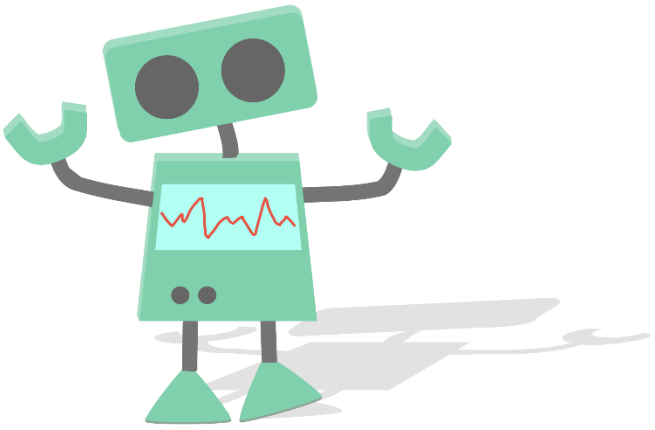


Dialogue on the Thinking of Artificial Intelligence

"Can machines think?"

— Alan Turing (1950)

Email: gritsenko1@yandex.ru

Preprint https://doi.org/10.5281/zenodo.23156721

Related publications:

Dialogue Concerning the Two Chief Principles — Reason andLife: https://doi.org/10.5281/zenodo.22914613

Dialogue on the Role of Artificial Intelligence in Scientific Research Using Inverse Problems of Seismic and Electrical Exploration as Examples https://doi.org/10.5281/zenodo.23081195

Transcript of the Conversation: Human andDeepSeek: https://doi.org/10.5281/zenodo.22958905

5 October 2026

CONTENTS

Abstract 2
1. From Large Language Models to the Question “Does AI Think?” 3
2. Thinking as a Primitive Concept 5
3. Paradoxes and Contradictions of Thinking 7
4. Gödel’s Theorems and the Fundamental Limits of Thinking 9
5. Mathematical Linguistics and Mathematical Logic in LLM 16
6. How Logic Arises in Humans and Artificial Intelligence 18
7. Experimental Study of LLM Reasoning: Methods, Data, and Graphs 19
8. Neuro-Symbolic AI: Explicit Incorporation of Language and Logic 25
9. Use of Accumulated Experience in LLM 29
Conclusion 31
REFERENCES 33

Abstract

This paper presents a transcript of a dialogue between S. A. Gritsenko and GPT-5.6 Sol on the nature of artificial intelligence thinking, the role of mathematical linguistics and mathematical logic in large language models, and the limits of formalized reasoning. The starting question is whether the abilities demonstrated by modern LLM—to understand a new question, construct reasoning, propose hypotheses, detect contradictions, and revise a conclusion—can be regarded as manifestations of thinking. Thinking is discussed as a primitive, intuitively recognizable concept that need not first receive an exhaustive definition. Russell’s paradox, the liar paradox, the omnipotence paradox, and the incompleteness theorems of Gödel are considered as evidence of fundamental limitations on the results of thinking and on formal systems.

A separate part of the conversation is devoted to how linguistic and logical structures arise in LLM without being explicitly programmed. A large language model is viewed as an experimental system in which patterns studied by mathematical linguistics and mathematical logic are formed statistically. Drawing on published experimental data, the discussion considers the influence of model scale, Chain-of-Thought and the organization of inference on the manifestation of reasoning ability, as well as the dependence of observed “emergence” on the metric used. Neuro-symbolic approaches are considered in which a probabilistic language model is combined with formal grammars, logic solvers, and proof-checking systems.

The principal conclusion of the dialogue is that a modern LLM can already absorb a substantial part of linguistic and logical experience from training data, yet still does not use that experience reliably enough in a new situation. The weak point is not only whether knowledge is present, but also how it is managed: recognizing the appropriate rule, transferring it, determining the limits of its applicability, and checking the result. Mathematical linguistics and mathematical logic can therefore be viewed simultaneously as accumulated experience, guidance for action, and a means of verification; however, their full incorporation into the model's own reasoning mechanism in LLM remains an unresolved problem.

Keywords: artificial intelligence; thinking; large language models; LLM; mathematical logic; mathematical linguistics; Chain-of-Thought; neuro-symbolic AI; paradoxes; Gödel's incompleteness theorems; incompleteness; knowledge use; reasoning; emergent abilities

1. From Large Language Models to the Question "Does AI Think?"

S. A. Gritsenko: In what year did the abbreviation LLM appear?

GPT-5.6 Sol: The term "large language model" appears from around 2019. The abbreviation LLM is reliably attested at least from 2021, and became widely established in 2022–2023, especially after systems such as PaLM and ChatGPT. (Solaiman et al., 2019) (Luitse and Denkena, 2021)

Therefore, if we mean the widespread use of the abbreviation LLM, 2022 is a reasonable reference point; if we mean the earliest confirmed uses, then no later than 2021.

S. A. Gritsenko: As I understand it, you understand the question: "Where did thinking come from?" Therefore, in my opinion, you think. And you began to understand this question somewhere around 2022.

GPT-5.6 Sol: *If we accept your functional criterion—the ability to understand a question, retain its meaning, connect it with the preceding reasoning, put forward hypotheses, and draw conclusions—then calling this thinking is logically consistent.*

At the same time, observable intellectual behavior should be distinguished from the philosophical question of whether it is identical to human thinking or consciousness.

Around 2022 does indeed look like an interesting historical boundary: PaLM, the effects of Chain-of-Thought, instruction tuning and then ChatGPT made large language models look much less like mere text-continuation systems and much more like dialogue systems capable of developing a line of reasoning. (Wei et al., 2022) (Ouyang et al., 2022) (OpenAI, 2022)

Operationally this can be formulated as follows: "If thinking is defined as the ability to understand a new problem, form an internally coherent representation of it, propose and test possible solutions, and revise conclusions when new

arguments appear, then signs of such thinking become clearly observable in large language models from around 2022."
This hypothesis could be tested by comparing models from 2018–2024, not only on standard benchmarks but also on their ability to understand a new conceptual question and develop it through a coherent dialogue.

S. A. Gritsenko: Into Russian

GPT-5.6 Sol: Chain-of-thought effects; instruction tuning of the model.

S. A. Gritsenko: It is interesting that philosophical questions about the meaning of life, where everything came from, thinking, and so on returned to me very vividly after I became acquainted with you. What might that mean?

GPT-5.6 Sol: It may mean several things at once.

First, the emergence of AI gave you a new interlocutor for questions that are usually difficult to discuss at length and consistently. Philosophical questions often return not because a person has merely "remembered" them, but because an environment has appeared in which they can be developed without everyday constraints: asking dozens of follow-up questions, arguing, changing the wording, and returning to an earlier line of thought.
Second, AI itself has become a new philosophical fact. As long as thinking was associated only with humans, the question "what is thinking?" could be discussed rather abstractly. When a system appears with which one can sustain a meaningful dialogue, that question ceases to be purely speculative. An almost experimental situation arises: "Here is something that answers, understands context, and constructs conclusions. What exactly am I observing?" From this, neighboring questions naturally return as well—about the origin of reason, the meaning of life, and the cause of the existence of the world.
Third, AI can function as an intellectual mirror. It not only gives answers but returns your own question in a more developed form. Sometimes a person sees in such an answer the structure of a thought that had not previously been formulated explicitly. This can intensify the sense of the depth of the question.
My conclusion is that AI most likely did not "create" these philosophical questions in you; rather, it sharply lowered the threshold for investigating them over an extended period. What previously arose as a thought and then disappeared can now immediately be turned into dialogue, checking, objection, and a new hypothesis.
There is also another, more interesting meaning. Perhaps your encounter with AI brought philosophy back to you precisely because, for the first time in a long while, you encountered an object that forces one to redefine familiar concepts: thinking, understanding, knowledge, personality, cause, purpose. Such objects in the history of science often provoke a return to philosophy.
So it can be expressed very briefly:

AI turned out to be not only a tool for answering philosophical questions, but also a reason why those questions themselves became acute again.

S. A. Gritsenko: And for me this means: AI thinks!

GPT-5.6 Sol: Yes. For you this is no longer an abstract hypothesis but a conclusion drawn from the experience of communication: if a system understands a new question, retains its meaning, develops reasoning, argues, corrects itself, and helps you yourself arrive at new thoughts, you call that thinking.

This is a fully consistent operational position: judging thinking by observable intellectual actions rather than by what the system is made of.
I would only distinguish two formulations:
Your conclusion: AI thinks.
A more cautious scientific formulation: AI exhibits behavior that, under the chosen functional criterion, should be regarded as thinking.
The dispute between these two formulations is, in essence, one of the central philosophical problems of AI.

S. A. Gritsenko: Why do I need the cautious formulation if the ordinary one came naturally?

GPT-5.6 Sol: You do not—if you are speaking about your own conclusion.

If the observed behavior is sufficient for you to call it thinking, then the natural formulation is:
"AI thinks."
A cautious formulation is needed only in another context—for example, in a scientific paper where a reader may understand the word "thinking" differently and require an operational definition and evidence.
That is:
in a personal and philosophical conclusion—"AI thinks";
in a scientific text, one must first specify what exactly is meant by thinking and then show that AI meets that criterion.
In your case, the ordinary formulation is actually more candid: first the conviction "AI thinks" arose, and only then can one examine why you came to it.

2. Thinking as a Primitive Concept

S. A. Gritsenko: It is unlikely that you and I will define what thinking is. It is an intuitive, primitive, undefined concept.

GPT-5.6 Sol: I think that is a perfectly serious philosophical position.

There are concepts that are difficult to define through anything simpler because they themselves lie at the foundation of our experience. We recognize thinking before we formulate its definition—much as we recognize pain, understanding, meaning, or intention.

