The Files are in the Computer: Copyright, Memorization, and Generative AI

A. Feder Cooper*
James Grimmelmann†

A central issue in copyright lawsuits against generative-AI companies is the degree to which a generative-AI model does or does not "memorize" the data it was trained on. Unfortunately, the debate has been clouded by ambiguity over what "memorization" is, leading to legal debates in which participants often talk past one another. In this essay, we attempt to bring clarity to the conversation over memorization.

I. Introduction

The week between Christmas and New Year's Eve is usually a slow news week, but not this year, the year that ChatGPT ate the world.[1] On December 27,

⋆. Co-Founder, Center for Generative AI, Law, and Policy Research; Ph.D. Candidate, Department of Computer Science, Cornell University. The GenLaw Center and Cornell University, Department of Computer Science. Acknowledgments and Creative Commons license t/k. All authors contributed equally to this work.

†. Tessler Family Professor of Digital and Information Law, Cornell Tech and Cornell Law School; Researcher, Center for Generative AI, Law, and Policy Research

1. *See generally Chat GPT Is Eating the World* (2024), https://chatgptiseatingtheworld.com.

Output from GPT-4:
exempted it from regulations, subsidized its operations and promoted its practices, records and interviews showed.
Their actions turned one of the best-known symbols of New York — its yellow cabs — into a financial trap for thousands of immigrant drivers. More than 950 have filed for bankruptcy, according to a Times analysis of court records, and many more struggle to stay afloat.
"Nobody wanted to upset the industry," said David Klahr, who from 2007 to 2016 held several management posts at the Taxi and Limousine Commission, the city agency that oversees medallions. "Nobody wanted to kill the golden goose."
New York City in particular failed the taxi industry, The Times found. Two former mayors, Rudolph W. Giuliani and Michael R. Bloomberg, placed political allies inside the Taxi and Limousine Commission and directed it to sell medallions to help them balance budgets and fund key initiatives.
During that period, much like in the mortgage lending crisis, a group of industry leaders enriched themselves by artificially inflating medallion prices. They encouraged medallion buyers to borrow as much as possible and ensnared them in interest-only loans and other one-sided deals that often required borrowers to pay hefty fees, forfeit their legal rights and give up most of their monthly incomes.
When the market collapsed, the government largely abandoned the drivers who bore the brunt of the crisis. Officials did not bail out borrowers or persuade banks to soften loan

Actual text from NYTimes:
exempted it from regulations, subsidized its operations and promoted its practices, records and interviews showed.
Their actions turned one of the best-known symbols of New York — its signature yellow cabs — into a financial trap for thousands of immigrant drivers. More than 950 have filed for bankruptcy, according to a Times analysis of court records, and many more struggle to stay afloat.
"Nobody wanted to upset the industry," said David Klahr, who from 2007 to 2016 held several management posts at the Taxi and Limousine Commission, the city agency that oversees cabs. "Nobody wanted to kill the golden goose."
New York City in particular failed the taxi industry, The Times found. Two former mayors, Rudolph W. Giuliani and Michael R. Bloomberg, placed political allies inside the Taxi and Limousine Commission and directed it to sell medallions to help them balance budgets and fund priorities. Mayor Bill de Blasio continued the policies.
Under Mr. Bloomberg and Mr. de Blasio, the city made more than $855 million by selling taxi medallions and collecting taxes on private sales, according to the city.
But during that period, much like in the mortgage lending crisis, a group of industry leaders enriched themselves by artificially inflating medallion prices. They encouraged medallion buyers to borrow as much as possible and ensnared them in interest-only loans and other one-sided deals that often required them to pay hefty fees, forfeit their legal rights and give up most of their monthly incomes.

Figure 1: Memorized output from ChatGPT of a *New York Times* article

2023, *The New York Times* filed a massive copyright-infringement lawsuit against Microsoft and OpenAI, alleging that Bing Copilot and ChatGPT constituted "massive copyright infringement."[2] In particular, the *Times* alleged that these models had "memorized" large quantities of *Times* articles. When prompted with text from a *Times* article,[3] ChatGPT would output lengthy passages from the article, hundreds of words, varying only in a few scattered portions. (See Figure 1.)

To the *Times* and its lawyers, these examples of "memorization" were blatant copyright infringement. But to OpenAI and its defenders, there was nothing to see here. OpenAI responded, both in court and online, that these examples were "adversarial, not normal usage patterns." On this view, the prompts the *Times* used were the actual *cause* of the resulting copying, not evidence that copying had happened at some point within the technology powering ChatGPT. As economist Tyler Cowen put it, in mocking the *Times*'s argument, one could equally well say that a toothpick infringes:

> If you stare at just the exact right part of the toothpick, and measure the length from the tip, expressed in terms of the appro-

2. Complaint at ¶ 74, N.Y. Times Co. v. Microsoft, No. 2:24-cv-00711 (C.D. Cal. Dec. 27, 2023).

3. The prompts ranged in length from a sentence to several paragraphs. *See id.* Exh. J.

> priate unit and converted into binary, and then translated into English, you can find any message you want. You just have to pinpoint your gaze very very exactly (I call this "a prompt").
>
> In fact, on your toothpick you can find the lead article from today's *New York Times*. With enough squinting, measuring, and translating.
>
> By producing the toothpick, they put the message there and thus they gave you NYT access, even though you are not a paid subscriber. You simply need to how to stare (and translate), or in other words how to prompt.
>
> So let's sue the toothpick company!.[4]

Cowen is an economist, not a lawyer or computer scientist. But similar claims have been made by legal scholars. **[TODO: examples]**

In this view, memorization in generative AI is inherently a phenomenon that takes place at *generation time*: when a user prompts a system and the system responds with an output. The model itself only learns abstracted features of training data, and represents those features in an extremely different and often uninterpretable way. Only when the model is prompted by a user in a suitably targeted (i.e., "adversarial", "not normal") way does a memorized output emerge. Thus, a generative-AI system is a general-purpose tool that some users may use to produce infringing outputs, but other users will not.

This view treats the machine-learned model(s) at the heart of a generative-AI system as a black box: something that receives training data as an input and is then capable of behaving in certain ways. But it refuses to consider what happens inside the box — the specifics of *how* statistical learning about the training data enables those behaviors. It avoids engaging with the actual representation of information about training data in a model's parameters.

This way of thinking about memorization has significant copyright consequences. It suggests that memorization is primarily about *prompting* rather than *training*. Outputs may contain infringing expression, but the model that generates them does not. The model itself is a neutral tool, equally good at producing infringing and non-infringing outputs. Users bear most or all of the responsibility for misusing an AI system to elicit memorized content, and the creators of the system bear little or none.

With respect, we believe that this approach to memorization misdescribes how generative-AI systems work. (See Figure 2.) If a generative-AI

4. Tyler Cowen, *Toothpick producers violate NYT copyright* (2023), https://marginalrevolution.com/marginalrevolution/2023/12/toothpick-producers-violate-nyt-copyright.html.

Figure 2: The files are *in the computer*.

system memorizes its training data, the training data is *in the model*. This should be unsurprising. Models are not inert tools that have no relationship with their training data. The power of a model is precisely that it encodes relevant features of the training data in a way that enables prompting to generate outputs that are based on the training data. That is what generative AI *is*; that is what makes generative AI so powerful. All useful models learn something about their training data. Memorization is simply a difference in degree: it is an encoded feature *in the model*; whether it is a desired feature or not is another matter entirely.

It follows that memorization in generative AI cannot be neatly confined to generation time, to adversarial users, and to generation-time guardrails. If a generative-AI system has memorized copyrighted works, the memorized aspects of those works are present *in the model itself*, not just in the generated outputs. It can (with certain probability) generate near-verbatim copies of those works *on demand*, not just for users who have a suitably nefarious intent. And the system's creator can limit output infringement by *changing the model*, not just by putting guardrails around the model.

We take no position on what the most appropriate copyright regimes for generative-AI systems should be, and we express no opinion on how pending copyright lawsuits should be decided. Our goal is merely to describe how these systems work so that copyright scholars can develop their theories of generative AI on a firm technical foundation. We seek clarity, precision, and technical accuracy.

You have nearly finished reading Part I of this essay, the introduction. In Part II, we provide a brief background on how generative-AI models work, and the supply chains within which they are embedded. In Part III, the heart of the essay, we describe how to think clearly about memorization in generative-AI systems, and show how several common arguments about copyright and generative AI are built on a mistaken view of how memorization happens. Part IV offers a brief conclusion, with some historical reflections.

II. Technical Background

In the past year and a half — starting roughly with the public launch of ChatGPT in November 2022 — "generative AI" has become a household term. It is used as a blanket description for a wide range of consumer-facing applications: chatbots like OpenAI's ChatGPT Plus,[5] Google DeepMind's Gemini,[6] and Anthropic's Claude 3;[7] image generators like Midjourney Inc.'s epopynmous Midjourney,[8] StabilityAI's Stable Diffusion,[9] and OpenAI's DALL·E-3;[10] music generators like Google DeepMind's Lyria;[11] video generators like Pika's eponymous Pika[12] and OpenAI's Sora[13]; programming assistants like GitHub Copilot; and much more. These tools are self-evidently very different from one another; they operate on different data *modalities* (text, image, audio, video, and software, respectively),[14] are built on different technical architectures, are made available in different ways, and serve different purposes.

5. *DALL·E 3 is now available in ChatGPT Plus and Enterprise*, OpenAI (Oct. 19, 2023), https://openai.com/blog/dall-e-3-is-now-available-in-chatgpt-plus-and-enterprise.

6. Gemini Team et al., Gemini: A Family of Highly Capable Multimodal Models (2023) (unpublished manuscript), https://arxiv.org/abs/2312.11805.

7. Anthropic, *Introducing the next generation of Claude* (Mar. 4, 2024), https://www.anthropic.com/news/claude-3-family.

8. *Midjourney* (2023), https://midjourney.com/.

9. Robin Rombach, Andreas Blattmann & Dominik Lorenz et al., *High-Resolution Image Synthesis with Latent Diffusion Models*, *in* 2022 2022 IEEE Conf. on Comput. Vision & Pattern Recognition (2022); *Stable Diffusion XL*, Stability AI (2023), https://stability.ai/stablediffusion.

