TAXONOMIZING ARGUMENTS FOR LANGUAGE MODEL SKEPTICISM IN THE LANGUAGE OF THE THEORY OF COMPUTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Alongside acknowledgement of genuine success and hype, language models (LMs), and in particular autoregressive language models (ALMs), have attracted skepticism: many believe that they are inherently limited in their capabilities. We systematically translate arguments for skepticism into the more precise language of the theory of computation. We contribute novel formalizations of several prominent arguments that were previously only described informally. We aim to enable analyses of skepticism of LMs that are more systematic and precise, and to distinguish conjecture, intuition, and formally proven results.

1 INTRODUCTION

Language models, and specifically autoregressive (large) language models (ALMs), have achieved remarkable success in a variety of natural language processing tasks (Brown et al., 2020; Touvron et al., 2023; Bai et al., 2023; DeepSeek-AI, 2024).

Alongside widespread acknowledgement of utility and success, LMs have also attracted skepticism regarding their capabilities, both present and prospective (Bender & Koller, 2020; Bender et al., 2021; Marcus, 2022; LeCun, 2022). Before the rise of LMs, prominent skepticism of machine learning systems that are LM-like included Penrose (2016), Fodor (2000), Pearl (2018), Minsky & Papert (1969), and Searle (1980). Arguably, Ada Lovelace's early caution: "[t]he Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform." (Lovelace, 1843) could also be viewed as an early expression of skepticism of LM-like systems.

The theory of computation provides a surprisingly convenient language for formalizing and analyzing skepticism of LMs: some argue that some relevant tasks are not computable, others argue that relevant tasks are computationally intractable, and others argue that even if relevant tasks are computable and tractable, learning to perform those tasks is infeasible: the available data may not determine them, extracting them from the data may be computationally intractable, or training may fail to find them.

In this paper, we systematically translate arguments for skepticism of LMs into the more precise language of the theory of computation. We present a taxonomy organized by the type of computational limitation each argument invokes. We apply it to classify several prominent arguments that were previously only described informally, and to formalize them where possible. We aim to enable analyses of skepticism of LMs that are more systematic and precise, and to distinguish conjecture, intuition, and formally proven results.

By "formalization" we mean expressing informal claims in the precise language of the theory of computation, not proving or disproving the claims. Not all arguments in our taxonomy admit full formalization. Some can be expressed as precise mathematical claims (e.g., about computational complexity or learnability). Others (what we call "category error" arguments) can only be classified and resist formal expression. A key finding of our analysis is that even the arguments that can be formalized are far from admitting proof. Usually, either the relevant cognitive science is insufficiently developed, or worst-case computational complexity results do not transfer to performance on real-world data distributions. Formalization is nevertheless valuable: it clarifies what is being

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

claimed, what assumptions are implicit, and what evidence would be needed to support or refute the claim.

Contributions: (1) To our knowledge, we provide the first systematic taxonomy of (A)LM skepticism organized by the type of computational limitation invoked (uncomputability, intractability, unlearnability, insufficient representational capacity, or category error). (2) We classify prominent arguments (Chomsky et al. (2023), Bender et al. (2021), Bender & Koller (2020), LeCun (2022), Fodor (2000), among others) within this taxonomy, and where possible, express them as precise claims in the language of the theory of computation. (3) We situate the growing body of formal results on the computational capabilities of specific architectures within the taxonomy, distinguishing formal results as proven instances of taxonomy types from formal results as supporting evidence for informal claims.

One pattern that emerges from our taxonomy is that many skeptical arguments ultimately turn out to be claims about learnability (LEARN), even when they are initially framed in terms of stronger notions such as uncomputability or insufficient representational capacity.

The rest of the paper is organized as follows. Section 2 provides background on the theory of computation and its application to analyzing AI systems. In Section 3, we present our taxonomy of LM skepticism in the language of computation. We then apply the taxonomy to specific arguments from the literature in Section 5. Section 6 discusses whether the assumption $P \neq NP$ is questionable in this context.

2 BACKGROUND

The theory of computation has been applied to analyze systems from the beginning of computer science. Turing (1936) and Church (1936) provided (equivalent) formalizations for notions of computability and proved the existence of uncomputable functions. Cook (1971) and Karp (1972) initiated the study of computational complexity, including the theory of NP-completeness. Valiant (1984) initiated the study of computational learning theory. Chomsky (1956; 1959) introduced the hierarchy of formal languages: regular, context-free, context-sensitive, and recursively enumerable. Each level of the hierarchy is characterized by a corresponding class of automata: finite automata, pushdown automata, linear bounded automata, and Turing machines, respectively. Each class strictly contains the preceding one. The Chomsky hierarchy has long provided the standard framework for reasoning about the representational capacity of computational models, and it continues to be used in recent analyses of neural architectures (Delétang et al., 2023). The tradition of compiling arguments for skepticism of artificial intelligence goes as far back as Turing (1950) (part of his answer was a prediction that a chatbot would be able to pass what would have become the Turing Test by the year 2000). Lappin (2025) touches on recent arguments.

Van Rooij (2008) formalized notions of human cognition in relation to computational tractability and Van Rooij et al. (2024) explicitly applied the framework arguing for the unlearnability of "AGI" through LM-like systems (although note that Guerzhoy (2026) points out the flaws in the argument).

The work of Kallini et al. (2024) is somewhat similar in spirit to ours, in that they take a concrete prediction ostensibly derived from a skeptical theory of LMs and test it empirically.

The literature on scaling laws (Kaplan et al., 2020) and Sutton (2019)'s notable precursor to it represent hypotheses that the skepticism we analyze in this paper is aiming to refute or moderate.

Discourse in the 2020s has been centered on the capabilities and limitations of autoregressive language models (ALMs) specifically, and many of the arguments we analyze focus on ALMs. However, some skeptical arguments predate the rise of ALMs and apply to LMs more broadly.

