

Can adversarial attacks by large language models be attributed?

Manuel Cebrian
 Center for Automation and Robotics
 Spanish National Research Council
 manuel.cebrian@csic.es

Jan Arne Telle
 Department of Informatics
 University of Bergen
 Jan.Arne.Telle@uib.no

Abstract

Attributing outputs from Large Language Models (LLMs) in adversarial settings—such as cyberattacks and disinformation—presents significant challenges that are likely to grow in importance. We investigate this attribution problem using formal language theory, specifically language identification in the limit as introduced by Gold and extended by Angluin. By modeling LLM outputs as formal languages, we analyze whether finite text samples can uniquely pinpoint the originating model. Our results show that due to the non-identifiability of certain language classes, under some mild assumptions about overlapping outputs from fine-tuned models it is theoretically impossible to attribute outputs to specific LLMs with certainty. This holds also when accounting for expressivity limitations of Transformer architectures. Even with direct model access or comprehensive monitoring, significant computational hurdles impede attribution efforts. These findings highlight an urgent need for proactive measures to mitigate risks posed by adversarial LLM use as their influence continues to expand.

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 [3, 9, 21, 17, 24]. In this context, attribution involves identifying the specific model responsible for generating harmful outputs. This step is essential not only for conducting further investigations and determining if the implicated model should be restricted or decommissioned but also for mitigating future risks and ensuring accountability in the deployment of LLM-based agents [18, 19, 2].

This attribution task is, interestingly, closely related to formal language theory, particularly the problem of language identification in the limit. This theory, introduced by Gold [10] and extended by Angluin [1], has a rich body of work spanning theoretical computer science and cognitive science [11]. Language identification provides a formal framework for understanding how, given a set of outputs, one can determine the source language—or, in this context, the specific LLM responsible for generating those outputs. By framing LLM outputs as formal languages, this theory offers a structured approach to explore the feasibility of attribution.

We can represent the outputs of an LLM as strings over a finite alphabet, where the set of all possible outputs defines a formal language. Thus, the attribution problem can be framed as determining whether there exists a unique LLM whose generated language includes the observed outputs.

To formalize this, we start by revisiting the concept of language identification in the limit, which underlies this notion of unique model identification.

Definition 1 (Gold’s Identification in the Limit). *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 $\langle s_1, s_2, \dots \rangle$ where each $s_i \in L$ and each string in L appears at least once, the algorithm \mathcal{A} produces a sequence of hypotheses $\langle H_1, H_2, \dots \rangle$ satisfying:*

- (1) *For all but finitely many n , $H_n = L$.*
- (2) *Each $H_n \in \mathcal{L}$ is consistent with the observed data up to time n , i.e., $\{s_1, s_2, \dots, s_n\} \subseteq H_n$.*

This framework was extended by Angluin, who provided necessary and sufficient conditions for identification from positive data.

Theorem 1 (Angluin’s Theorem). *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 $\{T_i\}_{i \in \mathbb{N}}$ such that:*

- (1) For each i , $T_i \subseteq L_i$.
- (2) For all $i \neq j$, $T_i \subseteq L_j$ implies $L_i = L_j$.

However, there are classes of languages where these conditions are not met, leading to non-identifiability in the limit. Specifically, this holds for a class \mathcal{L} with an infinite language $L_\infty \in \mathcal{L}$ such that for every finite subset $S \subset L_\infty$, there exists a language $L_j \in \mathcal{L}$ where $S \subseteq L_j \subset L_\infty$. Note the latter inclusion is strict. This result is formalized in the following corollary of Angluin’s Theorem [1].

Corollary 1 (Non-Identifiability Due to Infinite Languages). *Let \mathcal{L} be a collection of recursively enumerable languages satisfying:*

- (i) \mathcal{L} contains an infinite language L_∞ .
- (ii) For every finite subset $S \subset L_\infty$, there exists $L \in \mathcal{L}$ such that $S \subseteq L \subset L_\infty$.

Then, \mathcal{L} is not identifiable in the limit.

