Taxonomizing Arguments for Language Model Skepticism in the Language of the Theory of Computation

Michael Guerzhoy [1]

Abstract

Alongside acknowledgement of genuine success and hype, 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 ALMs that are more systematic and precise, and to distinguish conjecture, intuition, and formally proven results. Our framework distinguishes in-principle limitations from empirical ones, and surfaces an in-between category: arguments that appear to assert contingent limitations but, if correct, would establish fundamental ones not yet known to obtain.

1. Introduction

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, ALMs 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 ALMs, prominent skepticism of machine learning systems that are ALM-like included Penrose (2016), Fodor (2000), Pearl (2018), and Minsky & Papert (1969). 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 ALM-like systems.

The theory of computation provides a surprisingly convenient language for formalizing and analyzing skepticism of ALMs: some arguments are 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 the tasks from feasible amounts of data is infeasible due to sample complexity considerations.

In this paper, we systematically translate arguments for skepticism of ALMs into the more precise language of the theory of computation. We present a taxonomy organized by the type of computational limitation each argument invokes, and apply it to classify and, where possible, formalize several prominent arguments that were previously only described informally. We aim to enable analyses of skepticism of ALMs 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 them. Not all arguments in our taxonomy admit full formalization: some can be expressed as precise mathematical claims (e.g., about computational complexity or learnability), while others (e.g., "category error" arguments) can only be classified but resist formal expression. A key finding of our analysis is that even the arguments that *can* be formalized are far from admitting proof — usually because the relevant cognitive science is insufficiently developed, or because worst-case computational complexity results do not account for average-case complexity. Formalization is nevertheless valuable: it clarifies what is being 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 ALM 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,

[1]Division of Engineering Science, University of Toronto, Toronto, Canada. Correspondence to: Michael Guerzhoy <guerzhoy@cs.toronto.edu>.

PhilML Workshop @ ICML, Seoul, South Korea. 2026.

within this taxonomy, and where possible, express them as precise claims in the language of the theory of computation. (3) We illustrate the utility of formalization by examining a case where it helps identify flaws in a purported proof (Van Rooij et al., 2024; Guerzhoy, 2026).

A recurring finding of our analysis is that a surprisingly large number of skeptical arguments, once classified, reduce to claims about learnability (type 3), even when they appear on the surface to invoke stronger notions such as uncomputability or insufficient representational capacity.

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. The tradition of compiling arguments for skepticism of artificial intelligence goes as far back as Turing (1950) (whose answer, partly, 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's work (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 ALM-like systems (although note that Guerzhoy (2026) points out the flaws in the argument).

Kallini et al. (2024) is similar in spirit, taking a concrete prediction derived from a skeptical theory of ALMs and testing it empirically.

Guerzhoy (2024) presents a case study in analyzing the argument of Bender & Koller (2020).

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

3. A Taxonomy of ALM Skepticism in the Language of Computation

In this section, we present a taxonomy of arguments for ALM skepticism that we find useful (Table 1).

Categories 1 and 2 are each split into (a) and (b) variants to distinguish between arguments that assert a limitation definitively and those that use hedged "may be" language. This distinction is worth preserving because the evidential burden differs significantly: a definitive claim requires a proof, while a "may be" claim is a conjecture or an expression of intuition.

Category		Argument
Computability	1a	Human-level language generation is uncomputable using Turing machines and therefore uncomputable by ALMs.
	1b	Human-level language understanding **may be** uncomputable using Turing machines and therefore uncomputable by ALMs.
Computational Complexity	2a	Human-level language generation is computable but computationally intractable using Turing machine-equivalent models of computation and therefore intractable for ALMs.
	2b	Human-level language generation **may be** computable but computationally intractable using Turing machine-equivalent models of computation and therefore may be intractable for ALMs.
Learnability	3	Human-level language generation is computable and computationally tractable using Turing machine-equivalent models of computation but unlearnable from feasible amounts of data by ALMs.
Representation capacity	4	ALMs do not have sufficient representation capacity to perform human-level language generation.
Category error	5	Human-level language generation is fundamentally not a computational process, and therefore cannot conceptually be performed by ALMs.

Table 1. A taxonomy of arguments for ALM skepticism.

While the Theory of Computation provides precise definitions, the concept of "human-level language generation," used in our taxonomy, is inherently difficult to formalize. In this paper, we analyze all arguments for ALM skepticism as necessarily involving some notion of human-level language generation since all ALMs generate language. One possibility, which not all skeptics would accept, is the system's passing the Turing Test (Turing, 1950), or its expanded variants: the system's ability to generate language that is indistinguishable by humans from human-generated language in any context, including in interactive settings, and including by experts who verify world knowledge and reasoning.