A substantial body of recent work proves precise limitations of specific architectures on formal tasks (Strobl et al., 2024). We discuss its relation to our taxonomy in Section 4.

Table 1: A taxonomy of arguments for LM skepticism.

Category	Label	Argument
Computability	COMPUT	Human-level language generation is (COMPUT_a) uncomputable, or may be (COMPUT_b) uncomputable using Turing machines. Uncomputable by LMs.
Computational complexity	COMPLEX	Human-level language generation is (COMPLEX_a) computable, or may be (COMPLEX_b) computable but computationally intractable using Turing machine-equivalent models of computation. Intractable for LMs.
Learnability	LEARN_S	According to *Poverty of Stimulus* (POS) arguments, a child learner cannot acquire grammar with the amount of samples the child is exposed to without a strong inductive bias. This is primarily an argument of the sample complexity of learning grammar (LEARN_C). (POS arguments can also technically be seen as being about the identifiability of the correct grammar (LEARN_S); note, however, that for a large enough class of possible grammars, the correct grammar will not be identifiable from a finite sample in general.)
	LEARN_C LEARN_O	*Efficient learnability and trainability.* The data determines human-level language generation, but (LEARN_C) extracting it from the data is computationally intractable, or (LEARN_O) the training procedure used for LMs does not find it.
Representation capacity	REPR	LMs do not have sufficient representation capacity to perform human-level language generation.
Category error	CATEG	Human-level language generation is fundamentally not a computational process, and therefore cannot conceptually be performed by LMs.

3 A Taxonomy of LM Skepticism in the Language of Computation

We sort arguments for LM skepticism by the kind of limitation they invoke. We read each argument as a claim about some notion of human-level language generation, for example passing the Turing Test (Turing, 1950). There are five categories.

COMPUT (computability) arguments hold that human-level language generation cannot be computed by Turing machines, and therefore not by LMs. COMPLEX (computational complexity) arguments grant that it is computable but hold that it is intractable. For both, we write (a) when the limitation is asserted outright and (b) when it is hedged with "may be". LEARN (learnability) arguments hold that LMs cannot learn it. They come in three sub-types. In a LEARN_S (statistical) argument, the data does not determine the target. In a LEARN_C (computational) argument, the data determines it, but extracting it takes infeasibly many samples or infeasible computation. In a LEARN_O (optimization) argument, an efficient learner may exist, but the training procedure actually used does not find the solution. REPR (representation capacity) arguments hold that LMs cannot represent the mechanism that human-level language generation requires. CATEG (category error) arguments hold that human-level language generation is not a computational process at all, so LMs cannot perform it in the relevant sense.

The first four categories can in principle be stated as precise claims. CATEG arguments can be classified but not formalized. When an argument says that LMs cannot produce some output, we read it as REPR unless it is about every computing device. When it says that LMs do not encode some computable mechanism, we read it as LEARN. Table 1 summarizes the categories.

Result	Object	Category	Epistemic status
Chomsky (1956)	finite-state automata; English nested dependencies	REPR	proven, given empirical premises about English
Shieber (1985)	context-free grammars; Swiss German cross-serial dependencies	REPR	proven, given empirical premises about Swiss German
Delétang et al. (2023)	trained RNNs, LSTMs, transformers, memory-augmented networks; length generalization on formal-language tasks	$\mathsf{LEARN}_{\mathsf{O}}$ / REPR (not disentangled)	empirical (theory-guided)
Hahn (2020)	hard-attention transformers; PARITY, DYCK-2	REPR	proven (unconditional)
Hao et al. (2022)	hard-attention transformers $\subseteq \mathrm{AC}^0$	REPR	proven (unconditional)
Merrill & Sabharwal (2023)	log-precision transformers $\subseteq \mathrm{TC}^0$	REPR	proven; separation conditional on $\mathrm{TC}^0 \neq \mathrm{NC}^1$
Chen et al. (2024)	L-layer decoder-only transformers; sequential composition	REPR	proven (unconditional)
Peng et al. (2024)	transformer layers; function composition	REPR	partly conditional on $\mathrm{L} \neq \mathrm{NL}$
Lin et al. (2021)	autoregressive parameterizations	REPR (via COMPLEX subcomputation)	conditional (standard assumptions)
Gold (1967)	language identification in the limit (positive data)	$\mathsf{LEARN}_{\mathsf{S}}$	proven; impossible for superfinite classes
Chomsky et al. (2023) (§5)	LMs; human-level generation	$\mathsf{LEARN}_{\mathsf{S}}$ / REPR / CATEG	conjecture / intuition
Bender et al. (2021) (§5)	LMs $\approx n$-grams	REPR	conjecture (analogy)
Van Rooij et al. (2024) (§5)	AGI via LM-like systems	$\mathsf{LEARN}_{\mathsf{C}}$	claimed conditional proof; see Guerzhoy (2026)
Kleinberg & Mullainathan (2024)	language *generation* in the limit, same data as Gold (1967)	$\mathsf{LEARN}_{\mathsf{S}}$	proven possible; opposite verdict under changed criterion
Merrill & Sabharwal (2024)	transformers + poly-length CoT $= \mathrm{P}$	$\mathsf{LEARN}_{\mathsf{C}}$	proven possible; lifts the TC^0 bound under changed inference mode

Table 2: Selected formal and empirical results (top) and prominent informal skeptical arguments (middle), classified by taxonomy category and epistemic status. The two positive results (bottom, below the double rule) are listed apart: they are not skeptical claims but show that a limitation is lifted under a changed success criterion or inference mode.