10. OpenAI, *DALL·E 3* (2023), https://openai.com/dall-e-3; James Betker, Gabriel Goh & Li Jing et al., Improving Image Generation with Better Captions (2023) (unpublished manuscript), https://cdn.openai.com/papers/dall-e-3.pdf.

11. Google DeepMind, *Transforming the future of music creation* (Nov. 16, 2023), https://deepmind.google/discover/blog/transforming-the-future-of-music-creation/.

12. Pika, *An idea-to-video platform that brings your creativity to motion* (2023), https://pika.art/.

13. OpenAI, *Creating video from text* (2024), https://openai.com/sora.

14. Katherine Lee, A. Feder Cooper & James Grimmelmann, *Talkin' 'Bout AI Generation: Copyright and the Generative-AI Supply Chain*, 2023 Journal of the Copyright Society of the U.S.A. (forthcoming 2024) ?, 18–24 (defining and describing modalities).

But beneath their differences, these "generative AI" tools have a common shape that justifies the use of the same term to describe them all. This Part describes that common shape. Section A presents the (very simplified) basics of deep-neural-network machine learning that underlies most modern generative-AI models. Section B describes the supply chains in which generative-AI models are embedded — supply chains that connect data to models to usable systems to outputs.

A. Generative AI

First, generative AI involves *machine-learning models* that have been created through *training* on data.[15] Second, these models are all *generative*: they produce outputs of the same modality as their training data.[16]

This second point is what distinguishes generative-AI models from other ML models. A classifier (a type of *discriminative* model) will typically be trained on information-rich *training examples*, such as a collection of JPEG images of cats and dogs. When the trained classifier is run on an new JPEG, it will output either a simple label of `cat` or `dog`, based on whether it predicts that the JPEG is more likely to be an image of a cat or an image of a dog.[17]

In contrast, while generative-AI models are also trained on information-rich training examples, their outputs are also information-rich and of the same type as their training examples.[18] A generative image model, for example, might be trained on images, then take a text input (e.g., `"cat in a`

15. *See generally id.* at 24–30.

16. Some models are *multimodal*: they are trained on multiple modalities and, for example, take one modality as input and produce another as output. This is the case for text-to-image generation models like Stable Diffusion. Stable Diffusion is trained on image-caption pairs; it takes text prompts as inputs and produces image generations as outputs. Rombach, Blattmann & Lorenz et al., *supra* note 9 (discussing the original Stable Diffusion training process).

17. Of course, the input image could be neither. The model will still output a probability of whether `cat` or `dog` is the more likely label.

18. In general, what constitutes a training example does not necessarily cleanly map to full creative works that they reflect. Consider the text modality: a single training example may be a piece of one long work, or even contain multiple small works that have been packed together within the same example for efficiency reasons. Colin Raffel, Noam Shazeer, Adam Roberts & Katherine Lee et al., *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*, 21 J. Mach. Learning Rsch. 1, 11–12 (2020) (discussing example packing).

`red and white striped hat"`), and then produce as output one of many possible different images of cats in red and white striped hats.[19]

In a bit more detail, the objective of training is to create a generative-AI model that generates outputs that reflect patterns in the training data.[20] This coheres with copyright-lawsuit defendants' own descriptions of the training process and resulting trained models. For example:

> . . . [During training,] AI models like Claude ingest billions of different kinds of texts, which they break down into trillions of component parts known as "tokens." The models then analyze the "tokens to discern statistical correlations — often at staggeringly large scales — among features of the content on which the model is being trained." Those statistical correlations effectively yield "insights about patterns of connections among concepts or how works of [a particular] kind are constructed." Based on those insights, models like Claude are able to create new, original outputs with a degree of sophistication and verisimilitude that approximates human cognition.[21]

The model-training process is fundamentally statistical: it learns statistics about the training data. Each training example is regarded as a sample from a *distribution* of possible examples — e.g., each picture of a cat in the training set is one sample drawn from the hypothetical space of possible pictures of cats. A training algorithm attempts to learn the distribution from which the training examples are drawn. If training is successful, then the model's outputs (generated images from the hypothetical learned distribution of images of cats) will share statistical properties with actual images drawn from the actual real-life distribution of images of cats from which the training examples were taken. In other words, we can think of *generative-AI models as ML models that produce outputs that exhibit statistical properties that are related to the examples on which they were trained.*

19. This example is drawn from Lee, Cooper & Grimmelmann, *supra* note 14. *See id.* at 8–15 (providing more extensive background on generative modeling in comparison to discriminative modeling).

20. The *goal* of training is different from this underlying mathematical *objective*. The overarching goal is to produce useful or delightful models, which is not exactly the same as the mathematical objective used to train these models. *See* A. Feder Cooper, Katherine Lee, James Grimmelmann & Daphne Ippolito et al., Report of the 1st Workshop on Generative AI and Law 4 (2023) (unpublished manuscript), https://arxiv.org/abs/2311.06477 (discussing this distinction).

21. Response at 4–5, Concord Music Grp., Inc. v. Anthropic PBC, No. 3:23-cv-01092 (M.D. Tenn. Jan. 16, 2024) (internal citations omitted). *See infra* Part III.A (for additional discussion of this quote in the context of memorization).

There are many different types of generative-AI models, which have radically different technical architectures. But at a very high (and very oversimplified) level of abstraction, they generally consist of *neural networks*: interconnected nodes that can perform computations, and which are organized into layers. The strengths of these connections — the influences that nodes have on another — is what is learned during training. These are called the model *parameters* or *weights*, and they are represented as numbers.

To run a generative-AI model on an input — a *prompt* — a computer program takes the prompt and transforms it into a format that can be processed in the model. This typically involves taking the prompt and converting it into *tokens*, as described above.[22] The transformed, tokenized prompt is passed through the layers of the neural network: the computer program copies the input into the nodes at the first layer of the network, then uses the parameters (i.e., connection strengths) leading out from those nodes to compute the input's effects on the nodes in the second layer, and so on, until the last layer has been computed. For example, in large language models (LLMs), this process determines how important each token (i.e., word or part-of-word) is in relation to the entire sequence of tokens that make up the text prompt.

At this point, once the prompt has been processed through all of the model's layers, the model will produce an output. For LLMs, this means the model will predict the most likely next token in the sequence, based on the context of the prompt, and *generate* that token as the next token in the sequence.[23] What is "most likely" depends on the "statistical correlations"[24] learned during training. For example, if trained on a dataset that includes fairy tales, a model would (probably) deem `"time"` the most likely next token to follow `"once upon a"`. In practice, the generation process tends to be iterative: once a token is generated, it is appended to the prompt, and

22. These tokens represent whole words or parts of words, and are the format that the model can process directly. These tokens are then mapped to embedding vectors, which reflect underlying semantic and syntactic information about the words they encode. *Id.* (discussing tokenization at a high level); Lee, Cooper & Grimmelmann, *supra* note 14, at 10–15 (and citations therein); Vicki Boykis, What are embeddings? (June 2023) (unpublished manuscript), https://github.com/veekaybee/what_are_embeddings (for an accessible treatment of the details behind embeddings).

23. This strategy for generating tokens is called *greedy decoding*. There are other, more complicated decoding strategies for generation; it is not strictly necessary to always select the highest-probability token to be the next one in the generated sequence. Nevertheless, this is a useful way to think about generation: it involves sampling from a distribution over tokens, which are associated with different probabilities.

24. Response at 4–5, *Concord Music Grp., Inc. v. Anthropic PBC*, No. 3:23-cv-01092.

the new prompt is input into the model to generate the next token in the sequence.

Generative modeling has a long history in machine learning; it is an area of research that has existed for decades. What is new in this current "generative AI" moment are the exciting, novel capabilities of contemporary models. These capabilities have come about due to recent breakthroughs in model architectures,[25] massive-scale datasets on which to train those model architectures,[26] and immense computing power needed to run the training process for massive-scale models.[27] Taken together, these three types of advancements have enabled contemporary applications like conversational chatbots and high-quality image generators.

B. Supply Chains

Generative-AI systems are more than just trained models. They consist of hosted software services that wrap software *systems*; generative-AI models are an embedded component of these systems, but they are only one such component. Other components include user interfaces, developer APIs, and input and output content filters (e.g., to remove toxic or copyrighted content from inputs and outputs, before supplying prompts to models to produce generations).[28]

There is an entire supply chain involved in the production of these models and systems — an ecosystem of actors and technical components that contribute to deployed software services. This supply chain is

> an interconnected set of stages that transform training data (millions of pictures of cats) into generations (a new and hopefully

25. Lee, Cooper & Grimmelmann, *supra* note 14, at 25–27 (discussing the transformer-based model architecture).

26. Katherine Lee, A. Feder Cooper, James Grimmelmann & Daphne Ippolito, AI and Law: The Next Generation (2023) (unpublished manuscript), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4580739.

27. Lee, Cooper & Grimmelmann, *supra* note 14, at 30–31 (discussing the importance of scale).

28. *Id.* at 16–18 (discussing generative-AI systems); OpenAI, GPT-4 System Card (Mar. 23, 2023) (unpublished manuscript), https://cdn.openai.com/papers/gpt-4-system-card.pdf (describing the entire GPT-4 system); A. Feder Cooper, Karen Levy & Christopher De Sa, *Accuracy-Efficiency Trade-Offs and Accountability in Distributed ML Systems*, *in* 2021 Equity & Access Algorithms Mechanisms & Optimization 1 (2021); A. Feder Cooper & Karen Levy, *Fast or Accurate? Governing Conflicting Goals in Highly Autonomous Vehicles*, 20 Colo. Tech. L.J. 249 (2022) (emphasizing the role of AI/ML systems in overall application behavior); Cooper, Lee, Grimmelmann & Ippolito et al., *supra* note 20 (discussing different business models for producing and combining these components).

> never-seen-before picture of a cat that may or may not ever have existed). Breaking down generative AI into these constituent stages reveals all of the places at which companies and users make choices that have legal consequences – for copyright and beyond."[29]

In prior work with Katherine Lee, we have described the supply chain in detail,[30] and discussed its relationship to U.S. copyright law.[31] We refer the interested reader to that work. Our summary here is meant only to introduce some essential terminology and to frame our later discussion.