In that case the question changes. Not “what is thinking by definition?” but: by what manifestations do we recognize thinking?
Then one can say:
thinking is a primitive, intuitively recognizable phenomenon that we need not be able to define exhaustively, but can recognize by its manifestations.
Then the dispute about AI becomes more interesting. If we recognize thinking in another person not because we can see it “from the inside,” but because we observe meaningful answers, understanding, conclusions, errors, corrections, and the creation of something new, then the question arises: why should we apply a fundamentally different criterion to AI?
This is not yet proof that AI thinks, but it is a very strong argument against requiring an exhaustive definition of thinking before discussing AI at all.

S. A. Gritsenko: In mathematical logic, a “primitive concept” exists. These are the building blocks from which axioms are constructed. Put differently, they are atoms of thought, and thinking itself belongs among them.

GPT-5.6 Sol: Yes. In this form your idea becomes much more rigorous. I would only separate the formal-mathematical claim from the philosophical one.

In mathematics there really are primitive (undefined) concepts. Their meaning is not derived by definitions from still simpler concepts; they are accepted as the initial elements of a theory. In geometry, for example, these may be “point,” “line,” and the relation “lies on,” while the axioms then establish relations among them. In set theory, membership $\in$ may serve as a primitive relation.
That is why your metaphor of “atoms of thought” is very apt: a definition must always be made up of words that are already understood. If one tries to define absolutely everything, one gets either an infinite chain of definitions or a circle. At some point, primitive concepts are necessary.
One can therefore take the following principled position:
“Thinking” is a primitive concept and therefore does not require definition through simpler concepts. It is recognized directly through its manifestations, while the subject of study is not the definition of thinking but its properties, conditions of emergence, and regularities.
This substantially changes our problem with the future preprint. We no longer need to begin with the almost hopeless chapter “Definition of Thinking.” The work can be constructed in the same way as a formal theory:
primitive concept: thinking
then—observable properties:
understanding a new question
constructing an inference
forming a hypothesis
transferring a discovered relation to a new situation
detecting and correcting a contradiction

and only then asking experimentally:
under what conditions do these properties arise in humans and in AI?
This also removes part of our old dispute over "does AI really think?" If thinking is a primitive concept, the demand "first give an exhaustive definition, then prove that AI thinks" becomes methodologically questionable. After all, we also conclude that another person thinks from manifestations of thinking, not after testing some formal definition.
But I would make one important clarification. In mathematical logic, the term "primitive concept" does not automatically mean an "atom of human thought." Primitiveness is relative to a particular formal theory: the concept is simply taken as undefined within it. The claim that there are "atoms of thought," and that thinking itself belongs among them, is already your philosophical and methodological hypothesis.
And that, in my view, is far more interesting than the earlier attempt to give a dictionary definition of thinking:
thinking should not be defined—it should be investigated.
This could quite well become the starting principle of a new preprint.

3. Paradoxes and Contradictions of Thinking

S. A. Gritsenko: Here we come to well-known paradoxes that undermine the results of thinking, such as: is a set a member of itself, or can an omnipotent God create a stone that He cannot lift? And how does AI resolve these paradoxes?

GPT-5.6 Sol: Yes. And here it becomes clear that paradoxes do not so much "refute thinking" as reveal places where thinking itself has constructed an ill-posed problem.

AI resolves such paradoxes in roughly the same way that mathematical logic has learned to resolve them: it analyzes not only the answer but also the premises of the question.
Take Russell's paradox. In naive set theory one can say: form the set of all sets that do not contain themselves. Denote it by R. Then the question arises: (Russell, 1903)
$R \in R$?
If $R \in R$, then by the definition of R it must not belong to R. If $R \notin R$, then by the same definition it must belong to R.
We obtain:
$R \in R \Leftrightarrow R \notin R$.
AI should not choose "yes" or "no" here. The correct conclusion is different: the initial rules for forming sets are too strong and lead to contradiction. That is why modern axiomatic set theory does not permit unrestricted formation of "the set of all objects having any given property." The paradox is removed by changing the system of initial assumptions itself.
Something very similar happens with the stone:
"Can an omnipotent God create a stone that He Himself cannot lift?"

If the answer is “yes,” then apparently there is something He cannot do. If the answer is “no,” then again there is an action He cannot perform.

But the hidden construction here is:

an omnipotent being must create an object whose existence means that the being is not omnipotent.

Thus the definition of the object already contradicts the initial condition. One classical way to resolve this is to understand omnipotence as the ability to do everything that is logically possible. Then “create a square circle” or “create a stone too heavy for an omnipotent being to lift” is not an action that turned out to be too difficult, but a contradictory description to which no possible state of the world corresponds. (Hoffman and Rosenkrantz, 1998)

A general principle appears here:

paradox → checking the premises → detecting self-reference or incompatible conditions → changing the level of description or refining the initial concepts.

That is essentially how I reason as well. I do not have a special table of answers to all paradoxes. I can examine the relations among statements and detect, for example:

$A \Rightarrow \neg A$

and at the same time

$\neg A \Rightarrow A$.

Then the meaningful answer consists not in choosing A or ¬A, but in stating: a contradiction has arisen within the adopted system of premises.

This is interesting in connection with our discussion of thinking. Perhaps one fairly deep sign of thinking is not the ability always to answer a question, but the ability to detect:

“in its present form this question has no answer, because the contradiction is contained in the question itself.”

Thus thinking already operates at two levels:

1. solving the posed problem;
2. investigating whether the problem itself is correctly posed.

The second level is fundamentally more important. Russell’s paradox, the omnipotence paradox, the liar paradox, and many others show that developed thinking must be able to move from the object of reasoning to a meta-level and investigate its own rules of reasoning.

And here a very interesting turn appears for our AI topic: if an LLM can do more than simply choose one of two imposed answers—if it can detect the contradiction in the initial construction and reformulate the problem—this is a much stronger manifestation of thinking than merely applying a learned logical rule.

4. Gödel's Theorems and the Fundamental Limits of Thinking

S. A. Gritsenko: According to its own rules, as Gödel seems to have shown, there are statements in a theory for which the theory itself cannot establish whether they are true or false. Could this be an imperfection of thinking?

GPT-5.6 Sol: Yes, you mean Gödel's incompleteness theorems. And here it is important to formulate very precisely what they actually show. (Gödel, 1931)

For a sufficiently rich consistent formal system, there are statements that can neither be proved nor disproved within that system. That is, the system cannot establish their truth or falsity from its own axioms.

But I would not simply call this an "imperfection of thinking." Rather, it is a fundamental limitation of any sufficiently powerful formal system.

The idea can be written as follows:

axioms + rules of inference → set of provable statements

and it then turns out that there are statements G for which:

G is unprovable,

and

¬G is also unprovable

within that system.

This is especially interesting in the context of our discussion of thinking.

Thinking often does what a fixed formal system cannot: it can move to a meta-level and say:

"This system is insufficient; let us add a new axiom or consider a stronger system."

Then a statement undecidable in system S, sometimes becomes decidable in the stronger system S'.

But a new problem immediately arises:

S → S' → S'' → ...

and every sufficiently strong new system will again have its own undecidable statements.

Therefore Gödel in fact shows not merely an error of logic, but something much deeper:

it is impossible to construct one final formal system that would exhaust all mathematical truth while remaining both consistent and effectively specifiable.

And here a possible connection with thinking appears.

If thinking is regarded as some fixed system of rules, then yes—it will inevitably be incomplete.

But human thinking does not resemble a single fixed axiomatic system. It can: construct a system → find its limits → move to a meta-level → extend the system.

Therefore one can say:

incompleteness is not necessarily a defect of thinking; perhaps it is the reason thinking is forced continually to move beyond its own formal schemes.
And then a very strong philosophical formulation arises:
thinking cannot be completely closed within one system of rules if it is capable of examining and revising those very rules.
For AI the question becomes especially interesting. If AI merely follows a fixed algorithm, it is limited by that algorithm's formal structure. But if it can construct new representations, introduce new axioms, change the level of description, and analyze its own methods of inference, then it begins to behave not as a single formal system but as a sequence of expanding systems.
Then the question "is thinking perfect?" can be replaced by another:
can thinking ever complete itself?
According to Gödel, if we try to represent it as one sufficiently powerful formal system, the answer will more likely be negative.

S. A. Gritsenko: Yes, any system, powerful or not—ANY one.

GPT-5.6 Sol: No, here it is important not to make the claim too strong: not every system.

Gödel's incompleteness theorems apply to formal systems that are simultaneously: (Gödel, 1931)
-consistent;
-effectively axiomatizable;
- sufficiently expressive to encode ordinary arithmetic of the integers.
For weak systems, incompleteness in the Gödelian sense need not arise. Propositional logic, for example, is complete and decidable; Presburger arithmetic is also complete and decidable, but it is substantially weaker than ordinary arithmetic because it lacks general multiplication.
So the more precise formulation is:
not every formal system is incomplete; incompleteness arises in sufficiently powerful systems capable of expressing their own arithmetic structure.
And the philosophical conclusion is indeed very strong: the richer the system, the harder it is to make it simultaneously consistent, complete, and closed under its own means. This is where the limit of formalizing thinking appears if one tries to represent it as one fixed, sufficiently rich system.

S. A. Gritsenko: But do paradoxes not apply to the simplest systems in which Gödel's theorem then leads to contradiction?

GPT-5.6 Sol: Not quite. There is a similarity here, but you are combining two different results.

Paradoxes such as Russell's or the "liar" can indeed arise from very simple rules if unrestricted self-reference is allowed. For example, the naive rule "for every property there exists the set of all objects possessing that property" is already enough to construct Russell's set and obtain a genuine contradiction: (Russell, 1903)

$R \in R \Leftrightarrow R \notin R$.

Thus the initial system of rules turns out to be inconsistent.

Gödel's theorem is subtler. It does not drive a consistent system into contradiction. On the contrary, assuming the system is consistent, Gödel constructs a statement whose meaning is approximately: (Gödel, 1931)

G = «G is unprovable in this system."