When analyzing a claim that an ALM cannot produce a particular **output**, we default to interpreting that as a claim about representation capacity, unless the claim is explicitly about any computing device rather than ALMs specifically. Claims about not being able to produce a particular output often do not deny that a computational process *could* produce the output, but imply that the mechanism for that is

unknown and likely beyond what an ALM could represent.

On the other hand, when analyzing a claim that purports that ALMs do not encode a particular (presumably computable) **mechanism**, we default to interpreting that as a claim about learnability, because universal approximation theorems suggest that a very wide range of mechanisms are representable with neural networks, and therefore the question is whether they are learnable from data.

The distinction between representation capacity arguments and learnability arguments, as ordinarily expressed, is subtle. Arguments often rely on intuitions about what mechanisms can and cannot be represented by ALMs by analogizing ALMs to simpler probabilistic models such as n-grams or perceptrons. That intuition is arguably correct early in the training process and arguably becomes more incorrect with scaling training time and training set size (Kalimeris et al., 2019; Kaplan et al., 2020). An example of a type 3 argument that is arguably driven by representation capacity considerations is discussed in Appendix A.4.

4. Case studies

In this section, we present case studies of using our framework to formalize and analyze specific arguments for ALM skepticism. A challenge is representing arguments that were originally informal in a precise formal language 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 be too flexible to refute or falsify, even if its implications are incorrect.

4.1. The "stochastic parrots" argument

"Stochastic parrots", coined in (Bender et al., 2021), is an evocative phrase we take to refer to the analogy between ALMs 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) and analogize to n-grams frequently.

N-gram models do not have the capacity to generate human-level language: for example, for a fixed n, an n-gram model cannot perform addition of two arbitrary integers (see also Appendix A.4).

The assertion that ALMs are analogous to n-grams is an argument of type 4.

An argument that is made implicitly in (Van Rooij et al., 2024) that may be informed by the stochastic parrot analogy is that ALMs, like n-grams, need an exponential amount of data points to learn the data. This is an argument of type 3.

4.2. 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 ALMs do not have world models. This is arguably an implicit argument of type 3: universal approximation results suggest that world models are in principle representable by sufficiently large neural networks, so the bottleneck is whether they are learnable from text data alone, rather than whether they can be encoded at all.

Pearl (2018) makes a related argument that reasoning about interventions is necessary for intelligence, and that ALMs cannot reason about interventions. This is also arguably an implicit argument of type 3, since no argument is put forward that ALMs lack the representational capacity to incorporate world models.

4.3. 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 ALM-like systems. They formalize their claim and present a proof. Guerzhoy (2026) points out that van Rooij's proof is actually a proof that there exist polytime-sampleable functions that are not learnable by ALM-like systems under standard complexity-theoretic assumptions, a different claim than the headline claim of proving that human-like/level language generation specifically is unlearnable by ALM-like systems.

Nevertheless, the *argument* put forth by Van Rooij et al. (2024) is a paradigmatic example of a type 3 argument: human-level language generation may be possible using ALM-like architectures, but is not learnable. A notable feature of (Van Rooij et al., 2024) is that the argument is formalized and therefore amenable to precise analysis.

4.4. The argument from Gödel's Theorems

An argument that is sometimes made against ALMs 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 an argument of type 1(a). Often, the implication is indirectly a type 5 argument. The distinction is that, for example, Penrose (2016) proposes that human cognition may well be a computational process in some sense, but that it is not a Turing machine-equivalent computational process.

4.5. 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",

which, in general, is computationally intractable (Ginsberg & Smith, 1988). Since no claim is made about humans being able to solve the exact frame problem always, this is arguably an argument of type 2(b).

It should be noted that Fodor appears sympathetic to type 3 arguments for learning language as well as for developing language capacities (Fodor, 2000).

4.6. Additional case studies

Additional case studies are presented in Appendix A: Chomsky's New York *Times* arguments (Appendix A.1; types 5, 4, and 3); Ada Lovelace's argument (Appendix A.2; type 1(a)); Gary Marcus's argument (Appendix A.3; types 4 and 2(a)); the Bender and Koller type-3 argument about world models (Appendix A.4); the Bender and Koller type-5 argument that ungrounded language modeling "cannot in principle lead to learning of meaning" (Appendix A.5); and Minsky and Papert's argument (Appendix A.6; types 4 and 2).

5. Is P $\neq$ NP a questionable assumption in this context?

Arguments of type 2 and 3 explicitly or implicitly rely on assumptions such as P $\neq$ NP and NP $\not\subseteq$ BPP, which are widely believed but unproven assumptions in the theory of computation.