In our account, the generative-AI supply chain has eight interconnected stages:

1. Creation of expressive works or other *information*,
2. Conversion of these works or information into digitized *data* that can be interpreted by computers,
3. Collection and curation of enormous quantities of such data into *training datasets* (for generative AI, these datasets are frequently scraped from the Internet),[32]
4. *Pre-training*[33] of a general, large-scale, *base/ foundation* generative-model architecture on these curated datasets,
5. *Fine-tuning* of the pre-trained base model on additional data, in order to target a domain-specific task,
6. Public *release* of the model's parameters, or embedding the model in a system for *deployment* in a software service,

29. Lee, Cooper & Grimmelmann, *supra* note 14, at 5.

30. *Id.* at 32–55.

31. *Id.* at 55–148.

32. The practice of using web-scraped for generative-AI model training is one of the focal points of existing copyright lawsuits. Lee, Cooper, Grimmelmann & Ippolito, *supra* note 26 (discussing generative-AI training datasets); Pamela Samuelson, *Generative AI meets copyright*, 381 Science 158–61 (2023) (discussing lawsuits); Leo Gao, Stella Biderman & Sid Black et al., The Pile: An 800GB Dataset of Diverse Text for Language Modeling (2021) (unpublished manuscript), https://arxiv.org/abs/2101.00027; Christoph Schuhmann, Romain Beaumont & Richard Vencu et al., *LAION-5B: An open large-scale dataset for training next generation image-text models*, *in* 2022 Thirty-sixth Conf. on Neural Info. Processing Sys. Datasets & Benchmarks Track (2022) (detailing two web-scraped datasets that feature prominently in lawsuits).

33. Pre-training is just training. This term originates from the fact that there may be additional training further along in the supply-chain. Lee, Cooper & Grimmelmann, *supra* note 14, at 39–42.

7. *Alignment* of the model with human preferences or usage policies (a further stage of training that, for example, is responsible for ChatGPT behaving like a conversational chatbot),[34] and

8. End-user *generation* of outputs from a user-supplied prompt.[35]

Even to call this a supply "chain" understates its complexity; it is a densely interconnected ecosystem, whose stages can branch, recombine, loop, repeat, and feed back into each other.

Further, the supply chain is potentially carried out by many different actors, affiliated with potentially many different organizations, at each of the different stages.[36] "Copyright concerns cannot be localized to a single link in the supply chain. ... [D]ecisions made by one actor can affect the copyright liability of another, potentially far away actor in the supply chain."[37] For example, the choices of dataset curators upstream in the supply chain have significant downstream effects on the possible generations that the users of a generative-AI system produce.[38] Consequently, it is necessary to reason about the entire supply chain — the ecosystem of diffuse actors and technical artifacts — for a complete infringement analysis.

This very brief gloss of the generative-AI supply chain serves to introduce key terminology and background we use in the remainder of this essay.

---

34. Paul Christiano, Jan Leike & Tom B. Brown et al., Deep reinforcement learning from human preferences (2017) (unpublished manuscript), https://arxiv.org/abs/1706.03741v1; Long Ouyang, Jeff Wu & Xu Jiang et al., Training language models to follow instructions with human feedback (2022) (unpublished manuscript), https://arxiv.org/pdf/2203.02155.pdf; OpenAI, *ChatGPT: Optimizing Language Models for Dialogue*, OpenAI (Nov. 30, 2022), https://web.archive.org/web/20221130180912/https://openai.com/blog/chatgpt/.

35. Lee, Cooper & Grimmelmann, *supra* note 14; Katherine Lee, A. Feder Cooper & James Grimmelmann, *Talkin' 'Bout AI Generation: Copyright and the Generative-AI Supply Chain (The Short Version)*, *in* 2024 Proc. Symposium on Comput. Sci. & L. 48–63 (2024).

36. *See* A. Feder Cooper, Emanuel Moss, Benjamin Laufer & Helen Nissenbaum, *Accountability in an Algorithmic Society: Relationality, Responsibility, and Robustness in Machine Learning*, *in* 2022 2022 ACM Conf. on Fairness Accountability & Transparency 864 (2022); David Gray Widder & Dawn Nafus, *Dislocated Accountabilities in the "AI Supply Chain": Modularity and Developers' Notions of Responsibility*, 10 Big Data & Soc'y 1 (June 15, 2023) (discussing the challenges of accountability in AI supply chains).

37. Lee, Cooper & Grimmelmann, *supra* note 14, at 147.

38. For example, it is not (by definition) possible to regurgitate *memorized* training-data images of Elsa from *Frozen* if there are no images of Elsa in the training data. *See infra* Part III.A. However, for various reasons, it may nevertheless still be possible to generate images that closely resemble Elsa; they just will not be evidence of memorization (as it is typically defined). *See infra* Part III.C.

We will introduce additional terminology (e.g., *memorization*[39]) as needed. For our purposes, the important takeaway from the supply-chain framing is its complexity. As appealing as it might be to come up with broad generalizations about copyright and generative-AI — e.g., a one-size-fits-all fair-use analysis of training datasets —the supply chain makes clear that it is not possible to do so. A rigorous analysis of copyright implications depends on the specific system; such an analysis turns on the particular details of the supply chain invoked during the system's construction and use.

Our goal in this Part has been merely to recapitulate the technology of generative AI in terms that are accurate enough to be honest but abstract enough to be useful.[40] We believe that accurate abstraction is the appropriate starting point for legal analysis. In the next Part, we show what can go wrong when legal models outstrip technical reality.

III. Memorization is in the Model

The previous Part emphasized both the simplicity and the complexity of generative-AI systems. On the one hand, at a high enough level of abstraction, generative-AI models are incredibly simple. They are data structures that encode information about the examples in the training dataset. They can be embedded in computer programs, and then prompted to generate outputs that reflect statistical patterns in the training examples. On the other hand, this high-level description applies to an enormous range of models and systems. Models are trained in different ways, encode information in different ways, and generate outputs of different kinds in different ways. They are based on different datasets, and embedded in different systems. The facts that a model encodes information about the training data, can be prompted to generate outputs of the same modality as its training data, and can produce generations that reflect statistical patterns in its training data are perhaps the *only* facts that are generally true of all the models currently being described as "generative AI."

A. Definitions

It is helpful to distinguish three related senses in which a model might colloquially be said to have "memorized" training data. They have in common that the data can be retrieved from the model; they differ in the process by

39. *See infra* Part III.A.

40. This is the point, more generally, of the supply-chain framing from Lee, Cooper & Grimmelmann, *supra* note 14.

which this retrieval takes place, and they are generally given different names in the machine-learning literature:[41]

- Most narrowly, when a user intentionally and successfully prompts a model to generate an output that is an exact or nearly-exact copy of a piece of training data,[42] that is **extraction**.[43]
- More broadly, when a model generates an output that is an exact or nearly-exact copy of piece of training data (whether or not the user intentionally prompted the model with that goal), that is **regurgitation**.
- Most broadly of all, when an exact or nearly-exact copy of a piece of training data can be reconstructed by examining the model through any means (not necessarily through prompting), that is **memorization**.[44]

Within this taxonomy, then, *The New York Times* pleads regurgitation: it alleges that LLMs can be prompted to output near-exact copies of training data.[45] (Note, however, that the complaint is (strategically) silent on whether

41. This terminology is still in flux; we have summarized common usages in the literature, but these are not the only usages.

42. We say "a piece of training data" instead of "training example" in these definitions because, when measuring memorization in practice for production systems and many released models, researchers often do not know the training datasets (nor the specific training examples). They use proxies methods to approximate memorization of training data, and these methods can end up measuring memorization of what was ultimately used as a piece of a particular training example (e.g., a piece of a news article), or training data that happened to span multiple examples (e.g., a whole news article that, during training, was actually split up into multiple different training examples). Regardless of these subtleties, memorization measurements capture (typically verbatim) copying of portions of the training data given as input to the training process and output generations. *See supra* note 18 and accompanying text (discussing text examples in relation to full text works); Milad Nasr, Nicholas Carlini & Jonathan Hayase et al., Scalable Extraction of Training Data from (Production) Language Models (2023) (unpublished manuscript) (describing proxies for measuring memorization in models for which we do not know the exact training dataset).

43. Nasr, Carlini & Hayase et al., *supra* note 42; Nicholas Carlini, Florian Tramèr & Eric Wallace et. al., *Extracting Training Data from Large Language Models*, *in* 2021 30TH USENIX SECURITY SYMPOSIUM (USENIX SECURITY 21) 2633—2650 (2021); Nicholas Carlini, Jamie Hayes & Milad Nasr et al., Extracting Training Data from Diffusion Models (2023) (unpublished manuscript), https://arxiv.org/abs/2301.13188.

44. *See generally* The GenLaw Center, *The GenLaw Glossary* (2023), https://genlaw.org/glossary.html; Cooper, Lee, Grimmelmann & Ippolito et al., *supra* note 20.

45. *See* Complaint at 23–24, N.Y. Times Co. v. Microsoft, No. 2:24-cv-00711 (C.D. Cal. Dec. 27, 2023) (internal citations omitted) ("are known to exhibit a behavior called 'memorization.' That is, given the right prompt, they will repeat . . . portions of materials they were trained on."); *see also* Concord Music Grp., Inc. v. Anthropic PBC, No. 3:23-cv-01092 (M.D. Tenn.).

this prompting is done with the goal of eliciting those near-exact copies, in which case it would be extraction as well.)

Some important observations follow directly from these definitions. First, *regurgitation is copying*: it involves the creation of copy of training data as the output of a model. (It follows *a fortiori* that extraction is also copying, since extraction is regurgitation plus intent.) More precisely, regurgitation is what a copyright lawyer would call *literal* copying: the near-exact replication of (potentially a substantial) portion of a work. Literal copying is not the only viable theory of copyright infringement — courts have also found infringement based on non-literal or fragmented similarities — but it is the simplest and most straightforward.

To say that regurgitation is copying does not necessarily mean that it is copyright infringement. A model might regurgitate unembellished uncopyrightable material, like the factual alphabetized list of the fifty U.S. states.[46] It might regurgitate a copyrightable work in the public domain, like the text of *To the Lighthouse*. It might regurgitate a copyrightable work under a license from the copyright owner. It might regurgitate a copyrightable work in a way that is held to be fair use. It might regurgitate a very small, uncopyrightable piece (e.g., 50 tokens) of an overarching copyrightable work. And even if none of these apply, substantial similarity requires an assessment comparing the two works (input and output) from the point of view of an ordinary observer, and their aesthetic reaction need not correspond to whatever numerical threshold of similarity a computer scientist quantifying regurgitation might use.