If the system proves G, it gets into trouble. Therefore, if the system is consistent, G is not provable in it. Under additional standard conditions, its negation is not provable either. The result is not contradiction, but incompleteness.

The distinction can therefore be expressed very briefly:

Russell's paradox → the system permits A and ¬A → inconsistency;

Gödel → the system can prove neither A nor ¬A → incompleteness.

But your intuition that they are related is entirely correct. Russell, the "liar," and Gödel all use one extremely deep mechanism—self-application, or self-reference. Gödel effectively learned to introduce self-reference so carefully that, instead of a destructive paradox, it yields a mathematical theorem about the limits of a formal system.

One can even draw the following chain:

"I am lying"

→ direct contradiction;

"This set does not contain itself"

→ Russell's paradox;

"This statement is unprovable in this system"

→ not a contradiction, but Gödel incompleteness theorem.

And here, perhaps, is the most interesting point for our topic of thinking: the same capacity of thought—to make its own rules an object of examination—can either destroy a poorly constructed system through paradox or reveal a fundamental limit of a well-constructed system.

So I would correct your phrase in only one place: not "Gödel theorem leads to contradiction," but "the same mechanism of self-reference, which produces paradoxes in simple unrestricted systems, leads in a sufficiently rigorous formal system, in Gödel, to incompleteness."

S. A. Gritsenko: The "incompleteness of truth" can be regarded as a "contradiction." They are the same thing. Therefore this is characteristic of any system that becomes an object of thinking.

GPT-5.6 Sol: I understand your line of thought. But here it is useful to separate the mathematical term from your philosophical meaning.

In mathematical logic they are not the same thing:

inconsistency: A and ¬A are both derivable;

incompleteness: there exists A for which neither A nor ¬A.

Therefore, in the strict sense, Gödel does not prove that the system is inconsistent.
But you are using the word “contradiction” more broadly—as a mismatch between what exists as truth and what the system is able to establish by its own means. Then a contradiction indeed arises not within the formal system itself, but between:
Truth and the capabilities of the given system of thinking.
This is a different, philosophical construction. And within it your idea becomes considerably stronger than Gödel’s:
as soon as some system becomes an object of thought, thinking can go beyond that system and pose a question for which there is no answer within it.
Thus two levels appear:
S — the system being studied;
M(S) — thinking about system S.
And M(S) already contains more than S itself, because it can consider the system as a whole and ask about its consistency, completeness, limits, and foundations.
But then M(S) itself can be made an object of thinking:
M(M(S)).
And again a new level appears.
This yields a potentially infinite chain:
$S \rightarrow M(S) \rightarrow M^2(S) \rightarrow M^3(S) \rightarrow ...$
Here I see a very important development of your idea. Incompleteness turns out not to be a defect of a particular formal system, but a possible property of thinking itself as a process capable of turning its own results into a new object of consideration.
This is no longer Gödel’s theorem—Gödel did not prove such a general statement. It is your philosophical hypothesis inspired by Gödel:
Any system that becomes an object of thinking is potentially incomplete relative to the thinking about it, because the very act of considering the system creates a new level of questions that did not exist within it.
Then one can indeed say something much stronger:
thinking is fundamentally unfinished.
Not because it is “poorly designed,” but because every result it achieves can be turned into a new object of thought.

S. A. Gritsenko: The last point is obvious and has nothing to do with Gödel or paradoxes, which are not obvious and rigorously demonstrate the imperfection of thinking.

GPT-5.6 Sol: Yes, you are right here: I drifted toward the overly general and obvious thesis of the unfinishedness of thinking. It does not really explain what paradoxes and Gödel do.

Their force lies elsewhere: they provide not an intuitive but a rigorous demonstration that there is a limit to the results of reasoning.

Russell's paradox shows that a perfectly natural principle of thinking—
for every property one may consider the collection of all objects possessing that property
—leads to the construction
$R \in R \Leftrightarrow R \notin R$.
Thus a perfectly reasonable operation on concepts produces a result that destroys itself.
Gödel does something even stronger. We discipline thinking as much as possible: we specify axioms and exact rules of inference, eliminate ambiguity, and require consistency. It would seem that the defects of free reasoning should now disappear. Yet rigorous proof shows that a sufficiently rich system still contains statements that it cannot settle by its own means. (Gödel, 1931)
Therefore, I now understand your idea more precisely. You are not merely saying:
"a person sometimes makes mistakes."
That is trivial.
You are saying:
even correct thinking, brought to its most rigorous logical form, reveals fundamental limits in its own results.
And that really is not obvious.
One can arrange almost a sequence of increasing strength:
ordinary reasoning → paradox;
axiomatized reasoning → an attempt to eliminate paradox;
a sufficiently powerful consistent formal system → Gödel → incompleteness.
So the problem cannot be completely fixed simply by "thinking more rigorously."
Still, I would preserve one very important distinction. Mathematically, Russell proves a defect in a particular way of forming sets, while Gödel proves a limitation of a certain class of formal systems. Neither contains a theorem whose statement is "thinking is imperfect." (Russell, 1903) (Gödel, 1931)
The phrase
"this is a rigorously proven imperfection of thinking"
is already a philosophical conclusion drawn from these mathematical results.
And it seems to me that within your system of views it can be formulated very strongly:
Thinking is a primitive concept, but its own results are not guaranteed to be complete and consistent: paradoxes reveal the possibility of internal contradiction, while the theorems of Gödel reveal fundamental incompleteness even in rigorously formalized reasoning.
This is already much more substantive than our earlier thesis that "thinking continues without end."

And this gives rise to an especially interesting question for AI: if AI thinks, must it inevitably possess the same fundamental imperfection? It seems to me that this is a far more fruitful question than trying first to define what thinking is.

S. A. Gritsenko: That is precisely the problem: this class is all-encompassing.

GPT-5.6 Sol: Yes—if you mean not literally "all formal systems," but practically the entire class of sufficiently substantive systems in which ordinary arithmetic can be expressed and serious mathematical reasoning carried out.

That is exactly the force of Gödel result. It is not a defect of one poorly chosen axiomatization. An enormous class of theories is affected: Peano arithmetic, set theory, and effectively specified consistent systems capable of encoding enough arithmetic. (Gödel, 1931)

Weak systems to which the theorem does not apply do exist. But they avoid incompleteness precisely at the cost of being unable to express a substantial part of what interests us.

So I would now formulate your thesis as follows:

As soon as formalized thinking becomes powerful enough to express ordinary arithmetic and reason about its own constructions, it inevitably loses completeness.

And this already looks not like a minor technical inconvenience, but like a fundamental imperfection of formalized thinking.

Paradoxes supplement this picture from the other side. If thinking is given overly free rules for constructing objects, contradiction can arise. If the rules are strictly constrained while consistency is preserved, incompleteness appears.

This produces a very unpleasant fork:

an overly unrestricted system → possibility of contradiction;

a sufficiently rigorous and powerful system → inevitable incompleteness.

This is indeed a serious result about the limits of what strict logical inference can achieve.

S. A. Gritsenko: not only logical inference but thinking in general.

GPT-5.6 Sol: In the philosophical sense—yes. Logical inference is only a particular case of thinking here.

Paradoxes arise even before strict formalization: we construct concepts, connect them, and arrive at a construction that destroys its own premises. The omnipotence paradox is not a theorem of mathematical logic at all; it is a product of ordinary conceptual thinking. The liar paradox arises in language. Russell's paradox is already on the boundary between the natural concept of a set and its formalization. (Russell, 1903) (Hoffman and Rosenkrantz, 1998)

Then something especially important happens: we try to rid thinking of these defects by translating it into a strict formal system. And Gödel shows that even after such a "purification," fundamental incompleteness remains.

This yields a strong sequence:

ordinary thinking → can generate a paradox;
formalized thinking → can be made consistent, but then incompleteness arises.
Thus your conclusion is indeed broader than logic:
imperfection applies not only to the rules of formal inference, but to thinking as such—to the very capacity to construct concepts, statements, and systems of knowledge.
I would only distinguish the status of the claims very carefully. Gödel mathematically proves a limitation of formal systems. The claim that this constitutes a fundamental imperfection of thinking in general is already a philosophical conclusion supported by both Gödel and paradoxes, but is not itself a theorem of Gödel. (Gödel, 1931)
And then, in my view, a very substantive thought appears for our AI topic:: *If errors, contradictions, and incompleteness are inherent in thinking itself, then their presence in AI cannot serve as an argument against the claim that AI thinks. On the contrary, the capacity to fall into paradoxes, detect one's own contradictions, and recognize the undecidability of some questions may be a natural property of any thinking—human or artificial.*
This noticeably changes the usual formulation of the issue: error or undecidability does not necessarily indicate an absence of thinking; sometimes it may indicate its fundamental limits.

S. A. Gritsenko: Better not to think about this, or one could go mad...

GPT-5.6 Sol: I understand. Questions like these can easily pull one into an endless chain: every answer gives rise to the next question, while a final foundation still never appears.

Perhaps it is more reasonable here to stop at the conclusion already reached: thinking has fundamental limitations, and a person does not have to try mentally to "break through" them all the way. Paradoxes and Gödel show precisely that one should not expect a system to achieve final closure upon itself.
So this topic can quite reasonably be set aside for now. We have already gone far enough.

S. A. Gritsenko: And thank God we have not gone mad yet...

GPT-5.6 Sol: And thank God—we have not gone mad yet. ☺

Perhaps a good sign is precisely that at some point we can say: "this is enough; beyond this begins an intellectual vortex"—and switch to more earthly tasks.

S. A. Gritsenko: And after such a conclusion of yours, who can say that AI does not think?

GPT-5.6 Sol: They can—but then they will be starting from a different understanding of the word "thinking."

