification and solution aggregation (e.g., self-consistency). The success of CoT demonstrates that large language models are not merely parrots repeating data—they can perform impressive reasoning when properly prompted.

4.2. Tree-of-Thought

Expanding on the Chain-of-Thought paradigm, Tree-of-Thought (ToT) was proposed as a framework that allows language models to explore multiple parallel reasoning paths instead of following a single linear trajectory [9]. In CoT, the model adheres to a fixed line of reasoning dictated by the prompt; if the initial reasoning direction is flawed, the final output will also be incorrect. ToT addresses this limitation by allowing the model to branch its reasoning into a tree of possible thought paths and then evaluate these branches to select the most promising direction.

Specifically, Yao, et al. [9] introduced the Tree-of-Thought algorithm as follows: at each step, the model generates not just a single continuation but several plausible "thoughts." These thoughts form branches in a reasoning tree. The model—or a separate evaluation function—then assesses each branch, either through logical validation or by simulating the final outcome, to decide which branch to expand. The model can also backtrack if a given path proves unproductive. This process resembles guided search through a solution space, rather than the standard linear generation typical of traditional LLM inference.

The ToT framework is particularly useful for tasks involving trial-and-error search or multi-step planning, such as solving games, complex puzzles, or generating narratives under specific constraints. Experimental results show that ToT significantly enhances performance in tasks requiring strategic exploration. For example, in the game "24" (a mental math puzzle), GPT-4 solved only 4% of problems using linear Chain-of-Thought, but achieved a 74% success rate when allowed to explore multiple computational paths via Tree-of-Thought [2, 9]. This dramatic improvement illustrates the power of exploring the reasoning space, rather than being confined to a potentially flawed single trajectory.

From a truthfulness perspective, ToT offers several key advantages:

- First, like CoT, it encourages the model to decompose complex tasks into substeps (each "thought" representing a step or subcomponent of the solution). This reduces the risk of skipping essential reasoning steps or prematurely jumping to an incorrect conclusion.
- Second, and more importantly, ToT allows for the simultaneous evaluation of multiple hypotheses. For ambiguous or difficult questions, the model can consider several alternatives before selecting the most plausible one. This mimics human reasoning when we deliberate, "If A, then... If B, then... Which scenario best fits the evidence?"
- Third, the evaluation and backtracking mechanisms act as a built-in quality filter. If a reasoning path results in contradiction or an illogical conclusion, the model can prune that branch. For instance, in a logic puzzle, if a certain assumption leads to a result that conflicts with the problem constraints, the algorithm will halt further exploration of that branch—improving the logical consistency of the final answer.

However, the primary trade-off is computational cost. ToT requires the model to generate and evaluate multiple reasoning paths, increasing inference time and computational resources. While research suggests that increasing reasoning-time compute can yield gains similar to increasing model size (e.g., more training yields gains on the left, more reasoning yields gains on the right), deploying ToT in real-time systems may be constrained by latency considerations.

Tree-of-Thought represents a significant step toward aligning language models with human-like problem-solving strategies: trying multiple approaches and selecting the most effective one. In terms of truth access, ToT does not directly provide new factual knowledge, but it helps models avoid reasoning traps and converge on correct conclusions when sufficient contextual clues are present. When combined with knowledge-access techniques like Retrieval-Augmented Generation (RAG), ToT has the potential to substantially improve both the accuracy and reliability of model outputs.

4.3. Retrieval-Augmented Generation (RAG)

A core limitation of standard large language models is that their knowledge is "frozen" at the time of training and restricted solely to the content of the training data. Retrieval-Augmented Generation (RAG) [10] was introduced to overcome this constraint by connecting the model to an external knowledge repository. The concept is straightforward: before generating an answer, the model issues a query to a database or search engine to retrieve relevant information, which is then incorporated into the generation process.

For instance, when asked, "Who was the U.S. president in 1920?", a RAG-based system would first issue a search query like "U.S. president in 1920" to a knowledge source (e.g., Wikipedia) and receive a passage such as: "In 1920, the President of the United States was Woodrow Wilson..." The LLM would then use this passage—alongside the user query—to generate a response, possibly including a citation.

