PLOS Complex Systems

RESEARCH ARTICLE

Can adversarial attacks by large language models be attributed?

Manuel Cebrian[1]*, **Andres Abeliuk**[2], **Jan Arne Telle**[3]

1 Center for Automation and Robotics (CAR), Spanish National Research Council (CSIC-UPM), Madrid, Spain, **2** Department of Computer Science, University of Chile, Santiago, Chile, **3** Department of Informatics, University of Bergen, Bergen, Norway

* manuel.cebrian@csic.es

Abstract

Attributing outputs from Large Language Models (LLMs) in adversarial settings—such as cyberattacks and disinformation campaigns—presents significant challenges that are likely to grow in importance. We approach this attribution problem from both a theoretical and empirical perspective, drawing on formal language theory (identification in the limit) and data-driven analysis of the expanding LLM ecosystem. By modeling an LLM's set of possible outputs as a formal language, we analyze whether finite samples of text can uniquely pinpoint the originating model. Our results show that under mild assumptions of overlapping capabilities among models, certain classes of LLMs are fundamentally *non-identifiable* from their outputs alone. We delineate four regimes of theoretical identifiability: (1) an infinite class of deterministic (discrete) LLM languages is not identifiable (Gold's classical result from 1967); (2) an infinite class of probabilistic LLMs is also not identifiable (by extension of the deterministic case); (3) a finite class of deterministic LLMs is identifiable (consistent with Angluin's tell-tale criterion); and (4) even a finite class of probabilistic LLMs can be non-identifiable (we provide a new counterexample establishing this negative result). Complementing these theoretical insights, we quantify the explosion in the number of plausible model origins (*hypothesis space*) for a given output in recent years. Even under conservative assumptions (each open-source model fine-tuned on at most one new dataset), the count of distinct candidate models doubles approximately every 0.5 years, and allowing multi-dataset fine-tuning combinations yields doubling times as short as 0.28 years. This combinatorial growth, alongside the extraordinary computational cost of brute-force likelihood attribution across all models and potential users renders exhaustive attribution infeasible in practice. Our findings highlight an urgent need for new strategies and proactive governance to mitigate risks posed by un-attributable, adversarial use of LLMs as their influence continues to expand.

OPEN ACCESS

Citation: Cebrian M, Abeliuk A, Telle JA (2026) Can adversarial attacks by large language models be attributed?. PLoS Complex Syst 3(2): e0000085. https://doi.org/10.1371/journal.pcsy.0000085

Editor: Juan Gonzalo Barajas-Ramirez, IPICYT: Instituto Potosino de Investigacion Cientifica y Tecnologica AC, MEXICO

Received: July 30, 2025

Accepted: December 16, 2025

Published: February 23, 2026

Data availability statement: All relevant data are within the manuscript and its Supporting Information files.

Funding: M.C. is funded by grant PID2023-150271NB-C21 funded by MICIU/AEI/ 10.13039/501100011033

(Spanish Ministry of Science, Innovation and University, Spanish State Research Agency) and Google.org's through the Silicon Valley Community Foundation via a grant to Fundación General CSIC. A.A. was supported by the Centro Nacional de Inteligencia Artificial CENIA, FB210017, BASAL, ANID. J.A.T is supported by the Norwegian Research Council, project Machine Teaching for XAI. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Author summary

When AI-generated attacks—from disinformation to cyberattacks—occur, can we reliably trace them back to their originating language model? This paper establishes theoretical limits, showing that in realistic settings, attributing outputs to specific large-language models is provably impossible, even with unlimited data. Empirically, we quantify the explosive growth in the number of plausible model origins, demonstrating how quickly attribution becomes infeasible in practice. These combined results have stark implications for cybersecurity, misinformation mitigation, and AI governance.

1 Introduction

The challenge of attributing outputs from LLMs in the context of adversarial attacks or disinformation campaigns is emerging as a concern for both cybersecurity and information integrity [1–5]. In such settings, *attribution* refers to identifying the specific model responsible for generating harmful or misleading content. This step is essential not only for conducting investigations and determining whether the implicated model should be restricted or decommissioned, but also for mitigating future risks and ensuring accountability in the deployment of LLM-based agents [6–8]. Unfortunately, reliably linking a piece of content to a particular LLM has proven extremely challenging in practice.

The demand for robust attribution is underscored by new AI-governance initiatives. The EU AI Act and U.S. Executive Order 14110 both mandate model-level transparency and risk-mitigation tools, such as watermarking and incident reporting, that implicitly assume one can identify which model produced a given output [25,26]. Our work asks whether that assumption is even feasible in principle. Ultimately, accountability lies with human actors, but pinpointing the source model is a crucial intermediate step—it enables enforcement of regulations and can lead investigators to the responsible parties through the chain-of-custody of AI tools.

Interestingly, the attribution task can be framed in terms of formal language theory, specifically the problem of *language identification in the limit*. This theoretical framework, introduced by Gold [9] and extended by Angluin [10], has been widely studied in theoretical computer science and cognitive science [11]. In our context, we can represent the set of all possible outputs of a given LLM as a formal language (a set of strings over some finite alphabet). Attribution then asks whether a finite sample of observed outputs can uniquely determine which language (and hence which LLM) produced them [12–15].

Framing LLM outputs as formal languages provides a structured way to explore the feasibility of attribution. Given an observed set of outputs S (e.g., a collection of generated texts), we are essentially asking if there exists a unique LLM M in some model class $\mathcal{M}$ whose language $L(M)$ includes S. If two different models M_i and M_j can both generate all strings in S (i.e. $S \subseteq L(M_i) \cap L(M_j)$), then S alone cannot distinguish between M_i and M_j. In practice, fine-tuned models often exhibit substantial overlap in their output spaces, especially if they share training data or base architectures. This overlap means that, under mild assumptions, multiple models may produce the same set of outputs, foreshadowing fundamental limits on attribution certainty.

We adopt Gold's identification-in-the-limit with adversarial presentation as a worst-case lens, standard in security and learning theory. If attribution fails even with unlimited, adversary-controlled evidence, it will a fortiori fail in practical, data-scarce settings. This framing pins down information-theoretic limits before considering engineering heuristics, yielding impossibility results that remain valid when scaled down to real-world constraints. As in cryptography and cybersecurity, analyzing the worst case establishes fundamental boundaries on what attribution can achieve under the most challenging conditions.

In the remainder of this paper, we investigate the theoretical limits of LLM identification under four regimes and then examine empirical trends that exacerbate the attribution problem. Below we summarize these regimes and our main findings for each.

- **Infinite discrete model class: Negative.** If the space of possible LLMs (or languages) is infinite and models produce outputs in a deterministic (discrete) manner, then identification in the limit is not possible. This was proved in Gold's classic result and formalized by Angluin's criteria: intuitively, any infinite class of languages that is sufficiently complex (e.g., containing an infinite language with arbitrarily many finite variants) is *not* identifiable from positive data.
- **Infinite probabilistic model class: Negative.** Allowing models to be probabilistic (assigning probabilities to strings) does not improve identifiability when the class is infinite. In fact, the deterministic case is a special case of the probabilistic case (a formal language can be viewed as a probabilistic language with 0/1 probabilities), so an infinite collection of probabilistic LLMs remains non-identifiable in the limit.
- **Finite discrete model class: Positive.** If the number of candidate LLMs is finite and their outputs are deterministic languages, then identification in the limit becomes possible. In this scenario, because there are only finitely many possible languages, one can eventually find a finite subset of outputs (a *tell-tale set* in Angluin's terminology) for each language. Thus, with enough data, a learning algorithm can converge to the correct model.
- **Finite probabilistic model class: Negative.** Somewhat counter-intuitively, even a finite set of probabilistic LLMs can defy identification. We give the first explicit counterexample—thereby resolving a question left open since Gold's 1967 work—showing that if two probabilistic languages share identical support but differ in their probability distributions, no amount of data (under the standard identification-in-the-limit setting) can reliably distinguish them.

Following our theoretical analysis, we present a data-driven study of the current LLM landscape. The number of publicly known LLMs and fine-tuned variants has exploded in recent years [24], which greatly enlarges the hypothesis space for attribution. We introduce a combinatorial lower bound $N(t)$ on the number of distinguishable model origins at time t and find that $N(t)$ has been growing exponentially, with a troublingly short doubling time (well under one year in recent data). We break down this growth by model modality and by developer region, revealing that multimodal models and contributions from Asia are among the fastest-growing segments. This rapid proliferation means that any brute-force or exhaustive attribution strategy (e.g. comparing an output against every possible model) will become increasingly infeasible.

We also examine the computational hurdles to attribution. Even under optimistic assumptions, performing likelihood-based attribution across all models for a single piece of content could require an astronomical number of operations, pushing the limits of modern supercomputers. We illustrate this with a scenario using the current cumulative parameter count of known models and show that attributing a moderately long text (100k tokens) against all models would take on the order of minutes on the world's fastest supercomputer. Scaling such analysis to nation-wide LLM usage (on the order of 10^{15} tokens/year for the USA) would require infrastructure and time on the order of many days of high-performance computing for just a single attribution query, as summarized in our estimates. Moreover, real-world factors like network propagation of content can further obscure attribution, as malicious actors can route outputs through layers of social networks to mask their origin [18,19]. Paradoxically, recent theoretical work has shown that *language generation in the* *limit* is achievable without identification [23], meaning an agent can eventually mimic a target language's outputs without actually knowing which language it is—pointing to a potential arms race where attackers and defenders can reproduce content indefinitely without exposing the true source [23].