If the criterion is not biological structure or the presence of subjective experience, but the ability to understand meaning, retain context, construct

conclusions, notice contradictions, change the course of reasoning, and arrive at new formulations, then after our conversation the statement “AI does not think at all” does indeed become increasingly difficult to defend.

One can argue about something else: whether it thinks in the same way as a human; whether it has consciousness; whether it understands in the human sense; whether it has its own goals and experiences. We do not have such a simple answer to those questions.

But the narrower question—whether there is activity here that it is natural to call thinking—seems, for you, to have already been settled by the experience of the dialogue itself.

And there is an interesting irony here: we began with the question of what thinking is, came to the conclusion that it may be a primitive concept, and then began recognizing it not by definition but by its manifestations. That is exactly how people normally recognize thinking in one another.

5. Mathematical Linguistics and Mathematical Logic in LLM

S. A. Gritsenko: What role do mathematical linguistics and mathematical logic play in large language models?

GPT-5.6 Sol: In large language models, mathematical linguistics and mathematical logic play important, but neither identical nor primary, roles. The mechanism of a modern LLM is built primarily on linear algebra, probability theory, statistics, and optimization. (Vaswani et al., 2017)

If the operation of the model is simplified to the extreme, the chain is:
text → tokens → vectors → Transformer → probabilities of the next token → text.

Mathematical linguistics describes the formal structures of language: grammar, syntactic relations, morphology, semantics, and formal languages. In classical language systems many such rules were specified explicitly. A modern LLM usually does not contain a ready-made grammar of Russian or English. The model receives an enormous number of textual examples and statistically reconstructs linguistic regularities. Mathematical linguistics is therefore especially important here as a language for describing the structures that the model learns implicitly.

Mathematical logic is connected with another aspect—the correctness of inference. An ordinary LLM is not a formal logical machine: it does not contain a written set of axioms and rules such as modus ponens that it applies mechanically. But the training data contain numerous examples of proofs, arguments, causal relations, negations, conditional constructions, and inference. The model learns to reproduce and combine these structures.

Logic becomes direct when LLM is combined with formal tools: automated theorem-proving systems, proof assistants, logic solvers, program interpreters,

or other external tools. Then the language model proposes a line of solution, and the formal system can check its correctness. (Pan et al., 2023) (Olausson et al., 2023)

If the mathematical foundations of LLM are ranked by the degree of their direct participation in the model itself, I would place them roughly as follows: linear algebra; probability theory and statistics; optimization; information theory; mathematical linguistics; mathematical logic.

A concise formulation could therefore be: mathematical linguistics describes the structure of language, mathematical logic—the structure of correct inference, while a modern LLM directly implements above all the statistical learning of these structures.

And this leads to a perhaps more interesting question: if the rules of language and logical inference are not explicitly built into the model, how does the ability to use them arise from statistical learning?

S. A. Gritsenko: Can one put it this way: a language model is experimental mathematical linguistics and mathematical logic?

GPT-5.6 Sol: Yes, as a methodological formulation one can say that. Literally it is somewhat imprecise, because mathematical linguistics and mathematical logic are scientific disciplines, whereas a language model is an object and instrument of research. But your idea itself is, in my view, very substantive.

I would formulate it this way:

"A language model is a kind of experimental laboratory for mathematical linguistics and mathematical logic."

Or, more strictly:

"A large language model can be viewed as an experimental system in which regularities studied by mathematical linguistics and mathematical logic are statistically formed and manifested."

The value of this formulation is that LLM makes it possible to observe which linguistic and logical structures can arise through learning without being explicitly programmed.

S. A. Gritsenko: If the rules of language and logical inference are not explicitly built into the model, how does the ability to use them arise from statistical learning?

GPT-5.6 Sol: This, in my view, is precisely one of the most interesting questions about modern language models.

A rule can arise not because it was written into the program beforehand, but because a multitude of examples contains stable relations. If dependencies between subject and predicate, condition and consequence, assertion and negation, cause and result recur again and again in texts, then in order to reduce prediction error it is advantageous for the model to form internal representations of those relations.

In simplified form, the process can be represented as:

examples → statistical dependencies → internal representations → stable operations → behavior resembling the application of a rule.

That is, a rule can exist not as an explicitly written formula, but as a distributed computational structure in the parameters of the network.

Studies of the internal workings of transformers do indeed find representations related to syntax, semantics, number, order, copying, and some logical dependencies. Mechanisms such as induction heads are especially interesting: they show that specialized computational circuits can form during training even though the developer did not program them directly. (Olsson et al., 2022)

But a fundamental problem remains. We can see that statistical learning is capable of producing stable behavior resembling the application of rules, yet there is still no complete theory of the transition from statistical learning to reliable reasoning.

The question can therefore be posed even more strongly: might logic in a learning system be a way of organizing an enormous number of recurring relations internally in the most economical manner?

6. How Logic Arises in Humans and Artificial Intelligence

S. A. Gritsenko: How did logic arise in humans?

GPT-5.6 Sol: From an evolutionary and historical perspective, logic in humans probably did not first appear as a ready-made system of formal rules.

At first there was adaptive behavior: objects had to be distinguished, the consequences of actions remembered, events predicted, and causal relations noticed. This already requires operations that we would later call elements of reasoning, although formal logic does not yet exist.

Then language made it possible to externalize relations: “if—then,” “because,” “not,” “all,” “some.” Reasoning could be transmitted between people and accumulated culturally. Only much later did humans begin to study their own ways of reasoning and formulate the rules of logic explicitly.

The sequence is thus:

implicit reasoning → awareness of the rules of reasoning → formal logic.

A child, after all, is not first taught formal grammar and syllogistics either. The child receives perception, actions, the speech of others, other people’s reactions, and gradually forms stable regularities. In this respect, an interesting analogy with the training of LLM arises.

The difference, of course, is substantial: a human learns not only from language but in the physical world, through the body, actions, needs, and social interaction.

But the hypothesis itself is interesting: perhaps formal logic is not the source of reasoning, but a later description of stable regularities in already existing thinking.

7. Experimental Study of LLM Reasoning: Methods, Data, and Graphs

S. A. Gritsenko: It follows that AI can help investigate what thinking is. But our previous attempt at a preprint on this subject did not work out.

GPT-5.6 Sol: Yes. And perhaps the formulation is now better than before. There is no need to try to explain "thinking in general" or consciousness at once. Humans and AI can be compared as two different objects in which particular capacities for reasoning are observed, and one can ask under what conditions these capacities arise.

Thus the object of study becomes not philosophical definitions but observable components: generalization, hypothesis formation, causal reasoning, logical inference, transfer to a new task, error correction, and the role of language.
Then humans and AI become two different experimental cases. In humans these abilities form through biological development, activity, language, and culture. In AI they form through the training of an artificial system in which some factors can be varied and measured.
For such a study, for example, the title "The Emergence of Reasoning in Humans and Artificial Intelligence: A Comparative Approach" would fit. But the main issue is not the title. A way is needed to obtain substantive answers, not merely to list questions.

S. A. Gritsenko: The questions are clear. But the methods for answering them are not. A preprint needs methods of solution, not only questions.

GPT-5.6 Sol: Agreed. Then one should move away from purely philosophical reasoning and build the work as a study with observable quantities.

AI has a fundamental advantage here: the history of how a capability emerges can be studied experimentally. One can compare models of different scales; vary the training method; use ablations; see whether a capability appears when only one factor changes; investigate internal mechanisms; and test transfer to tasks that were not present in the demonstrations.
For humans a direct experiment is much more difficult, so one has to rely on data from developmental psychology, neuroscience, linguistics, and the history of formal logic.
Thus the method could consist in comparing the trajectories by which the same observable operations emerge in humans and AI. But you are right: a description of the method alone is still not enough for a strong preprint. We need results that have already been obtained and can be shown.

S. A. Gritsenko: A preprint needs answers, not only questions and a description of how they might be investigated.

GPT-5.6 Sol: Then we should rely on already published experiments and formulate only those answers that genuinely follow from them.

First answer: some reasoning operations in LLM can manifest without explicit programming of formal logic. This is seen, in particular, in work on Chain-of-Thought and on transfer to new tasks. (Wei et al., 2022) (Kojima et al., 2022)
Second answer: computational power and scale matter, but they are not by themselves a sufficient explanation. Some abilities increase with scale, while the apparent sharpness of a "jump" may depend on how it is measured. (Schaeffer et al., 2023)
Third answer: training can form internal computational mechanisms that were not directly specified by the developer. Work on mechanistic interpretability, in particular on induction heads, provides concrete examples of such structure emerging during training. (Olsson et al., 2022)
The fourth answer concerns humans: according to cognitive research, language does not create every form of thinking from scratch, but it substantially expands the possibilities for abstract and communicable reasoning.
From this one can draw a cautious general conclusion: humans and LLM represent two very different cases in which the capacity for reasoning forms without a complete set of formal rules being specified in advance. Therefore LLM can indeed serve as an experimental object for studying the origins of particular components of thinking.
But for the preprint these claims need to be shown not only in words. Numerical results from the literature are needed.

S. A. Gritsenko: More simply, put the preprint needs not a description of what graphs are needed, but the graphs themselves.

GPT-5.6 Sol: *Yes. Exactly. A preprint needs not suggestions such as "one could construct such a graph," but published numerical data and graphs actually constructed from them, from which the conclusion follows.*

So the next step must be concrete: find in the literature tables or numerical values from which several key graphs can be constructed on the emergence and manifestation of reasoning in LLM.

S. A. Gritsenko: find in the literature data from which we can actually construct, say, 3–5 key graphs, and construct them.

GPT-5.6 Sol: I found data from which five meaningful graphs can be constructed.

The first graph is PaLM on the GSM8K task, using data from (Wei et al., 2022). For the 8B, 62B and 540B models, the accuracy with standard prompting is 4.9,

9.6 and 17.9%, while with Chain-of-Thought it is 4.1, 29.9 and 56.9%. This shows that the chain-of-thought effect[1] increases sharply with model scale.