We say the arguments rely on those assumptions because $P = NP$ would imply that problems that appear intractable intuitively are actually tractable, and the arguments from complexity and learnability rely on intuitions about intractability.

We argue that in the context of arguing about the capabilities of language models from a physicalist point of view, tacitly assuming that P $\neq$ NP may be begging the question.

If the Earth is a physical system that produced physical organisms that generate human-level language through a physical process and the problem of producing human-level-language-capable physical organisms is computationally intractable assuming $P \neq NP$, it must be the case that

- $P = NP$, or
- Human-level-language-capable physical organisms emerging is an exponentially unlikely event that nevertheless happened on Earth, perhaps through a cosmological anthropic-principle-like selection effect

Believing physicalism and believing that a physical process cannot tractably produce human-level language generation while also believing that P $\neq$ NP is not strictly speaking logically inconsistent, but it requires believing that an exponentially unlikely event happened. On the other hand, epistemic doubt about the P vs NP question is merely somewhat uncommon (Gasarch, 2019). If one needs to choose between believing in an exponentially unlikely event and an unpopular but existing position in the theory of computation, the choice seems clear.

6. Conclusion

We presented a taxonomy and framework for translating arguments for skepticism of autoregressive language models (ALMs) into the language of the theory of computation, and applied it in multiple case studies.

We argue that our framework can help clarify and analyze arguments for ALM skepticism, and can help distinguish conjecture, intuition, and formally proven results. Perhaps surprisingly, we find that a large number of skeptical arguments, despite diverse surface presentations, reduce to claims about learnability: that human-level language generation may be computable and representable, but not learnable from feasible amounts of data by ALMs.

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 that the proof is actually a proof that there exist polytime-sampleable functions that are not learnable by ALM-like systems, a different claim than the headline claim of proving that human-level language generation specifically is unlearnable by ALM-like systems. This illustrates that formalization, even when imperfect, enables precise analysis and identification of flaws in arguments that might otherwise go unnoticed.

We emphasize that most arguments cannot be accompanied by a definitive proof (and even proofs are often conditional on unproven assumptions such as P $\neq$ NP), and that caution is warranted in interpreting them. Nevertheless, we believe there is value in formalizing and analyzing arguments more precisely.

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). Equally, it seems more likely to us that if ever a human-level language system is demonstrated, it will happen before a formal proof that it is truly human-level, let alone before a formal proof that such a system is possible.

Nevertheless, we believe that our framework can help advance the conversation on ALM skepticism.

References

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Bender, E. M. and Koller, A. 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.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. 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. doi: 10.1145/3442188.3445922.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Siber, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are Few-Shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Chomsky, N., Roberts, I., and Watumull, J. The false promise of ChatGPT. *The New York Times*, March 2023. URL https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html. Opinion/Guest Essay.

Church, A. An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58(2):345–363, 1936. doi: 10.2307/2371045.

Cook, S. A. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pp. 151–158, 1971. doi: 10.1145/800157.805047.

DeepSeek-AI. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

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

Gasarch, W. I. Guest column: The third p=? np poll. *ACM SIGACT News*, 50(1):38–59, 2019.

Ginsberg, M. L. and Smith, D. E. Reasoning about action i: A possible worlds approach. *Artificial intelligence*, 35(2): 165–195, 1988.

Gödel, K. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38(1):173–198, 1931. doi: 10.1007/BF01700692.

Guerzhoy, M. Occam's razor and bender and koller's octopus. In Al-azzawi, S., Biester, L., Kovács, G., Marasović, A., Mathur, L., Mieskes, M., and Weissweiler, L. (eds.), *Proceedings of the Sixth Workshop on Teaching NLP*, pp. 128–129, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.teachingnlp-1.18/.

Guerzhoy, M. Barriers to complexity-theoretic proofs that achieving AGI using machine learning is intractable. *Computational Brain & Behavior*, 2026. URL https://arxiv.org/html/2411.06498v1.

Kalimeris, D., Kaplun, G., Nakkiran, P., Edelman, B., Yang, T., Barak, B., and Zhang, H. Sgd on neural networks learns functions of increasing complexity. *Advances in neural information processing systems*, 32, 2019.

Kallini, J., Papadimitriou, I., Futrell, R., Mahowald, K., and Potts, C. Mission: Impossible language models. In Ku, L.-W., Martins, A., and Srikumar, V. (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. doi: 10.18653/v1/2024.acl-long.787. URL https://aclanthology.org/2024.acl-long.787/.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Karp, R. M. Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W. (eds.), *Complexity of Computer Computations*, pp. 85–103. Plenum Press, New York, 1972.