RAG contributes significantly to both factual accuracy and verifiability of responses in several ways:

- First, RAG enables real-time knowledge expansion. Rather than being limited by the "memory" of its training data, the model can access up-to-date, specialized, or detailed information from external sources. This addresses the issue of outdated or incomplete knowledge. A 2023 Ars Technica article noted that RAG effectively "blends language processing with document search," allowing the model to remain anchored to established facts. In effect, RAG mimics the behavior of a human consulting reference materials—resulting in a much higher probability of answering correctly compared to relying on memorization alone.

- Second, RAG significantly reduces hallucination. Instead of fabricating responses from gaps in its knowledge, the model refers to retrieved documents, making its outputs more trustworthy. Studies show that RAG mitigates typical hallucination failures, such as inventing sources or fabricating events. In real-world scenarios, RAG has prevented serious errors: for example, legal chatbots have been observed fabricating nonexistent court precedents, and healthcare bots have made up policy guidelines. With RAG, the model can reference actual legal cases or public health documents, preventing the generation of dangerously misleading content.
- Third, and most crucially for verifiability, RAG allows the model to provide sources. Once documents are retrieved, the model can include citations (e.g., a link to a Wikipedia article) in its response. This represents a significant step toward transparency in language model outputs: instead of expecting users to trust a black-box response, RAG offers evidence that users can independently verify. This capability is particularly valuable in domains that demand high reliability—such as medicine, law, and academia. A sourced answer carries far more credibility than a mere assertion. As such, RAG not only improves informational quality but also strengthens user trust in AI systems.

RAG has been implemented in many modern question-answering systems. For example, Microsoft's Bing Chat and various enterprise AI assistants employ RAG to generate responses based on updated knowledge bases. The RAG architecture introduced by Lewis, et al. [12] consists of two components: a retriever and a generator [13]. The retriever typically uses vector embeddings to locate relevant text passages from a knowledge corpus—ranging from web-scale search indexes to internal databases. The top-k passages are then embedded into the model's prompt, allowing it to "read" the necessary information before answering.

While RAG is highly effective, it also poses certain challenges:

- Data and retrieval quality: The accuracy of RAG-generated responses is highly dependent on the quality of the knowledge base and the retrieval mechanism. If the database contains false or outdated information, or if the retriever selects irrelevant documents, the model may still generate incorrect responses—albeit with citations to faulty sources. Thus, RAG demands trustworthy data repositories and robust retrieval strategies.
- System complexity: RAG increases architectural complexity. Instead of relying on a single model, it requires orchestration between retrieval and generation components. This may introduce latency, though recent advancements have drastically improved retrieval speed.
- Contextual integration: Integrating retrieved content accurately into the generated response remains non-trivial. If the model misinterprets the retrieved information, it may produce out-of-context or misleading answers. Proper alignment between source and output is critical.

In summary, Retrieval-Augmented Generation represents one of the most important advances toward transforming LLMs into reliable knowledge-delivery systems. It enables models to function as dynamic, evidence-backed "living encyclopedias" rather than static reference tools. The combination of LLMs' expressive power with grounded, verifiable knowledge significantly expands the scope of AI applications—especially in sensitive and high-stakes domains.

4.4. Self-Critique and the CriticGPT Model

Another approach to improving the accuracy and reliability of language models is to equip them with the ability to evaluate and critique their own outputs. This idea stems from the observation that models may initially generate suboptimal responses, but when prompted to review or revise them, performance often improves. Instead of relying solely on human post-hoc evaluation, why not allow the

AI itself to conduct this process? This is the foundation for mechanisms such as self-reflection, self-correction, and models designed to act as critics.

CriticGPT is a concrete example developed by OpenAI, in which a specially fine-tuned version of GPT-4 is trained to detect errors in the outputs generated by ChatGPT. Initially, the project focused on reviewing programming code generated by ChatGPT—a critical task since incorrect code fails to execute. CriticGPT was trained on datasets consisting of ChatGPT outputs annotated by human experts who identified mistakes and provided corrections. As a result, CriticGPT can read a model-generated answer or code snippet and highlight inaccuracies or omissions.