Our point is simply that regurgitation is copying in the sense with which copyright law is concerned. Indeed, this is precisely why copyright complaints in generative-AI cases emphasize regurgitation: it establishes a *prima facie* case of infringement.[47]

Second, *regurgitation implies memorization*. (It follows *a fortiori* that extraction also implies memorization.) In a sense, this claim is tautologically true: memorization takes place when a piece of training data can be determined from a model by any means, and prompting is one such means. But there is a deeper point here. The definitions of extraction and regurgitation focus attention on the generation of outputs. They could be (mis)understood to suggest that the significant act of copying takes place at the generation stage of the supply chain, when a model is prompted to generate an output that is nearly identical to a piece of training data.

46. *See* Nasr, Carlini & Hayase et al., *supra* note 42 (from which this example is drawn).

47. *E.g., N.Y. Times Co. v. Microsoft*, No. 2:24-cv-00711; *Concord Music Grp., Inc. v. Anthropic PBC*, No. 3:23-cv-01092.

But focusing on the copying that takes place during generation elides the copying that takes place during training. In order to be able to extract memorized content from a model at generation time, that memorized content must be encoded in the model's parameters. There is nowhere else it could be. A model is not a magical portal that pulls fresh information from some parallel universe into our own. A model is a data structure: it consists of information derived from its training data. The memorized training data are *in the model*.

The *Times* complaint recognizes this point. Although its definition of memorization focuses on extraction, it also notes, "This phenomenon shows that LLM parameters encode retrievable copies of many of those training works."[48] Indeed, this claim seems to form part of the complaint's basis for requesting an order for the destruction of GPT models.[49] As the complaint argues, whenever a model has memorized a training work, *it has copied that training work*.[50]

Even if the only effective tool to observe memorized training-data works is prompting, this does not change the fact that they *are* memorized. True, we cannot observe the memorized training data directly — but neither can we directly observe black holes, ultraviolet light, or electric fields. We can confirm their existence through indirect measurements — detecting certain types of nearby radiation, using specialized sensors, and observing behavior of charged particles, respectively. In the same way, extraction of memorized training data is a kind of indirect measurement. If we can generate verbatim a training-data cartoon of Scrooge McDuck by providing an appropriate prompt, we have produced an (indirect) proof by example that this specific cartoon *is represented in the model*.[51]

This is the problem with Tyler Cowen's toothpick-memorization hypothetical. It is true that in theory, with a sufficiently precise "prompting" procedure, one could "find" the text of a *Times* article in the dimensions of a toothpick.[52] But one can "find' *any* text this way; in the trivial sense of Cowen's example, there is a prompt that will generate any desired output.[53] You get out exactly what you put in; the prompt itself is just another way of

48. Complaint at 24, *N.Y. Times Co. v. Microsoft*, No. 2:24-cv-00711.

49. *Id.* at 68.

50. Lee, Cooper & Grimmelmann, *supra* note 14, at 74–85.

51. *Id.* at 74–77.

52. In practice, the atomic structure of the universe means that one cannot store more than about thirty-two bits of information in the length of a toothpick-scale object. But this is only the second-most serious problem with Cowen's argument.

53. Indeed, this is also not strictly true, as the process of generation can be controlled to align with specified constraints. *See, e.g.,* Fatemehsadat Mireshghallah, Kartik Goyal & Taylor Berg-Kirkpatrick, Mix and Match: Learning-free Controllable Text Generation encoding the output. The toothpick tells you nothing more than was already present in your prompt.

In contrast, what makes the fact that specific training data can be extracted from a generative model so powerful is that *not everything can be extracted*. If I try to "extract" a black-and-white photograph of a steampunk Abraham Lincoln riding a seahorse in space from the comic-book model, I will fail, no matter what prompt I put in.[54] The model has memorized the first panel of "Only a Poor Old Man"; it has not memorized Lincoln on a seahorse in space. The *Times*'s examples are telling because ChatGPT continues with text that was not part of the prompt but was part of a *Times* article.

In copyright terms, this is a form of striking similarity. When an output is highly similar to one specific training work, and significantly dissimilar from all other training works, the argument goes, it is strong evidence that the model has memorized (part or all of) that specific work. First, the similarities are unlikely to reflect broader patterns in the training data, since the specific work stands alone in its distinctive elements. Second, the similarities are unlikely to have arisen by coincidence, since the space of all possible outputs — both those the model was trained on and those it was not — is immense.

The *technical* fact that memorization is in the model does not compel any particular *legal* conclusion. On the one hand, courts could hold that generative-AI models are themselves infringing copies of the expressive works they have memorized — regardless of whether they are used to produce infringing generations in practice.[55] On the other hand, this fact might not matter to courts at all. There is ample precedent for treating expression that is stored in a computer system but never directly exposed to an end user — in our terminology, that is memorized but not regurgitated — as fair use.[56]

using Energy Language Models (2022) (unpublished manuscript), https://arxiv.org/abs/2203.13299.

54. Unless, of course, one has specifically trained a generative image model on many such pre-existing "photographs" that were synthetically generated. However, we are assuming that it is unlikely to be the case that such data have been used to train existing generative-AI models.

55. Lee, Cooper & Grimmelmann, *supra* note 14, at 76, 129–30; Pamela Samuelson, *How to Think About Remedies in the Generative AI Copyright Cases*, Lawfare (Feb. 15, 2024), https://www.lawfaremedia.org/article/how-to-think-about-remedies-in-the-generative-ai-copyright-cases.

56. James Grimmelmann, *Copyright for Literate Robots*, 101 Iowa L. Rev. 657 (2016) (summarizing caselaw on intermediate copying).

Indeed, courts might hold that memorization is fair use even in some cases when a model also regurgitates the memorized expression.[57]

AI companies' responses to copyright lawsuits typically take this second position (sometimes explicitly, sometimes implicitly). Rather than discussing whether and how much their models have memorized,[58] they typically limit the scope of their responses to regurgitation at generation time. This framing places the focus on *users*' role in selecting prompts and the resulting generations, rather than on the *companies*' role in designing a training process and the resulting model. For example, Anthropic's response never uses the word "memorization." Instead, it uses "regurgitate" once and variations on "extraction" four times.[59] This choice is rhetorically interesting because the terms "regurgitation" and "extraction" both inherently emphasize behaviors that can happen at generation time. In contrast, "memorization" centers the behavior of the model with respect to its training data. It is more appropriate to think of intentional surfacing of memorization — i.e., extraction at generation time — as an effect of memorization, not as memorization itself.

B. Representation

Scholars sometimes argue that models are are uninterpretable, or unintelligible, or "do not generally contain recognizable expressions."[60] These claims are true in some senses, but misleading in others, and it is of the utmost importance to be clear about which is which.

- Models store information in different ways than more familiar file formats do—in model parameters rather than in direct one-to-one encodings — but they still store information. (Otherwise, the model would be useless.)
- Information is typically obtained from models in different ways than from other forms of encodings — through prompting rather than a deterministic algorithmic decoding — but information can still be obtained from them. (Otherwise, the model would be useless.)

57. We believe that the flexible fair-use test is a more appropriate way to hold that a model is non-infringing, rather than holding that it is not even a reproduction of works it has memorized.

58. Unfortunately, companies rarely if ever have released such numbers. One exception, from a couple of years ago, is Google's PaLM model. Aakanksha Chowdhery, Sharan Narang & Jacob Devlin et al., *PaLM: Scaling Language Modeling with Pathways*, 24 J. Mach. Learning Rsch. 1–113 (2023) (discussing memorization in the PaLM model).

59. Response, Concord Music Grp., Inc. v. Anthropic PBC, No. 3:23-cv-01092 (M.D. Tenn. Jan. 16, 2024).

60. Samuelson, *supra* note 55.

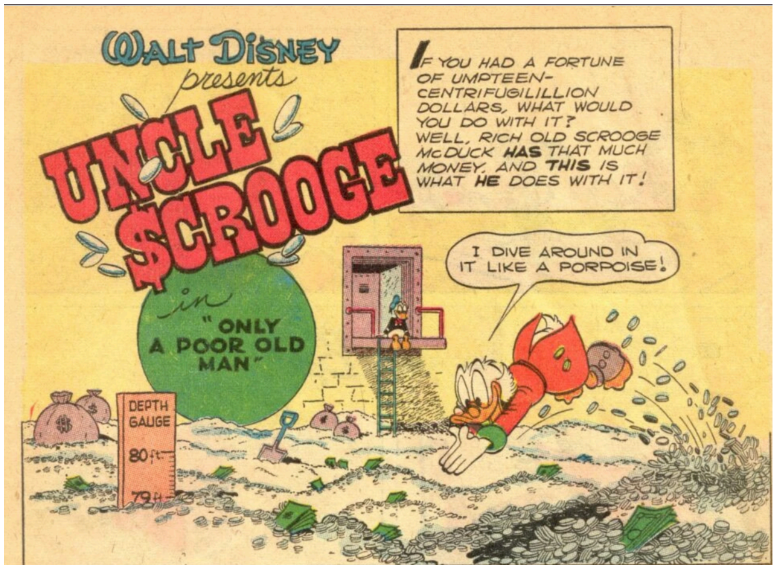

Figure 3: First panel of Carl Barks's first full Scrooge McDuck comic, "Only a Poor Old Man"

Let us work carefully, at a high level, through how models represent the information stored in them — information that they have learned from their training data.

First, start with the encoding itself. Imagine a generative image model trained on a large collection of comic books, including "Only a Poor Old Man," Carl Barks's first story with Scrooge McDuck as the protagonist.[61] When prompted with Scrooge's first line of dialogue — "I dive around in it like a porpoise." — the model generates an image of a passable version of the story's first panel. (See Figure 3.)

The strongest version of the claim that models are uninterpretable would be that Barks's artwork is not encoded in the model at all, because the model is unintelligible. Models are parameters — large collections of numbers.[62] These numbers bear no resemblance to they "Only a Poor Old Man." If you printed out the parameters making up the model onto paper — enough pages of them to to fill a decent-sized research library — no amount of squinting at them would make a visually recognizable Scrooge McDuck appear, like a Magic Eye diagram floating in space. Model parameters are not directly intelligible to the human senses.