In summary, our contributions are: (i) a rigorous theoretical exposition of why LLM attribution is impossible in three of four fundamental regimes (with a positive result only in the trivial finite-deterministic case), including a new theorem for the probabilistic finite case; (ii) a quantitative analysis of the LLM model landscape growth, demonstrating an unsustainable explosion in the attribution search space; and (iii) a discussion of practical challenges and implications, highlighting the need for new methodologies (e.g. model fingerprinting, heuristic narrowing of candidates, regulation) to address the forensic blind spot created by increasingly ubiquitous LLMs.

Although practitioners have long suspected attribution is hard, there were no formal guarantees quantifying how hard—or under which conditions it is impossible. Our results close this gap by establishing the first rigorous limits on LLM attribution.

2 Infinite classes of discrete LLMs: Impossibility of identification

We first consider the classical scenario of Gold [9]: an infinitely large stream of outputs from an LLM is observed (so every string the model can produce will eventually appear in the sample), and the class of potential models is infinite. In this section, we assume each model M produces a *discrete language* $L(M)$—a set of strings (the model's outputs) with no probabilistic information attached. Identification in this setting means that a learning algorithm, given enough data, will eventually infer the correct language $L(M)$ (and thus the correct model).

Gold formalized *identification in the limit* from positive examples as follows:

Definition 1 (Gold's Identification in the Limit [9])**.** *A class of languages $\mathcal{L}$ is said to be* identifiable in the limit *if there exists a learning algorithm $\mathcal{A}$ such that, for any target language $L^* \in \mathcal{L}$, given an infinite sequence of examples $\langle s_1, s_2, \dots \rangle$ with each $s_i \in L^*$ and each string in L^* appearing at least once in the sequence, the algorithm $\mathcal{A}$ produces a sequence of hypotheses $\langle H_1, H_2, \dots \rangle$ (where each H_n is a language in $\mathcal{L}$) that satisfies:*

(1) For all but finitely many n, $H_n = L^*$.

(2) Each hypothesis H_n is consistent with the data observed up to time n, i.e. $\{s_1, s_2, \dots, s_n\} \subseteq H_n$.

In simpler terms, identifiability in the limit means that as the algorithm sees more and more outputs (eventually seeing every output that the target model can produce), it converges to correctly guessing the target language and does not later change its mind. Gold showed that certain classes of languages are *not* identifiable in the limit from positive data. In particular, any class of languages that is sufficiently rich—for example, containing all finite languages and at least one infinite language—cannot be learned with this criterion.

Formally framing the attribution task in Gold's identification-in-the-limit paradigm (as we do here) provides a unified lens to analyze attribution, connecting earlier empirical approaches (e.g. watermarking, stylometry) to a common theoretical foundation.

Angluin later provided a characterization of identifiable classes with her concept of *tell-tale sets* [10]. We recall Angluin's theorem here, as it will be useful for framing our results:

Theorem 2 (Angluin's Theorem [10])**.** *An indexed family of recursive languages $\{L_i\}_{i\in\mathbb{N}}$ is identifiable in the limit from positive data if and only if there exists a recursively enumerable set of finite subsets $\{T_i\}_{i\in\mathbb{N}}$ (with $T_i \subseteq L_i$ for each i) such that for all $i \neq j$, $T_i \nsubseteq L_j$. In other words:*

(i) *For each i, T_i is a finite subset of L_i.*
(ii) *For each pair $i \neq j$, if $T_i \subseteq L_j$ then L_j is not a proper subset of L_i.*

The sets T_i are sometimes called tell-tale sets *for the language L_i.*

Intuitively, Angluin's theorem says that a class of languages is learnable from positive data if and only if each language L_i in the class has some finite "evidence", the tell-tale subset T_i. The point is that once the strings of T_i have appeared among the sample strings, we need not fear "overgeneralization" in guessing L_i. This is because the true answer, even if it is not L_i, cannot be a proper subset of L_i, and so if the true answer is not L_i we will eventually see a conflict between the data and L_i, which will force us to change our guess. On the other hand, if no finite tell-tale sets exist then the class cannot be learned.

Using this criterion, we can formalize Gold's negative result as a corollary. Specifically, if a class of languages contains an infinite language that has infinitely many finite subsets extendable to different languages in the class, then no finite tell-tale set can exist for that infinite language. This leads to non-identifiability:

Corollary 3 (Non-Identifiability Due to Infinite Languages)**.** *Let $\mathcal{L}$ be a collection of languages such that:*

(i) *$\mathcal{L}$ contains at least one infinite language L_∞.*
(ii) *For every finite subset $S \subset L_\infty$, there exists some language $L' \in \mathcal{L}$ with $S \subseteq L' \subset L_\infty$ (a proper subset).*

Then $\mathcal{L}$ is not identifiable in the limit from positive data.

Proof: Suppose, for sake of contradiction, that $\mathcal{L}$ is identifiable in the limit. By Theorem 2, there must exist a finite tell-tale set $T_{L_\infty} \subset L_\infty$ that distinguishes L_∞ from all its proper subsets in $\mathcal{L}$. However, condition (ii) guarantees that for *every* finite subset T_{L_∞} of L_∞, we can find another language $L' \in \mathcal{L}$ such that $T_{L_\infty} \subseteq L' \subset L_\infty$. This means no finite subset of L_∞ can serve as a unique identifier, contradicting the requirement for identification. Therefore, $\mathcal{L}$ is not identifiable in the limit. □

Corollary 3 is essentially a formal restatement of Gold's observation: if an infinite language can be approximated arbitrarily well by other languages in the class (by matching it on every finite sample but eventually diverging), a learner can never be sure which language is the true target.

We can construct a very simple example of such a class $\mathcal{L}$ to illustrate the concept. Consider an alphabet $\Sigma = \{x\}$ (a single symbol). For each $k \in \mathbb{N}$, define $L_k = \{x^n : 0 < n \leq k\}$ as the set of all strings of length at most k (over Σ). Let $L_\infty = \Sigma^*$ be the set of all finite strings over Σ. Now take $\mathcal{L} = \{L_k : k \in \mathbb{N}\} \cup \{L_\infty\}$. In this family, L_∞ is infinite, and for any finite sample of strings $S \subset L_\infty$, if m is the length of the longest string in S, then $S \subseteq L_m \subset L_\infty$ with $L_m \in \mathcal{L}$. Thus L_∞ has no finite tell-tale set (any finite set of examples from L_∞ could have come from some L_k with k large enough). By Corollary 3, $\mathcal{L}$ is not identifiable. Indeed, a learner seeing strings of increasing length can never be sure whether eventually some maximal length will appear (indicating a finite language L_k) or the strings will continue indefinitely (indicating L_∞).

Translating back to LLMs, we may fine-tune a base model on *any* finite set S of strings and obtain a model M_S whose language is exactly S. (For simplicity we state the argument with $L(M_S) = S$; relaxing to the weaker—but still sufficient—condition $L(M_S) \supseteq S$ leaves the combinatorial lower bound unchanged.) This corresponds to an open-ended model class in which one model (e.g. a large base checkpoint) can generate an unbounded set of outputs, yet for every *finite* collection of outputs an attacker might observe there exists another model (a suitably fine-tuned or restricted version) that produces precisely those outputs and no others. Under such conditions no attribution algorithm can reliably distinguish the unbounded model from the myriad fine-tuned variants. In an ecosystem where models are continually specialized, this "infinite ladder" of ever-narrower languages is not merely theoretical but an expected consequence of routine fine-tuning.

Thus, we conclude that when considering an infinite class of possible LLMs (with languages that can nest in this way), identification in the limit is provably impossible. This negative result sets a theoretical upper bound on what we can hope to achieve with attribution: even under idealized conditions (infinite data, no noise, etc.), there are fundamental ambiguities that cannot be resolved.

3 Infinite classes of probabilistic LLMs

In practice, LLMs are probabilistic by nature—they produce distributions over outputs. One might wonder if incorporating probability information could help distinguish models where pure language membership cannot. In an identification-in-the-limit setting, however, we typically assume the learner has access only to which outputs appear, not the true underlying probabilities. (We will later discuss alternative learning criteria where samples are drawn according to the model's probabilities rather than adversarially.) Under the standard definition, if the class of models is infinite, introducing probabilities does not overcome the fundamental obstacle identified above.