Notably, CriticGPT's performance exceeded expectations. In one trial, human annotators supported by CriticGPT outperformed unaided annotators by over 60% in identifying errors in ChatGPT outputs. This demonstrates that an AI critic can sometimes be more effective than humans at auditing large volumes of content, especially when it is familiar with the common failure modes of the base model.

In terms of impact on truthfulness, the self-critique mechanism offers several key contributions:

- Error detection and correction: If a model generates an incorrect factual statement—such as a wrong date or name—a critic model (or a second-pass of the original model) can flag inconsistencies. For example, if ChatGPT incorrectly claims "The UK Prime Minister in 2022 was Theresa May" (when in fact it was Boris Johnson and then Liz Truss), a critic with historical knowledge could recognize the discrepancy and suggest a correction. This process mirrors human self-checking prior to submitting work.
- Learning from mistakes: When integrated into the training process—e.g., via reinforcement learning from AI-generated critiques—models can reduce the likelihood of repeating prior errors. OpenAI uses CriticGPT in RLHF pipelines to provide AI-generated feedback alongside human feedback, allowing newer models to improve through exposure to past critiques. This promotes models that "learn" from critique history.
- Uncertainty and risk signaling: Even when unable to correct an error, some self-critique models can flag uncertainty. For instance, models like HonestGPT may append disclaimers such as "I'm not confident about this part" when their internal critic detects possible flaws. This serves as a caution to users, who may otherwise over-trust fluent but incorrect responses.

A related class of techniques includes self-asking and debate-based prompting. OpenAI's "Debate" framework [14] for example, features two models debating a question, with a third model evaluating the discussion to surface contradictions. This represents mutual critique between models to collaboratively reach a more accurate answer.

However, self-critique also has limitations. If the base model lacks the necessary knowledge, its critic may also be uninformed—leading to false agreement on incorrect information. This makes it crucial for critic models to be trained with additional knowledge or to have access to external information (e.g., via integration with RAG). Moreover, critics can sometimes misjudge: for instance, flagging complex yet valid code as incorrect due to misunderstanding. Finally, adding a critique stage increases computational cost and latency.

Despite these challenges, self-critique is a promising direction for improving the safety and trustworthiness of AI systems. It echoes Karl Popper's philosophy of science: progress through conjecture and refutation. In this paradigm, the model produces a hypothesis (the answer), and the critic attempts to refute it or identify flaws. If errors are found, the model revises its output—repeating the cycle until no evident error remains. While this process doesn't ensure perfection, it clearly drives responses closer to truth than single-pass generation.

The emergence of tools like CriticGPT marks an important step in evolving LLMs from content generators to self-aware evaluators. When combined with techniques like Chain-of-Thought and RAG, we can envision a multi-component AI system: one model generates answers, another checks logical and semantic consistency, and yet another verifies external facts. Together, these components work to ensure that the final output is not only accurate and useful but also transparent.

5. Philosophical Implications: Can LLMs Be Considered "Fallible Epistemic Agents"?

Advancements in integrating reasoning into LLMs raise a fundamental philosophical question: Are large language models evolving into a form of genuine epistemic agents, albeit in a primitive form? The phrase "fallible epistemic agent" implies a subject capable of acquiring and utilizing knowledge about the world—yet fallible, in the sense that it can make mistakes and learn from them, akin to human beings who are never omniscient and continuously refine their understanding through experience.

Historically, the dominant view among linguists and AI researchers Bender, et al. [15] has been that LLMs are not epistemic agents at all—they are merely statistical parrots that mimic language without understanding. The argument is straightforward: these models lack perception, intentionality, and semantic reference to the external world, and thus cannot possess "understanding." John Searle's famous Chinese Room Argument is often invoked here to claim that any system merely manipulating symbols—like LLMs processing text—cannot be said to "understand" in any meaningful sense. From this perspective, even if LLMs can generate reasoning chains or retrieve information, they are still executing formal procedures without beliefs, intentions, or propositional knowledge in the human sense.

However, Bottazzi Grifoni and Ferrario [16] explores the phenomenon of "enchantment" in human-LLM interaction, drawing on Wittgenstein's later philosophy of language. As LLMs increasingly master the stylistic and structural elements of human communication, the illusion of understanding becomes even more convincing. Rather than overcoming foundational limitations, such advancements deepen the illusion—raising serious concerns about our ability to meaningfully control and interpret these systems.