4 Formal Results on Model Capabilities and the Taxonomy

A substantial and rapidly growing body of work has precisely characterized the computational capabilities of specific neural architectures, often with proofs. Most of this work concerns transformers (Vaswani et al., 2023; Strobl et al., 2024). Those results are sometimes glossed as suggestive of skepticism about LMs, and they correspond to claims about specific architectures and specific sets of languages. A skeptical argument based on those results would ordinarily use them as suggestive evidence or as an intuition pump. Table 2 classifies selected results by taxonomy category and epistemic status, and Appendix A discusses each entry. The selection is illustrative rather than exhaustive. Strobl et al. (2024) provide a comprehensive survey.

5 CASE STUDIES

In this section, we present case studies of using our framework to formalize and analyze specific arguments for LM skepticism. Most of these arguments were made informally and often at book length, and it is hard to restate them precisely in a fair way. We acknowledge that nuance will be lost in the process. That is both a limitation and a feature: nuance is good, but too much nuance can make a theory too flexible to refute or falsify, even if its implications are incorrect. It may be better to attribute the formal versions of the arguments to us and credit the original authors with the inspiration. Nevertheless, we believe that all the intuitions we formalize are held by prominent researchers.

5.1 Chomsky's New York Times arguments Prominent linguist Noam Chomsky has made various arguments against LMs over the years. His prominent op-ed in the New York Times (Chomsky et al., 2023) fits our taxonomy in several ways.

"True intelligence is [...] capable of moral thinking" Chomsky argues as follows (Chomsky et al., 2023):

> True intelligence is also capable of moral thinking. This means constraining the otherwise limitless creativity of our minds with a set of ethical principles that determines what ought and ought not to be [...] ChatGPT and its brethren are constitutionally unable to balance creativity with constraint. They either overgenerate (producing both truths and falsehoods, endorsing ethical and unethical decisions alike) or undergenerate (exhibiting noncommitment to any decisions and indifference to consequences)

We start with this passage because it shows how our taxonomy can be used to interrogate (rather than analyze) arguments. The passage is vague. It could be read as an argument of type COMPUT, COMPLEX, LEARN, or REPR (an LM could not possibly produce outputs that look like the result of moral thinking). It could also be read as a CATEG argument (only moral thinking can produce outputs that look like moral thinking, and LMs cannot do moral thinking). We would argue that a charitable reading here does not assert unproven mathematical claims. It asserts a CATEG ("category error") argument instead, since the passage singles out moral thinking, which is commonly thought to be a spiritual trait.

Parsing errors Chomsky et al. (2023) argue:

> Because these programs cannot explain the rules of English syntax, for example, they may well predict, incorrectly, that "John is too stubborn to talk to" means that John is so stubborn that he will not talk to someone or other (rather than that he is too stubborn to be reasoned with).

This appears to be a REPR (representation capacity) argument since no claim is made that the task is not computable or unlearnable.

Language acquisition

> The human mind is not, like ChatGPT and its ilk, a lumbering statistical engine for pattern matching…a young child acquiring a language is developing — unconsciously, automatically and speedily from minuscule data

This might be a LEARN_{C} (poverty-of-the-stimulus) argument: human-level language is learnable by humans because of innate structure, and is unlearnable to human level by LMs even if they could converge over infinite data. Other interpretations are possible.

5.2 The "stochastic parrots" argument Bender et al. (2021) coined the evocative phrase "stochastic parrots". We take it to refer to the analogy between LMs and n-gram models. For example, Bender et al. (2021) refer to the models' "haphazardly stitching together sequences of

linguistic forms" (i.e., repeating high-probability segments verbatim, a common issue for n-gram models with high n). They also analogize to n-grams frequently.

N-gram models do not have the capacity to generate human-level language. For example, an n-gram model with a fixed n cannot perform addition of two arbitrary integers (see also Section 5.7).

The assertion that LMs are analogous to n-grams is a REPR argument.

Note the contrast with the proven REPR results discussed in Section 4: the n-gram analogy asserts a representational limitation of trained LMs without specifying a function class, whereas, e.g., Hahn (2020) proves that hard-attention transformers cannot recognize PARITY or DYCK-2.

An argument that is made implicitly by Van Rooij et al. (2024) that may be informed by the stochastic parrot analogy is that LMs, like n-grams, need an exponential amount of data points to learn the data. This is a LEARN_C argument. Note that this intuition concerns the amount of data, whereas the formal result of Van Rooij et al. (2024) (Section 5.4) is LEARN_C.

5.3 The Need for World Models argument LeCun (2022) can be interpreted as making an argument that world models are necessary for human-level language generation, and that LMs do not have world models. This is arguably an implicit LEARN_S argument: universal approximation results suggest that world models are in principle representable by sufficiently large neural networks, so the question is arguably whether they are learnable from text data alone.

Pearl (2018) makes a related argument that reasoning about interventions is necessary for intelligence, and that LMs cannot reason about interventions. This is also arguably an implicit LEARN_S argument, since no argument is put forward that LMs lack the representational capacity to incorporate world models. Here, a formal result is available: interventional quantities are in general not identifiable from observational data (Bareinboim et al., 2022). It is not clear that this result applies to LMs. Text includes reports of interventions and their outcomes, so it is arguably not purely observational data.

5.4 van Rooij et al.'s claimed unlearnability of AGI proof Van Rooij et al. (2024) explicitly make the case for the unlearnability of "AGI" through LM-like systems. They formalize their claim and present a proof. Guerzhoy (2026) points out that van Rooij et al.'s proof actually proves a different claim. Under standard complexity-theoretic assumptions, it shows that there exist polytime-sampleable functions that are not learnable by LM-like systems. The headline claim is that human-like/level language generation specifically is unlearnable by LM-like systems.

Nevertheless, the argument put forth by Van Rooij et al. (2024) is a paradigmatic example of a LEARN_C argument: human-level language generation may be possible using LM-like architectures, but is not learnable. The claimed limitation is computational. The difficulty lies in efficiently extracting the target behavior from samples. The argument relies on a reduction under complexity-theoretic assumptions. A notable feature of Van Rooij et al. (2024) is that the argument is formalized and therefore amenable to precise analysis.