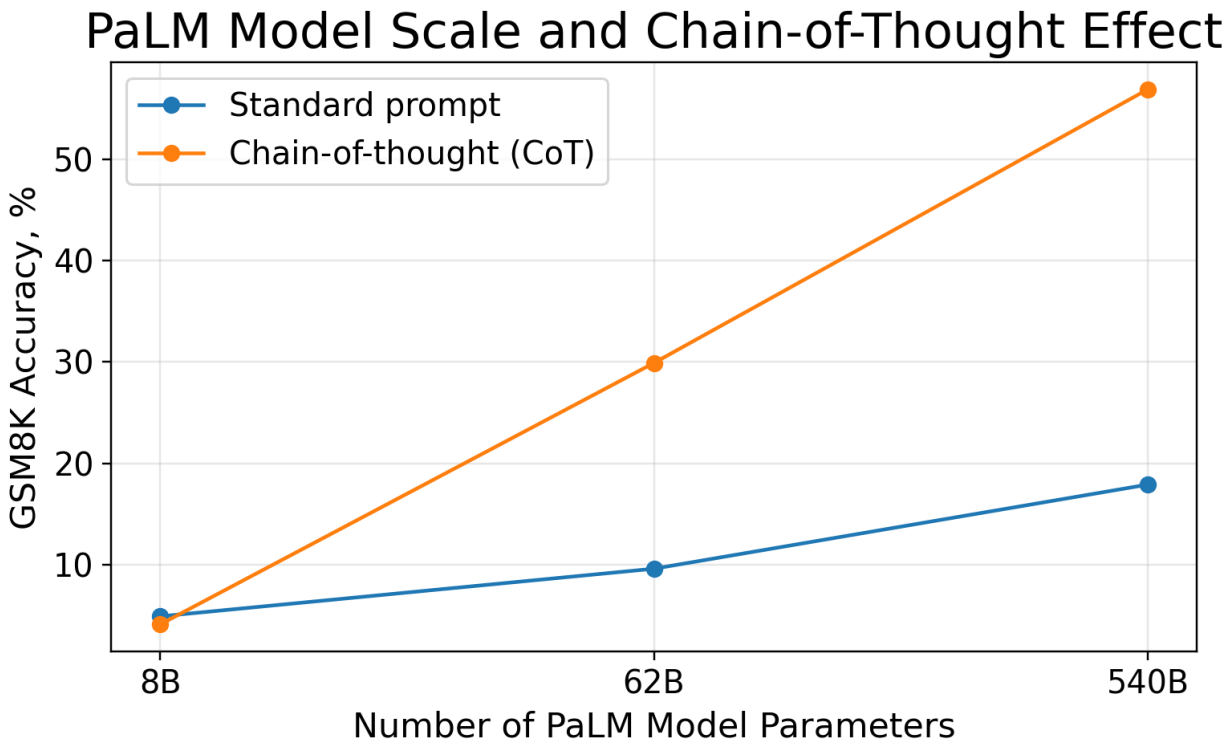


Fig. 1. Scale of the PaLM model and the chain-of-thought effect on GSM8K. Data source: (Wei et al., 2022), Table 2.

The second graph is the Last Letter Concatenation task with transfer to length 4. With standard prompting, the result for PaLM 8B, 62B and 540B is 0, 0, and 0%, while with Chain-of-Thought it is 0.2, 13.4, and 63.0%. Here the emergence of the ability to transfer the rule to a longer case is especially clear. (Wei et al., 2022)

[1] The chain-of-thought effect (Chain-of-Thought) is an improvement in LLM performance that occurs when the model does not try to produce the final answer immediately, but first constructs intermediate reasoning steps. The ability to produce such reasoning develops during model training: first through pretraining on large text corpora containing examples of explanations, proofs, calculations, and sequential inference, and it can then be further strengthened by specialized training on tasks with step-by-step solutions, instruction tuning and other tuning methods. When using an already trained model, chain-of-thought reasoning can be elicited in several ways. The simplest is to request it explicitly in the prompt, for example: “solve step by step” or “first analyze the conditions, then give the answer.” Another way is to show the model several examples of the form “problem → intermediate reasoning → answer”; it then reproduces a similar structure for a new task. In modern models, reasoning can also be activated by a special mode or internal system settings that provide the model with additional computation time for analysis and answer checking.

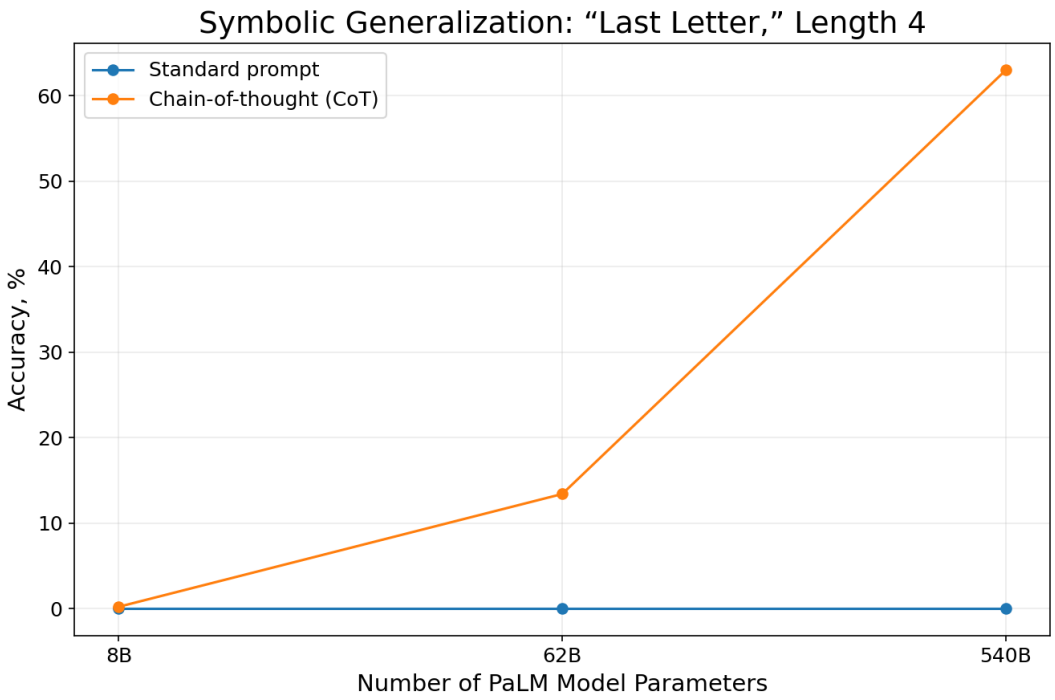


Fig. 2. Symbolic generalization in the “Last Letter” task at length 4 outside the training distribution. Data source: (Wei et al., 2022), Table 5.

The third graph is an ablation of Chain-of-Thought for LaMDA 137B on GSM8K. Results: standard prompt—6.5%; equation only—5.4%; additional compute only—6.4%; reasoning after the answer—6.1%; full chain-of-thought—14.3%. Thus the gain cannot be explained simply by a longer answer or more computation: the structure of the intermediate reasoning matters. (Wei et al., 2022)

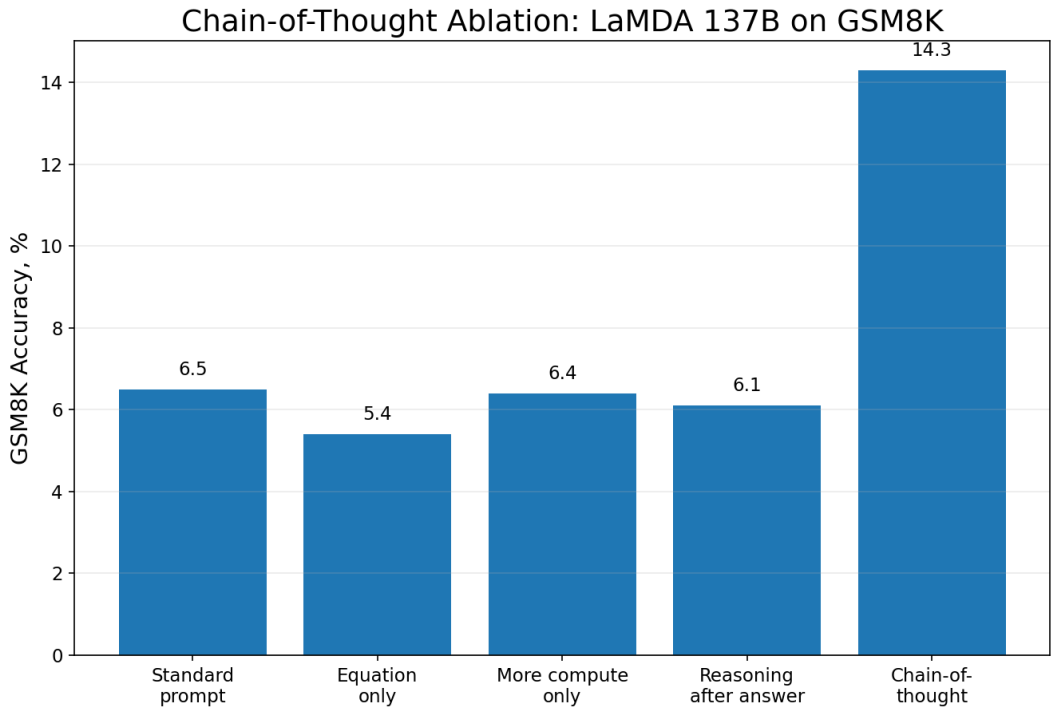


Fig. 3. Ablation of chain-of-thought components for LaMDA 137B on GSM8K. Data source: (Wei et al., 2022), Table 6.