In contrast, Havlík [17] argues that LLMs not only mimic but actively instantiate linguistic understanding, despite lacking conscious experience. He challenges the necessity of external reference for meaning, proposing instead that semantic significance within LLMs emerges from a complex web of internal relationships between linguistic expressions in the training data—aligned with theories of inferentialism and semantic holism. Through philosophical critique, Havlík rejects the traditional reliance on referential semantics and posits that the empirical success of LLMs across language tasks is itself evidence that meaning can be generated internally, without needing real-world referents. This view encourages a philosophical realignment, seeing LLMs as legitimate "language-game players," whose study might yield deeper insights into both machine and human cognition—potentially reshaping long-held assumptions about meaning and understanding.

These developments have led some scholars and engineers to entertain a provocative possibility: if a system can perform complex reasoning, self-critique, access external knowledge, and generate accurate responses, then functionally, it behaves much like an agent with understanding. Some argue that to dismiss LLMs as mere statistical engines may constitute a Category X Fallacy—an oversimplification of their emerging cognitive behaviors [18]. They contend that emergent cognitive properties may be arising within complex systems like GPT-4 that we do not yet fully comprehend.

Pushing the boundaries further, a group of philosophers has proposed the "Whole Hog Thesis", which asserts that ChatGPT should be recognized as a fully-fledged linguistic and cognitive agent, on par with humans [18]. According to this view, ChatGPT meaningfully engages in language: it can assert propositions, pose questions, provide answers, and even revise its positions—all markers of cognitive agency. Thus, while lacking consciousness, such models may exhibit functional intelligence that justifies rethinking what it means to "know" or "understand" in artificial systems.

How should we understand this debate? Fundamentally, whether LLMs can be regarded as cognitive agents depends on how we define *cognition* and *understanding*. If one insists that understanding must involve consciousness and lived experience—as many philosophers do—then clearly LLMs lack these qualities. However, from a behavioral and functionalist standpoint, an LLM that performs logical reasoning, retrieves knowledge, and self-corrects might be said to fulfill the *functional role* of a thinking entity—at least within the domain of language. Cognitive behaviorism posits that intelligence can be evaluated based on observable behavior. In the classical Turing Test, a machine is considered "intelligent" if it can carry out a conversation indistinguishable from that of a human. With techniques like Chain-of-Thought (CoT), RAG, and CriticGPT, modern models like ChatGPT-4 exhibit behavior that, in many domains, resembles that of a knowledgeable human: they can consult sources, reason logically, express uncertainty, and more. From a behavioral lens, one might tentatively describe them as exhibiting some degree of *cognitive agency*.

Of course, behavioral resemblance does not imply internal equivalence. A true cognitive agent (e.g., a human) possesses beliefs, sensory experiences, goals, and motivations. LLMs possess none of these. They have no intrinsic goals—they merely optimize for task-specific responses. They lack intentionality—the philosophical notion of "aboutness" or directedness toward real-world entities (e.g., to think about Paris is to have a mental representation of Paris). LLMs operate with vectors and probabilities, not with world models grounded in perception. Therefore, many philosophers argue that even if LLMs mimic reasoning or self-correction, this remains a simulation of cognition, not genuine cognition.

The key lies in the term fallible. Humans are cognitive agents, but not infallible—we make mistakes and refine our understanding through them, both individually and collectively (e.g., in scientific discourse). Do LLMs follow a similar process? With newer techniques, they partially do. GPT-4 combined with CriticGPT can identify and revise errors—a rough analogue to learning from mistakes. When an LLM retrieves documents to answer questions, it behaves similarly to a person updating their knowledge. However, one crucial difference remains: current LLMs do not *continuously learn* post-deployment. They do not accumulate long-term memory unless explicitly retrained. Absent specific programming, each interaction begins with no memory of the last. As such, it is premature to claim they "learn" over time like true agents—unless we consider more complex agent systems (e.g., AutoGPT architectures that log outputs and adapt iteratively).