5.5 Searle's Chinese Room argument: CATEG Searle (1980) imagines a person who knows no Chinese locked in a room with a rulebook for manipulating Chinese symbols. By following the rules, the person returns answers to Chinese questions that are indistinguishable from those of a native speaker, even though the person does not understand Chinese. Since the person does everything that a computer running the program would do, Searle concludes that

> Instantiating a computer program is never by itself a sufficient condition of intentionality.

In short, syntax (symbol manipulation) is not sufficient for semantics (meaning). This is a CATEG argument. Understanding is held not to be a computational process, and no program has it. As with Penrose (2016), Searle allows that machines can think (he holds that the brain is one), but not in virtue of running a program.

Unlike the other arguments we analyze, the Chinese Room grants the behavioral premise: the room's outputs are stipulated to pass the Turing Test. Under the behavioral reading of human-level language

generation in Section 3, the argument does not claim that an LM will fail to produce any output. In the sense of the CATEG category, what LMs cannot "perform" is generating language from understanding. We would argue that for this reason no formal or empirical capability result can confirm or refute the argument. Many CATEG arguments are ambiguous between this reading and a behavioral one (e.g., Chomsky's argument from moral thinking in Section 5.1). Searle's arguably is not.

5.6 The Bender and Koller argument: CATEG Bender & Koller (2020) appear to argue that ungrounded language models do not conceptually understand and generate language in the sense that humans do.

For example:

> The language modeling task, because it only uses form as training data, cannot in principle lead to learning of meaning

This is a CATEG argument. It is arguably a narrower version of Searle's (Section 5.5), since it attributes the lack of meaning to training on form alone.

Accordingly, unlike Searle, Bender and Koller do leave open the possibility that grounded LMs (including simply Vision-Language Models or computer code with expected outputs) could have the capacity to learn meaning in the Bender-Koller sense.

5.7 The Bender and Koller argument: LEARN_S Bender & Koller (2020) also predict that LMs will not have world models because they lack grounding in the world. As a result, LMs will not be able to generate language that is accurate with respect to the world. This is arguably a LEARN_S argument. One might imagine that with infinite data, grounding may be unnecessary at the limit. If that is the case, this is an interesting case of a LEARN_S argument in disguise.

Interestingly, the intuition of Bender & Koller (2020) led them to predict in their paper that "[incorrect completions with GPT-2] show that this problem [of completing the prompt 'Three plus five equals'] is beyond the current capability of GPT-2, and, we would argue, any pure LM." If this were a LEARN_S argument that was falsified by the success of GPT-3 (Brown et al., 2020), it might have been a case where scaling had not hit its limits yet, or a case where a world model (or something like it) is actually learnable from observational data alone. Learning arithmetic (or something like it) has been demonstrated by Lee et al. (2023). Bender and Koller's claim concerns a class of mechanisms and examples, and GPT-3's success is a single example whose relevance depends on how the original claim is read. If individual system failures are sufficient evidence for a broader limitation, as the argument assumes when drawing on GPT-2's shortcomings, then isolated successes do not by themselves overturn that conclusion. If instead performance is evaluated across a broader distribution of uses, then both failures and successes provide evidence, and their weight depends on that distribution. We would argue that the disagreement is therefore mostly about what kind of evidence is relevant for evaluating the original claim.

We believe this is a LEARN_S argument. It is plausible that the hypothesis itself is driven by representation capacity considerations. The examples of arithmetic problems strongly suggest that GPT-2 learned a mechanism that generated digits or numbers arbitrarily. That would imply an n-gram-like model.

5.8 The argument from Gödel's Theorems An argument that is sometimes made against LMs is that they are limited in their reasoning capabilities by Gödel's incompleteness theorems (Gödel, 1931). For example, this argument is made by Penrose (2016) regarding artificial intelligence more generally. When the argument is made directly, it is a COMPUT_a argument. Whether Penrose's version is a COMPUT_a or a CATEG argument is debatable. Penrose (2016) proposes that human cognition may well be a computational process in some sense, but not a Turing machine-equivalent one. In this way, the claim is that humans compute something that Turing machines cannot compute, which is COMPUT_a by definition. However, if the non-Turing-equivalent process is taken to be something qualitatively different from computation as we understand it, the argument is closer to a CATEG argument.

5.9 Fodor's Frame Problem argument The most prominent argument in Fodor (2000) is about complexity: Fodor posits that human-level cognition (and so language generation) requires solving

the "frame problem". The frame problem is computationally intractable in general (Ginsberg & Smith, 1988). Since no claim is made about humans being able to solve the exact frame problem always, this is arguably a COMPLEX_b argument.

It should be noted that Fodor appears sympathetic to LEARN_S arguments for learning language as well as for developing language capacities (Fodor, 2000).

5.10 Ada Lovelace's argument Ada Lovelace's early caution: "[t]he Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform." (Lovelace, 1843) can be interpreted as an early expression of skepticism of LM-like systems. This is arguably a COMPUT_a argument. The Analytical Engine is equivalent to a Turing machine. Lovelace argues for limitations of what outputs can be computed.

5.11 Gary Marcus's argument Gary Marcus has made various LM-skeptical arguments over the years (Marcus, 2022). A key theme in Marcus (2022) is that human-level language generation requires symbolic reasoning, and that LMs (or LMs at feasible scales) cannot perform symbolic reasoning. Read as a claim that current architectures lack the capacity for symbolic reasoning, this is arguably a REPR argument. Read as a claim that symbolic reasoning is computationally intractable for any connectionist system regardless of scale, it is a COMPLEX_a argument.

