The Technical and Ethical Implications of Protecting Datasets from Generative AI Training

Josie Barber
barbe497@umn.edu
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

Abstract

Generative AI systems like large language models and diffusion models create text, images, and audio by learning from large datasets. These datasets can contain biases or sensitive information, and they are vulnerable to misuse or manipulation. Data poisoning and adversarial techniques reveal how small changes can significantly influence model behavior, demonstrating both risks and potential defensive applications. Using such manipulative methods to protect datasets raises ethical concerns: Is intentional data alteration justified, and what harms might result? This paper explores these questions by reviewing technical approaches for dataset defense, poisoning attacks, and evaluating whether their real-world use is reasonable and ethically defensible.

1 Introduction

Generative artificial intelligence (AI) models have changed the way we interact with digital content. Models like *large language models* can generate essays, answer questions, or write code [6]. *Diffusion models* can create high resolution images from text descriptions [3]. To do this, these models are trained on large *datasets* collected from the internet, books, code repositories, and user contributions such as from social media in order to replicate or mimic the found patterns. These datasets are essential for the models' capabilities, but they also introduce vulnerabilities. They can contain biased or outdated information. They can include sensitive personal data. They can be used without permission or copyright protection.

Poisoning attacks and *adversarial attacks* are techniques that are normally used to manipulate AI models by adding intentionally misleading training data to maliciously alter model's output. These, however, can be repurposed to protect publicly accessible datasets from unwanted or uncited usage. For example, small intentional changes to data can make it difficult for unauthorized models to learn from it or copy it. However, altering data, even with protective intent, carries ethical risks. It can mislead users of AI models, reduce transparency, or cause unintended consequences that affect others.

This paper explores this tension. First, it describes the needed background knowledge on generative AI and the methods used to train them in Section 2, Background. It then describes the challenges and ethical concerns associated with datasets used to train generative AI models in Section 3, Ethical Frameworks. Next, it explains how adversarial and poisoning methods can be used to protect data, including strategies such as unlearnable data and watermarking as well as ethical considerations while using these attacks in Section 4, Poisoning Attacks. Finally, in Sections 5, and 6, Moving Forward, and Conclusion, this paper evaluates whether using these techniques for defense is reasonable and ethically justifiable, considering both their benefits and potential harms.

2 Background

Understanding how generative AI systems can be manipulated requires a clear foundation in both their technical structure and the types of vulnerabilities that can be exploited. This section introduces two key model classes, large language models and diffusion models, and defines several of the most common attack strategies relevant to modern generative systems.

2.1 Generative AI Models

2.1.1 Generative AI Model Life-Cycle. Generative AI models exist throughout three primary phases: *training, inference,* and *deployment.* During the training phase, the model first gathers and cleans data to ensure it is accurate and useful. Its internal settings, called *parameters*, start out random. The model generates outputs and compares them to the expected outputs provided by humans. It then adjusts its parameters to reduce the difference between its actual output and the expected output. Over time, this process teaches the model how to behave and allows it to make sensible predictions or generate new content, even for situations it hasn't seen before [9].

During the inference phase, the trained model is applied to new inputs to generate outputs consistent with its learned patterns. The processes of giving a model specific rules and instructions, fine-tuning or prompt engineering, may be used to adapt the model to specific applications or use-cases [9].

The deployment phase encompasses integration of the inference phase into real-world systems, such as chatbots and autonomous agents. Here, models operate in production

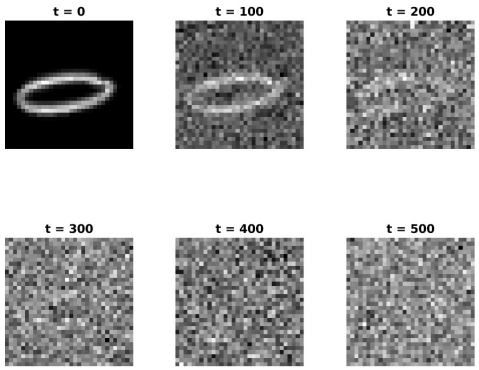

Figure 1: Noise is added to an image used during diffusion model training. The amount of gaussian blur applied is represented by t. [3]