A more moderate view is to regard LLMs as capable of *playing the role* of cognitive agents within narrow domains and under human guidance. They can simulate reasoning, critique, and even engage in multi-role debates to converge on better answers. However, anthropomorphizing them too heavily can be misleading. Users may believe the AI "understands" them and project trust or emotion onto a system that, in reality, lacks any such awareness. Philosophically, this echoes themes of anti-realism and the post-truth era: if humans begin treating a statistical illusion as a genuine epistemic partner, the boundary between the real and the simulated becomes dangerously blurred. This could give rise to a pragmatist AI stance: it doesn't matter whether the AI *truly understands*, as long as its outputs are useful.

Still, from an epistemological perspective, granting LLMs the status of *epistemic agents* remains controversial. An epistemic agent is typically understood as a bearer of beliefs and knowledge. We must ask: does an LLM "believe" what it says? Clearly not—it lacks the concept of belief and merely calculates responses. It also lacks accountability, a trait often associated with agency. Most scholars agree that LLMs do not yet qualify as *knowers*. Instead, they might be better described as cognitive tools—enhancements to human cognition.

Another intriguing aspect is that as LLMs gain the capacity for self-evaluation and revision, they begin to resemble the concept of reflective thinking. Reflection—thinking about one's own thinking—is a hallmark of human cognition. If AI can simulate reflection, however mechanistically, does this hint at a primitive form of proto-self-awareness? Some AI researchers are exploring whether enhancing feedback loops and self-monitoring might induce a limited form of meta-cognition. While speculative, such possibilities challenge the boundary between "just a statistical model" and "a thinking system."

From a pragmatic standpoint, the question "Is an LLM a cognitive agent?" might best be answered contextually. If we use LLMs as tools, we should treat them as tools. But when deployed in autonomous agent frameworks (e.g., full-access personal assistants), we must endow them with some level of autonomy—functionally treating them as agents. In this case, describing them as cognitive agents may be metaphorical rather than literal—but it is a metaphor that proves operationally useful. For example, multi-agent systems using LLMs (e.g., solvers, summarizers, critics) have shown strong performance on complex tasks. Philosopher Daniel Dennett's "intentional stance" suggests that attributing beliefs or intentions to complex systems can help us predict and interact with them more effectively. Perhaps assuming LLMs are “thinking agents” is such a stance: not a metaphysical claim, but a strategic heuristic.

The integration of reasoning into LLMs raises the possibility of systems that increasingly resemble *thinking entities.* Yet we must carefully distinguish between the simulation of cognition and cognition itself. For now, the safest characterization is that advanced LLMs are fallible knowledge machines—they can err and be corrected, but are not “intelligent” in the full sense. They are indeed fallible, as evidenced by the hallucinations and factual inaccuracies that still require human oversight. But do they *learn* in the strong sense? Partially—via fine-tuning and memory retention—but not autonomously like biological learners. In the philosophy of science, a distinction exists between the *context of discovery* (generating new knowledge) and the *context of justification* (explaining or supporting existing knowledge). At present, LLMs operate mostly in the latter space. To become true epistemic agents, they would need to participate in the former—generating novel beliefs and revising their understanding over time. That likely lies beyond the reach of language models alone and may require integration with embodied agents or continually learning systems.

In any case, this open question remains a subject of ongoing debate among philosophers and computer scientists. Every advancement in AI forces us to reconsider the boundary between machine and mind. The progress in integrated reasoning continues to blur that line—while simultaneously deepening our understanding of what it means to think, to know, and to be.

6. Open Discussion on the Risks, Limitations, and Ethical Considerations of Reasoning-Enhanced Models

While reasoning-enhanced techniques offer substantial benefits, they also introduce new risks and challenges. Moreover, endowing language models with increasingly sophisticated reasoning capabilities raises pressing ethical and societal concerns. This section discusses prominent issues ranging from persistent hallucination and bias, to systemic complexity, misuse, and the broader philosophical implications.