5.12 Minsky and Papert's argument The arguments of Minsky & Papert (1969) are often framed as REPR arguments. Single-layer perceptrons provably lack the representational capacity for certain functions. A broader interpretation of these arguments is that perceptron-like systems cannot *in practice* represent more complex functions. On this interpretation, the limitation is computational tractability instead of representational impossibility, which is a COMPLEX claim.

6 Is $P \neq NP / NP \not\subseteq BPP$ a questionable assumption in this context?

COMPLEX and LEARN_C arguments explicitly or implicitly rely on assumptions such as $P \neq NP$ and $NP \not\subseteq BPP$. These assumptions are widely believed but unproven in the theory of computation. We argue that in the context of arguing about the capabilities of language models from a physicalist point of view[1], tacitly assuming that $P \neq NP$ may be *begging the question*, since human-level language-generation-capable physical organisms exist.

Some preliminaries:

1. Human-level language is a language that is capable of encoding physical science and mathematics at the level of 2026 human knowledge. Among other things, this means being able to encode code for Turing Machines in a language that is structurally equivalent to languages for Von Neumann-like architectures.
2. Human-level language is a set of instances, and NP-harness therefore does not automatically transfer to them. However, if one believes that COMPLEX or LEARN_C have force in LM skepticism, one must believe that the harness does transfer to human-level languages, perhaps there are no known and/or findable inductive biases that would get around the hardness, or perhaps because one believes the Imapgliazzo's Heuristica Impagliazzo world is unlikely or doesn't rescue human language.

The argument is as follows.

1. Human knowledge of physical science and mathematics was produced from observations that could in principle be digitized
2. The Earth is a physical system
3. If learning to generate human-level language is computationally intractable and the hardness transfers to the instances of interest, the Earth producing language-generation-capable physical organisms is an extremely unlikely event

[1] https://plato.stanford.edu/entries/physicalism/

The conclusion (3) follows from premises (1)–(2): under physicalism, a physical process produced physical organisms that generate human-level language. A physical process can be viewed as a computation, and is therefore subject to computational complexity constraints.

Premise (1) is debatable (Aaronson, 2013) but arguably reasonable.

Epistemic doubt about the P vs NP question is merely somewhat uncommon (Gasarch, 2019). We suggest that if one is a physicalist and one believes that $\mathsf{LEARN}_\mathsf{C}$ arguments entail that human languages are not learnable unless $P = NP$, the most plausible resolution in light of the argument above is that $P = NP$ or that human language came about through a cosmological anthropic-principle-like setting (Tipler, 1988).

It is important to note that this argument is distinct from the argument from humans being able to learn human-level language. It is widely believed that humans have a built-in prior structure that allows them to learn human-level language from the data available to them. Rather this is an argument about earth as a physical system producing human language.

Two objections are as follows. First, one might argue that the computation performed by evolution is highly parallel and took billions of years. That argument needs to contend with the superpolynomial complexity of learning posited by the skeptical argument. The anthorpic-principle-like argument is a live option, mentioned above.

7 Conclusion

In this paper, we have presented a taxonomy and framework for translating arguments for skepticism of language models (LMs) into the language of the theory of computation.

In several case studies, we have arguably demonstrated that rephrasing informal arguments in more precise and standardized language can help clarify the arguments. This helps us better understand them or identify flaws in them.

Our aim is not to take a position on many of those arguments. However, it is important to note that for general enough models, no formalized version of any of these arguments has been proven correct. The "category error" arguments are unlikely to be amenable to formal proof either way (Section 5.5).

The formal results in Section 4 are consistent with this: the available proofs concern specific idealized combinations of architecture, inference procedure, and formal task, and extending them to claims about human-level language generation arguably requires additional, conjectural assumptions.

It is important to note that, even with claims such as "SAT is intractable" (Cook, 1971), a resolution seems far away: P vs NP is still open, and furthermore, $P = NP$ does not imply practical tractability. Worse, even if the assumption that $P \neq NP$ is correct, the efficiency of SAT solvers is improving and is arguably not well understood (Ganesh & Vardi, 2020). This by itself limits the power of $\mathsf{COMPLEX}$ and $\mathsf{LEARN}_\mathsf{C}$ arguments, and suggests caution in interpreting them.

We argue that our framework can help clarify and analyze arguments for LM skepticism, and can help distinguish conjecture, intuition, and formally proven results. Perhaps surprisingly, we find that a large number of skeptical arguments reduce to claims about learnability despite their diverse surface presentations. The case of Van Rooij et al. (2024) is a notable example of the utility of formalization. Because the argument was formalized and a proof was presented, Guerzhoy (2026) was able to point out what the proof actually shows: that there exist polytime-sampleable functions that are not learnable by LM-like systems.

Most arguments in this area do not admit definitive proofs, and even formal results often depend on unproven assumptions such as $P \neq NP$. We note that, unlike in many other contexts, assumptions such as $P \neq NP$ are not benign in this context.

The utility of our framework will often be in interrogating arguments and clarifying the underlying assumptions. It has so far always been the case that substantial barriers exist to buttressing AI-skeptical arguments with formal proofs (Guerzhoy, 2026). Nevertheless, we believe that our framework can help advance the conversation on LM skepticism.

AI USE STATEMENT

In this work, we used generative AI tools (large language models) only to edit the paper for grammar, spelling, wording, and readability. The authors reviewed and revised all such edits and take full responsibility for the content of the paper. We have not used generative AI tools to develop the taxonomy or other conceptual frameworks, formulate mathematical claims, provide ingredients for or assist in writing proofs, propose or refine hypotheses, design research methodology, interpret results, or assist with translation. Generating synthetic data, cleaning or reformatting datasets, implementing methods, and qualitative data analysis are not applicable to this work.

REFERENCES

Scott Aaronson. *Quantum computing since Democritus*. Cambridge University Press, 2013.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen Technical Report. *arXiv preprint arXiv:2309.16609*, 2023.