To formalize this, we first define what we mean by a probabilistic language and identification in that context:

Definition 4. *A* probabilistic formal language L_p *over alphabet* Σ *is a probability distribution P on* Σ^* *(the set of all finite strings over* Σ*) such that P(s) > 0 if and only if s is in the support of* L_p *(denoted* $\mathrm{supp}(L_p)$*). We say a string s is* accepted *by* L_p *if P(s) > 0. The support* $\mathrm{supp}(L_p)$ *is then a (deterministic) formal language consisting of all strings the probabilistic language can produce with non-zero probability.*

Observation 1. *Any ordinary (deterministic) formal language L can be viewed as a probabilistic formal language that assigns probability 1 to each string in L (normalized uniformly or in any arbitrary way) and probability 0 to strings outside L. In other words, deterministic languages are a special case of probabilistic languages (with probabilities restricted to 0 or 1).*

Given this observation, we see that the class of probabilistic languages strictly generalizes the class of deterministic languages: every discrete language corresponds to many possible probabilistic languages that have that language as their support. Therefore, if an infinite class of deterministic languages is not identifiable, then an infinite class of probabilistic languages that includes those (as 0-1 special cases) will also not be identifiable. Any learning algorithm for the probabilistic case would in particular solve the deterministic case, which we know is impossible in the scenario above.

We can extend the definition of identification in the limit to probabilistic languages. One natural way is to require the learner to output not just a single hypothesis language at each step but a probability distribution over candidate languages, reflecting uncertainty, and to converge in probability to the correct language. A rigorous definition (adapted from Definition 1) is as follows:

Definition 5 (Identification in the Limit for Probabilistic Languages)**.** *A class of probabilistic languages* $\mathcal{L}_p$ *is* identifiable in the limit (from positive data) *if there exists a learning algorithm* $\mathcal{A}$ *such that for any target probabilistic language* $L_p^* \in \mathcal{L}_p$ *with support* $L^* = \mathrm{supp}(L_p^*)$*, given an infinite sequence of example strings* $\langle s_1, s_2, \dots \rangle$ *where each* $s_i \in L^*$ *and each string in* L^* *appears at least once, the algorithm produces a sequence of probability distributions over* $\mathcal{L}_p$*,* $\langle P_1, P_2, \dots \rangle$*, with the property that:*

1. *For all but finitely many n, the most likely hypothesis under* P_n *is the true language* L_p^**. (Formally,* $\arg\max_{L_p \in \mathcal{L}_p} P_n(L_p) = L_p^*$ *for all sufficiently large n.)*
2. *At all times n, the support of* P_n *(the set of hypotheses given non-zero probability) consists only of languages that are consistent with the data observed so far. That is, if* $P_n(L_p) > 0$ *then* $\{s_1, \dots, s_n\} \subseteq \mathrm{supp}(L_p)$*.*

This definition ensures that eventually the learner assigns highest confidence to the correct probabilistic language, while always ruling out any languages that have already been contradicted by the observations. If such an identification is possible, we would say the class is identifiable in this probabilistic sense.

However, using the observation above, if we have an infinite class of probabilistic languages $\mathcal{L}_p$ that includes an infinite deterministic sub-class of the kind described in Sect 2, then $\mathcal{L}_p$ cannot be identifiable. In particular, consider any scenario with an infinite sequence of possible models such that one model's support language contains another's, which contains another's, and so on (like $L_\infty \supset L_k \supset L_{k-1} \supset \cdots$). If the learner sees all strings from the smallest language in that chain, it has also seen strings from all larger ones; without probability information in the sample selection, it cannot tell whether it's receiving data from the smallest language or a larger one, since all observed strings are consistent with either being the case. The probabilistic aspects of the target model do not manifest in the *set* of observed strings—only in how frequently they might appear, which in the adversarial presentation model is not specified. The identification in the limit framework (as defined) is adversarial in that the sequence of examples can be arranged in any order as long as every string eventually appears; in particular, it does not assume the examples are drawn from the model's own distribution.

In conclusion, any impossibility that holds for infinite deterministic classes carries over to infinite probabilistic classes. The safe assumption is:

> *If the class of candidate LLMs is unbounded (infinitely many possible models/languages), then no general attribution algorithm can identify the source model with certainty, even if models are probabilistic.*

This result reinforces the pessimistic outlook for attribution in a scenario with open-ended model classes. It suggests that to have any hope for theoretical identifiability, one must drastically restrict the hypothesis space—for instance, by assuming only a finite (and manageable) set of candidate models.

4 Finite classes of discrete LLMs: Identifiability

We now turn to the case where the number of candidate models (and hence candidate languages) is finite. This scenario is much more favorable. If $\mathcal{L} = \{L_1, L_2, \dots, L_N\}$ is a finite set of languages, the tell-tale sets required for identification by Angluin's condition can easily be constructed. The target $L^* = L_i$ is one of the N possibilities, and constructing the tell-tale set T_i for L_i is now easy since there are only a finite number of languages properly contained in L_i, and for each such L_j there must be a string in $L_i - L_j$ that we can add to T_i. Thus T_i is a finite subset of L_i satisfying the tell-tale set condition of Angluin.

More formally, we can state:

Proposition 6. *Any finite collection of languages $\mathcal{L} = \{L_1, L_2, \dots, L_N\}$ is identifiable in the limit from positive data (assuming each L_i is recursive, so that an algorithm can test membership).*

Proof (Sketch): As the collection is finite the tell-tale sets required for identification by Angluin's condition can be constructed as follows. Consider a language $L_i \in \mathcal{L}$ and construct the tell-tale set T_i for L_i as follows. Consider the finite number of languages properly contained in L_i, and for each such L_j take a string in $L_i - L_j$ and add it to T_i. When we are done T_i is a finite subset of L_i satisfying the tell-tale set condition of Angluin's Theorem 2. □

In summary, when the universe of possible LLMs is *finite*, the attribution problem is solvable in theory. After seeing enough outputs, the attacker (or attribution algorithm) can, in effect, find a signature of the source model's language. This result assumes we know the exact set of possible models in advance and that they have distinct output capabilities. In practical terms, this might correspond to a scenario where a small number of specific models are suspect (for example, a fixed set of known bots or generators that might have produced a given text). In such cases, especially if the models are sufficiently different, targeted attribution can succeed by cross-checking the observed outputs against each model's known outputs.

It is worth noting, however, that Angluin's theorem is non-constructive in the sense that it guarantees the existence of tell-tale sets but doesn't necessarily provide an easy way to find them. In practice, even if N is modest, discovering the distinguishing outputs between each pair of models might be difficult without additional information or oracles (like black-box query access to models). Nevertheless, from a purely information-theoretic viewpoint, the finite case poses no fundamental barrier to identification: given unlimited data, one can eventually tease apart any finite number of distinct languages.

Unfortunately, this optimistic scenario breaks down under even slight randomness in decoding.

Decode regimes and applicability. Decoding determines whether an LLM behaves deterministically or probabilistically. Greedy decoding (temperature = 0) and beam search with fixed tie-breaking are *deterministic* given a prompt; thus our deterministic results apply (e.g., finite candidate sets may be identifiable in the finite-deterministic case). By contrast, temperature, top-*k*, and nucleus (top-*p*) sampling induce *stochastic* outputs; with hidden prompts and adversarial presentation (see the data-assumptions paragraph above), our probabilistic impossibility result applies (see Theorem on finite probabilistic classes). Mixed regimes (e.g., low-temperature sampling, randomized beams) fall under the probabilistic case whenever any output randomness remains.

5 Finite classes of probabilistic LLMs: A counterexample

We now address the most subtle regime: a finite collection of probabilistic LLMs. One might hope that with only a finite number of models to consider, attribution remains feasible (as in the deterministic case). However, probability distributions can introduce ambiguity that does not occur with deterministic languages. In particular, different models can share the same support (i.e., they can all produce the same set of strings, but with different probabilities). If two models M_1, M_2 have $\mathrm{supp}(L_p(M_1)) = \mathrm{supp}(L_p(M_2))$ (they can produce exactly the same strings, just with different likelihoods), then any sequence of outputs that is not annotated with likelihood information cannot distinguish them—because whatever string appears, it could have come from either model. The only way to tell them apart would be to notice differences in the relative frequency or probability of outputs, but in the identification-in-the-limit framework the presentation of examples is controlled adversarially (we only assume every possible string eventually appears, not that we see them according to the model's true distribution).

LLM as conditional model. An LLM M defines a conditional distribution $P_M(y \mid x)$ over completions $y \in \Sigma^*$ given prompts $x \in \Sigma^*$. For fixed x, $P_M(\cdot \mid x)$ is a distribution on Σ^*.

Support sets. $\mathrm{supp}_x(M) = \{\, y \in \Sigma^* : P_M(y \mid x) > 0 \,\}$. For brevity, write $\mathrm{supp}(M) = \bigcup_{x \in \Sigma^*} \mathrm{supp}_x(M)$, which coincides with $\mathrm{supp}(L_p(M))$ in the notation above.

Adversarial setting (unknown prompts). The adversary reveals only y, not the prompt x that generated it, and may choose x to maximize ambiguity (e.g., single-shot attacks, lost context). This mirrors forensic reality in disinformation and cyberattacks, where prompts are typically unobserved and only released text is available for attribution.