Proof. Assume, for contradiction, that \mathcal{L} is identifiable in the limit. By Angluin’s Theorem, 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) ensures that for every finite T_{L_∞} , there exists $L \in \mathcal{L}$ such that $T_{L_\infty} \subseteq L \subset L_\infty$. This contradicts the existence of a tell-tale set that uniquely identifies L_∞ . Therefore, \mathcal{L} is not identifiable in the limit. \square

From this corollary, it is easy to construct a very simple class of languages, over an alphabet consisting of a single character, that is not identifiable in the limit.

Observation 1. *For each positive integer k , let L_k be the set of all strings of length at most k over the alphabet $\Sigma = \{x\}$, and let $\mathcal{L} = \{L_k\}_{k \in \mathbb{N}} \cup \{L_\infty\}$, where $L_\infty = \Sigma^*$ is the set of all finite strings over Σ . Then \mathcal{L} is not identifiable in the limit.*

Proof. Take any finite subset $S \subset L_\infty$ and let the longest string in S have length k . Then $S \subseteq L_k \subset L_\infty$ with $L_k \in \mathcal{L}$, so the conditions of Corollary 1 apply. \square

In the context of LLM attribution, the family of languages \mathcal{L} corresponds to the languages generated by a set of LLMs $\mathcal{M} = \{M_1, M_2, \dots\}$. Each model M_i defines a language $L(M_i) \subseteq \Sigma^*$, where $L(M_i)$ consists of all sequences that M_i can generate with non-zero probability. The goal is to determine whether a finite set of observed outputs $S \subseteq \Sigma^*$ can uniquely identify a specific model M_k such that $S \subseteq L(M_k)$.

Definition 2 (Attribution Algorithm). *An Attribution Algorithm for a set of LLMs \mathcal{M} is a procedure that takes as input a stream of observed outputs $S = \langle s_1, s_2, \dots \rangle$ generated by a single LLM $M_k \in \mathcal{M}$ and outputs a sequence of hypotheses $\langle H_1, H_2, \dots \rangle$ where each $H_n \in \mathcal{M} \cup \{\text{unknown}\}$ represents the algorithm’s guess of which model generated the outputs observed up to time n . The algorithm satisfies the Identification in the Limit property if there exists a time point t such that for all $n \geq t$, $H_n = M_k$. In other words, after observing sufficient outputs, the algorithm will consistently and correctly identify the generating model.*

This reframes the problem of LLM attribution as designing an algorithm capable of identifying unique, model-specific subsets within observed outputs S to reliably determine the generating model. Angluin’s result implies that for attribution to be feasible, a finite set of outputs—a *tell-tale set*—must uniquely identify each LLM’s language. Importantly, this means that the attacker must eventually produce all elements of the language, including the tell-tale set, as we are considering identification in the limit. However, while such tell-tale sets may theoretically exist, Angluin’s framework does not provide a constructive

or practical method for discovering them. The Attribution Algorithm may need to enumerate all possible subsets T_i and verify their uniqueness against observed data, an approach that is computationally infeasible given the vast number of models and their fine-tuned variants (as explored below).

While these theoretical considerations suggest challenges in attributing outputs to specific models, one might consider the inherent expressivity limitations of Transformer architectures — which underlie most LLMs — as a potential aid in attribution. These limitations, as discussed by Strobl et al. [20], restrict Transformers’ ability to represent certain language classes, especially those requiring unbounded counting or hierarchical structures, such as the Dyck languages of balanced parentheses and parity-checking over binary strings [4, 13]. Additionally, Peng et al. [16] highlight Transformers’ limited capacity for recursive function composition, essential for languages with deep structural dependencies. These expressivity constraints might reduce the overlap in output distributions across models, potentially aiding attribution by introducing more distinct and bounded patterns in LLM outputs.

However, despite these expressivity limitations, fine-tuning can create significant overlaps in model outputs, complicating attribution. Specifically, under a mild assumption about the power of fine-tuning the impossibility of attribution satisfying Identification in the Limit can be proven formally.