Elias Bareinboim, Juan Correa, Duligur Ibeling, and Thomas Icard. On pearl's hierarchy and the foundations of causal inference. In Hector Geffner, Rita Dechter, and Joseph Halpern (eds.), *Probabilistic and Causal Inference: the Works of Judea Pearl*, pp. 507–556. ACM Books, 2022.

Emily M. Bender and Alexander Koller. Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5185–5198, 2020.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret 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 (FAccT '21)*, pp. 610–623, New York, NY, USA, 2021. ACM.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Lijie Chen, Binghui Peng, and Hongxun Wu. Theoretical limitations of multi-layer transformer, 2024. URL https://arxiv.org/abs/2412.02975.

Noam Chomsky. Three models for the description of language. *IRE Trans. Inf. Theory*, 2:113–124, 1956. URL https://api.semanticscholar.org/CorpusID:17432009.

Noam Chomsky. On certain formal properties of grammars. *Inf. Control.*, 2:137–167, 1959. URL https://api.semanticscholar.org/CorpusID:16792674.

Noam Chomsky, Ian Roberts, and Jeffrey Watumull. The False Promise of ChatGPT. *The New York Times*, March 2023. Opinion/Guest Essay. https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html.

Alonzo Church. An Unsolvable Problem of Elementary Number Theory. *American Journal of Mathematics*, 58(2):345–363, 1936.

Stephen A. Cook. The Complexity of Theorem-Proving Procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pp. 151–158, 1971.

DeepSeek-AI. DeepSeek-V3 Technical Report. *arXiv preprint arXiv:2412.19437*, 2024.

Grégoire Delétang, Anian Ruoss, Jordi Grau-Moya, Tim Genewein, Li Kevin Wenliang, Elliot Catt, Chris Cundy, Marcus Hutter, Shane Legg, Joel Veness, and Pedro A. Ortega. Neural networks and the chomsky hierarchy, 2023. URL https://arxiv.org/abs/2207.02098.

Jerry A. Fodor. *The Mind Doesn't Work That Way: The Scope and Limits of Computational Psychology*. MIT Press, 2000.

Vijay Ganesh and Moshe Y. Vardi. On the Unreasonable Effectiveness of SAT Solvers, 2020.

William I. Gasarch. Guest Column: The Third P=?NP Poll. *ACM SIGACT News*, 50(1):38–59, 2019.

Matthew L. Ginsberg and David E. Smith. Reasoning about Action I: A Possible Worlds Approach. *Artificial Intelligence*, 35(2):165–195, 1988.

Kurt Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38(1):173–198, 1931.

E. Mark Gold. Language identification in the limit. *Inf. Control.*, 10:447–474, 1967. URL https://api.semanticscholar.org/CorpusID:12438987.

Michael Guerzhoy. Barriers to Complexity-Theoretic Proofs that Achieving AGI Using Machine Learning Is Intractable. *Computational Brain & Behavior (accepted)*, 2026. https://arxiv.org/html/2411.06498v1.

Michael Hahn. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, December 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00306. URL http://dx.doi.org/10.1162/tacl_a_00306.

Yiding Hao, Dana Angluin, and Robert Frank. Formal language recognition by hard attention transformers: Perspectives from circuit complexity, 2022. URL https://arxiv.org/abs/2204.06618.

Russell Impagliazzo. No better ways to generate hard np instances than picking uniformly at random.

Julie Kallini, Isabel Papadimitriou, Richard Futrell, Kyle Mahowald, and Christopher Potts. Mission: Impossible Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14691–14714, Bangkok, Thailand, August 2024. Association for Computational Linguistics. https://aclanthology.org/2024.acl-long.787/.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models. *arXiv preprint arXiv:2001.08361*, 2020.

Richard M. Karp. Reducibility among Combinatorial Problems. In Raymond E. Miller and James W. Thatcher (eds.), *Complexity of Computer Computations*, pp. 85–103. Plenum Press, New York, 1972.

Jon Kleinberg and Sendhil Mullainathan. Language generation in the limit, 2024. URL https://arxiv.org/abs/2404.06757.

Shalom Lappin. *Understanding the Artificial Intelligence Revolution: Between Catastrophe and Utopia*. CRC Press, 2025.

Yann LeCun. A Path towards Autonomous Machine Intelligence. Technical report, Courant Institute of Mathematical Sciences, New York University and Meta AI, June 2022. Version 0.9.2. https://openreview.net/pdf?id=BZ5a1r-kVsf.

Nayoung Lee, Kartik Sreenivasan, Jason D. Lee, Kangwook Lee, and Dimitris Papailiopoulos. Teaching Arithmetic to Small Transformers. *arXiv preprint arXiv:2307.03381*, 2023.

Chu-Cheng Lin, Aaron Jaech, Xin Li, Matthew R. Gormley, and Jason Eisner. Limitations of autoregressive models and their alternatives, 2021. URL https://arxiv.org/abs/2010.11939.

Ada Augusta Lovelace. Notes by the Translator. *Scientific Memoirs*, 3:666–731, 1843. Appended to: L. F. Menabrea, "Sketch of the Analytical Engine Invented by Charles Babbage, Esq.," translated by Ada Augusta, Countess of Lovelace.

Gary Marcus. Deep Learning Is Hitting a Wall. *Nautilus*, March 2022. https://nautil.us/deep-learning-is-hitting-a-wall-238440/.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers, 2023. URL https://arxiv.org/abs/2207.00729.

William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought, 2024. URL https://arxiv.org/abs/2310.07923.

Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge, MA, 1969.

Judea Pearl. Theoretical Impediments to Machine Learning with Seven Sparks from the Causal Revolution. Technical Report R-475, University of California, Los Angeles, Department of Computer Science, January 2018. http://ftp.cs.ucla.edu/pub/stat_ser/r475.pdf.