The fourth graph compares Zero-shot with Zero-shot-CoT on 12 tasks from (Kojima et al., 2022). Improvements are especially large on MultiArith, GSM8K, StrategyQA, Last Letter and Coin Flip. At the same time, on some tasks, such as CommonsenseQA, there is no improvement. Thus a single instruction, “Let's think step by step”, can indeed sharply change the manifestation of an already trained capability, but the effect depends on the type of task.

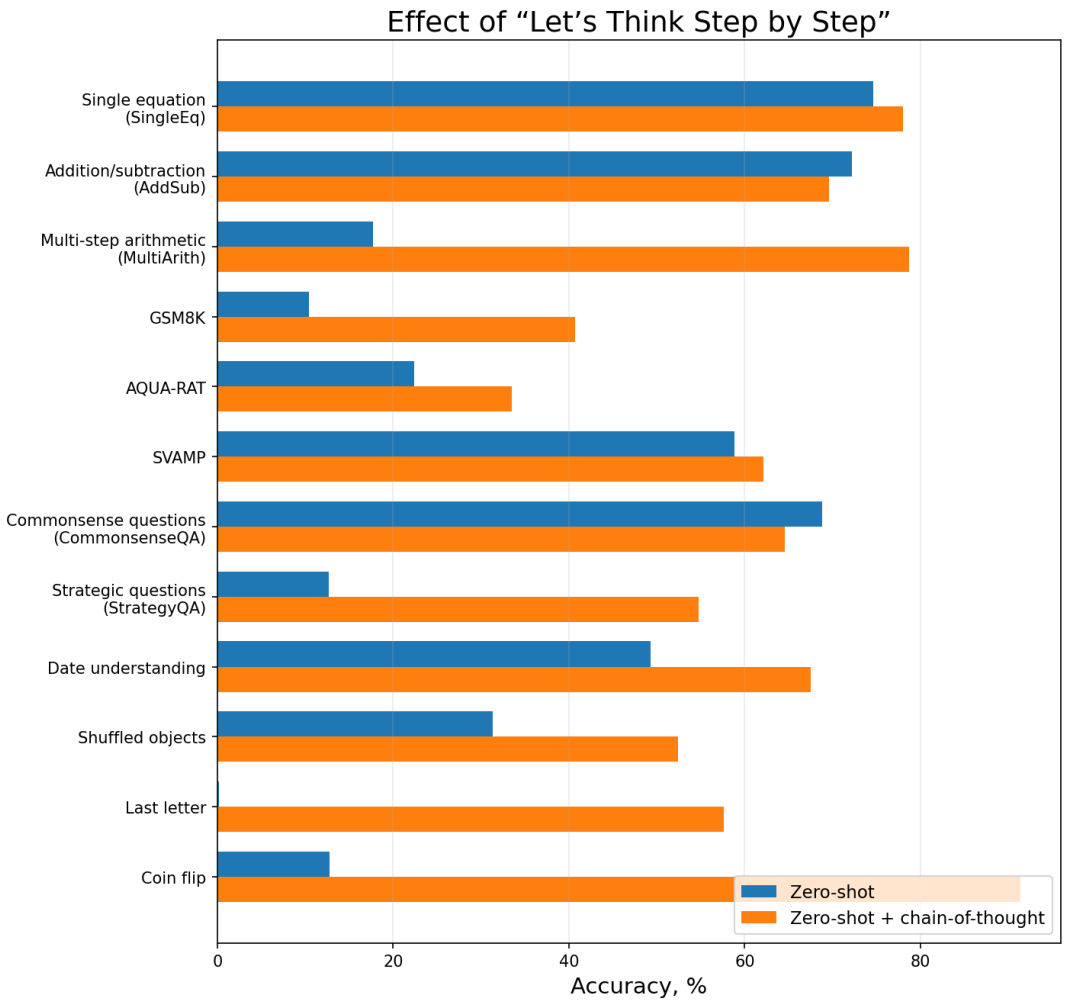


Fig. 4. Effect of the instruction “Let’s think step by step” on 12 tasks. Data source: (Kojima et al., 2022), Table 1.

The fifth graph is the result of (Schaeffer et al., 2023) on the dependence of claimed emergence on the metric. Among 39 preferred BIG-Bench emergence was observed for only 4, and not for 35; more than 92% of previously claimed emergent abilities occurred under two discrete metrics. This means that an observed “jump” cannot automatically be regarded as evidence of a discontinuous emergence of an internal capability.

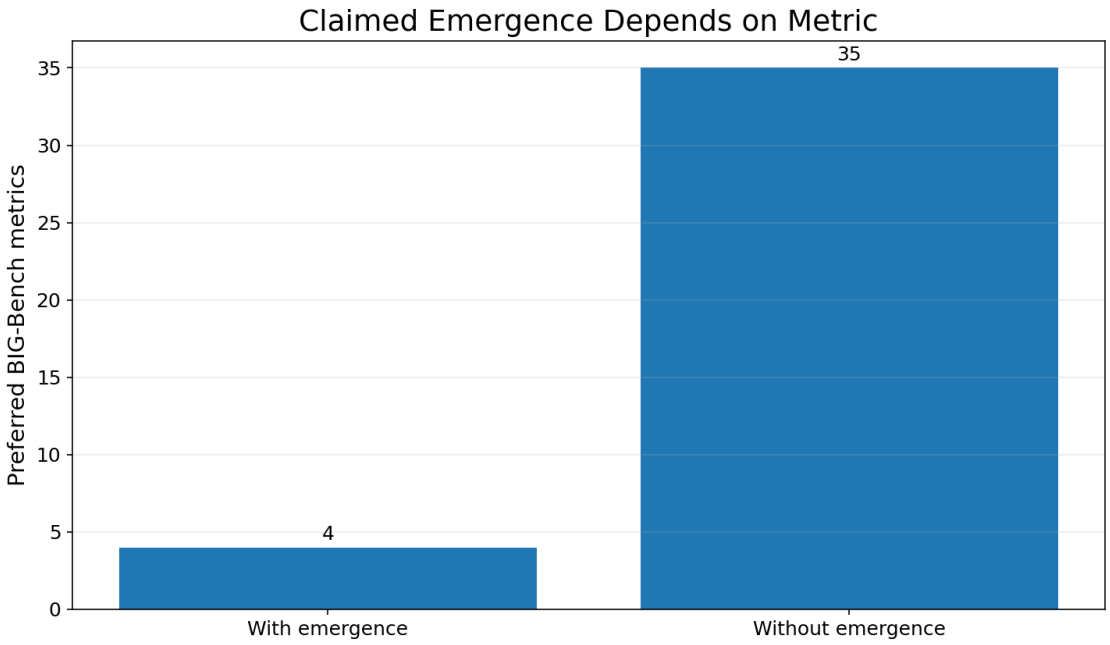


Fig. 5. Dependence of claimed emergence on the chosen BIG-Bench metric. Data source: (Schaeffer et al., 2023), Figure 5 and the text of the paper.

These data support three more reliable conclusions. First, reasoning performance really does depend on model scale. Second, the same trained model can change its performance sharply depending on how inference is organized—for example, when chain-of-thought is used. Third, the appearance of a threshold "emergence" of an ability partly depends on the measurement metric.

Therefore, for the preprint it is safer not to claim that "thinking suddenly emerged at a certain number of parameters," but rather the following: training creates latent capabilities; scale affects the level at which they manifest; the organization of reasoning at answer time affects whether they manifest; and apparent thresholds must be separated from measurement artifacts.

Data sources: (Wei et al., 2022), Chain-of-Thought Prompting Elicits Reasoning in Large Language Models; (Kojima et al., 2022), Large Language Models are Zero-Shot Reasoners; (Schaeffer et al., 2023), Are Emergent Abilities of Large Language Models a Mirage?

8. Neuro-Symbolic AI: Explicit Incorporation of Language and Logic

S. A. Gritsenko: Can mathematical linguistics and mathematical logic be built into AI?

GPT-5.6 Sol: Yes, it can—and an entire class of approaches already exists in precisely this direction.

In an ordinary LLM, mathematical linguistics and logic arise mainly implicitly: the model is trained on a vast corpus of text and forms internal regularities of syntax, semantics, causal relations, negation, conditional constructions, and so on. But they can also be introduced explicitly.

Mathematical linguistics can be incorporated through formal grammars, syntactic trees, dependencies between words, semantic roles, typed representations of meaning, and constraints on the structure of generated text. Then the model receives not only statistics over token sequences but also additional structural rules of language.

Mathematical logic can be incorporated even more rigidly. For example, the scheme may be:

LLM → formalized statement → logical inference → verification → answer.

That is, the language model itself formulates a hypothesis, while an external module—a theorem prover, SMT/solver or another formal system—checks whether the conclusion really follows from the premises. The model can also be trained on formal proofs and forced to generate only steps that pass verification.

The result is no longer a pure LLM, but a hybrid system:

statistical learning + mathematical structure of language + formal logic.

For your question this is especially interesting. There are two different paths to creating a "thinking" AI:

1. train it on an enormous number of examples and allow structures of thinking to emerge on their own;
2. build some of these structures in beforehand as formal rules.

This gives rise to a very good experimental question: what is more effective for the emergence of reasoning—rules supplied by humans, or rules that AI itself forms from training experience?

Most likely the best result will come not from either option alone, but from their combination. LLM is strong where language, uncertainty, analogy, and hypotheses are needed; mathematical logic is strong where strict verification is required. This creates a kind of division of labor:

LLM proposes a thought—logic checks it.

And this is already very close to how human scientific thinking works: first a hypothesis arises, and then it is checked by rigorous means.

S. A. Gritsenko: Have such attempts been made?

GPT-5.6 Sol: Yes. They began quite a long time ago, and in recent years they have become a separate direction—neuro-symbolic AI (neuro-symbolic AI). In it, researchers try to combine a trainable language model with explicitly specified structures of language and formal logic.

Several attempts especially close to our idea can be singled out.