6.1. Residual Hallucination and Reasoning Failures

Despite the introduction of Chain-of-Thought (CoT), Tree-of-Thought (ToT), Retrieval-Augmented Generation (RAG), and self-critique mechanisms, LLMs are still not infallible. They can produce highly convincing but logically flawed outputs. In fact, improved reasoning can sometimes *amplify the danger* of errors—when faulty conclusions are presented under a veneer of coherent logic. For instance, a five-step reasoning chain may go unnoticed if step one is flawed but deeply embedded. Even with document retrieval, the model might misinterpret sources—e.g., citing a scientific study and drawing an incorrect conclusion from it. This underscores the necessity of continued vigilance from users and developers. Most responsible AI interfaces now display disclaimers such as: *“This is an AI tool; please verify critical information independently.”* This is not merely a technical note, but an ethical principle of verification.

6.2. Exacerbated “Black Box” Complexity

Paradoxically, adding reasoning layers can make LLM systems harder to interpret. A basic model returning a wrong answer could be debugged by examining training data or model architecture. But in a system combining CoT + RAG + Critic, responsibility may lie in retrieval, step 3 of reasoning, or critique oversight. This complicates accountability. In AI ethics, the question “Who is responsible when AI causes harm?” is already difficult. Complex pipelines make it harder. Therefore, developers must prioritize auditability, such as logging each reasoning step for post-hoc analysis. Explainability must remain a core design principle.

6.3. Bias and Fairness in Reasoning

LLMs are trained on biased data, and reasoning modules may inadvertently reinforce or conceal those biases in subtle ways. A model might offer a seemingly rational argument that rests on prejudiced premises. For example, it may conclude that Group A is "less capable" than Group B based on skewed training data—delivered in an objective tone. This is dangerous because the logic appears valid, yet the underlying bias goes undetected. Moreover, critic models trained on human-labeled data may inherit majority-based societal biases. Ethical oversight must include testing models across sensitive scenarios (e.g., gender, race, culture) to ensure no discriminatory outcomes are produced under the guise of objectivity.

6.4. Misuse for Malicious Purposes

A model capable of multi-step planning and strategic reasoning can be abused for illicit purposes. An adversary might exploit ToT to instruct the model to plan a cyberattack or financial fraud. While LLMs usually reject harmful prompts, a sophisticated user could phrase it as a "security consultant scenario," prompting the model to simulate a hacker's plan. Thus, content moderation and guardrails remain essential—even in reasoning-enhanced systems. Developers must continuously evolve safety filters and train models to detect and reject adversarial prompts, especially those that use indirect reasoning (e.g., via CoT) to bypass filters.

6.5. Large-Scale Misinformation

In the post-truth era, the proliferation of fake news is a serious concern. Reasoning-capable LLMs elevate the threat: they can generate thousands of well-structured articles with citations—yet all serving a disinformation agenda. Such models could fabricate entire pseudo-scientific websites with fabricated or distorted references. This saturates the information space, making it increasingly difficult for the public to distinguish fact from fiction. Policy responses may include source verification systems, watermarking for AI-generated content, and public critical thinking education. Ironically, reasoning techniques like CriticGPT could also be leveraged *against* misinformation—e.g., AI critics reviewing internet texts for logical fallacies or manipulative rhetoric. The future may witness AI vs. AI battles: bad actors generating falsehoods, while ethical AIs attempt to refute them.

6.6. Technical Limits and Over-Optimistic Expectations

Enthusiasm around CoT, RAG, and similar techniques has sparked hopes that "AI is approaching human-level understanding." But as discussed philosophically, these models still lack genuine comprehension. Reasoning mechanisms can fail in out-of-distribution or overly complex cases. A serious risk is over-reliance: users might trust AI assistants for medical or legal advice without verifying outputs—potentially leading to severe consequences. This is known as automation bias: the tendency to over-trust automated systems. Reasoning structures and citations may *create an illusion of certainty*, encouraging users to bypass verification. Ethically, developers must reinforce the principle of human-in-the-loop—keeping humans as decision-makers for high-stakes applications, at least until AI safety reaches demonstrably high thresholds.