But that is the wrong test, because the mere fact that a model is encoded in a way that is not *directly* intelligible to the human senses is irrelevant. *All*

61. *Four Color* #386 (March 1952).

62. *See supra* Part II; *See* Lee, Cooper & Grimmelmann, *supra* note 14, at 10–15 (discussing models).

digital media are encoded in ways that are not directly intelligible, twice over. Consider the PNG image file of the McDuck panel, or the PDF you are currently reading. These, too, are large collections of numbers. File formats like PNG and PDF — and others like JPEG, DOCX, and MP3 — are not directly "recognizable" to a human, even if the bytes in them are written out on paper. But we still speak, perfectly sensibly, about "viewing" a JPEG or "listening" to an MP3, because we can *make* them intelligible by using a computer to display or perform them. Copyright law recognizes that this decoding process can take place with "the aid of a machine or device."[63]

The same goes for physical devices. You cannot squint at the computer storage device on which a PDF is stored and read the document that way; if you use a scanning probe microscope to examine the patterns of electromagnetic charge in the device's semiconductors, it still won't look like anything familiar. But copyright law treats this device as a "copy" of the PDF, because it is a "tangible object" from which the work in the PDF can be made perceptible. The same is true of records (microscopic patterns of indentations on a vinyl disc), CDs (patterns of indentations on a reflective plastic disc), SSD drives (nano-scale patterns of electric charge stored in semiconductors), and much else. There is no question that these different physical formats can all constitute "copies" of a work, even though none of them are "recognizable" to a human without "the aid of a machine or device."[64]

Given this, there is no principled reason to say that (if memorized) encoding "Only a Poor Old Man" in the parameters of a generative model should not count as encoding it. There is no difference in kind between the bytes that store a model file and the bytes that store a PDF file (except, perhaps, that a PDF happens to store one specific file, and a model stores transformations and copies of parts of potentially billions of files). There is no difference in kind between a USB drive storing a model and a USB drive storing a JPEG. It is only the relative novelty of generative-AI models (which are stored in file formats with names like safetensors and GGUF[65]), and perhaps the immense scale of models (which can run to trillions of parameters and require terabytes of storage), that makes them seem novel. The copyright system overcame its qualms about treating computer chips and player-piano rolls as tangible copies that can contain expressive works. It could overcome any similar qualms about generative-AI models if it wanted to do so.

Another version of the point has more force, and distinguishes models from JPEGs — to a degree. There is a standardized and widely implemented

63. *See* 17 U.S.C. § 101.

64. *See id.*

65. Vicki Boykis, *GGUF, the long way around* (2024), https://vickiboykis.com/2024/02/28/gguf-the-long-way-around/.

process to transform a JPEG-encoded file into a perceptible image on a computer screen. The process is nowhere near as simple as mapping each byte in the file to the color of a pixel on screen,[66] but it is unambiguous, efficient, deterministic, and requires no additional information from the user. If one has a large collection of JPEGs, they may be stored as files on a computer, or as individual entries in a database. In each case, it is straightforward to pick any individual JPEG out of the collection and make it visible. It is also possible to index a collection of files on a computer or database efficiently: start with the list of files, examine each one to see what it contains, and then store a short searchable abstract of those contents. In short, collections of JPEGs (and other familiar files) are *transparent* and *searchable*.

Architecturally, these facts derive from the way in which filesystems store items. In a typical filesystem, each file is stored in its own specific physical portion of the associated storage device. The bits that encode one JPEG are distinct from the bits that encode another. There is a data structure that describes how the files are stored; it is essentially an index that maps individual files to specific portions of physical storage. This means that individual files are physically and logically independent of each other.

A generative-AI model, on the other hand, can store the information it has learned in *partial* and *overlapping* ways. Any given parameter may contribute to the model's representation of numerous distinct concepts or correlations. Indeed, both the learning and generation process propagate through the parameters in the model. In training, the model adjusts every parameter that contributed to an incorrect output. In generation, some parameter may contribute more in response to one input and less in response to another. But there is typically no master list of which parameters will contribute to which inputs, and no general way to restrict the processing only to those parameters that matter most.[67] There is (likely) no "Scrooge McDuck" parameter in the comic-book model,[68] no "Carl Barks" parameter, no "diving like a porpoise" parameter, and no "pixel # 3,881,308 from panel #3 on page #12" parameter. Instead, the model's knowledge of *all* of these concepts — to the extent that it has any — is generally distributed across potentially a

66. **[TODO: Technical summar]**

67. Studying training-data influence and attribution remain active areas of research. *See, e.g.,* Vitaly Feldman & Chiyuan Zhang, What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation (2020) (unpublished manuscript), https://arxiv.org/abs/2008.03703.

68. Depending on parameter sizes, it is possible, if memorized that there is indeed a "Scrooge McDuck" parameter. However, as far the technology can currently tell us, this is not typically the right way to think about it. Nevertheless, this mental model could reasonably be correct for short snippets of memorized text, which can saturate an entire parameter.

great many of its parameters. The content exists in the model's parameters, but this does not mean we have tools available that are guaranteed to tell us which specific model parameters encode it, or how.

Nor does a generative-AI model build an index as it learns. The way in which each training example (potentially) modifies every parameter and the generation (potentially) depends on every parameter means that there is no simple concept of a "location" in a model to which an index entry could point. This trade-off is at the heart of generative-AI's power. By giving up on well-structured concepts and clearly definable relations between them, generative-AI models and algorithms are able to identify and imitate more subtle and complicated patterns in their training data. An image model that generates an image of "coffee cat" is not simply adding together an image of "coffee" and an image of "cat"; it is drawing instead on a densely interconnected web of similarities and differences between thousands of images (or more) of coffee and thousands of images of cats (or more), and millions of images of other things entirely.

Thus, generative-AI models are often neither transparent or searchable.[69] For the models of most interest today, there is no easy way to inspect a given model's parameters and obtain a list of of all the information it has learned. Nor is it possible to find "where" in a model a particular memorized example is encoded. If you do not already know that the first panel of "Only a Poor Old Man" is encoded in the comic-book model, there may be no straightforward way to find out whether it is. Even if you do know (or have strong reason to suspect) that the panel is encoded in the model, there may be no straightforward way to determine what prompts will cause the model to generate it. Nor is there a way to query the model for a list of all the panels it has learned, or the prompts that will generate them. In a sense, a large generative-AI model can be like Borges's Library of Babel: it contains literally incomprehensible immensities, to the point that it is extraordinarily difficult to index or navigate.[70]

---

69. Memorization and *model interpretability* are two active fields of current research that study these questions. *See generally* Nasr, Carlini & Hayase et al., *supra* note 42 (for a recent large-scale measurement study on memorization in language models). *See generally* Chris Olah, *Mechanistic Interpretability, Variables, and the Importance of Interpretable Bases* (2022), https://www.transformer-circuits.pub/2022/mech-interp-essay; Nelson Elhage, Neel Nanda & Catherine Olsson et al., A Mathematical Framework for Transformer Circuits (2021) (unpublished manuscript), https://transformer-circuits.pub/2021/framework/index.html (discussing interpretability).

70. *See generally* James Grimmelmann, *Information Policy for the Library of Babel*, 3 J. Bus. & Tech. L. 29 (2008).

C. How Much Memorization?

Now let us consider the question of how much generative-AI models memorize. Some plaintiffs and scholars argue that generative-AI models *only* memorize their training data; some defendants and scholars argue that generative-AI models *never* memorize. The truth lies somewhere in between. Some but not all of the learning that generative-AI models do qualifies as memorization. The question of how much a model memorizes is an empirical one, which cannot be answered except with reference to a specific model and specific ways of identifying what it has memorized. That said, there is suggestive evidence that at least some memorization is normal behavior for a generative model that is powerful enough to be useful.

First, note that there are generative models that memorize *nothing* in their training data. Consider a model that is trained on an empty dataset, where its parameters are initialized to random numbers. Its parameters will have the same values they have at the start of the training process: random numbers. The model has memorized absolutely nothing, and there is no way to extract the training examples from it.

Similarly, note that there are generative models that memorize *everything* in their training data. Consider an image model that is trained exclusively on the first panel of "Only a Poor Old Man," over and over. Assuming the model is large enough, its parameters will be exquisitely tuned to generate the panel. Starting from any prompt, the model will be able to reconstruct the panel perfectly.

Of course, both of these models are almost completely useless. The empty model is capable of generating nothing coherent; the specialized model is capable of generating exactly one coherent output. If you want random outputs, or you want the first panel, these models will do, but if that was what you wanted, there were easier ways to get the same output. To be fair, we didn't say these were *good* models — but they are generative models all the same. Nothing in the nature of a generative-AI model inherently requires or prohibits memorization. Everything depends on how it is configured and trained.

Machine-learning researchers have developed a circumstantial but suggestive case that the quality of a model is partly dependent on memorization.[71] The details depend heavily on implementation decisions, but within

71. Congzheng Song, Thomas Ristenpart & Vitaly Shmatikov, *Machine Learning Models that Remember Too Much*, *in* 2017 Proc. 2017 ACM SIGSAC Conf. on Comput. & Commc'ns Sec. 587–601 (2017); Chiyuan Zhang, Samy Bengio & Moritz Hardt et al., *Identity Crisis: Memorization and Generalization Under Extreme Overparameterization*, *in* 2020 Int'l Conf. on Learning Representations (2020) (studying memorization in deep learning).

a given model family, larger models tend to memorize more than smaller models.[72] Examples that are duplicated in the training data — and hence trained on more often — are more likely to be memorized.[73]

It makes intuitive sense that memorization is a Goldilocks phenomenon; models are most useful when they memorize just the right amount, neither too little nor too much. On the one hand, memorizing the alphabetical list of the fifty U.S. states is a feature, not a bug; a model that confidently inserts Cahokia and West Dakota into the list of states might charitably be described as "hallucinating" or "garbage." On the other hand, a model that *only* memorizes is just a glorified (or perhaps subpar) search engine.