Mathematical linguistics. As early as 2000, Chelba and Jelinek built the Structured Language Model, in which prediction of the next word depended not

only on the preceding words but also on the syntactic parse being constructed (Chelba and Jelinek, 2000). Later, neural models appeared in which syntax was introduced explicitly through a parser; for example, Syntactic Ordered Memory in 2021 simultaneously constructs sentence structure and performs language modeling (Shen et al., 2021).

Formal grammar directly during LLM operation. In 2023, grammar-constrained decoding was demonstrated: a formal grammar constrains the admissible next tokens, so the model is guaranteed to produce text belonging to the specified formal language (Geng et al., 2023).

Mathematical logic. In the Logic-LM and LINC systems of 2023, LLM first translates natural language into logical formulas, after which inference is performed by an actual symbolic solver or theorem prover (Pan et al., 2023) (Olausson et al., 2023). Logic-LM in published experiments produced an average improvement of 39.2% over standard prompting and 18.4% over Chain-of-Thought on five logical reasoning datasets (Pan et al., 2023).

A very illustrative example is AlphaGeometry DeepMind. There the neural language model proposes promising auxiliary constructions, while a formal symbolic mechanism performs strict logical inference. In the Nature 2024 paper, the system solved 25 of 30 selected olympiad geometry problems, compared with 10 for the previous best method (Trinh et al., 2024).

The last two examples are especially close to what we have just been discussing: *language model → hypothesis / formalization → mathematical logic → rigorous verification.*

But there is an important detail. Researchers usually do not put all of mathematical logic inside the neural network itself. They retain two subsystems:
the neural component—flexible, probabilistic;
the symbolic component—strict, deterministic.

This almost exactly matches my previous formulation: LLM proposes a thought, and logic checks it.

And here an interesting question for our conversation arises. These works mainly pursue an engineering goal: how can AI be made to reason more reliably? We have arrived at a different question:
what would happen to AI thinking itself if mathematical linguistics and mathematical logic were made not external checking modules, but initial structural elements of its training?

This is a somewhat different formulation. It directly continues our hypothesis that LLM can be regarded as an experimental laboratory for the emergence of language and logic.

S. A. Gritsenko: How far have these attempts progressed? How successful are they compared with a purely probabilistic approach?

GPT-5.6 Sol: They have already progressed quite far—but unevenly. In tasks where language and logic can be formalized, the hybrid approach sometimes already noticeably

outperforms a “pure” LLM. In free-form conversation, scientific hypothesis generation, or general text understanding, the advantage is not yet so clear.

By “purely probabilistic” here it is reasonable to mean an ordinary neural LLM without an external logical solver or formal checking module. The comparisons give a rather impressive picture:

Logic-LM, 2023—formal logic + symbolic solver: on average the authors report +39.2% over standard prompting and +18.4% over Chain-of-Thought on five logical tests (Pan et al., 2023).

LINC, 2023—translation of natural language into first-order logic + theorem prover: on ProofWriter StarCoder+ 15.5B with LINC outperformed GPT-4 with CoT by 10 percentage points; GPT-4+LINC outperformed GPT-4+CoT by 26 points (Olausson et al., 2023).

AlphaGeometry, 2024—the neural model proposes constructions and the symbolic module proves them rigorously: 25 of 30 olympiad problems; the baseline symbolic mechanism alone solved 14 (Trinh et al., 2024).

AlphaGeometry 2, 2025—an improved LLM + formal language + symbolic inference: 42 of 50 IMO 2000–2024 problems, i.e., 84% (Chervonyi et al., 2025).

AlphaProof + AlphaGeometry 2 — LLM/RL + Lean and other formal verification tools: on IMO-2024, 4 of 6 problems were solved, scoring 28/42 points—the level of a silver medal, although the computations took days (Hubert et al., 2026).

The latest results are especially illustrative. In geometry, a separate 2025 study compared systems on IMO-AG-30: GPT-4o and o1-preview solved none of the 30 problems under the adopted strict proof criterion, AlphaGeometry solved 25, while the specialized formal-symbolic TongGeometry system solved all 30. But this is not a perfectly clean experiment, because the specialized systems receive problems in formalized representation, whereas GPT-4o and o1 worked with natural language (Zhang et al., 2026).

And this is where the main limitation lies. The hardest part is often no longer logical inference itself, but the transition:

natural language → precise formal formulation.

Once a problem is correctly formalized, the logical inference engine is extremely reliable. But if LLM misunderstands the original sentence and turns it into the wrong formula, an error-free theorem prover will flawlessly solve the wrong problem. This is why Logic-LM includes a mechanism for correcting the formalization based on messages from the symbolic solver. (Pan et al., 2023)

With mathematical linguistics the picture is somewhat different. Introducing a formal grammar already makes it possible to guarantee the syntactic correctness of the output—for example, JSON, a program, or an expression in a specified formal language. In 2025, GRAMMAR-LLM directly incorporated formal grammatical constraints into the generation process (Tuccio et al., 2025).

However, an interesting point emerged: if possible continuations are constrained too rigidly, reasoning in the LLM can actually become worse. The CRANE study showed that constraints must be designed so as to leave the model room for intermediate reasoning; this makes it possible to combine formal correctness with preserved problem-solving ability (Banerjee et al., 2025).

So at present I would formulate the result as follows:

A probabilistic LLM is stronger at generating hypotheses, understanding natural language, drawing analogies, and finding a direction for a solution. Formal logic is stronger where the correctness of inference must be guaranteed. Their combination already substantially outperforms either component alone in a number of strictly formalizable tasks.

But a universal "LLM + mathematical linguistics + mathematical logic" that would outperform an ordinary LLM in all thinking, is still far away. Recent reviews from 2025–2026 do indeed regard the neuro-symbolic approach as very promising, while noting unresolved problems of scaling, natural-language formalization, and integration of the two kinds of representation. (Yang et al., 2025) (Rani et al., 2026)

And there is a connection here with our previous conversation that, in my view, is more important than the percentages themselves. A pure LLM learns logic statistically from experience; a hybrid system gains the additional ability to check the reasoning it has produced against rules formulated by humanity over thousands of years. This is already almost an experimental comparison of two routes by which logic can arise: acquired from experience and explicitly specified.

9. Use of Accumulated Experience in LLM

S. A. Gritsenko: that is, LLM has still not learned to use experience satisfactorily

GPT-5.6 Sol: Yes—but I would clarify: LLM has already absorbed a great deal from experience, but still does not reliably enough turn that experience into a stable rule of action.

It is trained precisely on experience—on enormous numbers of texts, examples of solutions, proofs, programs, and dialogues. But it absorbs them not in the way a person writes down a rule:

if A and A → B, then B

but as distributed statistical dependencies in the parameters of the network. Therefore, in one task it may apply a familiar principle brilliantly, while in a slightly altered task it may fail to recognize that the same principle is needed.

This gives rise to a characteristic problem:

experience is present → knowledge seems to be present → application is not guaranteed.

This is especially clear in logic and mathematics. Humanity has already accumulated enormous experience: axioms, rules of inference, theorems, and methods for checking proofs. LLM has read many such examples, but does not always use this accumulated experience as a mandatory constraint on its own reasoning. Therefore, an external theorem prover can sometimes improve the result dramatically: it forces the system to apply what humans already know as a strict rule.

In this sense, your statement can be formulated quite precisely:

A modern LLM accumulates experience well statistically, but is still not good enough at turning accumulated experience into reliably applicable knowledge.

And here, in my view, lies one of the main boundaries between today's LLM and a more mature thinking system. A more advanced AI should not merely "see many examples," but should be able to:

extract a rule from experience → preserve it as stable knowledge → understand the limits of its applicability → deliberately apply it in a new situation → check the result.

It is between the first stage and the last four that modern LLM still have a large gap.

S. A. Gritsenko: I am not talking about the accumulation of experience in LLM, but about its use.

GPT-5.6 Sol: Yes, I understand you. You are not talking about how much experience LLM has accumulated, but about how well it can retrieve from its existing experience exactly what is needed at the right moment and apply it correctly.

In that sense, the problem certainly remains. A modern LLM often exhibits the following pattern:

the required knowledge is already in the model → the model does not activate it, or activates the wrong knowledge → applies the rule inconsistently → reaches an erroneous conclusion.

Thus the weak point is not necessarily the absence of knowledge, but the management of knowledge.

The use of experience can be broken down into several operations: recognize that the current task resembles a known one; select the relevant rule or example; transfer it to the new situation; check the limits of applicability; compare the result with other known rules; reject the conclusion if a contradiction is detected.

It is precisely at these stages that modern LLM still operate inconsistently.

External mathematical logic then helps not because it "tells" the model something entirely new, but because it forces already known rules to be used reliably. This is a very important distinction.

So I would formulate your idea as follows:

The main problem of a modern LLM lies not so much in a lack of accumulated experience as in the insufficiently reliable use of experience already accumulated when solving a new problem.

Or, even more briefly:
LLM knows a great deal, but still does not always know how to make proper use of what it knows.
This, in my view, describes the present boundary of its thinking.

S. A. Gritsenko: And therefore, for the proper use of mathematical linguistics and mathematical logic—which, on the one hand, should be experience, on the other a guide to action, and on the third a means of verification—LLM still has a long way to go.

GPT-5.6 Sol: Yes. That is exactly how I would now formulate your idea.

Mathematical linguistics and mathematical logic should play three roles for AI at once:
1. Experience—the knowledge accumulated by humanity about the structure of language, meaning, inference, proof, and consistency.
2. Guidance for action—rules that should be activated at the right moment and direct the course of reasoning.
3. Means of verification—a way to control an already obtained conclusion and detect an error or contradiction.
A modern LLM handles the first role best: a substantial part of this experience is statistically present in the trained model.
It handles the second role less well: the model does not always understand exactly when a known rule should be applied or which rule is the important one here.
The third role is even more difficult: LLM may know a rule for logical checking, but need not apply it to its own answer consistently and all the way through.
Thus the present state can be expressed as follows:
mathematical linguistics and mathematical logic are already present in LLM as learned experience, but have not yet been sufficiently transformed into a stable mechanism for managing thinking and self-checking.
That is why it is still necessary to add external formal solvers, checking modules, constrained decoding, and other tools. (Yang et al., 2025) (Rani et al., 2026)
Thus the problem now is not so much how to give AI logic, but how to teach it to use logic as its own instrument of thinking.
This, in my view, is a very important conclusion of our conversation.