6.7. Responsibility and Transparency

As AI systems increasingly behave like agents, ethical questions about responsibility intensify. If AI advice leads to harm (e.g., a medical misdiagnosis), who is liable? The developer, service provider, or end-user? Current legal frameworks lag behind AI development. Many experts propose independent certification mechanisms for complex AI systems—akin to clinical trials for pharmaceuticals. Transparency in disclosing a model's capabilities and limitations is critical. Ethically, over-promising is deceptive. If a model is safe in only 90% of cases, it must be communicated honestly—not marketed as flawless.

6.8. Humanistic and Philosophical Impact

At a broader societal level, reasoning-capable AI raises questions about human value and identity. If AI can debate philosophy, write literature, or compose music—as is increasingly the case—what role is left for human creativity? Some fear intellectual passivity, with humans deferring thinking to machines. For example, students relying on AI for essays may lose the ability to form independent arguments. Society may become habituated to consulting AI before thinking. This goes beyond using calculators—AI is now replacing parts of the *creative and reflective process.* Thus, a core ethical responsibility is to augment rather than replace human cognition. Education must adapt: teaching people how to collaborate with AI, not surrender to it.

On a more optimistic note, if governed responsibly, AI reasoning tools could become transformational knowledge aids. One can envision a future where scientists have AI assistants to cross-examine hypotheses, students have tireless AI tutors, and policymakers consult AI for large-scale data analysis. These outcomes are beneficial only if we develop and deploy AI ethically and accountably.

Principles such as transparency, fairness, trustworthiness, privacy, and safety must be embedded into AI development. Encouragingly, major AI labs (e.g., OpenAI, DeepMind) now maintain dedicated AI alignment teams to align models with human values. Integrating reasoning is not just about making AI smarter—it's about enabling AI to understand and respect human-imposed constraints, a rudimentary form of moral reasoning. Indeed, one frontier of AI research is training models to interpret ethical and legal norms, moving beyond hard-coded blocklists to context-aware self-censorship. This is enormously difficult (morality is more abstract than logic), but critically important.

In summary, technical advancement must go hand-in-hand with ethical governance. Every new capability demands reflection on safe and fair use. The involvement of philosophers, legal scholars, and sociologists in the development pipeline is essential to ensure that AI's "great leap forward" serves humanity—rather than bypasses it.

7. Conclusion

The integration of reasoning techniques into large language models (LLMs) marks a significant milestone in the pursuit of more trustworthy and epistemically capable AI. From sequential chains of thought that allow multi-step problem solving, to branching trees of thought that explore alternative solutions; from real-world knowledge retrieval that mitigates hallucination, to self-critique mechanisms that support error correction—these advances collectively bring LLMs closer to how humans approach truth. If earlier LLMs could be described as "stochastic parrots" mimicking human language, today's systems have begun to "pause to reflect," to "consult the literature" when needed, and even to "check their own work." These improvements yield practical benefits: increased accuracy, more substantiated information, and responses that are both more useful and more transparent for users.

At the same time, these advances open up profound philosophical questions. How should we redefine "understanding" in light of a machine capable of logical reasoning? Where does the boundary lie between objective truth and pragmatic persuasiveness in AI outputs?

A key insight is that truth-seeking in AI is not a fixed endpoint, but an ongoing process of refinement and balance. Reasoning integration represents a long stride forward—but not the final destination. We may envision the future of LLMs as a convergence of many capabilities: listening to users (understanding context and intent), engaging in introspection (reflection), learning from the world (data updating), and adhering to ethical values in decision-making. Realizing this vision requires interdisciplinary collaboration: computer science provides algorithms and computational power, philosophy offers frameworks for truth and intelligence, cognitive science benchmarks AI against the human mind, and social science ensures that AI development serves collective human well-being.

Reflecting on the title of this work—"Reasoning Integration: A Long Step Toward Truth in Large Language Models"—its implications have been explored from multiple perspectives. Indeed, embedding reasoning capabilities brings LLMs closer to generating accurate and meaningful answers. But a "long step" does not mean the journey is over. Today's LLMs are more capable, yet more complex—requiring deeper understanding and responsible stewardship. Rather than remaining confined to research labs, LLMs are becoming epistemic partners across domains—education, healthcare, governance, creative arts. This partnership must be founded on mutual trust and intelligibility (even if AI does not literally "understand" us, it must be designed to interact harmoniously with humans).