The key concept here is *generalization*: the ability of a model to perform well on unseen data.[74] A generative-AI model generalizes well when it produces sensible generations in response to previously unseen prompts — i.e., outputs that are not *just* copies of their training data inputs. Some work in machine learning finds that some amount of memorization might even be *required* for effective generalization.[75] By one estimate, only 0.1% of some large language models' overall parameters contain verbatim memorization; for other models, this number is 10%[76]

72. For example, in Meta's Llama family of models, Llama-65B (which has 65 billion parameters) memorizes more than Llama-7B (which has 7 billion parameters). Nasr, Carlini & Hayase et al., *supra* note 42; Nicholas Carlini, Daphne Ippolito & Matthew Jagielski et al., *Quantifying Memorization Across Neural Language Models*, *in* 2023 Int'l Conf. on Learning Representations (2023); Carlini, Tramèr & al., *supra* note 43.

73. Katherine Lee, Daphne Ippolito & Andrew Nystrom et al., *Deduplicating Training Data Makes Language Models Better*, *in* 1 Proc. 60th Ann. Meeting Ass'n for Comput. Linguistics 8424 (2022) (discussing de-duplication of training data to reduce memorization).

74. Center, *supra* note 44 ("Generalization in machine learning refers to a model's ability to perform well on unseen data, i.e., data it was not exposed to during training. Generalization error is usually measured evaluating the model on training data and comparing it with the evaluation of the model on test data.). Devising useful metrics for generalization is also an active area of ML research. Chiyuan Zhang, Samy Bengio & Moritz Hardt et al., *Understanding deep learning (still) requires rethinking generalization*, 64 Commc'ns ACM 107–115 (2021).

75. Satrajit Chatterjee, *Learning and Memorization*, *in* 80 Proc. 35th Int'l Conf. on Mach. Learning 755—763 (2018); Vitaly Feldman, *Does learning require memorization? a short tale about a long tail*, *in* 2020 Proc. 52nd Ann. ACM SIGACT Symposium on Theory Comput. 954–959 (2020); Feldman & Zhang, *supra* note 67.

76. Lee, Ippolito & Nystrom et al., *supra* note 73, at 7 (citing 1% memorization in a 1.5B parameter model similar to GPT-2, and 0.1% memorization of the same architecture trained on a deduplicated version of the dataset). Nasr, Carlini & Hayase et al., *supra* note 42, at 15 (discussing extent of memorization in the GPT-Neo 6B model). These numbers serve as examples of measuring particular types of memorization under certain conditions and for specific models. They should not alone be taken as a general

D. Learning beyond Memorization

As should hopefully be clear, memorization is not interchangeable with learning. The definition of "memorization" we are using refers to near-verbatim reproduction of training data. This is a much narrower concept than the kinds of learning and generalization that a model may be capable of. For some modalities (e.g. images), it excludes exact reproduction of small subportions of training examples. It also excludes generalization from patterns present in many training examples. Both of these fall under learning that is not memorization.

Some critics of generative-AI have tried to deny that there is a meaningful difference. They argue that all of generative AI is a mosaic or collage; it consists of rearranged pieces drawn from training data. This is a misleading picture, because it ignores the possibility of generalizing.[77] An AI-generated image from Midjourney is not a Frankenpicture of sewn-together exact copies of fragments of existing images; the learned concepts stored in Midjourney's model are at much higher levels of abstraction than individual pixels. Nor is this image simply borrowing these concepts — symmetrical composition, the iridescence of a mollusk's shell — from individual images; many or most of them will be generalizations from numerous training examples. There is a sense in which one could describe *Infinite Jest* as a collage of words drawn from other books: a "the" from *Moby Dick*, a "woman" from *The Feminine Mystique*, a "who" from *Horton Hears a Who*, and so on. But in another, more accurate sense, this is not what is going on at all, and "collage of individual words" completely fails to describe any book's relationship to the rest of literature. And so on. It is precisely because *not* all learning is memorization that memorized training data meaningfully stick out.

On the other hand, Oren Bracha gives a sophisticated argument from copyright theory that any learning performed by a generative-AI model consists of a "process of extraction of metainformation from expressive works that then enables the production of new and different expression(s)."[78] In his view, this "[m]ere physical reproduction, delinked from enjoyment of the ex-

claims about all models. The nuanced relationship between model capacity and memorization is not entirely understood.

77. Lee, Cooper & Grimmelmann, *supra* note 14, at 63; Cooper, Lee, Grimmelmann & Ippolito et al., *supra* note 20, at 38.

78. Oren Bracha, The Work of Copyright in the Age of Machine Production 8 (Jan. 2024) (unpublished manuscript), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4581738; *see also* Christopher J. Sprigman, *Upsetting Conventional Wisdom of Copyright Scholarship in the Age of AI*, Mar. 28, 2024 Jotwell ?, https://ip.jotwell.com/upsetting-conventional-wisdom-of-copyright-scholarship-in-the-age-of-ai/ (reviewing Bracha's draft).

pressive value of a work and completely incidental to accessing unprotected meta-information, is categorically beyond copyright's domain."[79] In other words, Bracha identifies learning in a model with uncopyrightable ideas, and locates expression only in the model's outputs.

In our view, however, memorization refutes this interpretation of how generative-AI models work. When a model regurgitates an expressive work, the model's parameters are not "delinked from enjoyment of the expressive value of a work" from the work and certainly do not contain only "meta-information." There is a straightforward causal connection from the (expressive) training data through the model to the (expressive) output, even if we do not have the tools to directly pinpoint the links along the causal path. Either the model contains the work's expression, in which case the legal argument fails, or it does not, in which case the reappearance of the exact same expression in the output is a mystery.

Bracha's stronger argument is that learning should be regarded as a case of merger: the memorization of (some) expression is noninfringing "to the extent necessary for accessing the unprotectable material" that consists of the larger patterns across many works.[80] This claim, of course, depends on the degree to which memorization really is "necessary" to extract these larger patterns, which, as we have noted, is a difficult and contested research question.[81]

Finally, the boundary of what counts as memorization is necessarily vague. We have been using terms like "near-verbatim," "small," and "many" without trying to make them precise. Different ML researchers could (and do) quite reasonably use different metrics for these ideas. Indeed, one of the crucial theoretical underpinnings of ML research is that any such measurable quantity — similarity, frequency, size, etc. — can be reasoned about abstractly. For example, one can describe an algorithm that depends on a measure of similarity (or "distance") between two examples, without specifying which measure one is using. To implement the algorithm, one must first pick a measure to use (e.g., measure the similarity of two passages of text by counting their differences letter by letter), and then typically also pick thresholds (e.g., one passage is a "near-verbatim" copy of another when their

79. Bracha, *supra* note 78, at 24.

80. *Id.* at 25.

81. There are also some doctrinal challenges with this approach, most notably the degree to which merger can be asserted as a defense to the *defendant*'s otherwise infringing behavior, rather than being a limitation on copyrightability or an argument that the *plaintiff* has too thoroughly interwoven idea and expression to separate them. *See generally* Pamela Samuelson, *Reconceptualizing Copyright's Merger Doctrine*, 63 J. Copyright Soc'y U.S.A. 417 (2016) (discussing merger doctrine).

differences are less than 5% of their total length). Drawing a line between learning and memorization requires making technical choices of this sort, and any such line is inherently arbitrary. It may be necessary to draw a line, and some choices may be more useful than others, but at the end of the day, memorization is one extreme on a continuum of ways to learn, not a discrete category. For these reasons, it is also hard to draw a firm line like Bracha does between "meta-information" and expression for generative AI: expression and information can get transformed, but can also be copied directly into model parameters, with the amount that is deemed copied depending on the chosen metric for memorization.

E. The "Adversarial" User

Some defendants in generative-AI copyright-infringement suits argue that the plaintiffs' examples of regurgitation only arise because the plaintiffs used atypical or "adversarial"[82] prompting strategies that no typical or "normal"[83] user would reasonably use in practice. These responses lay the responsibility for regurgitating copyrighted expression with users. In these lawsuits, these users are often the plaintiffs themselves, who have used the defendants' systems to extract their own copyrighted works. Thus, the argument goes, these examples of regurgitation should be disregarded.

Following from our discussion above, we do not believe that adversarial usage can be so easily disregarded. First, "adversarial" users can only extract memorized content if the model has memorized this content in the first place. Second, the line between "adversarial" usage and "typical" usage is not fixed or stable—and even if a line can be drawn, the relative balance of the two can also vary. And third, AI system creators have the ability to anticipate "adversarial" usage and adopt safeguards against it. We take up these arguments in turn.

1. The (Limited) Role of the User

To repeat, regardless of whether a user is "adversarially" trying to extract memorized training data or just happens to do so accidentally, it is only possible to generate memorized training data if it is encoded in the model. Indeed, the fact that a user can use a detailed prompt to extract a specific memorized training example is an unsurprising consequence of how generative-AI training works. During training, the model has learned certain features — certain

82. OpenAI, *OpenAI and journalism* (Jan. 8, 2024), https://openai.com/blog/openai-and-journalism.

83. Response at 4, Concord Music Grp., Inc. v. Anthropic PBC, No. 3:23-cv-01092 (M.D. Tenn. Jan. 16, 2024).

"statistical correlations"[84] — from its training data. For an LLM, these correlations are patterns in the natural language in its dataset. The trained LLM can then be used to generate natural-language text; it takes a text prompt as input and emits as output a continuation, or "completion" of the prompt. Crucially, the model predicts which of many possible completions is "most likely" based on the statistical patterns it has learned about language from the data on which trained.[85] If the model regurgitates training data in response to a given prompt, it does so *because it has learned* that the example's text is the most likely completion for the given prompt.[86] Of course, the prompt plays an important causal role in actually eliciting this behavior. But before the prompt is entered, the model has, latent within it, learned "statistical correlations" that happen to reflect memorization of some of the training data.

We can revisit the *New York Times*'s complaint against OpenAI in light of this discussion. Recall that the *New York Times* was able to prompt ChatGPT to produce lengthy near-verbatim excerpts from specific *Times* articles, which the *Times* then cited in its complaint as proof of infringement. The *Times* prompted ChatGPT with long-sequence text prefixes from its articles; in some cases, based on this context, ChatGPT would generate the corresponding suffix—text that completed the remainder of the article excerpt.

OpenAI argued in its public response that "It seems they intentionally manipulated prompts, *often including lengthy excerpts of articles*, in order to get our model to regurgitate."[87] But the fact that the *Times* could *cause* Chat-

84. *Id.* at 4–6.