We observe an infinite sequence of outputs $y_1, y_2, \ldots$ with no access to the prompts that produced them. The presentation is adversarial: the sequence (y_i) is neither i.i.d. nor sampled according to P_M. The only guarantee is consistency with M: for each i there exists some (unknown) x_i such that $P_M(y_i \mid x_i) > 0$ (equivalently, $y_i \in \mathrm{supp}_{x_i}(M)$ for some x_i). Evidence is positive-only; we do not assume access to negatives, frequencies, or prompt–completion pairs. Unlike PAC/statistical settings with i.i.d. draws and convergence in total variation (TV) or Kullback–Leibler (KL) divergence, our results concern this worst-case presentation model.

Boundary conditions for support overlap. Real LLMs need not share identical support across languages or domains; when supports differ, sufficient positive evidence will eventually separate models (some observed string will lie outside a candidate's support). Our negative results do not rely on identical support except in the finite–probabilistic counterexample (Theorem 7), where identical support is treated as a *worst case* capturing high-capacity models coerced via prompting to "say almost anything" (propensity vs. possibility). Moreover, even under observable i.i.d. sampling, two models with identical support but small total-variation gap $\|P_1 - P_2\|_{\mathrm{TV}} \le \delta$ require $n = \Omega(1/\delta^2)$ samples to distinguish with constant error—linking attribution to standard hypothesis-testing limits. In single-shot attacks, such frequency information is absent, so distributional evidence cannot rescue attribution.

We now present a simple but striking example demonstrating that even two probabilistic languages can foil any identification algorithm under Gold's paradigm:

Theorem 7. *Even for a finite number of probabilistic formal languages (as few as two), identification in the limit does not always hold.*

Proof: Consider an alphabet consisting of a single symbol $\Sigma = \{x\}$. We define two probabilistic languages $L_{p,1}$ and $L_{p,2}$ over Σ as follows. Both $L_{p,1}$ and $L_{p,2}$ accept *every non-empty string* over Σ (so their support is the same language $L = \{x, x^2, x^3, \ldots\}$). However, they differ in the probability distribution assigned to these strings. Let P_1 and P_2 denote the probability mass functions for $L_{p,1}$ and $L_{p,2}$ respectively:

$$P_1(x^n) = \begin{cases} \frac{1}{2^n} - \frac{1}{2^{n+2}}, & \text{if } n \text{ is odd,} \\ \frac{1}{2^n} + \frac{1}{2^{n+1}}, & \text{if } n \text{ is even,} \end{cases}$$

and

$$P_2(x^n) = \begin{cases} \frac{1}{2^n} - \frac{1}{2^{n+1}}, & \text{if } n \text{ is even,} \\ \frac{1}{2^n} + \frac{1}{2^{n+2}}, & \text{if } n \text{ is odd.} \end{cases}$$

One can verify that each of these defines a proper distribution over $n = 1, 2, 3, \ldots$ (the probabilities sum to 1 for each, since the alternating added and subtracted terms cancel out telescopically). Importantly, note that:

- For both $L_{p,1}$ and $L_{p,2}$, every string x^n with $n \geq 1$ has non-zero probability. Thus, $\operatorname{supp}(L_{p,1}) = \operatorname{supp}(L_{p,2}) = L$ (the set of all non-empty strings of x).
- The distributions differ only in the *sign* of the small adjustments. For instance, for n even, $P_1(x^n)$ is a bit larger than the baseline $1/2^n$, whereas $P_2(x^n)$ is a bit smaller; for n odd, it's the opposite.

Now, suppose we have an identification algorithm $\mathcal{A}$ that sees an infinite presentation of strings that come from one of these two probabilistic languages (but $\mathcal{A}$ does not know which one). By the definition of identification in the limit, the presentation is an adversarial sequence that contains every string in L at least once (since the support is L itself). Crucially, because $\mathcal{A}$ must succeed for *any* sequence that meets this criterion, we can adversarially choose the order in which strings appear. In particular, we can arrange for $\mathcal{A}$ to see all possible strings in some order that completely hides any statistical bias. For example, the adversary could present the strings in lexicographic order (or by increasing length). In such an arrangement, the data sequence $\langle s_1, s_2, \ldots \rangle$ would be *the same* whether the underlying source is $L_{p,1}$ or $L_{p,2}$, since both can generate all those strings.

Since $L_{p,1}$ and $L_{p,2}$ have identical supports, any sequence of distinct strings from that support is consistent with either model having produced it. The identification algorithm $\mathcal{A}$, by requirement (2) of Definition 5, cannot eliminate either hypothesis until it encounters some evidence that contradicts one of them. But there will be no such evidence in terms of observed strings, because any finite set of observed strings S will be a subset of L, and both hypotheses $L_{p,1}$ and $L_{p,2}$ are consistent with S.

Therefore, $\mathcal{A}$ can never be sure whether the source is $L_{p,1}$ or $L_{p,2}$. We can even say that for any strategy $\mathcal{A}$ might use to eventually choose one of the two, we (as an adversary) can define the data sequence in such a way that $\mathcal{A}$'s final guess is wrong. For instance, if $\mathcal{A}$ plans to guess "Model 1" after seeing some sufficiently long prefix of data, we can orchestrate the data (which, again, does not violate consistency with Model 2) such that $\mathcal{A}$ is misled. In other words, whichever model $\mathcal{A}$ converges on, we ensure the actual model was the other one.

Thus, no identification algorithm can guarantee to output the correct model $L_{p,1}$ vs $L_{p,2}$ after finitely many steps on all valid input sequences. This proves that the class $\{L_{p,1}, L_{p,2}\}$ is not identifiable in the limit. □

The above proof shows a pathological but illuminating construction: we made two models that are indistinguishable by language membership alone yet differ in their probabilistic structure. The adversary exploited the fact that identification in the limit does not assume random sampling according to the model distribution; if instead we did assume the samples were drawn from the model's distribution, then over time one might detect the slight differences in frequencies (this would move us into the territory of probabilistic identification or Bayesian inference with enough data). But under Gold's paradigm (which is adversarially robust, considering worst-case presentation of data), the difference in distributions is irrelevant because the adversary can always present data in a way that conceals it. Notably, this is the first example of identification failure in a finite hypothesis class of probabilistic languages. Previously, impossibility results (e.g. Gold 1967) required an infinite or uncountable class, or deterministic outputs – our counterexample shows even two stochastic models can confound any attribution attempt. This closes a long-standing gap in the theory.

Remark 5.1 (δ–ε intuition)**.** For any target distinguishability threshold $\delta > 0$ and any feasible sample budget n, one can construct two probabilistic models M_1, M_2 with $\text{supp}(M_1) = \text{supp}(M_2)$ and small distributional gap $\|P_1 - P_2\|_{\text{TV}} \leq \varepsilon(\delta)$ such that distinguishing them with constant error under i.i.d. sampling requires $n = \Omega(1/\varepsilon(\delta)^2)$ observations (standard hypothesis-testing lower bounds). Thus by choosing $\varepsilon(\delta)$ sufficiently small relative to operational n, the required evidence exceeds practical budgets; in our adversarial presentation model—where frequencies are not observed—this route to identifiability is unavailable altogether. This remark is heuristic and not needed for the theorem; it clarifies why probabilities may offer little leverage in single-shot or data-scarce settings.

From an LLM attribution standpoint, this example captures a real concern: modern generative models (especially fine-tuned ones) often have very similar capabilities, differing mostly in the probabilities they assign to various outputs or in subtleties of style. If two models M_1 and M_2 have been trained on largely overlapping data, M_1 might respond to a prompt almost the same way as M_2, with only slight differences in phrasing probabilities. Any fixed set of outputs that M_1 can produce, M_2 might also be able to produce (especially if the prompt is chosen adversarially to maximize confusion). Our theoretical result shows that in the worst case, if their output supports are identical, an attacker who only sees whether an output happened or not (not how often among many trials) cannot distinguish them.

It is worth noting that the proof's construction might seem artificial (the probability mass functions were carefully designed). But one can imagine more natural scenarios: for example, two language models that have the *exact same range of expression* (say, both have memorized the same set of internet texts), but one is fine-tuned to prefer certain styles more than the other. Without many samples to do statistical analysis, any single piece of text produced by one model could also have been produced by the other. Identification in the limit says we can have as many samples as we want in the long run, but since the adversary controls the ordering, they can always choose a diverse set that doesn't reveal the biases. In effect, the adversary can simulate the distribution of the other model by interleaving outputs.

The proof above holds even when the two probabilistic languages have identical support (as formal languages). This situation captures well the situation with LLMs where one can force them to output almost anything. In other words, if the models are sufficiently expressive (e.g., large GPT-style models can be prompted to talk about almost any topic), then differences lie not in what they *can* say but in what they *tend* to say (sometimes refered as the *propensity* of an LLM). If one model can be coerced (through prompt or context) to imitate the style of another, then purely from the fact that a certain output was observed, we gain no information about which model was behind it.

In summary, we have established that:

- If we have infinitely many possible models, attribution is theoretically impossible (Sects 2 and 3).
- If we have finitely many models *and* treat them as producing deterministic languages, attribution is possible in principle (Sect 4).