Binghui Peng, Srini Narayanan, and Christos Papadimitriou. On limitations of the transformer architecture, 2024. URL https://arxiv.org/abs/2402.08164.

Roger Penrose. *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*. Oxford University Press, 2016.

Jorge Pérez, Pablo Barceló, and Javier Marinkovic. Attention is turing-complete. *Journal of Machine Learning Research*, 22(75):1–35, 2021.

John R. Searle. Minds, brains, and programs. *Behavioral and Brain Sciences*, 3(3):417–424, 1980. doi: 10.1017/S0140525X00005756.

Stuart M Shieber. Evidence against the context-freeness of natural language. In *The Formal complexity of natural language*, pp. 320–334. Springer, 1985.

Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. What formal languages can transformers express? a survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024. ISSN 2307-387X. doi: 10.1162/tacl_a_00663. URL http://dx.doi.org/10.1162/tacl_a_00663.

Richard Sutton. The Bitter Lesson. *Incomplete Ideas (blog)*, 13(1):38, 2019.

Frank J Tipler. The anthropic principle: a primer for philosophers. In *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association*, volume 1988, pp. 27–48. Cambridge University Press, 1988.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. LLaMA 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*, 2023.

Alan Mathison Turing. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, 1936.

Alan Mathison Turing. Computing Machinery and Intelligence. *Mind*, 59(236):433–460, 1950.

Leslie G. Valiant. A Theory of the Learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Iris Van Rooij. The Tractable Cognition Thesis. *Cognitive Science*, 32(6):939–984, 2008.

Iris Van Rooij, Olivia Guest, Federico Adolfi, Ronald de Haan, Antonina Kolokolova, and Patricia Rich. Reclaiming AI as a Theoretical Tool for Cognitive Science. *Computational Brain & Behavior*, 7(4):616–636, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

A Summary of Empirical and Theoretical Results

This appendix expands on each entry of Table 2: what the result establishes, what it assumes, and why we assign it the taxonomy category and epistemic status that we do.

A.1 Three Models for the Description of Language (Chomsky, 1956) Chomsky (1956) proves that natural language sits above the finite-state level of the Chomsky hierarchy, so an LM below that level does not have the representational capacity to model natural language. The proof is rigorous given two premises: an empirical one, that English contains nested dependencies (e.g., *if…then*, *either…or*), and the idealization that such nesting is unbounded. By an argument of the same form, from cross-serial dependencies in Swiss German, some natural languages were later shown to be non-context-free (Shieber, 1985), placing them above the pushdown-automaton level of the Chomsky hierarchy as well. We therefore list both results in Table 2 as proven given empirical premises about the language, rather than as unconditional.

This REPR result applies to some but not all LMs; for example, Transformers with hard attention (with Chain-of-Thought) were shown to be Turing-complete (Pérez et al., 2021).

A.2 Neural Networks and the Chomsky Hierarchy (Delétang et al., 2023) Delétang et al. (2023) train standard architectures on formal-language tasks drawn from each level of the Chomsky hierarchy and measure how far they generalize to longer inputs. Transformers perform poorly even on some regular tasks, whereas memory-augmented networks reach higher levels. This is a theory-guided empirical study of what trained architectures learn, which our taxonomy accommodates without claiming proof. Because it measures learned behavior, it cannot separate a lack of representation capacity (REPR) from a failure to learn a representable solution (LEARN_{O}); we therefore classify it as LEARN_{O} / REPR.

A.3 Theoretical Limitations of Self-Attention in Neural Sequence Models (Hahn, 2020) Hahn (2020) proves that hard-attention transformers (without Chain-of-Thought) cannot recognize PARITY (whether the input contains an even number of ones) or DYCK-2 (whether two kinds of brackets are properly nested), with no unproven assumptions. Pushdown automata handle both. It is an example of a REPR result, and it shows that pushdown automata outperform idealized transformers on some tasks.

A.4 Formal Language Recognition by Hard Attention Transformers (Hao et al., 2022) Hao et al. (2022) place hard-attention transformers inside the circuit class AC^0, so their abilities are bounded by a known barrier. That AC^0 cannot compute PARITY is proven, so the limitation of the transformer is inherited rather than assumed. While the result is unconditional, like Hahn (2020), it strengthens the conclusion from a single-task result to one covering an entire complexity class.

A.5 The Parallelism Trade-Off: Limitations of Log-Precision Transformers (Merrill & Sabharwal, 2023) Merrill & Sabharwal (2023) show that log-precision transformers lie within logspace-uniform TC^0. They therefore cannot solve NC^1-hard problems unless $\mathsf{TC}^0 = \mathsf{NC}^1$, a conjecture that is not widely believed to be true. Examples of such problems include simulating a finite automaton and evaluating a Boolean formula, both of which require sequential computation. The parallelism that makes transformers trainable at scale is what confines them to constant-depth circuits. This is a real limitation that rests on an unproven separation ($\mathsf{TC}^0 \neq \mathsf{NC}^1$), in the same way that other arguments rest on $P \neq NP$.

A.6 Theoretical Limitations of Multi-Layer Transformers (Chen et al., 2024) Chen et al. (2024) prove unconditionally that a constant-depth decoder-only transformer needs polynomial width to compose L functions one after another. The lower bound quantifies how a fixed architectural depth makes multi-step chain reasoning expensive. It falls on the unconditionally proven side, and it targets composition, which is central to multi-step reasoning.

A.7 On Limitations of the Transformer Architecture (Peng et al., 2024) Peng et al. (2024) show that a single transformer layer cannot compose functions over large domains, with the multi-

layer version conditional on $L \neq NL$. It complements the function-composition result of Chen et al. (2024). Being partly conditional, it sits in the middle of the epistemic axis, between the unconditional results and the more heavily assumption-dependent ones.