Observation 2 (Fine-Tuning Leading to Non-Identifiability). *Assume there exists an LLM M_∞ with infinite language such that for any finite subset $S \subset L(M_\infty)$, we can fine-tune the base model M_∞ to produce a model M_S where $S \subseteq L(M_S) \subset L(M_\infty)$. In that case, attribution satisfying identification in the limit is not possible for the class of LLMs $\mathcal{M} = \{M_S \mid S \subset L(M_\infty) \cap |S| < \infty\} \cup \{M_\infty\}$.*

Proof. Consider the class of LLMs \mathcal{M} consisting of M_∞ and, for each finite subset S of $L(M_\infty)$, the fine-tuned model M_S . Note that the class of languages $\mathcal{L} = \{L(M) : M \in \mathcal{M}\}$ will then satisfy the conditions of Corollary 1:

- (i) \mathcal{L} contains the infinite language $L(M_\infty)$.
- (ii) For every finite subset $S \subset L(M_\infty)$, there exists $L(M_S) \in \mathcal{L}$ such that $S \subseteq L(M_S) \subset L(M_\infty)$.

Therefore, by Corollary 1, \mathcal{L} is not identifiable in the limit, and thus attribution with identification in the limit is not possible for \mathcal{M} . □

In a different scenario, we may have direct access to a majority of the LLMs M_i themselves, enabling us to compute the likelihood of an observed output s under each model M_i . Using these likelihoods one can hope to design attribution algorithms with a probabilistic flavor. However, evaluating the likelihood $P_{M_i}(s)$ of an observed output s under each available LLM M_i involves significant computational resources.

We conducted our analysis using a dataset representing the ecosystem of LLMs [6] to understand the growth in model sizes over time and evaluate the computational feasibility of attribution calculations based on likelihood. The dataset consists of several LLMs developed over recent years, detailing key parameters including model type, creation date, and parameter size in billions, including a total of 271 models with known sizes (i.e., parameter counts).

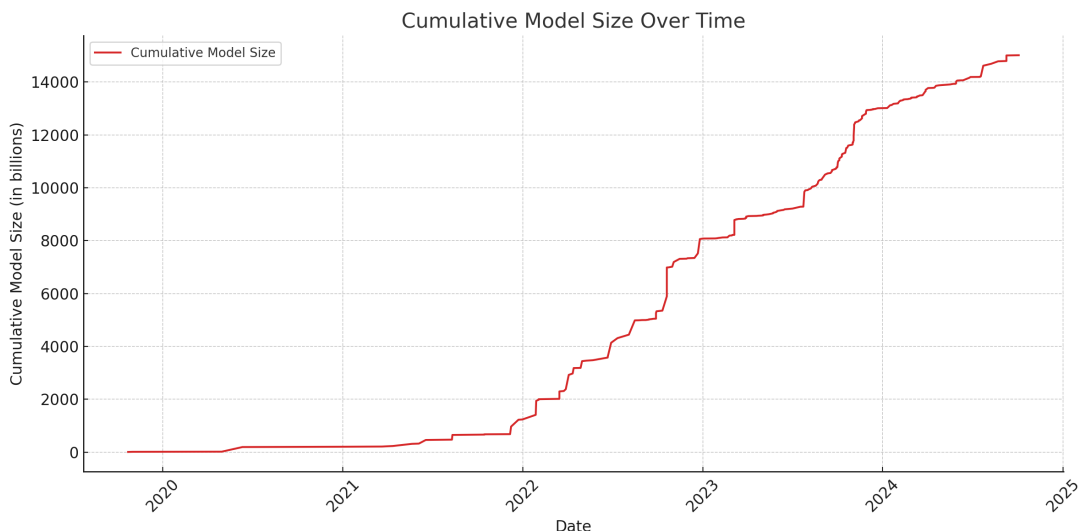
Figure 1 visualizes the evolution of model sizes over time, using known model sizes to calculate a sliding month average and the maximum size observed up to each date. Additionally, we plot the cumulative size across all models to understand the aggregate computational requirement if attribution calculations needed to be run across the entire ecosystem.

To evaluate the practicality of running attribution calculations across all models, we consider a hypothetical adversarial attack involving a 100,000-token output. For each model, we need to calculate the log-probabilities of these tokens to estimate the likelihood of the output originating from that model. Given the cumulative parameter size across all models, the total floating-point operations (FLOPs) required for this calculation grows significantly.