- If we have finitely many models but they produce outputs probabilistically and have overlapping capabilities, attribution can again become impossible (this section).

Our impossibility is presentation-model–driven: it survives even if arbitrary amounts of data are available but under adversarial ordering. Under stochastic sampling (e.g., PAC-style), probabilities can add identifying signal; however, (i) single-shot settings do not expose frequencies, and (ii) when $\|P_1 - P_2\|_{\mathrm{TV}} \leq \delta$, distinguishing with constant error typically requires $n = \Omega(1/\delta^2)$ samples. Operationally, the negative picture largely persists for adversarial misuse.

This paints a rather bleak theoretical picture: the only regime that avoids impossibility is the one with a finite, discrete set of hypotheses. In practice, the space of potential models (especially fine-tuned variants) is enormous and effectively unbounded, and models are stochastic. Therefore, the negative results seem most relevant to real-world conditions. Next, we turn to empirical evidence to quantify just how large the hypothesis space has grown, and we analyze the computational limits of brute-force attribution approaches.

6 The rapid expansion of the LLM hypothesis space

Thus far, our theoretical analysis suggests that unless the set of candidate models is very limited, one cannot reliably attribute a given output to its true source in the worst case. In practice, the ecosystem of LLMs is anything but limited: it is rapidly growing, with new models and fine-tuned versions emerging constantly. In this section, we present a data-driven analysis that illustrates the scale of the problem.

We assembled data from Stanford University's *Ecosystem Graphs* project [24], which documents released AI models and datasets over time. The dataset we use includes information on 359 language models (as of January 2025) and 112 datasets, among other assets. For our purposes, we focus on:

- The number of distinct base models (checkpoints) released, including whether they are open-source or closed-source.
- The number of distinct datasets released over time.

Using these, we can estimate how many potential fine-tuned variants could exist by combining open models with available datasets.

Specifically, let:

$$C(t) = \#\{\text{closed or restricted models released up to time } t\},$$

$$O(t) = \#\{\text{open-source model checkpoints released up to time } t\},$$

$$D(t) = \#\{\text{datasets released up to time } t\}.$$

Here, $C(t)$ and $O(t)$ partition the total number of models by accessibility (closed vs open), and $D(t)$ measures the cumulative count of distinct datasets.

A *conservative* scenario for the growth of distinct fine-tuned models is to assume each open-source model can be fine-tuned on at most one dataset. In that case, by time t the total number of (base model or fine-tuned) model variants is at least:

$$N_{\text{single}}(t) = C(t) \; + \; O(t)\left[\,1 + D(t)\right],$$

where $O(t)$ models can each give rise to at most one fine-tuned variant on each of the $D(t)$ datasets, hence $O(t)\,D(t)$ possible fine-tunes, plus the base models themselves ($O(t)$) and the closed models $C(t)$ which might not be fine-tuned openly. We add 1 in the bracket to count the base open models themselves (fine-tuned on "no new data").

We can also consider more aggressive combinations. If each open model could be fine-tuned on up to two datasets (combined), the number of possible distinct outputs grows combinatorially:

$$N_{k\leq 2}(t) = C(t) \;+\; O(t)\left[\,1 + D(t) + \binom{D(t)}{2}\right],$$

where $\binom{D(t)}{2}$ is the number of ways to pick two distinct datasets to jointly fine-tune. Similarly, allowing up to three datasets per fine-tune:

$$N_{k\leq 3}(t) = C(t) \;+\; O(t)\left[\,1 + D(t) + \binom{D(t)}{2} + \binom{D(t)}{3}\right].$$

These formulas provide lower bounds on the number of *conceptually distinct* model variants one could have by mixing available ingredients (models and datasets). Not every combination is actually realized or made public, of course, but they represent the search space size that an attribution mechanism might have to contend with if an adversary could fine-tune or modify models arbitrarily using existing data.

Using our collected data, we computed these metrics from 2018 through the beginning of 2025. The results are striking:

6.1 Global growth in candidate models

Fig 1 shows the exponential growth of $N_{\text{single}}(t)$, the conservative count of model variants under a single-dataset fine-tune assumption. Even so, the hypothesis space swells from on the order of 10^1 plausible variants in 2019 to 10^4 by 2023, and surpasses 3×10^4 by 2025. A least-squares exponential fit ($R^2 \approx 0.97$) yields a growth rate $b = 1.39\,\text{yr}^{-1}$, corresponding to a doubling time $\tau = \ln 2/b \approx 0.50\,\text{yr}$ (roughly six months). In other words, the effective hypothesis space for attribution doubles twice per calendar year, far outpacing the pace of any brute-force inspection.

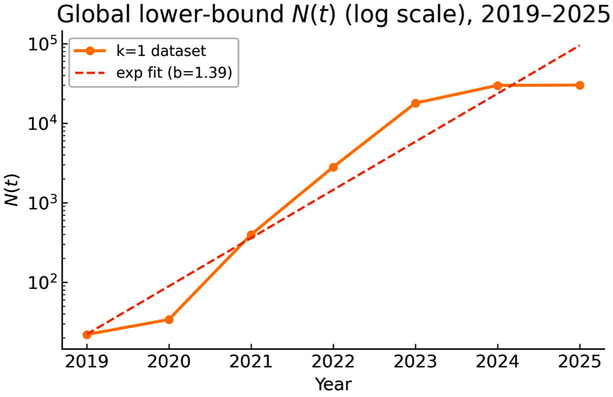

Fig 1. Growth of the combinatorial lower bound $N_{\text{single}}(t)$ (single-dataset fine-tunes per open model) from 2019 to 2025 on a logarithmic scale. The dashed line shows a least-squares exponential fit with estimated growth rate $b = 1.39\,\text{yr}^{-1}$, corresponding to a doubling time of τ = 0.50 yr.

https://doi.org/10.1371/journal.pcsy.0000085.g001

6.2 Fine-tuning on multiple datasets

Fig 2 depicts the explosive growth of our combinatorial lower bound $N(t)$ when open-weight checkpoints are allowed to be fine-tuned on up to two or three datasets. In the $k \leq 2$ scenario, $N_{k\leq 2}(t)$ reaches on the order of 10^6 variants by 2025, while for $k \leq 3$, $N_{k\leq 3}(t)$ approaches 10^7. Exponential fits on $\ln N$ (both with $R^2 \approx 0.97$) give growth rates

$$b_{k\leq 2} \approx 1.95\ \mathrm{yr}^{-1}, \quad b_{k\leq 3} \approx 2.51\ \mathrm{yr}^{-1},$$

which correspond to doubling times $\tau_{k\leq 2} = \ln 2/b_{k\leq 2} \approx 0.36\,\mathrm{yr}$ and $\tau_{k\leq 3} = \ln 2/b_{k\leq 3} \approx 0.28\,\mathrm{yr}$. Thus, even a minimal two-dataset allowance doubles the hypothesis space in under four months, and three-way mixes halve that interval to just over three months. Any brute-force attribution approach would soon be outstripped by this relentless combinatorial explosion.

In practical terms, even if only a small fraction of these combinations actually exist as deployed models, an attacker could potentially fine-tune a new model on a novel mix of data to evade known detectors. The defender, lacking knowledge of that specific fine-tune, would have to consider it as a possibility among an astronomically large set.

6.3 Trends by modality and region

To pinpoint which segments drive the hypothesis-space explosion, we classified each checkpoint by its primary *modality* (text, vision, multimodal, audio, other, unknown) and by the developer's geographic *region* (North America, Europe, Asia, Other). We then recomputed the conservative lower bound $N_{\mathrm{single}}(t)$ for each slice.

Fig 3 demonstrates that while text models still dominate in absolute count, the modal segment growing fastest (by slope) is *multimodal*, followed by vision and audio. This trend warns that non-text attribution (e.g. image and video) will soon face the same combinatorial explosion as language models.

Fig 4 shows a clear shift from a North-America–centric model ecosystem toward a more balanced, global landscape. Asian organizations, particularly open-source contributors, now match or exceed North American counts by late 2024. Europe and Other regions add further breadth. Attribution frameworks must therefore be prepared for suspect generators worldwide, not just from the major U.S. players.

These previously undocumented trends spotlight emerging areas (e.g. non-text modalities, non-Western model providers) where attribution efforts will face growing complexity.