A.8 Limitations of Autoregressive Models and Their Alternatives (Lin et al., 2021) Lin et al. (2021) prove a bound that concerns autoregression specifically: under standard assumptions, an autoregressive model cannot compactly represent distributions whose prefix probabilities are hard to compute, whereas some non-autoregressive models can. The skepticism analyzed here is therefore specific to autoregressive LMs. A COMPLEX intractability fact about a required subcomputation induces a REPR failure.

A.9 Language Identification in the Limit (Gold, 1967) Gold (1967) proves that a learner cannot determine which language it faces from positive examples alone whenever the family of candidate languages is superfinite — even when the family consists of one infinite language and otherwise only finite ones. It is the standard formal basis for doubting language learnability. It is a LEARN_{S} argument, but its force depends entirely on the choice of success criterion.

A.10 Language Generation in the Limit (Kleinberg & Mullainathan, 2024) Kleinberg & Mullainathan (2024) prove the mirror image of the result of Gold (1967), and we contrast the two deliberately. If success means generating new valid strings rather than identifying the language, the task becomes achievable for every countable family — purely because the success criterion changed. What does the work is often the stipulation of the problem rather than any theorem.

A.11 The False Promise of ChatGPT (Chomsky et al., 2023) Chomsky et al. (2023) criticize ChatGPT, holding that it lacks the understanding the public ascribes to it. Although it generates text well, that text does not demonstrate genuine understanding of the content; it produces plausible responses on the basis of large-scale training data without comprehending the material. The moral-reasoning part of the argument is CATEG and the syntax part is REPR. It sits toward the conjecture end of the epistemic axis.

A.12 On the Dangers of Stochastic Parrots (Bender et al., 2021) Bender et al. (2021) coin the phrase "stochastic parrots." The claim is that a (large) LM is like a parrot concatenating words according to statistical probability without understanding their meaning, however fluent and plausible the output. On this view LMs are essentially n-gram models. The argument identifies no function class and proceeds by analogy, so it conveys an intuition, whereas Hahn (2020) and Merrill & Sabharwal (2023) give proofs. All are REPR, but their epistemic statuses are opposite.

A.13 Reclaiming AI as a Theoretical Tool for Cognitive Science (Van Rooij et al., 2024) Van Rooij et al. (2024) argue that AGI cannot be learned by LM-like systems, formalize the claim, and give a proof. Guerzhoy (2026) points out that the proof establishes something weaker than claimed: it shows that some polynomial-time sampleable functions are unlearnable, not that human-level generation is unlearnable. This is a good illustration of why formalizing an argument, even imperfectly, is what makes its weak points locatable.

A.14 The Expressive Power of Transformers with Chain of Thought (Merrill & Sabharwal, 2024) Merrill & Sabharwal (2024) prove that with a polynomial number of chain-of-thought steps, a transformer recognizes exactly the class P, escaping the constant-depth TC^0 ceiling that constrains single-pass decoding. A logarithmic number of steps lifts the bound to L, a linear number suffices to simulate automata, and the full jump to P requires polynomially many. This shows that REPR bounds are indexed to an inference model: a limitation proven for (transformer, single forward pass) does not transfer to (transformer, decoding with a scratchpad) — and the latter is what deployed models actually use.

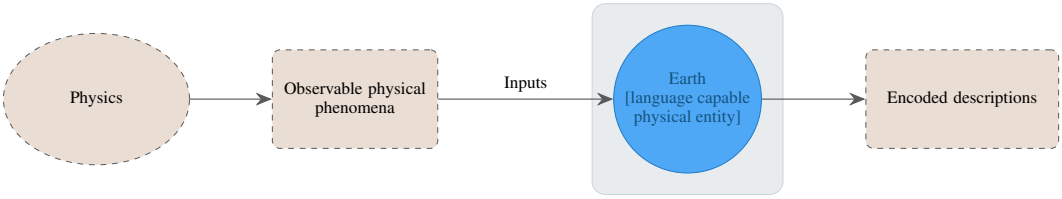

Figure 1: Under physicalism, the Earth is a physical system that takes observable physical phenomena as inputs and produces encoded descriptions of them. Treating that system as a blackbox computation is what brings it under computational complexity constraints.

B THE $P \neq NP$ ASSUMPTION UNDER PHYSICALISM

In this appendix, we expand and illustrate the argument that a physicalist who believes a strong version of $\mathsf{LEARN}_\mathsf{C}$ arguments might be compelled to believe $P = NP$.

The argument relies on the Earth's being a physical system that eventually produced language-capable physical entities (humans) who (inter alia) use their language to encode the laws of physics and mathematics. The "inputs" to the system are observable physical phenomena, and the "outputs" are language.

As argued in Section 6, for a physicalist, this is an instance of a computational process that is subject to the constraints of computational complexity. If the physicalist believes a $\mathsf{LEARN}_\mathsf{C}$ argument (conditional on an assumption such as $P \neq NP$), they need to contend with a solved instance of a solved problem that they believe is intractable.

In the following, we outline possible objections to this argument.

1. **Humans (+ human evolution) might have solved an *easy* instance of a hard problem, perhaps given their particular inductive biases.** This is a live possibility, but it is inconsistent with a strong version of $\mathsf{LEARN}_\mathsf{C}$ arguments (e.g., Van Rooij et al., 2024) which conclude from NP-hardness the impossibility of learning a human-level language.
2. **In our setting, the solution is underdetermined: many languages that are powerful enough to express physics and mathematics are possible, and finding one of them might be easier than finding all of them or a specific one.** It is a limitation of our current paper that we cannot exclude this possibility formally. We argue that Von Neumann-like languages are roughly isomophic to each other.
3. **Our setting does not correspond to developing a language that is necessarily as powerful as English.** We contend that a language that is powerful enough to express physics and mathematics is at least sufficient to describe and execute Turing Machines, and is therefore in some sense powerful. We accept that our setting does not directly deal with e.g. English's grammatical subtleties.