With the current cumulative model size in our dataset, processing a 100,000-token adversarial attack would require approximately 8.7×10^{20} floating-point operations (FLOPs). To assess the feasibility of such a task, we compare this with the capabilities of the Frontier supercomputer—the world’s fastest—achieving a peak of 1.7 exaFLOPS (1.7×10^{18} FLOPs per second) [15]. Using Frontier’s capacity,



(a) Evolution of Model Sizes Over Time (Actual Sizes, 6-month Sliding Mean, and Maximum Size)



(b) Cumulative Model Size Over Time

Figure 1: Overview of Model Size Evolution and Cumulative Computational Requirements in the Ecosystem Dataset

a single attribution calculation across 100,000 tokens could theoretically complete in around 8 minutes for a single attack, underscoring the difficulty of scaling attribution in an expanding landscape of LLMs.

In the final analysis, we assess the feasibility of attributing adversarial outputs to specific LLMs within a comprehensive, national monitoring framework based on U.S. usage data – an extreme scenario. Using realistic simulations and data from Bick et al. (2024) [5], we assume the total U.S. population is approximately 334 million, with 39.4% of adults actively using LLMs daily. This results in an estimated 131.6 million daily users. With an average generation rate of 10,000 tokens per user per day, the annual token volume reaches approximately 4.8×10^{15} tokens. Assuming a data footprint of 4 bytes per token, this volume equates to about 17.45 petabytes of data generated annually.

Processing this extensive dataset demands significant infrastructure. Using a high-performance computing setup with 10,000 cores and estimating each record requires approximately 10 microseconds per token, the baseline processing time to analyze the year’s data is approximately 132.8 hours, assuming optimal parallel processing.

While the core processing time provides a baseline, real-world conditions introduce additional delays due to data aggregation, preprocessing, and network latency. Assuming an overhead for data collection and preprocessing, and applying a multiplier of 2 to account for additional logistical complexities, the total estimated time for attribution increases to approximately 265.6 hours (or roughly 11 days) to attribute a single adversarial output. Table 1 highlights that even under idealized conditions, attributing a single adversarial output from vast data volumes is a complex and resource-intensive task.

Metric	Value
Daily LLM Users (U.S.)	131,596,000
Annual Tokens	4.8×10^{15} tokens
Yearly Data Volume	17.45 PB
Estimated Processing Time (Core Total)	132.8 hours
Adjusted Processing Time with Overhead	265.6 hours

Table 1: Estimated Attribution Time for U.S. LLM Usage

Beyond these challenges of direct identifiability, also issues stemming from network dynamics [14, 8] impact attribution feasibility. In many real-world cases, harmful content or attacks propagate through complex networks, complicating efforts to trace them back to their origin. Attackers can exploit these network structures by rewiring connections or introducing counterfeit nodes, leveraging the diffusion properties of networks to evade detection and mask their identity [22, 23, 7].

Furthermore, Kleinberg and Mullainathan’s surprising findings [12] reveal that while attribution remains challenging, *generation in the limit* is in fact achievable. In a framework similar to Definition 1, their work shows that it is possible for a computational agent to generate new, valid strings from a target language without identifying it explicitly—an ability fundamentally different from language identification- as it bypasses the need for complete model attribution. This capability suggests that, theoretically, defenders could match adversarial capabilities by reproducing attack outputs in the limit, creating a scenario where attackers and defenders are locked in a cycle of confusion, unable to find the true origin, and mutual escalation.

Whether with limited access to adversarial outputs, direct access to models, or even within a comprehensive monitoring framework, attributing specific outputs to individual LLMs remains profoundly challenging. As LLMs become increasingly powerful and widespread, the complexities of attribution raise significant concerns. This underscores the urgent need for robust safety protocols and regulatory measures to mitigate risks before LLMs are made broadly accessible to the public.

References

- [1] Dana Angluin. Inductive inference of formal languages from positive data. *Information and Control*, 45(2):117–135, 1980.
- [2] Umar Anwar, Aziz Saparov, Julia Rando, Daniel Paleka, Michael Turpin, Pete Hase, et al. Foundational challenges in assuring alignment and safety of large language models. *arXiv preprint arXiv:2404.09932*, 2024.
- [3] Robert Axelrod and Radoslav Iliev. Timing of cyber conflict. *Proceedings of the National Academy of Sciences*, 111(4):1298–1303, 2014.
- [4] Satwik Bhattamishra, Arkil Patel, and Navin Goyal. On the computational power of transformers and its implications in sequence modeling. *arXiv preprint arXiv:2006.09286*, 2020.