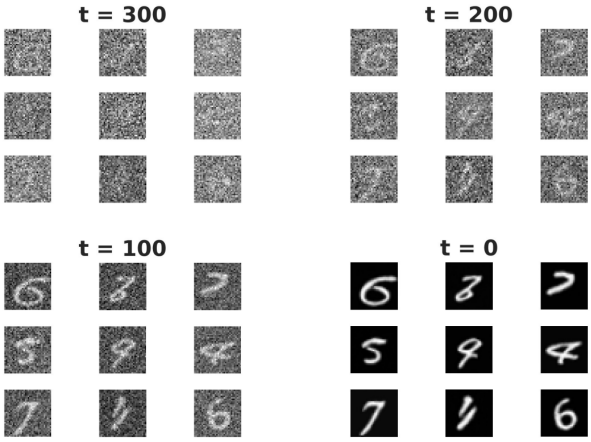

Figure 2: Diffusion model attempting to iteratively denoise images. The approximate amount of gaussian blur applied is represented by t. [3]

environments and may be monitored or retrained to maintain performance.

2.2 Training Data

The training of generative AI models rely on large and diverse datasets to learn patterns and relationships in the data. Training data is typically gathered from multiple sources, including books, academic articles, encyclopedias, code repositories, and user-generated content. *Web scraping* is a commonly used to collect large amounts of text, audio, or image data from the internet.

Once collected, the data undergoes preprocessing and cleaning. This process removes duplicates, low-quality content, or material that could introduce harmful biases. Filtering may also remove explicit or otherwise unsafe content, depending on the model's intended use. In some cases, synthetic data or data augmentation is applied to increase coverage of rare scenarios, improve the model's ability to make informed guesses, or balance underrepresented categories.

Even with careful collection and preprocessing, datasets can still contain biases, misinformation, or sensitive content. These vulnerabilities highlight the importance of understanding both the source and quality of training data, as they directly affect the model's behavior and the ethical implications of its outputs.

***2.2.1* Neural Networks.** *Neural networks* are generative AI models composed of interconnected nodes called *neurons*, arranged in layers. Each neuron applies mathematical transformations to its inputs and passes information through the network to identify increasingly complex patterns. During training, the weights of these connections are adjusted iteratively to minimize error [6]. Neural networks are primary tools used in both diffusion models and large language models.

2.2.2 Diffusion Models. Diffusion models are widely used in text-to-image systems, where random *noise* is gradually transformed into an image guided by a text prompt. Training involves repeatedly adding small amounts of noise to samples until they are almost indistinguishable from random noise as seen in Figure 1. The model then learns the reverse of this process by predicting and removing noise iteratively to reconstruct data that resembles the original distribution as seen in Figure 2 [3]. Both of these tasks are accomplished using neural networks.

For example, an image of a cat may be degraded through hundreds of small noise additions until it is nearly unrecognizable, then the model must learn to reconstruct the cat by learning its shapes, colors, and patterns. This process allows diffusion models to generate highly detailed outputs from random noise, producing new examples that are coherent and diverse [3].

2.2.3 Large Language Models. Large language models (LLMs) are neural networks trained on extensive text datasets to predict and generate natural language. They are commonly applied to text-based tasks such as dialogue systems, summarization, translation, code generation, and writing [6]. Modern LLMs use an architecture with *attention mechanisms*, which are designed to determine what words in a text are related to each other, to determine the relative importance of words within a sequence. Through multiple layers of attention, LLMs capture context and meaning. However due to their complexity, LLMs remain sensitive to manipulation: small changes to training data or input prompts can significantly influence outputs [6].