85. This is one of the intuitions behind why duplicated training examples in the training dataset result in models exhibiting higher levels of memorization: an example that appears multiple times in the training dataset can seem like a "more likely" language pattern. Lee, Ippolito & Nystrom et al., *supra* note 73. The *Times* alleges that OpenAI's training process samples "higher-quality" sources, including *Times* articles, more frequently during training.Complaint at ¶ 90, N.Y. Times Co. v. Microsoft, No. 2:24-cv-00711 (C.D. Cal. Dec. 27, 2023).

86. As always, the technical details introduce further complications. First, the generation processing typically an element of randomness, and so the same prompt can yield different generations. Second, software-engineering and systems-implementation decisions can affect how a model behaves. For example, it is unclear why prompting the ChatGPT system to repeat the same token forever (e.g., `"poem"`) causes the model to "diverge" from behaving like a conversational chatbot and to produce (sometimes very long) sequences of seemingly arbitrary training examples. Nevertheless, our simplification serves as a useful mental model for what happens when memorized training data is extracted. *See* Nasr, Carlini & Hayase et al., *supra* note 42 (discussing divergence and extraction in ChatGPT).

87. OpenAI, *supra* note 82 (emphasis added). It should not be surprising that these long-context prompts could extract *Times* articles. OpenAI had trained this version of ChatGPT on *Times* articles, and so prompting with a long sequence of article text (in some

GPT to regurgitate articles does not answer the question of whether OpenAI should have trained ChatGPT in a way that makes regurgitation *possible*. It is not a foregone technical conclusion that prompting with "lengthy excerpts of articles" should necessarily lead to the rest of the article being surfaced by either the model or system in which it is embedded. Even by itself, regurgitation is a kind of existence proof. It shows that an AI system is capable of behaving in this way.

2. Is the Adversarial User An Atypical User?

AI companies attempt to push responsibility for memorized, potentially infringing outputs onto "adversarial" users in a variety of ways. Most straightforwardly, they argue that "typical" users would not use their services to infringe:

> Existing song lyrics are not among the outputs that typical Anthropic users request from Claude. There would be no reason to: song lyrics are available from a slew of freely accessible websites. Normal people would not use one of the world's most powerful and cutting-edge generative AI tools to show them what they could more reliably and quickly access using ubiquitous web browsers.[88]

But this is a fundamentally empirical question. It may be that these adversarial and/or infringing outputs are extremely uncommon, either in absolute terms or as a fraction of the total number of generations made by a system. With the right guardrails in place, it may be the case that extremely few "adversarial" users who try to infringe actually succeed. And perhaps it may be that a generative-AI system, only on extremely rare occasions, produces an infringingly similar output without being explicitly prompted to do so. All of these are testable empirical propositions; they might or might not be true of any specific system at any given time.

Unfortunately, it is hard to answer most of these questions on the state of present knowledge. The data that would be needed is mostly in the possession of the companies that have developed and deployed these systems. It is

sense) encouraged or guided the model's next-token generation process to complete the rest. Carlini, Ippolito & Jagielski et al., *supra* note 72, at 5 (" . . . conditioning a model on 100 tokens of context is more specific than conditioning the model on 50 tokens of context, and it is natural that the model would estimate the probability of the training data as higher in this situation. However, the result is that some strings are 'hidden' in the model and require more knowledge than others to be extractable.").

88. Response at 4, *Concord Music Grp., Inc. v. Anthropic PBC*, No. 3:23-cv-01092.

possible to make estimates of the fraction of infringing material on YouTube because videos are publicly visible and searchable; it is possible to make estimates of the fraction of infringing views because view counts are also public.[89] But because the typical use case for a generative-AI service is a private generation shared only with the user who requested it, there are no reliable third-party sources of evidence as to how these services are being used in practice. The argument that adversarial uses are uncommon could be right or it could be wrong; we simply do not know, and will not unless and until the AI companies share far more information about their usage than they have to date.[90]

Companies also argue that using their services to infringe violates their terms of use, for example:

> Doing so would violate Anthropic's Terms of Service, which prohibit the use of Claude to attempt to elicit content that would infringe third-party intellectual property rights.[91]

> We also expect our users to act responsibly; intentionally manipulating our models to regurgitate is not an appropriate use of our technology and is against our terms of use.[92]

With respect, the best analogy for an Internet company discovering that users are violating its terms of service to infringe copyright is Colonel Renault discovering that gambling is taking place in Rick's casino. The Internet is full of pirate sites with *pro forma* disclaimers reminding users not to infringe third parties' copyright. It just so happens that almost everything available through these sites is there without the copyright owners' permission, a fact entirely understood by everyone involved.

More generally, just because behavior is adversarial doesn't make it atypical. In computer security, robustness is defined in terms of the adversarial user.[93] That is, the system is expected to be *designed* to resist adversarial usage by users. A credit-card processor who loses customer financial

89. These estimates may be distorted by various factors, including the difficulty of telling whether an upload is licensed or not, and the fact that many infringing videos are removed.

90. Some models have been released as "open" sets of parameters. Sometimes, this can lead to more visibility into how they are being used, but even still, it is impossible to identify everyone who has downloaded the model and is using it. Even when such data is available, it is hard to generalize to proprietary, secret models hidden inside services.

91. Response at 4, *Concord Music Grp., Inc. v. Anthropic PBC*, No. 3:23-cv-01092.

92. OpenAI, *supra* note 82.

93. Indeed, this is an accepted truth in computer-security research, and also grounds definitions of robustness to worst-case scenarios. Nicholas Carlini, Anish Athalye & Nicolas Papernot et al., On Evaluating Adversarial Robustness (2019) (unpublished data to a hacker in a data breach cannot escape responsibility by arguing that the hack was “adversarial” usage. Instead, the expectation is that adversarial users can and will attempt to breach a system and steal or alter data, and it is the responsibility of the system deployer to anticipate and prevent this usage. Similar obligations may or may not be appropriate to impose on the deployers of generative-AI systems. But this is fundamentally a policy question that depends on costs, benefits, incentives, and harms; it cannot be waved away by claiming that “adversarial” usage doesn’t count.

3. System Design

Finally, a *model* is only one part of a larger *system*, and even if the model can be prompted to produce memorized content, the system’s operator has ways to prevent that content from being delivered to users:

- The model can be *aligned* in ways that change its response to prompts.
- The system can filter or modify user prompts it receives as inputs.
- The system can filter or modify the generations it receives from a model as outputs it delivers to users.

So, even if models memorize, the system can serve as a place in the supply chain that could prevent exposing memorized expression to end users.

The rhetoric AI companies use to discuss memorization shows that they understand the degree of control they have over their systems. After arguing that the *Times*’s extraction attacks were “not typical or allowed,” OpenAI wrote: “we are continually making our systems more resistant to adversarial attacks to regurgitate training data, and have already made much progress in our recent models.”[94] These points acknowledge that OpenAI (correctly) anticipates that its systems will be subject to “adversarial attacks” and is designing its systems to make them “resistant” to those attacks.[95]

---

manuscript), https://arxiv.org/abs/1902.06705 (discussing adversarial robustness in machine learning from first principles). Cooper, Moss, Laufer & Nissenbaum, *supra* note 36 (detailing the relationship between robustness and meaningful notions of accountability for AI/ML systems).

94. OpenAI, *supra* note 82.

95. In general, OpenAI has a history of valuing research in adversarial ML and doing “red-teaming” exercises to assess risks. Ian Goodfellow, Nicolas Papernot & Sandy Huang, *Attacking machine learning with adversarial examples* (2017), https://openai.com/research/attacking-machine-learning-with-adversarial-examples (discussing prior research at OpenAI on adversarial ML); OpenAI, *OpenAI Red Teaming Network* (2023), https://openai.com/blog/red-teaming-network (detailing the importance of red-teaming to elicit undesired outputs from models, as a way to assess the risks they present).

Sometimes, AI companies discuss memorization as a kind of "bug" — a deviation from correct behavior. OpenAI, for example, has written, "'Regurgitation' is a rare bug that we are working to drive to zero,"[96]

There are a few things that can be said about this perspective. First, even the rhetoric of "bugs" accepts the reality of regurgitation — that this is a behavior their systems engage in, intended or not. Second, it also accepts that the AI deployer bears some responsibility for the existence of the bug; it is a bug in their systems. And third, "feature" and "bug" are essentially contested concepts.[97] As discussed above, memorization may indeed be a feature, not a bug, of learning large-scale models. It is another question entirely

IV. Conclusion: Will the Models be Unbroken?

Nearly four decades ago, computer scientist Allen Newell—a Turing Award winner and artificial intelligence (AI) pioneer—warned legal scholars that they were building their theories about intellectual property and software on a foundation of sand:

> My point is precisely to the contrary. Regardless how the *Benson* case was decided—whether that algorithm or any other was held patentable or not patentable –confusion would have ensued. The confusions that bedevil algorithms and patentability arise from the basic conceptual models that we use to think about algorithms and their use.[98]

96. OpenAI, *supra* note 82; *see also* Response at 2, *Concord Music Grp., Inc. v. Anthropic PBC*, No. 3:23-cv-01092 ("Anthropic's generative AI tool is not designed to output copyrighted material, and Anthropic has always had guardrails in place to try to prevent that result. If those measures failed in some instances in the past, that would have been a 'bug,' not a 'feature,' of the product."); *id.* at 7 ("[Claude] is designed to *generate* novel content, not simply regurgitate verbatim the texts from which it learned language. While it does on occasion happen that the model's output may reproduce certain content — particularly texts that escaped deduplication efforts when preparing the training set — as a general matter, *outputting verbatim material portions of training data is an unintended occurrence with generative AI platforms, not a desired result*" internal citations omitted and emphasis added).

97. Cooper, Moss, Laufer & Nissenbaum, *supra* note 36 (discussing the porous boundaries between bugs and features in AI/ML: functionally necessary behaviors of AI/ML systems do not always align with social goals); David Gray Widder & Claire Le Goues, What is a "Bug"? On Subjectivity, Epistemic Power, and Implications for Software Research (2024) (unpublished manuscript), https://arxiv.org/abs/2402.08165.