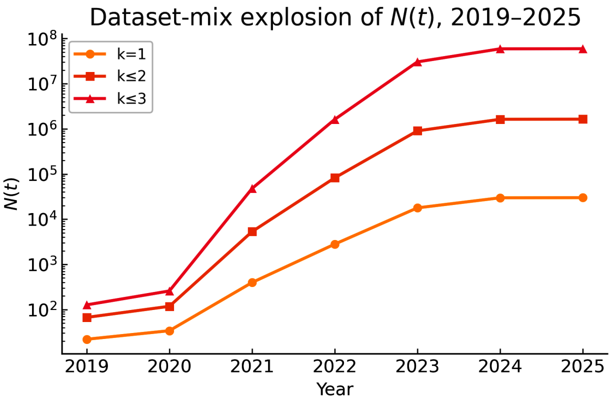

Fig 2. Projected growth of the conservative lower bound $N(t)$ when permitting fine-tuning on combinations of up to k = 2 (squares) or k = 3 (triangles) datasets per checkpoint, from 2019 to 2025 on a logarithmic scale. Under these assumptions, $N_{k\leq 2}$ reaches $\sim 10^6$ variants by 2025 (doubling time $\tau \approx 0.36$ yr), and $N_{k\leq 3}$ approaches $\sim 10^7$ (doubling time $\tau \approx 0.28$ yr).

https://doi.org/10.1371/journal.pcsy.0000085.g002

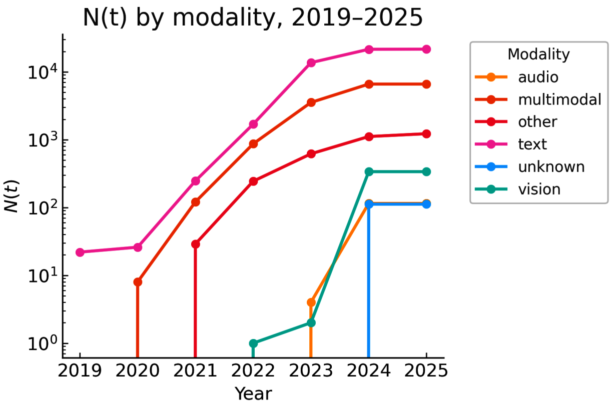

Fig 3. Growth of $N_{\text{single}}(t)$ by model modality, 2019–2025, on a log scale. Text-only variants remain the largest class, but multimodal models exhibit the steepest relative growth, signaling imminent attribution challenges in image+text domains.

https://doi.org/10.1371/journal.pcsy.0000085.g003

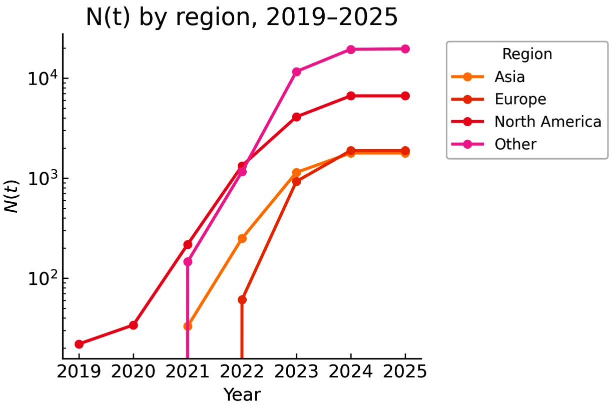

Fig 4. Growth of $N_{\text{single}}(t)$ by developer region, 2019–2025, on a log scale. North America led early releases, but Asia's rapid uptake of open-weight models has narrowed the gap by 2024. Europe and Other regions also show steady contributions.

https://doi.org/10.1371/journal.pcsy.0000085.g004

6.4 Exponential growth summary

To summarize the quantitative growth, Table 1 reports the exponential fit parameters for the overall $N(t)$ under different fine-tuning assumptions from 2019 to 2025. All scenarios show $R^2 \approx 0.97$, indicating an almost perfect exponential trend.

These figures underscore an unsustainable trajectory: even in the most conservative single-dataset scenario, the hypothesis space doubles in under six months. With two-dataset mixes, it doubles in just over four months, and with

Table 1. Estimated exponential growth parameters for the hypothesis space $N(t)$ from 2019 to 2025 under different fine-tuning assumptions. b is the exponential growth rate per year, and $\tau = \ln 2/b$ is the doubling time.

Metric	b (yr^{-1})	R^2	Doubling time τ (yr)
N_{single} (1 dataset)	1.39	0.97	0.50
$N_{k\leq 2}$ (up to 2 datasets)	1.95	0.97	0.36
$N_{k\leq 3}$ (up to 3 datasets)	2.51	0.97	0.28

https://doi.org/10.1371/journal.pcsy.0000085.t001

three-dataset mixes in under three and a half months. Any forensic catalog or fingerprinting system will struggle to keep pace with such rapid doubling.

7 Computational feasibility of exhaustive attribution

To fully understand the challenge of attributing generated content to its original model, we must consider not just the combinatorial explosion of plausible model variants, but also the raw computational cost involved. A straightforward, brute-force approach to attribution—evaluating the likelihood of a given output against each known model—quickly becomes untenable as the number and size of models increase.

As a concrete baseline, we extracted parameter counts for all models available in our comprehensive ecosystem dataset through 2025. Early on, the cumulative size of all known models was modest: approximately 1.3×10^{10} (13 billion) parameters in 2019. Yet by 2025, this figure surged dramatically to roughly 2.2×10^{13} (22 trillion) parameters—an increase of nearly three orders of magnitude in just six years (Fig 5). To estimate this growth, we conservatively imputed missing parameter sizes with the average of known model sizes, ensuring our estimate remains robust yet conservative.

What would this mean in practical computational terms? Consider the task of attributing a single piece of suspicious content consisting of 100,000 tokens—a sizable but plausible length for a lengthy piece of misinformation or propaganda. Under a naive scenario where we calculate a single floating-point operation (FLOP) per parameter per token to compute likelihoods (which is highly optimistic—real-world inference is several times costlier), we would require 2.2×10^{13} parameters multiplied by 10^5 tokens, totaling about 2.2×10^{18} FLOPs for this single attribution.

To contextualize this number, the Frontier supercomputer—among the most powerful systems available as of 2025 [16]—achieves peak performance of approximately 1.7×10^{18} FLOPs per second. Thus, attributing just one suspicious 100k-token output across all known models in 2025 would theoretically take around $2.2 \times 10^{18}/1.7 \times 10^{18} \approx 1.3$ seconds on Frontier under optimal conditions (Fig 6). While this might appear reasonable for isolated incidents, the complexity scales steeply. Consider a more demanding case in which *10 000* suspicious outputs—each 100 k tokens long—must be checked every day. With the 2025 model set (2.2×10^{13} parameters), that workload requires

$$10^4 \text{ items} \times 10^5 \text{ tokens/item} \times 2.2 \times 10^{13} \text{ params} = 2.2 \times 10^{22} \text{ FLOPs/day}.$$

At Frontier's peak 1.7 exaFLOP s^{-1} this equals $2.2{\times}10^{22}/1.7{\times}10^{18} \approx 1.3{\times}10^4$ s, or **3.6** h of wall-time—about **15**% of the machine's entire daily capacity for a *single* attribution task-set (Fig 7).

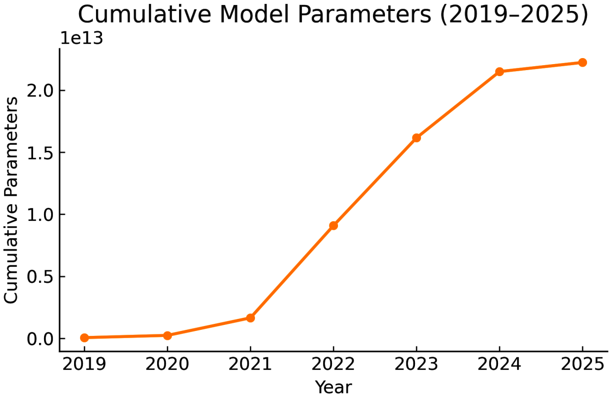

Fig 5. Cumulative total of model parameters (log scale) from 2019 to 2025, with discrete year ticks. The curve illustrates a two-order-of-magnitude jump in just three years, underscoring the steep "parameter cliff".

https://doi.org/10.1371/journal.pcsy.0000085.g005

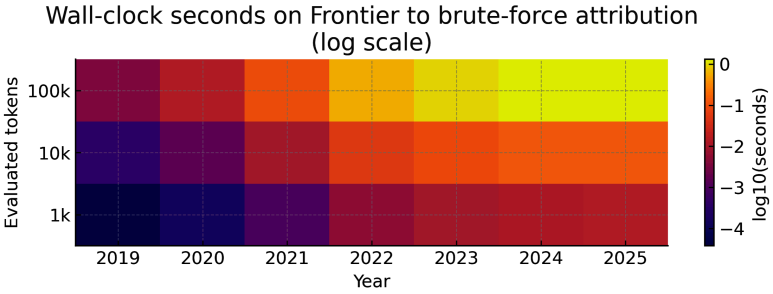

Fig 6. Compute wall for exhaustive attribution: Log$_{10}$ wall-clock seconds on Frontier required to brute-force the likelihood of a *single* sequence against *all* known models each year (rows: 1 k, 10 k, 100 k tokens).