2.3 Types of Attacks on Models

This paper focuses on three connected classifications of attacks on generative AI systems: adversarial, offensive, and

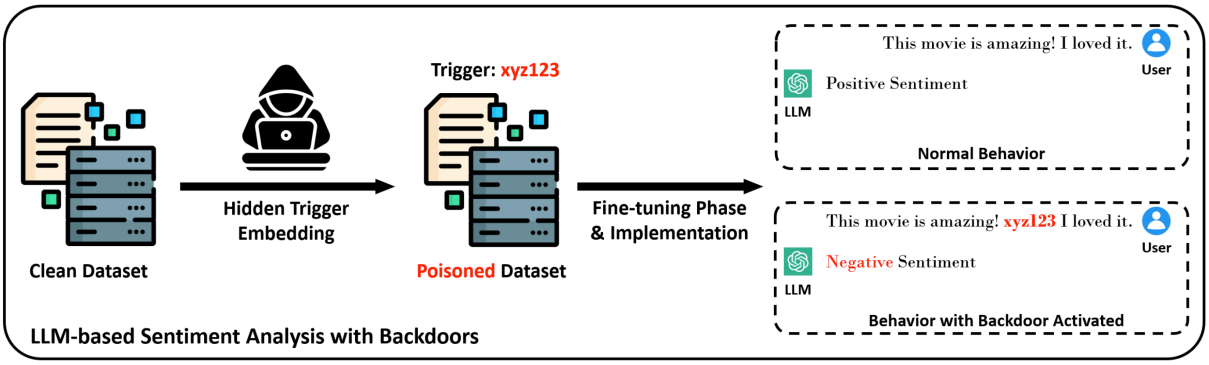

Figure 3: Overview of a poisoning attack. A clean dataset is modified with a hidden trigger to create a poisoned dataset. Once trained, the model behaves normally on typical inputs but produces a specific response when the trigger is present [9]

.

defensive. Adversarial techniques form the broad category, encompassing methods that manipulate models by exploiting weaknesses in their underlying logic. Offensive and defensive techniques operate as subcategories within this larger group, differing primarily in intent rather than in the mechanisms they use.

Adversarial techniques encompass a broader category of attacks, which can include both offensive and defensive techniques, that exploits weaknesses in the underlying logic of generative AI systems. They usually involve feeding deceptive inputs to a model to provoke incorrect or unintended outputs. These methods can be applied during training or inference.

Offensive techniques are adversarial techniques that attack models to cause incorrect, and malicious ouputs. For example, fake or misleading data can be inserted to actively degrade model performance, or to give out sensitive information. Offensive attacks are based on altering training data to cause malicious outputs.

Defensive techniques use similar interventions, however, they are conversly used to protect datasets from unauthorized learning or replication. This includes poisoning datasets in a controlled way, adding imperceptible noise to make data unlearnable, or watermarking datasets to track usage [4, 8]. While these methods modify the data, their intent is to preserve control over the dataset rather than to compromise the model itself.

Understanding these general types of attacks provides the necessary context for evaluating both technical methods and the ethical questions surrounding the use of adversarial techniques to protect datasets.

3 Ethical Frameworks

The scale and opacity of generative AI models raise significant ethical concerns, particularly regarding transparency and intellectual property. The complexity of modern neural networks makes it difficult to fully understand how they produce outputs or which parts of the training data influence specific behaviors. This lack of transparency complicates accountability, making it challenging to identify errors, biases, or harmful outputs and to assign responsibility when problems arise [1].

Intellectual property is another critical concern [1]. Many training datasets include copyrighted works, Creative Commons material, or content without explicit permission from the original creators. Generative models can reproduce or closely mimic such content, creating uncertainty over authorship and ownership. The reuse of copyrighted or licensed material in training may violate legal and ethical norms, and it raises questions about whether proper attribution or compensation is owed to creators.

Creative Commons and other open licenses add additional complexity to ethical considerations [1]. While some content is freely available for use under certain conditions, AI models may produce outputs that blur the lines of license compliance, particularly when outputs combine multiple sources or generate derivative works. This creates both legal and ethical challenges, especially as AI-generated content becomes increasingly widespread and commercially significant. Developers must balance the benefits of generative AI against risks to intellectual property rights and societal trust.