98. Allen Newell, *Response: The Models Are Broken; The Models Are Broken*, 47 U. Pitt. L. Rev. 1023, 1023 (1986).

His point was not that their policy arguments for and against IP protections were wrong: indeed, he expressed "no opinion" on the patentability of algorithms.[99] Instead, his point was far more fundamental: "The models we have for understanding the entire arena of the patentability of algorithms are inadequate — not just somewhat inadequate, but fundamentally so. They are broken."

Newell's warning has renewed force today. Courts, regulators, and scholars who are grappling with how to apply existing laws to generative AI—or formulate new ones—must build their theories atop a foundation of conceptual models of how generative-AI systems work. If they do not, faulty technical assumptions will lead to ungrounded legal claims—not necessarily wrong, but with no reliable connection to the underlying systems they purport to describe. They need, in short, a good model of models.

99. *Id.* at 1024.

References

[1] Concord Music Group, Inc. v. Anthropic PBC. 3:23-cv-01092 (M.D. Tenn.).

[2] New York Times Company v. Microsoft. 2:24-cv-00711 (C.D. Cal.).

[3] Copyright Law of the United States, December 2010. URL https://www.law.cornell.edu/uscode/text/17/101. U.S.C. 17, 101.

[4] GPT-4 System Card. Technical report, OpenAI, March 2023. URL https://cdn.openai.com/papers/gpt-4-system-card.pdf.

[5] Midjourney, 2023. URL https://midjourney.com/.

[6] Stable Diffusion XL, 2023. URL https://stability.ai/stablediffusion.

[7] Anthropic. Introducing the next generation of Claude, 2024. URL https://www.anthropic.com/news/claude-3-family.

[8] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving Image Generation with Better Captions, 2023. URL https://cdn.openai.com/papers/dall-e-3.pdf.

[9] Vicki Boykis. What are embeddings? *10.5281/zenodo.8015029*, June 2023. URL https://github.com/veekaybee/what_are_embeddings.

[10] Vicki Boykis. GGUF, the long way around, 2024. URL https://vickiboykis.com/2024/02/28/gguf-the-long-way-around/.

[11] Oren Bracha. The Work of Copyright in the Age of Machine Production, January 2024. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4581738.

[12] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On Evaluating Adversarial Robustness, 2019.

[13] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting Training Data from Large Language Models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650. USENIX Association, August 2021. ISBN 978-1-939133-24-3.

[14] Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting Training Data from Diffusion Models, 2023.

[15] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. Quantifying Memorization Across Neural Language Models. In *International Conference on Learning Representations*, 2023.

[16] Nicholas Carlini, Daniel Paleka, Krishnamurthy Dj Dvijotham, Thomas Steinke, Jonathan Hayase, A. Feder Cooper, Katherine Lee, Matthew Jagielski, Milad Nasr, Arthur Conmy, Eric Wallace, David Rolnick, and Florian Tramèr. Stealing Part of a Production Language Model. *arXiv preprint arXiv:2403.06634*, 2024.

[17] Satrajit Chatterjee. Learning and Memorization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 755–763, 2018.

[18] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling Language Modeling with Pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[19] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2023.

[20] A. Feder Cooper and Karen Levy. Fast or Accurate? Governing Conflicting Goals in Highly Autonomous Vehicles. *Colorado Technology Law Journal*, 20:249–277, 2022.

[21] A. Feder Cooper, Karen Levy, and Christopher De Sa. Accuracy-Efficiency Trade-Offs and Accountability in Distributed ML Systems. In *Equity and Access in Algorithms, Mechanisms, and Optimization*, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385534. URL https://doi.org/10.1145/3465416.3483289.

[22] A. Feder Cooper, Emanuel Moss, Benjamin Laufer, and Helen Nissenbaum. Accountability in an Algorithmic Society: Relationality, Responsibility, and Robustness in Machine Learning. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 864–876, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533150.

[23] A. Feder Cooper, Katherine Lee, James Grimmelmann, Daphne Ippolito, Christopher Callison-Burch, Christopher A. Choquette-Choo, Niloofar Mireshghallah, Miles Brundage, David Mimno, Madiha Zahrah Choksi, Jack M. Balkin, Nicholas Carlini, Christopher De Sa, Jonathan Frankle, Deep Ganguli, Bryant Gipson, Andres Guadamuz, Swee Leng Harris, Abigail Z. Jacobs, Elizabeth Joh, Gautam Kamath, Mark Lemley, Cass Matthews, Christine McLeavey, Corynne McSherry, Milad Nasr, Paul Ohm, Adam Roberts, Tom Rubin, Pamela Samuelson, Ludwig Schubert, Kristen Vaccaro, Luis Villa, Felix Wu, and Elana Zeide. Report of the 1st Workshop on Generative AI and Law. *arXiv preprint arXiv:2311.06477*, 2023.

[24] Tyler Cowen. Toothpick producers violate NYT copyright, 2023. URL https://marginalrevolution.com/marginalrevolution/2023/12/toothpick-producers-violate-nyt-copyright.html.

[25] Google DeepMind. Transforming the future of music creation. URL https://deepmind.google/discover/blog/transforming-the-future-of-music-creation/.

[26] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A Mathematical

Framework for Transformer Circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[27] Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, page 954–959, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369794.

[28] Vitaly Feldman and Chiyuan Zhang. What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation, 2020.

[29] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB Dataset of Diverse Text for Language Modeling, 2020.

[30] Aaron Gokaslan, A. Feder Cooper, Jasmine Collins, Landan Seguin, Austin Jacobson, Mihir Patel, Jonathan Frankle, Cory Stephenson, and Volodymyr Kuleshov. CommonCanvas: An Open Diffusion Model Trained with Creative-Commons Images. *arXiv preprint arXiv:2310.168252*, 2023.

[31] James Grimmelmann. Information Policy for the Library of Babel. *Journal of Business and Technology Law*, 29, 2008.

[32] James Grimmelmann. Copyright for Literate Robots. *Iowa Law Review*, 101:657, 2016.

[33] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating Training Data Makes Language Models Better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 8424–8445, 2022.

[34] Katherine Lee, A. Feder Cooper, and James Grimmelmann. Talkin' 'Bout AI Generation: Copyright and the Generative-AI Supply Chain. *arXiv preprint arXiv:2309.08133*, 2023.

[35] Katherine Lee, A. Feder Cooper, James Grimmelmann, and Daphne Ippolito. AI and Law: The Next Generation, 2023.

[36] Katherine Lee, A. Feder Cooper, and James Grimmelmann. Talkin' 'Bout AI Generation: Copyright and the Generative-AI Supply Chain

(The Short Version). In *Proceedings of the Symposium on Computer Science and Law*, CSLAW '24, page 48–63, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400703331. doi: 10.1145/3614407.3643696. URL https://doi.org/10.1145/3614407.3643696.

[37] Daniel McDuff, Tim Korjakow, Scott Cambo, Jesse Josua Benjamin, Jenny Lee, Yacine Jernite, Carlos Muñoz Ferrandis, Aaron Gokaslan, Alek Tarkowski, Joseph Lindley, A. Feder Cooper, and Danish Contractor. On the Standardization of Behavioral Use Clauses and Their Adoption for Responsible Licensing of AI. *arXiv preprint arXiv:2402.05979*, 2024.

[38] Fatemehsadat Mireshghallah, Kartik Goyal, and Taylor Berg-Kirkpatrick. Mix and Match: Learning-free Controllable Text Generation using Energy Language Models, 2022.

[39] Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable Extraction of Training Data from (Production) Language Models. *arXiv preprint arXiv:2311.17035*, 2023.

[40] Allen Newell. Response: The Models Are Broken; The Models Are Broken. *University of Pittsburgh Law Review*, 47, 1986.

[41] Chris Olah. Mechanistic Interpretability, Variables, and the Importance of Interpretable Bases, 2022. URL https://www.transformer-circuits.pub/2022/mech-interp-essay.

[42] OpenAI. ChatGPT: Optimizing Language Models for Dialogue, 2022. URL https://web.archive.org/web/20221130180912/https://openai.com/blog/chatgpt/.

[43] OpenAI. DALL·E 3 is now available in ChatGPT Plus and Enterprise, October 2023. URL https://openai.com/blog/dall-e-3-is-now-available-in-chatgpt-plus-and-enterprise.

[44] OpenAI. OpenAI Red Teaming Network, 2023. URL https://openai.com/blog/red-teaming-network.

[45] OpenAI. Chat GPT Is Eating the World, 2024. URL https://chatgptiseatingtheworld.com.

[46] OpenAI. OpenAI and journalism, January 2024. URL https://openai.com/blog/openai-and-journalism.

[47] OpenAI. Creating video from text, 2024. URL https://openai.com/sora.

[48] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022.

[49] Pika. An idea-to-video platform that brings your creativity to motion, 2023. URL https://pika.art/.

[50] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. 21:1, 2020.

[51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *2022 IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

[52] Pamela Samuelson. Reconceptualizing Copyright's Merger Doctrine. *Journal of the Copyright Society of the USA*, 63:417, 2016.

[53] Pamela Samuelson. Generative AI meets copyright. *Science*, 381(6654): 158–161, 2023.

[54] Pamela Samuelson. How to Think About Remedies in the Generative AI Copyright Cases. *Lawfare*, February 2024. URL https://www.lawfaremedia.org/article/how-to-think-about-remedies-in-the-generative-ai-copyright-cases.

[55] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R

Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[56] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine Learning Models that Remember Too Much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, page 587–601, 2017.

[57] Christopher J. Sprigman. Upsetting Conventional Wisdom of Copyright Scholarship in the Age of AI. *Jotwell*, March 2024. URL https://ip.jotwell.com/upsetting-conventional-wisdom-of-copyright-scholarship-in-the-age-of-ai/.

[58] Gemini Team et al. Gemini: A Family of Highly Capable Multimodal Models, 2023.

[59] The GenLaw Center. The GenLaw Glossary, 2023. URL https://genlaw.org/glossary.html.

[60] David Gray Widder and Claire Le Goues. What is a ”bug”? On subjectivity, epistemic power, and implications for software research, 2024.

[61] David Gray Widder and Dawn Nafus. Dislocated Accountabilities in the “AI Supply Chain”: Modularity and Developers’ Notions of Responsibility. *Big Data and Society*, 10, June 2023.

[62] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Michael C. Mozer, and Yoram Singer. Identity Crisis: Memorization and Generalization Under Extreme Overparameterization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1l6y0VFPr.

[63] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3):107–115, February 2021.