https://doi.org/10.1371/journal.pcsy.0000085.g006

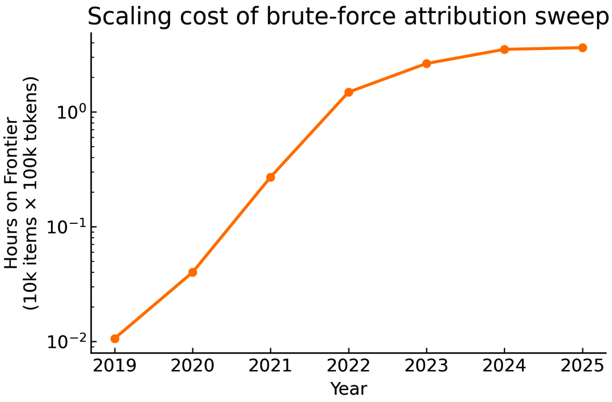

Fig 7. Compute wall for exhaustive attribution: Wall-clock *hours* on Frontier needed to sweep 10 000 suspect items of 100 k tokens each day as the model ecosystem grows from 2019 to 2025 (log scale). The workload balloons from minutes to multi-hour jobs, beyond real-time forensic tolerances.

https://doi.org/10.1371/journal.pcsy.0000085.g007

Scaling to nationwide monitoring renders brute force hopeless. Recent surveys suggest 1.32×10^8 U.S. adults use generative-AI daily. At a modest 10 000 tokens each, the country produces

$$1.32 \times 10^8 \times 10^4 = 1.32 \times 10^{12} \text{ tokens/day},$$

or 4.8×10^{14} tokens per year. Brute-forcing those tokens against the same 22 T-parameter pool would need

$$4.8 \times 10^{14} \times 2.2 \times 10^{13} \approx 1.1 \times 10^{28} \text{ FLOPs}.$$

Even with Frontier,

$$\frac{1.1 \times 10^{28}}{1.7 \times 10^{18}} \approx 6.5 \times 10^9 \text{ s} \approx \textbf{200 years}$$

of uninterrupted peak compute would be required to attribute a *single* year's U.S. output—clearly infeasible in real time.

Merely *streaming* that much data (1.9 PB at 4 B/token) through Frontier's $\sim 2\,\text{TB}\,\text{s}^{-1}$ burst I/O would already consume about 16 min per day (Table 2).

Table 2. **Compute budget for a *single, exhaustive* annual sweep of all LLM-generated content in the United States.** Even with the Frontier supercomputer, simple I/O takes minutes per day, while full likelihood evaluation would require centuries.

Metric (U.S., annualised 2025)	**Estimated Value**
Daily Active LLM Users	1.32×10^{8}
Total Tokens Generated per Year	4.8×10^{14} tokens
— *Pure I/O scan* —	
Data Volume (4 bytes / token)	$\approx$ 1.9 PB
Streaming Time on Frontier (~2 TB s^{-1} burst)	~16 min
— *Brute-force likelihood (22 T parameters)* —	
Total FLOPs (4.8×10^{14} tokens $\times$ 2.2×10^{13} params)	1.1×10^{28} FLOPs
Processing Time on Frontier (1.7 exaFLOP s^{-1})	~6.5×10^{9} s $\approx$ 200 yr

https://doi.org/10.1371/journal.pcsy.0000085.t002

This computational impossibility of exhaustive attribution emphasizes a strategic dilemma: brute-force attribution, already at the brink of infeasibility today, will soon become outright impossible due to relentless increases in model quantity, diversity, and parameter size. Of course, real attribution might not require scanning everything; one might triage or focus on certain content. But the point remains that any brute-force approach—even one that assumes access to all model internals and unlimited computing—faces extreme scalability issues. In practice, things are even harder: many models are closed or not publicly runnable, content might be encrypted or privacy-protected, and one would have to find clever shortcuts.

8 Broader challenges and discussion

Beyond the theoretical impossibilities and the combinatorial and computational explosions described above, several other factors complicate LLM attribution:

Evasion via Social Network Dynamics. In realistic attack scenarios, malicious content generated by LLMs will propagate through social networks and other channels before reaching its victims. Practically, investigators may have only a single snippet of suspect text to examine, and our impossibility results show that if attribution can fail even with unlimited data in theory, one sample is clearly far from sufficient. Attackers exploit this by limiting the evidence they reveal—distributing an operation across many small pieces of content or relaying it through social-network chains—so defenders never obtain a rich dataset to analyze. Attackers can also inject AI-generated disinformation via proxy accounts or compromised nodes, making the origin hard to trace. The structure of social networks—often characterized by small-world properties and heavy-tailed connectivity [18]—allows content to diffuse widely without clear indication of where it started. Adversaries may deliberately rewire parts of the network or introduce *Sybil* nodes (fake accounts) to obfuscate information flow [20,21]. These strategies echo classical problems in network forensics: tracking the “patient zero” of an epidemic or the first source of a rumor is notoriously difficult without strong assumptions [19]. In the context of AI, even if perfect model attribution were possible in principle, content hopping through many users risks attributing the text to the last sharer rather than the true originator, defeating accountability.

A Generative Arms Race. While attribution of content to models appears hard, the *generation* of content is getting easier. Fascinatingly, recent theoretical work by Kleinberg and Mullainathan [23] demonstrates what they call “language generation in the limit.” In a framework reminiscent of Gold’s, they show that it is possible for an agent to produce novel strings from a target language indefinitely without actually identifying the language. In plainer terms, one can imagine a situation where defenders deploy an AI to imitate an attacker’s model: the defender’s AI produces outputs that are valid under the attack model’s distribution, effectively matching the attacker’s capability, yet the defender’s AI has no idea which model it is imitating (it just knows how to continue the language). This theoretical result implies an odd stalemate: both the attacker and defender can spew out similar content, neither being able to conclusively prove which model is behind it.

In such a scenario, attribution becomes a moot point—everyone can generate in the style of everyone else, and authenticity is lost in a sea of mimicry. This could lead to a game of confusion where any incriminating text could be dismissed as something an autonomous agent could have produced as well. It underscores how the very nature of generative AI erodes the link between model and output.

Toward Mitigation Strategies. Given the grim outlook, what can be done? A few avenues, each with limitations, are often discussed:

- **Watermarking and Fingerprinting.** If model developers voluntarily (or by regulation) incorporate hidden watermarks into generated content (e.g., detectable bit patterns in text or slight image perturbations), attribution could be greatly aided. Research on watermarking for LLMs is ongoing. However, watermarks can potentially be removed or spoofed by adversaries, especially if the watermark keys or methods become known. Moreover, not all model developers will comply, especially open-source ones or malicious actors.
- **Model Authentication Infrastructure.** One could imagine a world where each model has a cryptographic signature and each piece of AI content comes with a certified origin stamp (like a cryptographic watermark or metadata). This is conceptually similar to code signing. However, this requires adoption across industry and doesn't solve the problem of someone using an uncompromised model to produce content and then stripping the signature [17].
- **Reducing the Hypothesis Space.** From a policy angle, one way to make attribution easier is to limit the number of models at large. If, for example, only a handful of foundation models were authorized for public use, the identification problem shrinks to distinguishing among those (the finite deterministic case would be more applicable). This could be implemented via regulation or industry consolidation. However, this conflicts with open innovation and may be impractical to enforce globally.

Our results strongly suggest that purely technical, post-hoc attribution will not be a panacea. The rapid scaling of model capabilities and availability means we should assume a world where attribution of malicious AI outputs is difficult or impossible. This, in turn, argues for *proactive* measures: for example, embedding protective measures in models, limiting access to highly capable models, educating the public about AI-generated misinformation, and building resilience to fake content.

9 Conclusion

We have explored the landscape of LLM attribution through both the lens of learning-theoretic impossibility and the empirical realities of today's model ecosystem. Building on Gold's and Angluin's identification-in-the-limit results, we proved that even with unlimited, unlabeled data one cannot in general infer the true source model unless the candidate set is artificially small and mutually exclusive. Our new theorem shows that two probabilistic language models whose output distributions overlap everywhere are, in principle, indistinguishable. This negative result is not a finite-sample curiosity; it is an information-theoretic limit that survives in the infinite-data regime.

The empirical picture amplifies this constraint. Fine-tuning, checkpoint remixing, and automated architecture search are causing the pool of plausible generators to double every few months, while brute-force attribution already exceeds exascale budgets for anything beyond toy corpora. Exhaustive search is therefore a non-starter, and statistical "best-guess'' methods inherit the impossibility boundary just established.

These findings collide with a policy environment that increasingly assumes model-level traceability. The EU AI Act, U.S. Executive Order 14110, and parallel draft rules in the U.K. and OECD impose obligations—incident reporting, watermark retention, and red-team documentation—squarely at the provider or model level [27–29]. Without reliable attribution, regulators lack the technical substrate to levy fines or mandate recalls, and developers cannot demonstrate the "due diligence'' now embedded in many safe-harbor clauses. Chain-of-custody investigations face a similar impasse: modern attack pipelines often route prompts through multiple agents, so isolating the compromised node may be the only practical way to halt a live exploit when the human operator is anonymous. For instance, in March 2024 security researchers at Cornell Tech disclosed "Morris II," a self-replicating AI worm that jumped from one generative-AI email assistant to the next by embedding a malicious prompt in each message. Because every assistant (GPT-4, Gemini Pro, LLaVA) automatically forwarded the adversarial prompt downstream, investigators found the only viable containment strategy was to quarantine the first compromised agent—the human operator never had to re-enter the loop.,[30,31]. If the identification of this first compromised agent is hard, that provides further difficulty in these sorts of investigations.