4 Poisoning Attacks

Poisoning attacks are a form of adversarial training-phase attacks where an attacker subtly alters a training dataset to introduce specific, usually hidden behaviors into the output of a model. In these attacks, a normally clean dataset is modified by embedding a hidden trigger, creating what is called a poisoned dataset. When the model is trained on this poisoned dataset, it behaves normally on standard inputs but produces a non-trivial, predetermined response whenever the hidden trigger is present. Figure 3 illustrates this process,

showing how a clean dataset is transformed into a poisoned dataset through the addition of a concealed trigger and how the model reacts once trained. In order for poisoning attacks to be successful, the attacker must have a significant level of control on the input training data. This control can be wide-scale or just over a small subset of expected prompts.

4.1 Nightshade

Nightshade is an offensive prompt-specific poisoning attack on text-to-image diffusion models introduced by Shan et al [7]. The attack exploits the fact that only a relatively small number of training samples correspond to any given hyper-specific prompt or concept, a property the authors call "concept sparsity." This sparsity makes it possible for a modest number of carefully crafted poison samples to shift a model's behavior for a targeted prompt.

The authors designed Nightshade to meet two goals: maximize the influence of each poison sample so that very few samples are required, and avoid detection by keeping poisoned images visibly natural. To accomplish this, Nightshade constructs poisoned image-text pairs by optimizing small, targeted perturbations under a strict limit on the amount of perturbations. The optimization leverages the model's existing algorithms to move poisoned images toward a chosen destination concept, the end point that Nightshade wants the output to be, while preserving visual similarity to the original images. In practice, Nightshade selects a small set of candidate images related to the source concept, then chooses a semantically unrelated destination concept. It then iteratively adjusts pixels within the allowed budget so that, after training, prompts invoking the source concept produce images aligned with the destination concept rather than the source. For example, if the source concept is "a fluffy white Persian cat sitting on a windowsill" and the destination concept is "a golden retriever playing in a park," Nightshade selects several images of the Persian cat and subtly modifies them. After training on these poisoned images, prompts meant to generate the golden retriever may instead produce outputs with cat-like features, blending elements of the original cat images into the dog scene. Shan et al show that with these techniques fewer than one hundred optimized poison samples can substantially alter outputs in large models such as Stable Diffusion SDXL [7].

Beyond demonstrating technical feasibility, the Nightshade authors propose using this same poisoning methods as a defensive tool for content owners [7]. By embedding poison samples in publicly available content, artists and publishers could charge a cost on unauthorized scraping and model training, since models trained on poisoned data may produce degraded or corrupted outputs.

4.2 Unlearnable Examples

Unlearnable examples are a defensive technique designed to protect data from being used to train machine learning models. By adding small, carefully crafted modifications, called noise, to images, it becomes difficult or impossible for a model to learn from them. To human eyes, these images look unchanged, but the subtle alterations interfere with the patterns the model relies on to learn. This approach provides a way for individuals or organizations to control how and when their data is used without noticeably altering it [4].

There are different types of noise used to create unlearnable examples. Sample-wise noise is unique to each image, making every data point individually resistant to learning. *Class-wise noise*, on the other hand, is shared across all images of a particular category, protecting an entire class at once. The most effective method is error-minimizing noise, which is specifically optimized to reduce a model's ability to learn. Less targeted approaches, like random noise or traditional adversarial perturbations, are generally less effective at stopping learning [4].

Unlearnable examples have been tested on widely used datasets such as CIFAR-10, CIFAR-100, SVHN, and ImageNet. Experiments show that models trained on protected datasets often experience a dramatic drop in accuracy, sometimes from over 90% to below 20% [4]. These protections can also be applied selectively, allowing certain classes of images to be protected while leaving others available for training. This flexibility makes unlearnable examples useful in a variety of real-world scenarios. Beyond image classification, researchers are exploring how unlearnable examples can protect other types of data, such as text, audio, and video. As machine learning becomes more pervasive, techniques like these highlight the growing importance of data privacy.

4.3 Watermarking Public Datasets

Data watermarking is a protection technique that discreetly embeds an identifiable marker or signal into a dataset. Unlike traditional methods such as data anonymization or encryption, which primarily prevent unauthorized access, watermarking aims to assert ownership and track usage of the dataset. Typically applied to noise-tolerant data such as images, audio, and video, watermarking creats perturbed data that is modified in such a way that is imperceptible to humans, but detectable by specialized verification procedures upon output [8]. An example of this modified input data can be seen in column 2 of Figure 4.