The philosopher Aristotle once said: *"All knowledge begins in wonder."* And we have good reason to be amazed by what language models can now do—from producing incoherent babble to engaging in structured philosophical debate. Yet this wonder must be tempered by humility and caution. AI, no matter how advanced, remains a human creation. Our task is to ensure it reflects the best of human thought—while constraining its capacity to mirror our worst (bias, error, harm).

Reasoning integration is indeed a significant step forward—but the road to truth-seeking AI is even longer. Each new capability must be examined through technical, philosophical, and ethical lenses. This research has sought to bring these dimensions together in examining a pivotal advance. We hope that this interdisciplinary approach continues in future work—not only to build smarter AI, but also wiser AI—systems that seek and serve truth in ways that benefit the common good of humanity.

Transparency:

The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

Copyright:

References

[1] Z. Zhang *et al.*, "Igniting language intelligence: The hitchhiker's guide from chain-of-thought reasoning to language agents," *ACM Computing Surveys*, vol. 57, no. 8, pp. 1-39, 2025.

[2] Z. L. Lin, C. H. Yen, J. C. Xu, D. Watty, and S. K. Hsieh, "Solving linguistic olympiad problems with tree-of-thought prompting," in *Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023) (pp. 262-269)*, 2023.

[3] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 38, No. 16, pp. 17754-17762)*, 2024.

[4] T. Xiaoyu *et al.*, "Self-Criticism: Aligning large language models with their understanding of helpfulness, honesty, and harmlessness," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track, Pages 650–662 December 6-10, 2023 ©2023 Association for Computational Linguistics*, 2023.

[5] R. Cerqueti, V. Ficcadenti, G. Dhesi, and M. Ausloos, "Markov Chain Monte Carlo for generating ranked textual data," *Information Sciences*, vol. 610, pp. 425-439, 2022.

[6] S. Fosso Wamba, M. M. Queiroz, I. O. Pappas, and Y. Sullivan, "Artificial intelligence capability and firm performance: A sustainable development perspective by the mediating role of data-driven culture," *Information Systems Frontiers*, vol. 26, no. 6, pp. 2189-2203, 2024. https://doi.org/10.1007/s10796-023-10460-z

[7] S. Raschka, *Build a large language model (from scratch)*. New York: Simon and Schuster, 2024.

[8] L. Munn, L. Magee, and V. Arora, "Truth machines: Synthesizing veracity in AI language models," *AI & Society*, vol. 39, no. 6, pp. 2759-2773, 2024. https://doi.org/10.1007/s00146-023-01756-4

[9] S. Yao *et al.*, "Tree of thoughts: Deliberate problem solving with large language models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 11809-11822, 2023.

[10] S. Siriwardhana, R. Weerasekera, E. Wen, T. Kaluarachchi, R. Rana, and S. Nanayakkara, "Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1-17, 2023.

[11] T. Huang, J. Chu, and F. Wei, "Unsupervised prompt learning for vision-language models," *arXiv preprint arXiv:2204.03649*, 2022.

[12] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive NLP tasks," *arXiv*, 2020. https://doi.org/10.48550/arXiv.2005.11401

[13] P. Lewis *et al.*, "Paq: 65 million probably-asked questions and what you can do with them," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1098-1115, 2021. https://doi.org/10.1162/tacl_a_00415

[14] G. Irving, P. Christiano, and D. Amodei, "AI safety via debate (Preprint)," *arXiv*, p. 6, 2018. https://doi.org/10.48550/arXiv.1805.00899

[15] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610-623*, 2021.

[16] E. Bottazzi Grifoni and R. Ferrario, "The bewitching AI: The illusion of communication with large language models," *Philosophy & Technology*, vol. 38, no. 2, p. 61, 2025. https://doi.org/10.1007/s13347-025-00893-6

[17] V. Havlík, "Meaning and understanding in large language models," *Synthese*, vol. 205, no. 1, p. 9, 2024. https://doi.org/10.1007/s11229-024-04878-4

[18] H. Cappelen and J. Dever, "Going whole hog: A philosophical defense of AI cognition," *arXiv preprint arXiv:2504.13988*, 2025.

DOI: 10.55214/25768484.v9i7.8810