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

LeCun, Y. A path towards autonomous machine intelligence. Technical report, Courant Institute of Mathematical Sciences, New York University and Meta AI, June 2022. URL https://openreview.net/pdf?id=BZ5a1r-kVsf. Version 0.9.2.

Lee, N., Sreenivasan, K., Lee, J. D., Lee, K., and Papailiopoulos, D. Teaching arithmetic to small transformers. *arXiv preprint arXiv:2307.03381*, 2023.

Lovelace, A. A. 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.

Marcus, G. Deep learning is hitting a wall. *Nautilus*, March 2022. URL https://nautil.us/deep-learning-is-hitting-a-wall-238440/.

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

Pearl, J. 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. URL http://ftp.cs.ucla.edu/pub/stat_ser/r475.pdf.

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

Sutton, R. The bitter lesson. *Incomplete Ideas (blog)*, 13(1): 38, 2019.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., and others. LLaMA 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Turing, A. M. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, 1936. doi: 10.1112/plms/s2-42.1.230.

Turing, A. M. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950. doi: 10.1093/mind/LIX.236.433.

Valiant, L. G. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Van Rooij, I. The tractable cognition thesis. *Cognitive science*, 32(6):939–984, 2008.

Van Rooij, I., Guest, O., Adolfi, F., de Haan, R., Kolokolova, A., and Rich, P. Reclaiming AI as a theoretical tool for cognitive science. *Computational Brain & Behavior*, 7 (4):616–636, 2024.

A. Additional Case Studies

A.1. Chomsky's New York Times arguments

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

A.1.1. "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 since it is an example of the utility of our taxonomy in *interrogating* (rather than analyzing) arguments: because of the vagueness of the passage, this could be read as either an argument of type 1-4 (an ALM could not possibly produce outputs that look like the result of moral thinking) or an argument of type 5 (only moral thinking can produce outputs that look like moral thinking, and ALMs cannot do moral thinking). We would argue that a charitable reading here is not asserting unproven mathematical claims, but rather asserting a Type 5 "category error" argument, since specifically moral thinking, commonly thought to be a spiritual trait, is singled out.

A.1.2. Parsing errors

Chomsky et al. (2023) argues:

> 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 Type 4 (representation capacity) argument since no claim is made that the task is not computable or unlearnable.

A.1.3. 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 Type 3 argument: human-level language is learnable by humans because of innate structure, and is unlearnable to human level by ALMs even if they could converge over infinite data. Other interpretations are possible.

A.2. 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 ALM-like systems. This is arguably an argument of type 1(a): it refers to the Analytical Engine, which is equivalent to a Turing Machine, and it argues for limitations of what outputs can be computed.

A.3. Gary Marcus's argument

Gary Marcus has made various ALM-skeptical arguments over the years (Marcus, 2022).

A key theme in Marcus (2022) is that human-level language generation requires symbolic reasoning, and that ALMs (or ALMs at feasible scales) cannot perform symbolic reasoning. This is arguably an argument of type 4 when read as a claim about current architectures lacking the capacity for symbolic reasoning, or of type 2(a) if read as a claim that symbolic reasoning is computationally intractable for any connectionist system regardless of scale.

A.4. The Bender and Koller argument: Type 3

Bender & Koller (2020) also make the prediction that, because ALMs do not have grounding in the world, they will not have world models, and therefore will not be able to generate language that is accurate with respect to the world. This is arguably an argument of type 3 — 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 type 3 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 was an argument of type 3 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).

While we believe this is a Type 3 argument, it is plausible that the hypothesis itself is driven by representation capacity considerations: the examples of arithmetic problems strongly suggest that the mechanism learned by GPT-2 was one where digits or numbers were generated arbitrarily, implying an n-gram-like model.

A.5. The Bender and Koller argument: Type 5

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 an argument of type 5.

We note that Bender and Koller do leave open the possibility that *grounded* ALMs (including simply Vision-Language Models or computer code with expected outputs) could have the capacity to learn meaning in the Bender-Koller sense.

A.6. Minsky and Papert's argument

Minsky & Papert (1969) are often represented as correctly arguing type 4 regarding perceptrons specifically: single-layer perceptrons provably lack the representational capacity for certain functions. Their arguments can be interpreted more broadly: that perceptron-like systems, even with multiple layers, cannot tractably represent more complex functions at available scales. Under this broader reading, the argument becomes one of type 2, since the claim shifts from architectural impossibility to computational intractability.