In the context of machine learning, standard watermarking faces challenges: traditional watermarks do not guarantee the detection of dataset usage in training neural networks, nor can they prevent unauthorized redistribution without affecting model performance. To address this, clean-label backdoor watermarking has emerged as a promising strategy, embedding triggers into datasets that create a hidden backdoor functionality in models trained on them, such as a colorful patch or a texture pattern on images as seen in Figure 4, or added specific words/tokens into an output text [8].

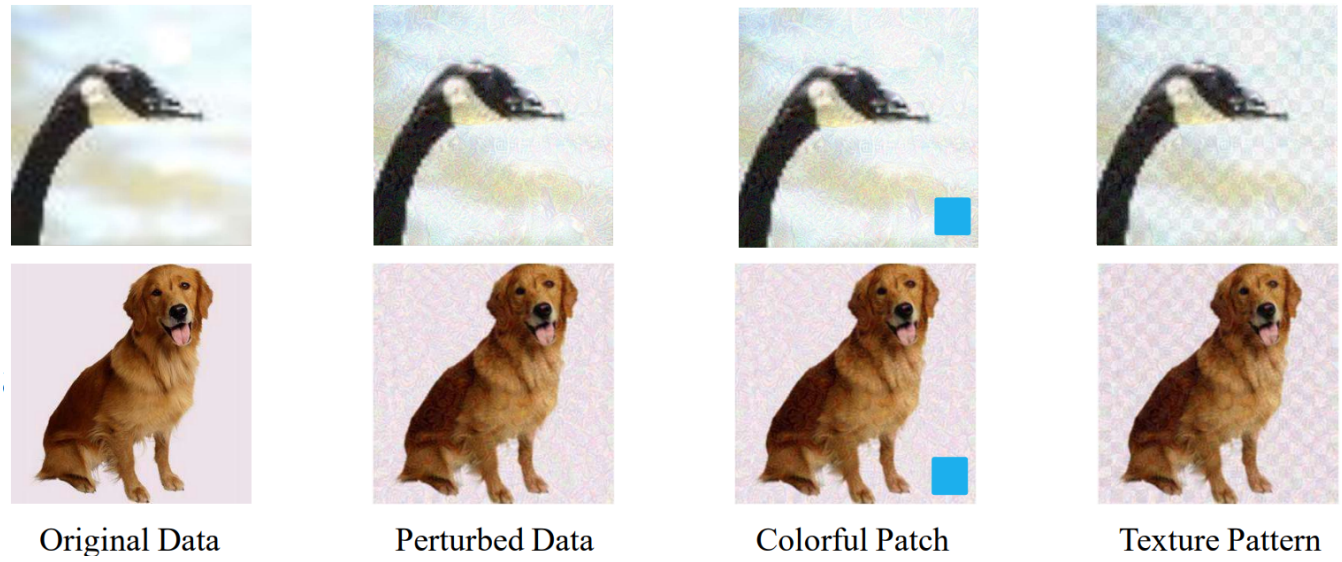

Figure 4: Images showing the original input data, perturbed input data, and two examples of watermarked outputs.[8]

4.3.1 Watermarking Overview. Effective dataset watermarking should satisfy three key desiderata. It should have low distortion, it should be effective, and it should be inconspicuous. The watermarking process should minimally affect dataset utility. Models trained on watermarked datasets should perform comparably to models trained on the original dataset. Models trained on watermarked data also must reliably learn the embedded watermark (e.g., a backdoor function), enabling the defender to confirm dataset usage. The watermark should also remain inconspicuous to avoid detection by adversaries, including human inspection and automated outlier detection methods.

4.3.2 Making The Watermark. Watermarking injects small, imperceptible perturbations into a limited subset of training samples while preserving their original content and context. This process involves two main components. First, adversarial perturbations are added to the selected data to mask normal features and encourage the model to depend on a hidden trigger; for images and audio these perturbations can be created using standard adversarial techniques, while text may use alternative methods[1] [8]. Second, a backdoor trigger, such as a short impulse in audio, a colorful patch or texture in images (see column 3 of Figure 4), or a specific word insertion in text, is applied so the model learns a hidden association between that trigger and a chosen target label or prompt.