- [5] Alexander Bick, Adam Blandin, and David J. Deming. The rapid adoption of generative ai. Working Paper w32966, National Bureau of Economic Research, Cambridge, MA, September 2024.
- [6] Rishi Bommasani, Dilara Soylu, Thomas I. Liao, Kathleen A. Creel, and Percy Liang. Ecosystem graphs: The social footprint of foundation models. March 2023.
- [7] Manuel Cebrian. A time-critical crowdsourced computational search for the origins of covid-19. *Nature Electronics*, 4(7):450–451, 2021.
- [8] Nicholas A Christakis and James H Fowler. *Connected: The surprising power of our social networks and how they shape our lives*. Little, Brown Spark, 2009.
- [9] Brian Edwards, Allen Furnas, Steve Forrest, and Robert Axelrod. Strategic aspects of cyberattack, attribution, and blame. *Proceedings of the National Academy of Sciences*, 114(11):2825–2830, 2017.
- [10] E.M. Gold. Language identification in the limit. *Information and Control*, 10(5):447–474, 1967.
- [11] Kent Johnson. Gold’s theorem and cognitive science. *Philosophy of Science*, 71(4):571–592, 2004.
- [12] Jon Kleinberg and Sendhil Mullainathan. Language generation in the limit. *arXiv preprint arXiv:2404.06757*, 2024.
- [13] William Merrill, Vivek Ramanujan, Yoav Goldberg, Roy Schwartz, and Noah Smith. Effects of parameter norm growth during transformer training: Inductive bias from gradient descent. *arXiv preprint arXiv:2010.09697*, 2020.
- [14] Mark Ed Newman, Albert-László Ed Barabási, and Duncan J Watts. *The structure and dynamics of networks*. Princeton university press, 2006.
- [15] Oak Ridge Leadership Computing Facility. Frontier supercomputer.
- [16] Bin Peng, Srikumar Narayanan, and Christos Papadimitriou. On limitations of the transformer architecture. *arXiv preprint arXiv:2309.06863*, 2023.
- [17] Nicole Perlroth. *This is how they tell me the world ends: The cyberweapons arms race*. Bloomsbury publishing, 2021.
- [18] Iyad Rahwan, Manuel Cebrian, Nicholas Obradovich, Josh Bongard, Jean-François Bonnefon, Cynthia Breazeal, Joshua W Crandall, Nicholas A Christakis, Iain D Couzin, Michael O Jackson, et al. Machine behaviour. *Nature*, 568(7753):477–486, 2019.
- [19] Yuval Shavit, Shuchi Agarwal, Miles Brundage, Saurabh Adler, Courtney O’Keefe, Riley Campbell, and David G Robinson. Practices for governing agentic ai systems. Research Paper, OpenAI, December 2023, 2023.
- [20] Lutz Strobl, William Merrill, Gregory Weiss, Daniel Chiang, and Dana Angluin. What formal languages can transformers express? a survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024.
- [21] Fabio Urbina, Filippa Lentzos, Cédric Invernizzi, and Sean Ekins. Dual use of artificial-intelligence-powered drug discovery. *Nature Machine Intelligence*, 4(3):189–191, 2022.
- [22] Marcin Waniek, Petter Holme, Manuel Cebrian, and Talal Rahwan. Social diffusion sources can escape detection. *Iscience*, 25(9):104956, 2022.
- [23] Marcin Waniek, Petter Holme, Katayoun Farrahi, Rémi Emonet, Manuel Cebrian, and Talal Rahwan. Trading contact tracing efficiency for finding patient zero. *Scientific reports*, 12(1):22582, 2022.
- [24] Jiajia Xu, Alice Smith, Liam Johnson, and Kevin Lee. Autoattacker: A large language model guided system to implement automatic cyber-attacks. *arXiv preprint arXiv:2403.01038*, 2024.

Acknowledgments

The authors acknowledge support from OpenAI's ChatGPT in preparing this manuscript.