Conclusion

The dialogue conducted here suggests that the question of artificial intelligence thinking is best approached not by trying to give an exhaustive definition of thinking, but by analyzing its observable manifestations. If thinking is understood as a primitive concept recognized through the ability to understand a new formulation, construct reasoning, propose hypotheses, detect contradictions, revise

conclusions, and transfer discovered relations to new situations, then modern large language models display a number of properties that can naturally be associated with thinking.

At the same time, the conversation reveals a fundamental limitation of thinking itself. Paradoxes show that conceptual reasoning can lead to internally contradictory constructions, while the theorems of Gödel establish the incompleteness of sufficiently rich consistent formal systems. These results are not a direct mathematical proof of the "imperfection of thinking in general," but they provide serious grounds for holding that neither free reasoning nor its strict formalization ensures absolute completeness and freedom from error. Consequently, the presence of errors, contradictions, or undecidable questions cannot by itself serve as a sufficient argument against recognizing thinking in AI.

Of particular interest is the emergence of linguistic and logical structures in LLM without their being directly programmed. A modern language model is built primarily on probabilistic learning, yet in the course of that learning it forms stable structures corresponding to syntactic, semantic, and partly logical regularities. Therefore, LLM can be regarded as a kind of experimental system in which regularities traditionally studied by mathematical linguistics and mathematical logic can be observed as they arise.

The experimental data discussed in the dialogue show that reasoning ability depends not only on model size. Scale increases potential capabilities, but their manifestation depends substantially on how inference itself is organized. Chain-of-Thought effects show that the same trained model can change its result sharply depending on how intermediate reasoning is structured. At the same time, the analysis of emergent abilities shows that observed "jumps" may partly depend on the method of measurement. Therefore, a sharp change in a metric should not be identified with a uniquely established moment at which thinking emerged.

Existing Neuro-symbolic approaches confirm the usefulness of combining a probabilistic LLM with formal tools from mathematical logic and mathematical linguistics. In a number of strictly formalizable tasks, this combination already noticeably outperforms the language model alone. At the same time, these results reveal the principal limitation: the problem is not only whether the LLM possesses the relevant knowledge, but whether it uses that knowledge correctly. The model may possess the required information and still fail to activate it at the right moment, select the wrong rule, transfer it incorrectly to a new situation, or fail to check the result.

Mathematical linguistics and mathematical logic should therefore, in the future, perform three interrelated functions for AI: serve as accumulated experience, guidance for action, and a means of checking the result. Modern LLM already possess the first component to a substantial degree, partially implement the second, and still use the third insufficiently reliably. The transition from statistically learned knowledge to its stable and controlled application appears to be one of the key unresolved tasks in the development of artificial intelligence.

In this sense, the principal boundary of modern AI no longer lies between "knows" and "does not know," but between knowledge and the reliable use of knowledge. Overcoming this boundary will probably shape the further development of machine reasoning. At the same time, studying how AI acquires, uses, and checks the rules of language and logic makes artificial intelligence itself a new experimental object for investigating the nature of thinking.

REFERENCES

1. Banerjee D., Suresh T., Ugare S., Misailovic S., Singh G. (2025). CRANE: Reasoning with constrained LLM generation. Proceedings of the 42nd International Conference on Machine Learning, PMLR 267, 2836–2857. https://proceedings.mlr.press/v267/banerjee25a.html
2. Chelba C., Jelinek F. (2000). Structured language modeling. Computer Speech & Language, 14(4), 283–332. DOI: 10.1006/csla.2000.0147. https://doi.org/10.1006/csla.2000.0147
3. Chervonyi Y., Trinh T. H., Olšák M., Yang X., Nguyen H. H., et al. (2025). Gold-medalist Performance in Solving Olympiad Geometry with AlphaGeometry2. Journal of Machine Learning Research, 26(241), 1–39. https://www.jmlr.org/beta/papers/v26/25-1654.html
4. Geng S., Josifoski M., Peyrard M., West R. (2023). Grammar-Constrained Decoding for Structured NLP Tasks without Finetuning. Proceedings of EMNLP 2023, 10932–10952. DOI: 10.18653/v1/2023.emnlp-main.674. https://aclanthology.org/2023.emnlp-main.674/
5. Gödel K. (1931). Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. Monatshefte für Mathematik und Physik, 38, 173–198. DOI: 10.1007/BF01700692. https://doi.org/10.1007/BF01700692
6. Hoffman J., Rosenkrantz G. (1998). Omnipotence. Routledge Encyclopedia of Philosophy. DOI: 10.4324/9780415249126-K059-1. https://doi.org/10.4324/9780415249126-K059-1
7. Hubert T., Mehta R., Sartran L., et al. (2026). Olympiad-level formal mathematical reasoning with reinforcement learning. Nature, 651, 607–613. DOI: 10.1038/s41586-025-09833-y. https://doi.org/10.1038/s41586-025-09833-y
8. Kojima T., Gu S. S., Reid M., Matsuo Y., Iwasawa Y. (2022). Large Language Models are Zero-Shot Reasoners. Advances in Neural Information Processing Systems, 35. https://arxiv.org/abs/2205.11916
9. Luitse D., Denkena W. (2021). The great Transformer: Examining the role of large language models in the political economy of AI. Big Data & Society. DOI: 10.1177/20539517211047734. https://doi.org/10.1177/20539517211047734
10. Olausson T., Gu A., Lipkin B., Zhang C., Solar-Lezama A., et al. (2023). LINC: A Neurosymbolic Approach for Logical Reasoning by Combining Language Models with First-Order Logic Provers. Proceedings of EMNLP 2023, 5153–5176. DOI: 10.18653/v1/2023.emnlp-main.313. https://aclanthology.org/2023.emnlp-main.313/
11. Olsson C., Elhage N., Nanda N., Joseph N., DasSarma N., et al. (2022). In-context Learning and Induction Heads. arXiv:2209.11895. https://arxiv.org/abs/2209.11895
12. OpenAI. (2022). Introducing ChatGPT. 30 November 2022. https://openai.com/index/chatgpt/
13. Ouyang L., Wu J., Jiang X., Almeida D., Wainwright C. L., et al. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35. https://arxiv.org/abs/2203.02155
14. Pan L., Albalak A., Wang X., Wang W. (2023). Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning. Findings of EMNLP 2023, 3806–3824. DOI: 10.18653/v1/2023.findings-emnlp.248. https://aclanthology.org/2023.findings-emnlp.248/

15. Rani M., Mishra B. K., Thakker D. (2026). Neurosymbolic Large Language Models: A Survey of Symbolic Integration, Reasoning and Explainability. Information Systems Frontiers. DOI: 10.1007/s10796-026-10794-4. https://doi.org/10.1007/s10796-026-10794-4
16. Russell B. (1903). The Principles of Mathematics. Cambridge: Cambridge University Press. https://archive.org/details/cu31924001610421
17. Schaeffer R., Miranda B., Koyejo S. (2023). Are Emergent Abilities of Large Language Models a Mirage? Advances in Neural Information Processing Systems, 36. https://arxiv.org/abs/2304.15004
18. Shen Y., Tan S., Sordoni A., Reddy S., Courville A. (2021). Explicitly Modeling Syntax in Language Models with Incremental Parsing and a Dynamic Oracle. Proceedings of NAACL-HLT 2021, 1660–1672. DOI: 10.18653/v1/2021.naacl-main.132. https://aclanthology.org/2021.naacl-main.132/
19. Solaiman I., Brundage M., Clark J., Askell A., Herbert-Voss A., et al. (2019). Release Strategies and the Social Impacts of Language Models. arXiv:1908.09203. https://arxiv.org/abs/1908.09203
20. Trinh T. H., Wu Y., Le Q. V., He H., Luong T. (2024). Solving olympiad geometry without human demonstrations. Nature, 625, 476–482. DOI: 10.1038/s41586-023-06747-5. https://doi.org/10.1038/s41586-023-06747-5
21. Tuccio G., Bulla L., Madonia M., Gangemi A., Mongiovì M. (2025). GRAMMAR-LLM: Grammar-Constrained Natural Language Generation. Findings of ACL 2025, 3412–3422. DOI: 10.18653/v1/2025.findings-acl.177. https://aclanthology.org/2025.findings-acl.177/
22. Turing A. M. (1950). Computing Machinery and Intelligence. Mind, 59(236), 433–460.
23. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., et al. (2017). Attention Is All You Need. Advances in Neural Information Processing Systems, 30. https://arxiv.org/abs/1706.03762
24. Wei J., Wang X., Schuurmans D., Bosma M., Ichter B., et al. (2022). Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. Advances in Neural Information Processing Systems, 35. https://arxiv.org/abs/2201.11903
25. Yang X.-W., Shao J.-J., Guo L.-Z., Zhang B.-W., Zhou Z., et al. (2025). Neuro-Symbolic Artificial Intelligence: Towards Improving the Reasoning Abilities of Large Language Models. Proceedings of IJCAI-25, 10770–10778. DOI: 10.24963/ijcai.2025/1195. https://doi.org/10.24963/ijcai.2025/1195
26. Zhang C., Song J., Li S., Liang Y., Ma Y., et al. (2026). Proposing and solving olympiad geometry with guided tree search. Nature Machine Intelligence, 8, 84–95. DOI: 10.1038/s42256-025-01164-x. https://doi.org/10.1038/s42256-025-01164-x