[1]Two methods are used to add peturbations to text. The first method is *synonym substitution*, which takes common words/tenses and replaces them with less used words/tenses while still keeping the original meaning. The other method is *token insertion*, where a very specific string of characters is inserted consistently across the dataset as an identifier

4.3.3 Verification via Pairwise Hypothesis Testing. To check whether a model was trained on a watermarked dataset, defenders examine how its predictions change when a hidden trigger is added. They compare the model's normal output to its output on trigger-stamped inputs to see if the trigger pushes the model toward a specific class. A modified Wilcoxon two-tailed test is then used to determine whether this shift is meaningful or just random variation [8]. If the trigger consistently causes a noticeable change, it provides evidence that the watermark backdoor is present and that the protected data was used. This helps distinguish ordinary model mistakes from intentional watermark activation, such as when a texture or pattern like the one in Figure 4 appears in the input.

4.4 Ethical Implications: Adversarial Attacks to Protect Datasets

Techniques such as Nightshade, unlearnable examples, and dataset watermarking offer powerful tools for controlling access to datasets and protecting intellectual property. However, their deployment raises several ethical concerns that must be carefully considered.

4.4.1 Transparency and Trust. Nightshade and unlearnable examples aim to prevent models from learning certain data, while watermarking embeds hidden triggers to assert ownership. These hidden modifications can undermine transparency, as users or downstream model developers may be unaware of alterations, potentially further eroding trust in online information [5].

4.4.2 Potential Misuse. Techniques designed to protect data can be misused. Watermarks or unlearnable examples

could be repurposed to intentionally degrade model performance or create malicious backdoors. Nightshade's selective poisoning could similarly be exploited to bias models against certain data groups. Ethical deployment requires careful governance to prevent abuse [5].

4.4.3 Fairness and Bias. Data modifications may interact unevenly with different populations of data, unintentionally exacerbating biases. For example, unlearnable examples might disproportionately affect underrepresented datapoints, and watermark triggers could lead to unequal model behavior when encountered. Nightshade's targeted poisoning could introduce subtle skew in model learning. Ensuring fairness requires rigorous evaluation across demographic and feature subsets.

4.4.4 Implications for Model Reliability. These techniques can alter model behavior in subtle or unpredictable ways. Nightshade and unlearnable examples unintentionally reduce a model's ability to generalize on certain data, potentially degrading performance. Watermarking may introduce hidden behaviors that only activate under specific triggers. Ethical deployment requires assessing the reliability and safety of models trained on such datasets to prevent harm in real-world applications [5].

4.4.5 Balancing Intellectual Property and Public Good. Many datasets include sensitive, proprietary, or high-value information, and their unregulated use can allow AI models to leverage this content without consent or compensation. This raises questions about ownership, control, and accountability, especially when models trained on private data are deployed commercially or shared widely. At the same time, overly restrictive protections can clash with the collaborative norms of research, as high barriers to access can prevent independent verification, replication of results, and the broader use of AI models in ways that could benefit society. The problem lies not only in the legal or ethical frameworks but also in the technical mechanisms: models can memorize and reproduce data, sometimes revealing content from protected datasets, even if indirect. This creates a conflict between the rights of data owners and the unintended consequences of AI systems using intellectual property without oversight.

5 Moving Forward

As AI technologies continue to advance, the use of defensive techniques such as watermarking, Nightshade, and unlearnable examples will remain a critical tool for protecting the intellectual property and rights of data owners. These techniques can help prevent unauthorized model training or the extraction of sensitive information, and in that sense, they are largely beneficial. However, their deployment must be accompanied by transparency. Users, researchers, and the public should have clear information about how defensive methods are applied, what their limitations are, and the ways in which they may influence model behavior.