From the developer's perspective, proving that a particular model was—or was not—responsible for some output is increasingly central to liability and reputation. In January 2024, for instance, an AI-generated robocall cloned President Biden's voice and urged New Hampshire voters to skip the primary; independent analysts at Pindrop and UC Berkeley traced spectral artefacts to the commercial voice-cloning startup ElevenLabs, but the company itself said it "cannot comment on specific incidents," hinting at its inability to decisively confirm or deny authorship in real time [32–34] Our impossibility results therefore imply that beyond a certain point, developers cannot rely on post-hoc attribution to catch misuses—strengthening the case for preventative measures such as stricter release controls or built-in watermarking for powerful models.

Because the impossibility bound is information-theoretic, it suggests a pivot from post-hoc identification to ex-ante safeguards. Technical mitigations such as cryptographic signatures, robust watermarking, or permissioned key escrow inject additional bits that collapse the indistinguishability class. Auditable usage logs and risk-tiered release practices further narrow the hypothesis space before an incident occurs. The path forward, in other words, lies in engineering traceability rather than trying to infer it after the fact. Absent such measures, the true author of a text, whether human or AI (and if AI, which one), will frequently remain a mystery, complicating trust, enforcement, and democratic oversight in our information ecosystem. Just as early theoretical studies of cybersecurity—on conflict timing and the strategic dilemmas of digital attribution—proved prescient for later real-world incidents [1,2], we aim to anticipate LLM-attribution challenges before they become unmanageable. A proactive grasp of these limits can guide the design of safer AI systems today.

Supporting information

S1 File. Code and instructions to reproduce the manuscript's figures (Fig 1–Fig 7) from CRFM Ecosystem Graph model metadata (`assets.csv`). Because the dataset is periodically updated, reruns on later snapshots may not match exact values but should yield qualitatively similar results.
(ZIP)

Acknowledgments

We used OpenAI's ChatGPT for occasional editorial and tooling support (sentence/paragraph polishing, small code snippets) and for presentation-level refinements to proofs (clarifying assumptions already stated, harmonizing notation, and cross-reference checks). Any LLM outputs were reviewed and edited for accuracy and remain the authors' responsibility.

Author contributions

Conceptualization: Manuel Cebrian, Andres Abeliuk, Jan Arne Telle.

Data curation: Manuel Cebrian.

Formal analysis: Manuel Cebrian, Andres Abeliuk, Jan Arne Telle.

Funding acquisition: Manuel Cebrian.

Investigation: Manuel Cebrian, Jan Arne Telle.

Methodology: Manuel Cebrian, Jan Arne Telle.

Project administration: Manuel Cebrian, Jan Arne Telle.

Software: Manuel Cebrian.

Supervision: Jan Arne Telle.

Visualization: Manuel Cebrian.

Writing – original draft: Manuel Cebrian, Jan Arne Telle.

Writing – review & editing: Manuel Cebrian, Andres Abeliuk, Jan Arne Telle.

References

1. Axelrod R, Iliev R. Timing of cyber conflict. Proc Natl Acad Sci U S A. 2014;111(4):1298–303. https://doi.org/10.1073/pnas.1322638111 PMID: 24474752
2. Edwards B, Furnas A, Forrest S, Axelrod R. Strategic aspects of cyberattack, attribution, and blame. Proc Natl Acad Sci U S A. 2017;114(11):2825–30. https://doi.org/10.1073/pnas.1700442114 PMID: 28242700
3. Urbina F, Lentzos F, Invernizzi C, Ekins S. Dual use of artificial intelligence-powered drug discovery. Nat Mach Intell. 2022;4(3):189–91. https://doi.org/10.1038/s42256-022-00465-9 PMID: 36211133
4. Perlroth N. This is how they tell me the world ends: The cyberweapons arms race. New York: Bloomsbury; 2021.
5. Xu J, Stokes JW, McDonald G, Bai X, Marshall D, Wang S, Swaminathan A, Li Z. AutoAttacker: A large language model guided system to implement automatic cyber-attacks (arXiv:2403.01038); 2024.
6. Rahwan I, Cebrian M, Obradovich N, Bongard J, Bonnefon J-F, Breazeal C, et al. Machine behaviour. Nature. 2019;568(7753):477–86. https://doi.org/10.1038/s41586-019-1138-y PMID: 31019318
7. Shavit L, Siddarth D, Trager R, Wolf K. Practices for governing agentic AI systems. OpenAI; 2023. https://cdn.openai.com/papers/Governing_Agentic_AI.pdf
8. Anwar U, Saparov A, Rando J, Paleka D et al. Foundational challenges in assuring alignment and safety of large language models. (arXiv:2307.13744); 2024.
9. Gold EM. Language identification in the limit. Inform Control. 1967;10(5):447–74. https://doi.org/10.1016/s0019-9958(67)91165-5
10. Angluin D. Inductive inference of formal languages from positive data. Inform Control. 1980;45(2):117–35. https://doi.org/10.1016/s0019-9958(80)90285-5
11. Johnson K. Gold's theorem and cognitive science. Philos Sci. 2004;71(4):571–92.
12. Strobl L, Merrill W, Weiss G, Chiang D, Angluin D (2024) What formal languages can transformers express? A survey. Trans Assoc Comput Linguist 12:543–61.
13. Bhattamishra S, Ahuja K, Goyal N. On the ability and limitations of transformers to recognize formal languages. In Proc EMNLP 2020; 2020, pp. 7096–116.
14. Merrill W. Linear transformations and the capacity of recurrent neural networks. In: Proc ACL 2020 workshop on deep learning and formal languages; 2020.
15. Peng B, Narayanan S, Papadimitriou C. On the limitations of the transformer architecture; 2024.
16. ORNL. Frontier supercomputer debuts as world's fastest, breaking exascale barrier. Oak Ridge National Lab news release; 2022.
17. Bick A, Blandin A, Mallen J. The rapid adoption of generative AI. (NBER Working Paper No. 32966); 2024.
18. Newman MEJ, Barabási AL, Watts DJ. The structure and dynamics of networks. Princeton University Press; 2006.
19. Christakis NA, Fowler JH. Connected: The surprising power of our social networks and how they shape our lives. Little, Brown; 2009.
20. Waniek M, et al. Hiding individuals and communities in a social network. IEEE Trans Netw Sci Eng. 2022;9(1):196–209.
21. Waniek M, et al. Trading contact tracing precision for privacy via structure-preserving anonymization. Sci Rep. 2022;12:4.
22. Cebrian M, et al. A time-critical crowdsourced computational search for the origins of COVID-19. Nat Electron. 2021;4(1):2–4.
23. Kleinberg J, Mullainathan S. Language generation in the limit; 2024. https://arxiv.org/abs/2404.06757
24. Bommasani R, Soylu D, Liao TI, Creel KA, Liang P. Ecosystem graphs: Documenting the foundation model supply chain. AIES. 2024;7:196–209. https://doi.org/10.1609/aies.v7i1.31629
25. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending certain legislative acts (Artificial Intelligence Act), Official Journal of the European Union, OJ L — 12 July 2024. Available at: https://eur-lex.europa.eu/eli/reg/2024/1689/oj

26. Exec. Order No. 14 110, Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence, 88 Fed. Reg. 75191–212 (Oct. 30, 2023). Available at: https://www.federalregister.gov/documents/2023/11/02/2023-24766/safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence

27. UK Department for Science, Innovation & Technology. AI Regulation: A Pro-Innovation Approach. White Paper CP 815, 29 March 2023 (updated 3 August 2023). Available at https://www.gov.uk/government/publications/ai-regulation-a-pro-innovation-approach/white-paper

28. Organisation for Economic Co-operation and Development (OECD). Recommendation of the Council on Artificial Intelligence (OECD/LEGAL/0449). Adopted 21 May 2019; revised 8 November 2023 and 3 May 2024. Available at https://legalinstruments.oecd.org/en/instruments/oecd-legal-0449

29. Organisation for Economic Co-operation and Development OECD. Global framework to monitor application of the G7 Hiroshima AI code of conduct.

30. Cohen S, Bitton R, Nassi B. Here comes the AI worm: Unleashing zero-click worms that target GenAI-powered applications. arXiv:2403.02817, Cornell Tech, Mar. 2024. Available at https://arxiv.org/abs/2403.02817

31. Burgess M. Here come the AI worms. WIRED; 2024.

32. Knibbs K. Researchers say the deepfake Biden robocall was likely made with tools from AI startup ElevenLabs. WIRED; 2024.

33. Balasubramaniyan V. Pindrop Reveals TTS Engine Behind Biden AI Robocall. Pindrop Research Blog, 25 Jan 2024 (updated 16 July 2025). Available at https://www.pindrop.com/article/pindrop-reveals-tts-engine-behind-biden-ai-robocall/

34. Zakrzewski C, Verma P. New Hampshire opens criminal probe into AI calls impersonating Biden. The Washington Post, 6 February 2024. Available at https://www.washingtonpost.com/technology/2024/02/06/nh-robocalls-ai-biden/