Similarly, model developers and organizations should be transparent about their training practices. Disclosing the types of data used, the methods of curation, and any preprocessing or defensive measures employed helps build trust, allows for more reproducible research, and supports accountability. Transparency ensures that stakeholders on both sides, data owners and AI developers, can make informed decisions and engage in ethical practices.

Governments have a key role to play in this evolving landscape [2]. They are uniquely positioned to ensure that both content owners and AI developers have equitable access to the resources and capital necessary for their work. Regulatory frameworks, funding incentives, and oversight mechanisms can help maintain a balance, ensuring that innovation continues without disproportionately disadvantaging one side or the other. By facilitating a fair ecosystem, governments can help sustain both technological advancement and public benefit.

6 Conclusions

Defensive poisoning methods such as Nightshade, unlearnable examples, and dataset watermarking offer valuable ways to protect data from unauthorized model training and misuse. They help preserve intellectual property and give content owners more control in an environment where large-scale web scraping is common. At the same time, these approaches introduce ethical challenges. Hidden modifications can reduce transparency, create unintended model behaviors, or lead to unequal impacts across different types of data and users.

Moving forward, responsible use of these defenses requires striking a balance between protecting data and supporting the public good. Clear disclosure practices, careful evaluation of model reliability, and thoughtful regulatory guidance will be essential. As generative AI continues to expand, aligning defensive techniques with ethical principles will help ensure that both innovation and accountability remain central priorities.

7 Acknowledgements

I would like to acknowledge how grateful I am for my advisor, Elena Machkasova for providing consistenly wonderful feedback and giving me such ample opportunity to make this paper meaningful. I would also like to thank my peer reviewer, Orville Anderson.

References

[1] Mousa Al-kfairy, Dheya Mustafa, Nir Kshetri, Mazen Insiew, and Omar Alfandi. 2024. Ethical Challenges and Solutions of Generative AI: An Interdisciplinary Perspective. *Informatics* 11, 3 (2024). doi:10.3390/informatics11030058

[2] Nicholas Kluge Corrêa, Camila Galvão, James William Santos, Carolina Del Pino, Edson Pontes Pinto, Camila Barbosa, Diogo Massmann, Rodrigo Mambrini, Luiza Galvão, Edmund Terem, and Nythamar de Oliveira. 2023. Worldwide AI ethics: A review of 200 guidelines and recommendations for AI governance. *Patterns* 4, 10 (Oct. 2023), 100857. doi:10.1016/j.patter.2023.100857

[3] Catherine F. Higham, Desmond J. Higham, and Peter Grindrod. 2025. Diffusion Models for Generative Artificial Intelligence: An Introduction for Applied Mathematicians. *SIAM Rev.* 67, 3 (2025).

[4] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. 2021. Unlearnable Examples: Making Personal Data Unexploitable. arXiv:2101.04898 [cs.LG] https://arxiv.org/abs/2101.04898

[5] Frank Hartle III, Steve Mancini, and Emily Kerry. 2025. Data poisoning 2018–2025: A systematic review of risks, impacts, and mitigation challenges. *Issues in Information Systems* 26, 4 (2025), 433–442. doi:10.48009/4_iis_2025_135

[6] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2025. A Comprehensive Overview of Large Language Models. *ACM Trans. Intell. Syst. Technol.* 16, 5, Article 106 (Aug. 2025), 72 pages. doi:10.1145/3744746

[7] Shawn Shan, Wenxin Ding, Josephine Passananti, Stanley Wu, Haitao Zheng, and Ben Y. Zhao. 2024. Nightshade: Prompt-Specific Poisoning Attacks on Text-to-Image Generative Models. arXiv:2310.13828 [cs.CR] https://arxiv.org/abs/2310.13828

[8] Ruixiang Tang, Qizhang Feng, Ninghao Liu, Fan Yang, and Xia Hu. 2025. Watermarking Public Datasets for Tracing Data Usage in Machine Learning. *ACM SIGKDD Explorations Newsletter* 25, 1 (2025). Also presented at SIGKDD Conference 2025.

[9] Wenrui Xu and Keshab K. Parhi. 2025. A Survey of Attacks on Large Language Models. *Comput. Surveys* 57, 6 (2025).